FREE DVD

Brand New!
Ubuntu 18.04
DVD Inside

ECLIPSE TOOLS
for IoT projects

INTERNET OF THINGS
Tools for home automation

LINUX PRO MAGAZINE

JULY 2018

INTERNET OF THINGS

Tools for home automation

Bluetooth Tricks
Track movements within your house

Linux on Old Hardware

Tablets on Linux
Configure graphics settings with xsetwacom

Raspberry Pi 3 B+
Does the world need a new Rasp Pi?

Python Gambling
Simulating games of chance

Logism
Design and draw digital circuits

LINUXVOICE

- **soundKonverter and fre:ac:** Full-featured tools for ripping audio CDs
- **Mermaid:** Create diagrams using simple commands
- **maddog:** Government paranoia about encryption

FOSSPicks
- GnuCash 3
- XLEngine

Tutorials
- Recoll
- Minetest

Issue 212
July 2018
US$ 15.99
CAN$ 17.99
0 74820 58049 3 07

WWW.LINUXPROMAGAZINE.COM

HETZNER
ONLINE
SAY HEI
to the new Hetzner Online
DATA CENTER PARK
HELSINKI

WHITHER BLOCKCHAIN?

Dear Reader,

Is blockchain finally getting interesting? I guess it always has attracted some attention – if not as an investment at least as a generator of headlines. Who could forget the Mount Gox bankruptcy [1] or Newsweek's unfortunate "unmasking" of a Bitcoin creator who turned out to be the wrong guy [2].

The value of a Bitcoin has compounded generously in recent years, and the people who got in early got lots of money (unless they lost track of their key [3]). The volatility is certainly insane; the cryptocurrency hit an all time high in December 2017, then lost two thirds of its value in two months, and then sprang back to about half of what it was in December. I'm not an economist, but it seems to me that when the price of anything jumps around that much, buying into it is a game of chance. One invests in Bitcoin the way one might "invest" in a lottery ticket.

Some of the shine is even off Bitcoin mining right now. The cottage industry that has developed around the task of finding Bitcoins using high-performance parallel computers is consuming insane amounts of energy, thus giving the once-innocuous digital currency an unexpectedly negative reputation with environmentalists [4].

But blockchain is more than just Bitcoin. Other recent developments are starting to reveal the real power behind this fast-evolving tech. Some leading jewelry companies, for instance, recently announced they would be using a blockchain-based ledger called Tracr to track and verify the authenticity of diamonds. Tracr is built on the IBM Blockchain platform, which is intended to let companies and industry groups build their own solutions around blockchain technology. The Hyperledger project, which is sponsored by the Linux Foundation, is building a similar collection of blockchain development tools that have found their way into a number of different initiatives, including the Greenstream project, which describes itself as "a blockchain disruptor focused on increasing efficiency and record keeping in the emerging Canadian legal cannabis industry."

Microsoft is developing the Coco framework as a platform for building blockchain solutions, and Amazon, Oracle, and other vendors are working on similar products. This new class of blockchain developer tools will let vendors and enterprise companies build their own applications to secure the supply chain and verify the authenticity of transactions. In the long run, watching the fortunes of this new breed of developer tools will tell you more about the direction of blockchain than following the wild fluctuations in the value of Bitcoin.

Joe

Joe Casad,
Editor in Chief

Info

[1] Wikipedia on Mt. Gox: *https://en.wikipedia.org/wiki/Mt._Gox*

[2] "After Newsweek Reveal, Man Insists He Isn't Bitcoin Creator: *http://nymag.com/daily/intelligencer/2014/03/bitcoin-creator-denies-newsweek-story.html*

[3] "I Forgot My PIN: An Epic Tale of Losing $30,000 in Bitcoin": *https://www.wired.com/story/i-forgot-my-pin-an-epic-tale-of-losing-dollar30000-in-bitcoin/*

[4] "Bitcoin's Energy Use Is Huge": *https://www.theguardian.com/technology/2018/jan/17/bitcoin-electricity-usage-huge-climate-cryptocurrency*

LINUX MAGAZINE

WHAT'S INSIDE

The Internet of Things revolution is tailor made for Linux. The ultimate hacker system is certainly well suited for automating the home. This month we report on some tools and techniques for Linux and the Internet of Things. You'll learn about ioBroker – a universal tool for managing different types of devices from a single interface. You'll also take a tour of the Eclipse IoT project, a collection of libraries and frameworks with the common goal of keeping the Internet of Things open and free.

Other highlights include:

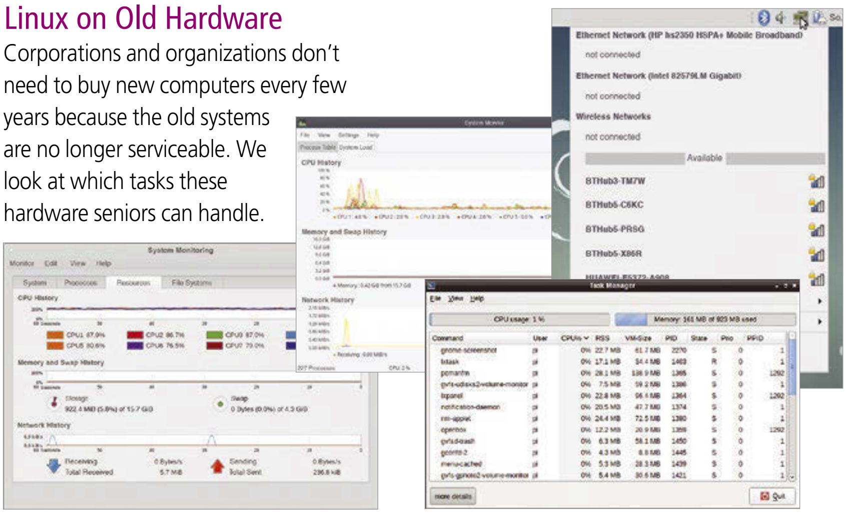
- **Linux on Old Hardware** – no reason to recycle that old computer if you have access to a suitable Linux system (page 36).
- **Raspberry Pi 3 B+** – faster networking, more CPU power, and enhanced wireless support bring new powers to the latest edition of the world's favorite tiny computer (page 54).

Turn ahead to this month's LinuxVoice for a tutorial on the Recoll search engine and a look at some cool tools for digitizing audio CDs.

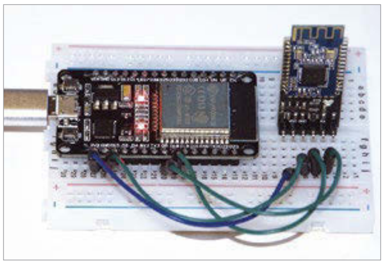
SERVICE

NEWS

REVIEWS

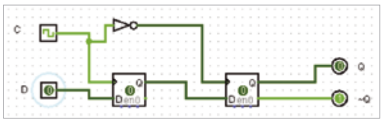


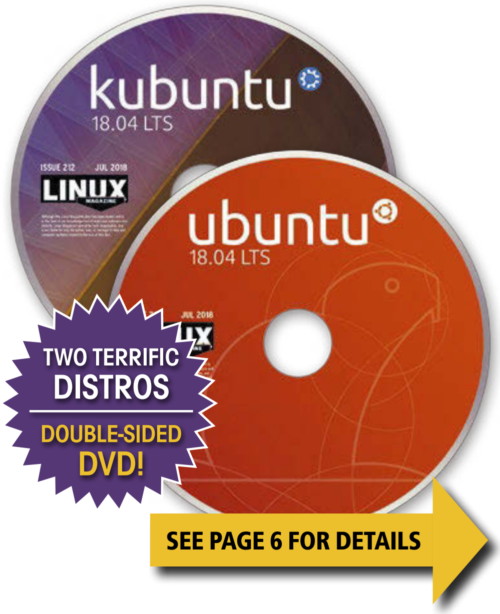
COVER STORIES



IN-DEPTH

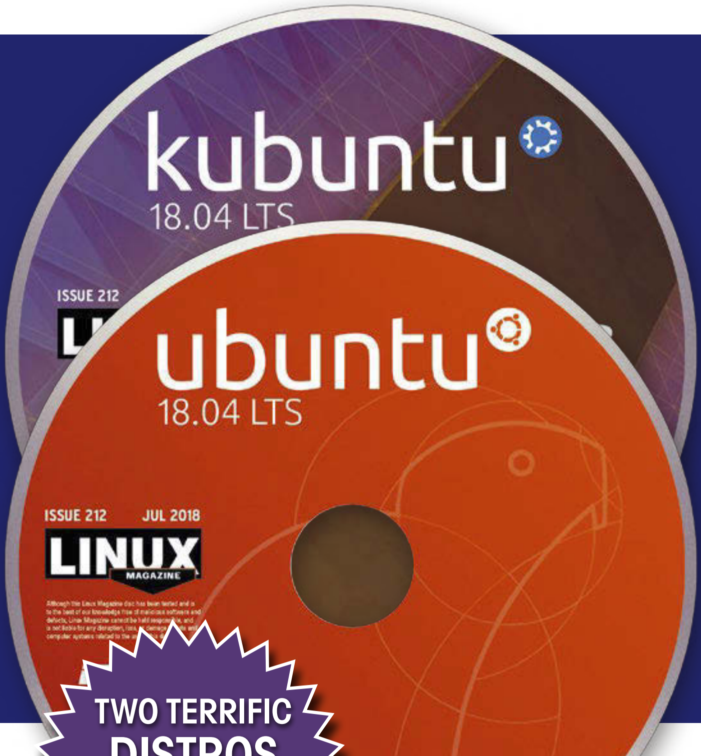
MAKERSPACE





LINUXVOICE

On the DVD

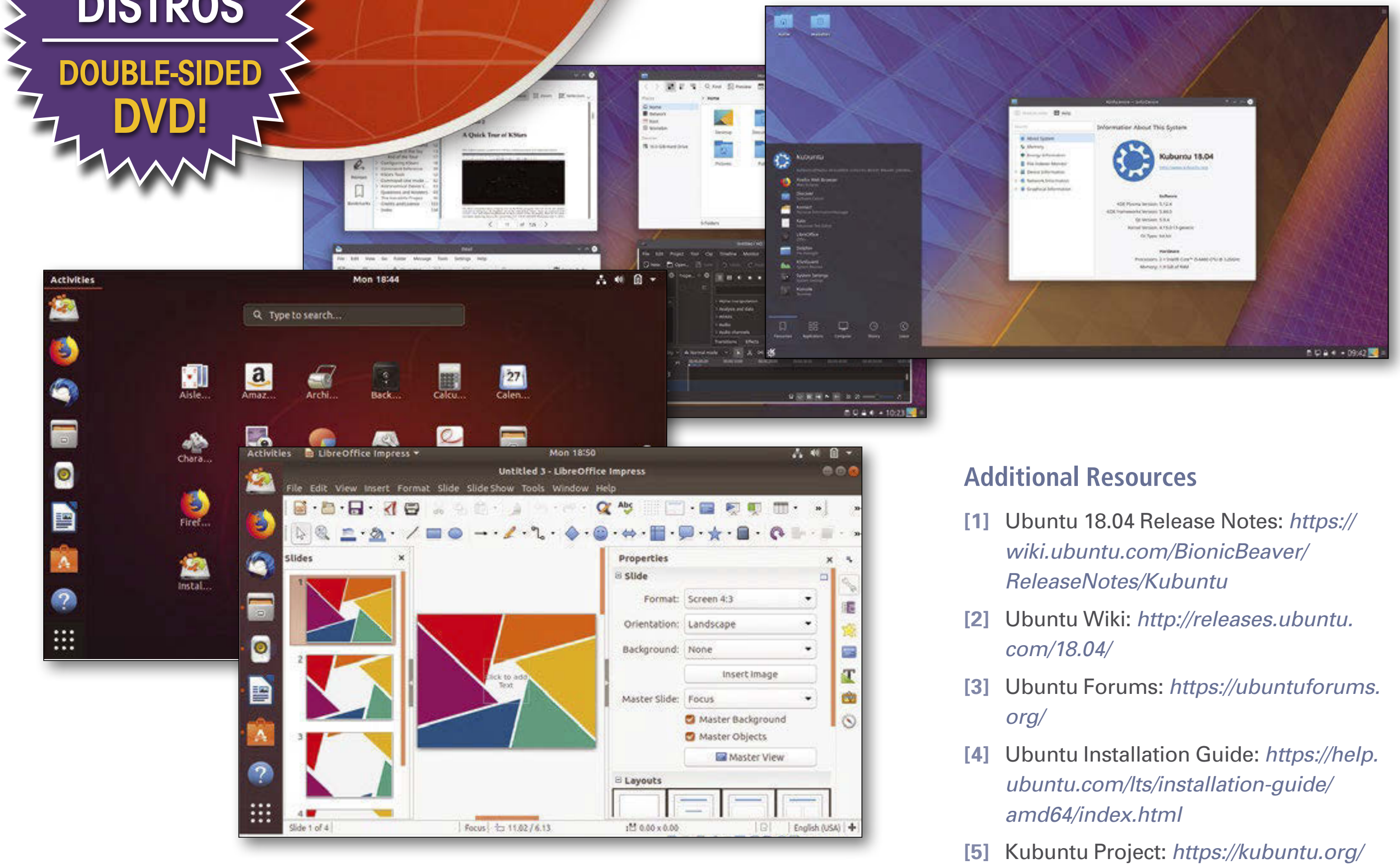


Ubuntu 18.04 "Bionic Beaver" LTS (64-bit)

Ubuntu 18.04 "Bionic Beaver" is a new Long Term Support (LTS) edition, which means it will receive five years of updates and technical support. The latest Ubuntu ships with Linux Kernel 4.15, which includes improved power management and support for AMD secure memory encryption. The new edition also comes with migrations to protect against the Spectre and Meltdown attacks. A new *minimum install* option lets you set up a basic desktop environment with only a web browser and core system utilities.

Kubuntu 18.04 "Bionic Beaver" LTS (32-bit)

Kubuntu is an Ubuntu flavor based on the popular KDE Plasma desktop. The latest version comes with Kubuntu 5.12 LTS, which includes stability improvements and "updates to major packages like Krita, KStars, KDE Connect, Firefox, and LibreOffice." KDE Plasma is built with the Qt toolkit and supports easy integration with mobile phones and tablets.



Defective discs will be replaced. Please send an email to subs@linux-magazine.com.

Additional Resources

[1] Ubuntu 18.04 Release Notes: *https://wiki.ubuntu.com/BionicBeaver/ReleaseNotes/Kubuntu*

[2] Ubuntu Wiki: *http://releases.ubuntu.com/18.04/*

[3] Ubuntu Forums: *https://ubuntuforums.org/*

[4] Ubuntu Installation Guide: *https://help.ubuntu.com/lts/installation-guide/amd64/index.html*

[5] Kubuntu Project: *https://kubuntu.org/*

[6] Kubuntu Forums: *https://kubuntu.org/*

NEWS

Updates on technologies, trends, and tools

THIS MONTH'S NEWS

Ubuntu 18.04 Released

Canonical has announced the release of Ubuntu 18.04, aka Bionic Beaver. It's a long term support (LTS) release that's suitable for enterprise customers and servers. Ubuntu 18.04 LTS will be supported for five years for Ubuntu Desktop, Ubuntu Server, and Ubuntu Core.

Ubuntu 18.04 is also the first Ubuntu LTS, after years, that ships with Gnome as the default desktop shell. In recent releases, Canonical has used its own Unity shell on top of Gnome to gain more influence and control over the user experience.

Ubuntu 18.04 comes with a customized version of Linux Kernel 4.15 that adds support for the latest hardware and peripherals. Some of the hardware-focused improvements that this kernel brings to Ubuntu include a CPU controller for the cgroup v2 interface, AMD secure memory encryption support, the latest AMD driver with software RAID enhancements, and management for systems with SATA Link Power Management.

Java users will continue to use OpenJDK 8, which has moved to universe and will remain available for the life of 18.04. The move is intended to help developers with migration issues for packages, custom applications, or scripts that can't be built with OpenJDK 10 or 11.

Security is one of the core features of Ubuntu 18.04. In a conference call, Mark Shuttleworth, the founder and CEO of Canonical, said that Ubuntu 18.04 is fully protected against Spectre and Meltdown.

Richard Stallman Calls Azure Sphere OS a Positive Step

Earlier this month, Microsoft shook the world with the announcement of Azure Sphere OS, an operating system designed to run on Internet of Things (IoT) devices. The OS is powered by a customized version of the Linux kernel. This marks the first product by Microsoft that runs on the Linux kernel. What's even more interesting is that Microsoft will be shipping a product that has code released under the GNU GPLv2. I talked to Galen Hunt, partner managing director, Microsoft Azure Sphere, who confirmed that Microsoft will comply with the licenses used for the code.

Last week, I met with Richard M. Stallman in his hometown of Boston to get his reaction to the announcement, "That's good. That program is free software. It's released under a free software license I wrote, which requires redistributors to respect the freedom of users and Microsoft is even going to respect the users freedom with regards to that particular program. Well, I'm glad about that," said Stallman.

However, he also criticized the community that coined the term "open source" to make copyleft licenses more acceptable for companies. He said that open source has done harm to the mission and goal of the Free Software Foundation: "...they coined the term open source to disconnect our software from our ethical ideas and they were fairly effective at that. Since then, we have to work hard to teach people, even the users of our software, that there's such a thing as the Free Software movement. That it's a movement for their freedom, that this is not just a matter of more convenient, more reliable software. Those are secondary desirable things, but they're not as important as freedom. Freedom is crucial," he said. Stallman said that he didn't use the term as it would have buried the idea behind the Free Software movement.

When asked how he really feels about Microsoft releasing code that's guided by the license he wrote, Stallman sounded positive, "It's just a beginning; I'm sure there's going to be a lot of non-free software in it. I see it as a small step. Remember, the goal is to kick non-free software out of your life; kick it out of our society. We should all be free. So when you compare this step with that goal, I see it as a small positive step."

RHEL 7.5 Released

Red Hat has released Red Hat Enterprise Linux (RHEL) 7.5, which has a strong focus on hybrid cloud. As the market is evolving, so is Red Hat. In 2014, Red Hat signaled a shift in focus from data centers to mobile and cloud. Red Hat acquired companies like FeedHenry and CoreOS to strengthen its mobile and cloud portfolio.

Now the cash cow of Red Hat, RHEL, is reflecting their changing focus. RHEL 7.5 offers enhanced security and compliance controls, in addition to better integration with Microsoft Windows infrastructure both on-premise and in Microsoft Azure.

Companies are mixing environments – spanning across on-prem, public cloud, and private cloud. RHEL 7.5 tries to reduce the complexity, especially in terms of security, that comes with such a hybrid environment. Red Hat Enterprise Linux 7.5 has enhanced software security controls to mitigate risk.

Red Hat said that a major component of these controls is security automation through the integration of OpenSCAP with Red Hat Ansible Automation. This is designed to enable the creation of Ansible playbooks directly from OpenSCAP scans, which can then be used to implement remediations more rapidly and consistently across a hybrid IT environment. Sensitive data can also now be better secured across varied environments with enhancements to Network-Bound Disk Encryption that support automatic decryption of data volumes.

RHEL 7.5 also comes with production ready container solutions. RHEL 7.5 includes support for Buildah, an open source utility designed to help developers create and modify Linux container images without a full container run time or daemon running in the background.

RHEL 7.5 is available for multiple architectures including x86, IBM Power, IBM z Systems, and 64-bit ARM. Although RHEL is available for subscription, you can download the 30-day evaluation version for free.

Sources: *https://www.redhat.com/en/about/press-releases/red-hat-strengthens-hybrid-clouds-backbone-latest-version-red-hat-enterprise-linux* and *https://access.redhat.com/products/red-hat-enterprise-linux/evaluation*

MORE ONLINE

Linux Magazine
www.linux-magazine.com

ADMIN HPC
http://hpc.admin-magazine.com/

Autonomous File Recovery • Jeff Layton
Have you ever deleted a file and immediately thought, "Ah! I needed that!"?

ADMIN Online
http://www.admin-magazine.com/

Integrating scripts into Group Policy
Thomas Joos
Various kinds of scripts are available for automating tasks in Windows. These scripts start automatically when you log on or log off and when you boot up or shut down the computer.

Dialing up security for Docker containers
Martin Loschwitz
Container systems like Docker are a powerful tool for system administrators, but Docker poses some security issues you won't face with a conventional virtual machine environment.

Preparing to move to the cloud • Udo Seidel
If you believe the advertisements, it's easy to use the services of cloud providers like Amazon, Google, Microsoft, and others. At the same time, the marketing by these companies suggests that the way into the cloud is completely uncomplicated.

ADMIN DevOps Focus
http://www.admin-magazine.com/DevOps

Ansible Configuration Management
Chris Binnie
Among configuration management tools, the most well known are Puppet, Chef, and Ansible, the latter of which was acquired by Linux powerhouse Red Hat at the end of 2015.

Automate AWS AMIs • Chris Binnie
Automation is the long-standing, presiding champion in any DevOps arena, and even more so in cloud environments, where the emphasis is on short-lived, ephemeral resources that can be safely discarded when they've run their course and completed their predetermined task.

Microsoft Releases a Linux-Based OS

Microsoft has announced a new project that is going to be powered by the venerable Linux kernel. At the RSA Conference 2018, the company shed some light on Microsoft Azure Sphere, a new platform to help create secured, Internet-connected microcontroller (MCU) devices.

Microsoft Azure Sphere is an end-to-end solution, all the way from Azure Cloud to actual chips found on the targeted IoT device. Microsoft Azure Sphere is comprised of three components: Azure Sphere certified microcontrollers (MCUs), Azure Sphere OS, and Azure Sphere Security Service.

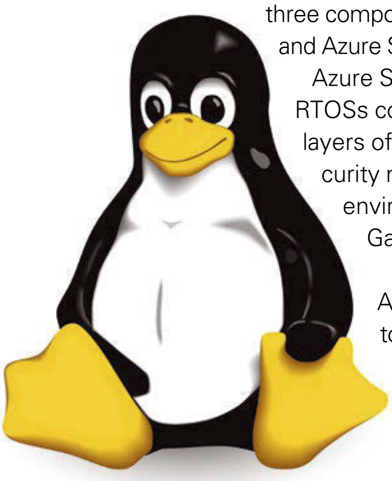
Azure Sphere OS is a custom OS aimed at security and agility. "Unlike the RTOSs common to MCUs today, our defense-in-depth IoT OS offers multiple layers of security. It combines security innovations pioneered in Windows, a security monitor, and a custom Linux kernel to create a highly-secured software environment and a trustworthy platform for new IoT experiences," wrote Galen Hunt, partner managing director, Microsoft Azure Sphere.

Developers can use Microsoft Visual Studio Tools to write applications for Azure Sphere. These tools include application templates, development tools and the Azure Sphere software development kit (SDK).

Visual Studio is not exclusive anymore to Windows. Microsoft open sourced a version of Visual Studio called Visual Studio Code, which is available for Linux.

The news was not surprising; the Microsoft Azure team has long favored better Linux integration. Not only does Linux run on more than 50 percent of Azure machines, the company has been using Linux to build components such as Azure Cloud Switch (ACS) and SONiC.

Source: *https://azure.microsoft.com/en-us/blog/introducing-microsoft-azure-sphere-secure-and-power-the-intelligent-edge/*



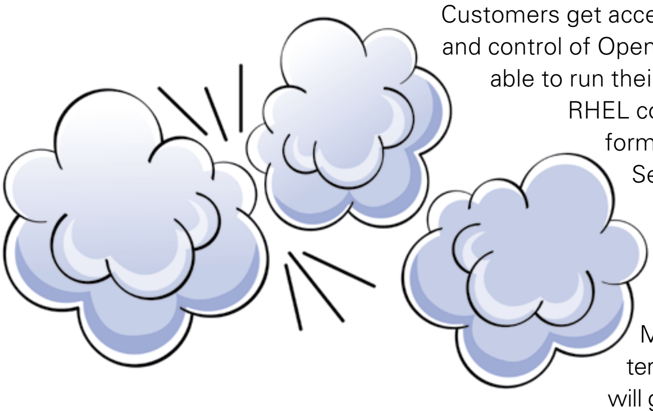
Red Hat, Microsoft Join Forces for Hybrid Cloud

At the Red Hat Summit 2018, Red Hat and Microsoft expanded their relationship with a new agreement to bring managed OpenShift services to Microsoft Azure. OpenShift is Red Hat's own distribution of Kubernetes.

From a technology perspective, the two companies will be working together on OpenShift on Azure to reduce the complexity of container management for customers. From a customer perspective, Red Hat and Microsoft will jointly manage the solution for customers, with support from both companies.

Customers get access to the Azure public cloud with the flexibility and control of OpenShift on-premises on Azure Stack. Users will be able to run their workloads on Windows containers alongside RHEL containers, using a uniform orchestration platform. Not only that, users will be able to use SQL Server as a Red Hat–certified container for deployment on Red Hat OpenShift on Azure and Red Hat OpenShift Container Platform across the hybrid cloud.

Under the partnership, users will be able to use Microsoft tools with Red Hat, and Visual Studio Enterprise and Visual Studio Professional subscribers will get RHEL credits. For the first time, developers can work with .NET, Java, or the most popular open source frameworks on this single supported platform.



Red Hat OpenShift on Azure is anticipated to be available in preview in the coming months. Red Hat OpenShift Container Platform, RHEL on Azure, and Azure Stack are currently available.

Zack's Kernel News

Chronicler Zack Brown reports on the latest news, views, dilemmas, and developments within the Linux kernel community. *By Zack Brown*

The NOVA Filesystem

As flash drives and other forms of nonvolatile RAM continue to supplement or replace traditional optical and magnetic technologies, new filesystems emerge to support them. Recently Andiry Xu, Lu Zhang, and Steven Swanson posted an update to their nascent NOn-Volatile memory Accelerated (NOVA) filesystem.

NOVA uses a log-based approach, which means that it writes new data sequentially onto the drive, leaving old data to sit until it's reclaimed by the system. One benefit of this is that older versions of files can be kept as "snapshots." Traditional filesystems written for drives based on rotating magnetic disks tend to update files by seeking to the file's location and updating its data in place. That reduces file fragmentation on those drives, which reduces the amount of time spent on seeking to different parts of the file. Nonvolatile RAM doesn't mind fragmenting all its writes into sequential chunks, because seek times aren't an issue for that kind of hardware.

The NOVA filesystem has reached the stage where it can do some good things, but its developers aren't yet ready to add it into the kernel for general consumption. This time around, they asked for feedback from the kernel folks. There was not a huge amount of discussion, but enough bug reports and general encouragement to indicate that NOVA is likely to go into the kernel as soon as it's ready.

Making System Calls Userspace Only

Lately, there's been a lot of interest in isolating user space from kernel space, partly because of some severe flaws in Intel hardware that have been discovered recently, affecting the security of a large number of computers on planet Earth. The Linux developers want to work around these flaws, while sacrificing as little as possible, especially in the area of speed.

One area affected by this has been the kernel's use of system calls. Traditionally, a system call is the way user software requests services from the kernel. You use system calls to open and close files, write data to disk, and so on. But the kernel itself has also traditionally used system calls to do all those things. To keep system calls entirely a user-space tool, and not a mechanism for hostile actors to gain access to kernel space, the massive task of removing all syscall invocations from the kernel has become necessary.

There's also some urgency, because the best current workarounds to Intel's hardware problems involve taking a sizable speed hit. The kernel developers want to gradually migrate away from those workarounds and replace them with sleeker, hipper solutions that know how to dance.

Like other efforts to massively change the way the kernel does things, this one can't be done all at once. Many system call invocations can be removed from the kernel relatively eas-

Author

The Linux kernel mailing list comprises the core of Linux development activities. Traffic volumes are immense, often reaching 10,000 messages in a week, and keeping up to date with the entire scope of development is a virtually impossible task for one person. One of the few brave souls to take on this task is **Zack Brown**.

ily, but some have subtleties that will take time to figure out and resolve. This is reminiscent of the effort to get rid of the Big Kernel Lock (BKL), in which many instances could be easily replaced by simpler locks, but some had specific requirements that needed more complicated solutions.

Like the BKL effort, this will be an ongoing project. Some parts of the kernel will have to masquerade as user space in order to retain permission to use system calls, at least for now. And of course, the whole effort is being done in conjunction with many related projects, all trying to find better ways of dealing with the Intel hardware flaws.

Extending Module Support to Plain Executables

It may soon be possible to load and run plain ELF executables as kernel modules. Alexei Starovoitov has been working on this, and it seems to offer some security benefits. But it also may introduce new risks that will require careful implementation.

Alexei posted some patches recently for this and listed off some of the benefits. Unlike a regular kernel module, the ELF executable would run as a user-mode process, which means that if it crashed, it wouldn't necessarily bring the entire system down with it. It would also be subject to all the normal controls placed on user processes, including being subject to the out of memory (OOM) killer code, which steps in when a system is almost out of memory and tries to identify and kill whichever process is most likely to be the cause. Exactly how well the OOM killer is able to do that is a tough question that is the subject of much ongoing work; however, with Alexei's patch, it would have a new pool of processes to consider. Debugging, testing, and profiling ELF executables is also something that can be done with regular user tools, which are more plentiful and possibly also more familiar.

These are all good reasons to include the feature in the kernel, and initially Linus Torvalds was very enthusiastic about it. His only initial suggestion was to increase the amount of logging that Alexei's code performed, so it would never be possible for a module to be loaded without the user seeing it.

There were a couple of criticisms at first that turned out not to be much of a problem. For example, Andy Lutomirski suggested that instead of coding this feature into the kernel itself, it could simply be added to the `modprobe` program and exist entirely in user space. Traditionally, anything that can go into user space should go into user space. Why not this?

But Linus felt that the module loading logic had become a real mess when it had been left out of the kernel and was susceptible to poor decisions by its maintainers, with which the kernel would then have to live. Moving the module logic inside the kernel, he said, had greatly improved the situation. The only thing `modprobe` should do, he said, was to track dependencies between modules and load them in without performing any additional checks or changes. Let the kernel handle that stuff. He remarked, "I do *not* want the `kmod` project that is then taken over by systemd, and breaks it the same way they broke firmware loading. [...] Right now `kmod` is a nice simple project. Lots of testsuite stuff, and a very clear goal. Let's keep `kmod` doing one thing, and not even have to care about internal kernel decisions like 'oh, this module might not be a module, but an executable'. If anything, I think we want to keep our options open, in the case we need or want to ever consider short-circuiting things and allowing direct loading of the simple cases and bypassing `modprobe` entirely."

So that objection turned out to be a nonissue. Likewise, Kees Cook had an objection that didn't go very far – he was concerned that Alexei's patch might make security exploits easier, in the event of certain types of bugs appearing in the kernel. Specifically, if it ever inadvertently became possible for a hostile user to break a module out of a virtual system running on top of the kernel, Alexei's feature would allow the ELF module to execute arbitrary code deep within the kernel.

But Linus didn't find that argument compelling either. Specifically, as he and others pointed out, in the circumstance Kees mentioned, a regular kernel module would be much more powerful and dangerous to let loose than an ELF binary. So Alexei's code wouldn't make the risk any worse than it already was. Additionally, they said, Kees's objection depended on there already being an exploitable security hole in the container code that might allow the hostile user to break a module out of that confinement in the first place. Any such security hole would be treated as a bug and fixed. So to Linus, it was also a nonissue.

But over the course of defending Alexei's patch from these criticisms, additional problems came up that were not so easily dismissed.

Linus raised one of these himself, pointing out that when a module was loaded, first its signature was checked, and then if it was OK, the module file would be loaded and run. But there was a moment in between checking the signature and loading the module, where "the `execve()` will end up not using the actual buffer we checked the signature on, but instead just re-reading the file." In which case, he went on, "somebody could maybe try to time it and modify the file after-the-fact of the signature check, and then we execute something else."

There were a couple of things standing in the way of doing this. And at first, it had seemed to Linus that anyone with sufficient privileges on the running system to modify files in the modules directory would already be able to simply run anything as root to begin with. If they could do that, there would be no reason for them to mess with modules, so there was no reason to try to guard against that possibility.

But on reflection, he'd realized that the hostile user wouldn't need to modify files in the modules directory; they would only need to copy files – something they could do with less permissions, yet still use the exploit to run arbitrary code in the kernel.

So, Linus said, this had to be addressed before Alexei's patch could go into the tree.

Additionally, Andy noticed something that was not a security hole in Alexei's patch, but was something almost as bad – a break in the kernel's application binary interface (ABI) backward compatibility. That refers to the ability of an arbitrary piece of compiled code to rely on the kernel behaving the way it did in the past. If the kernel breaks ABI compatibility, then certain pieces of existing compiled user code would break and need to be recompiled from source. And since not all source code is available for all the binaries running on Linux systems today, that represents an unacceptable change. Linus wants everything that can currently run on a Linux system to continue to be able to run.

It's not inconceivable that Linus might allow something to break part of the ABI, but it would be an exceptional circumstance. For example, if it were discovered that a certain part of the ABI contained a security hole that could only be fixed at the expense of ABI compatibility, Linus would make the change without hesitation. Other circumstances are much more iffy, though at times various kernel developers have argued in favor of lumping a bunch of hotly desired ABI changes into one big patch, like taking a swig of medicine and getting it over with.

So Andy's assertion that Alexei's code broke ABI compatibility was also a potential showstopper. Specifically, he explained to Alexei, "Without your patch, `init_module` doesn't keep using the file, so it's common practice to load a module and then delete or unmount it. With your patch, the unmount case breaks. This is likely to break existing user space."

Both Linus's and Andy's issues spawned a lot of discussion over how to fix them in Alexei's patch. And while no absolutely clear solutions seemed to emerge from the discussion, certainly something will be found to address those issues, and the code will eventually go into the kernel. It just seems like a feature a lot of people want, including Linus. Even Kees, who didn't like the security risks, felt that the feature itself was good. So there's plenty of motivation to iron out the details.

An interesting aspect of the whole issue is what does and does not constitute a security problem. As we saw, Kees had a security issue at the start of the discussion that Linus did not consider important. But Kees is the module security person, with authority (granted by Linus) to veto patches that introduce security holes in the kernel's module support. He's clearly very knowledgeable. And yet Linus did not agree with him about the significance of the issue he raised.

This is not just a question of Linus being right and Kees being wrong. A number of high powered security people in the world would still agree with Kees that reducing the available attack surface of a given vulnerability is an important thing to do. And so Kees's initial objection – that if certain bugs appeared in the kernel then Alexei's feature could offer a large and tempting target to hostile attackers – is one with which many of those security people would agree. They would probably argue that by reducing the attack surface, it becomes easier to test for and guard against any such attack. By exposing a larger attack surface, an attacker might find an exploit that would be more difficult for the kernel developers to identify and fix.

Linus does not share that view. He would probably argue that a security hole is a security hole – if it exists, it should be closed; if it doesn't exist, then what's the problem? He has also said in the past that once an attacker is able to gain a certain level of privileges on a system, trying to mitigate what they can do with those privileges is sort of a waste of time, given that the attacker could use those privileges to work around whatever mitigations are put in place. In other words, once a hostile user gains root on a system, that's the ball game, and adding more patches won't change that.

It's a different approach to security. And it's one that leads some security-minded users to claim that Linux is less secure than, for example, some of the free software BSD systems.

My personal view is these two views don't represent such vastly different positions as their advocates may think. If a security hole were to be discovered on either a BSD system or a Linux system, it would immediately be closed by the developers, and patches released to users. It's never the case that either the BSD or Linux projects just blithely release new versions with known security exploits in them. In BSD and Linux development, security issues trump all other considerations. The developers of any of these projects would remove whole subsystems without a second thought, rather than include a known security exploit. ■■■

Hofburg Palace, Vienna

Red Hat's Jim Perrin contrasts the company's three sponsored Linux projects

Three in One

Swapnil sorts through the complex relationships of CentOS, Fedora, and RHEL with Red Hat's Jim Perrin. *By Swapnil Bhartiya*

The Fedora project began in 2003. At the time, Red Hat had just canceled the old Red Hat Linux distro and was getting started with the commercial product that would be known as Red Hat Enterprise Linux (RHEL). The company wanted to continue to work with the open source community on a free Linux edition, and they started Fedora as a community-driven upstream contributor to RHEL.

Because Linux was open source, however, Red Hat could not distribute RHEL without making the source code available to others in the Linux community. A new category of distros emerged around the practice of removing trademarked material from RHEL and then compiling the source code to create a new, independent distribution. The most famous and most popular of these RHEL spin-offs was CentOS, which existed as an independent project for several years. Then in 2014, Red Hat surprised many experts by announcing that it would hire the CentOS developers and take over sponsorship of CentOS, thus making CentOS an in-house Red Hat project that was very close to RHEL but available for no cost.

The result is a constellation of three Linux distributions – two community-based and one commercial, each with a slightly different role, but all fitting together somehow to achieve a greater purpose for Red Hat. Swap sat down with Jim Perrin, Community Platform Engineering Manager, CentOS/Fedora at Red Hat, to sort through the complicated relationships of Fedora, CentOS, and RHEL.

Jim Perrin has been involved with the CentOS project for over 14 years and has spent eight years as a member of the CentOS Governing Board. When Red Hat acquired CentOS back in 2014, Perrin joined Red Hat as an employee. Perrin currently works at Red Hat as Community Platform Engineering Manager of CentOS and Fedora. He seems to have a complicated role. His primary focus is on the CentOS and Fedora communities, but as an engineering manager, he is also responsible for Fedora and CentOS infrastructure. In the past, he has been involved with the 64-bit ARM port for CentOS.

CentOS vs. Fedora: Friend or Foe

You could think of CentOS as a project that's eating RHEL's lunch just the way Linux Mint is eating Ubuntu's dinner. But the reality is, despite it being a clone of RHEL, CentOS maintains a very friendly and healthy relationship with its cousin Fedora.

"We tried to work as closely with the Fedora community as we could," said Perrin. "One of the most successful Fedora projects is the Extra Packages for Enterprise Linux (EPEL) repository for CentOS. Some of that code comes from Fedora. Most of the people who sit on the EPEL council are CentOS representatives, but the project itself belongs in Fedora. So that's one of the many areas where the two projects are collaborating with each other."

CentOS and Fedora are also sharing their Continuous Integration (CI) infrastructure. At the moment the CI initiative is more about continuous testing and not continuous integration, although they are working towards the goal of CI. The infrastructure helps in testing every time there are any updates. "We have a suite of tests for different projects that we run to try and validate as much as we can," he said. The project started at CentOS some two years ago, and today even Fedora is using some of that infrastructure for their test cases.

I Am Your Father

The popular belief is that CentOS and Fedora are like cousins, but there is more to the story. They also share code with each other. Rationally, we draw a conclusion that RHEL is upstream for CentOS and Fedora is upstream for RHEL, so technically Fedora is also upstream for CentOS. But things are not as they appear. "Both Fedora and CentOS are upstream for RHEL," said Perrin.

CentOS has a system of Special Interest Groups (SIGs) – groups that work on new topics or features to add functionality to CentOS that it doesn't inherit directly from RHEL. The SIG system allows for the upstream development of products that target enterprise Linux. In some cases, it is better to develop new community-drive features for RHEL through the CentOS SIG system than through Fedora. If you want to develop something for the current version of RHEL, using Fedora means you are running Linux kernel 4.14 or 4.15, whereas RHEL 7.x (and CentOS) are running a 3.10 based kernel. In that case, "Using Fedora as a development platform to target RHEL isn't going to work," said Perrin.

The better course would be to do that upstream development on CentOS, because it has an equivalent kernel version. As a result, there is a dual – downstream and upstream – relationship between CentOS and RHEL. "Fedora is entirely upstream in all ways, as it should be, but on the CentOS side, we are upstream for some things and downstream for others," said Perrin.

Whereas RHEL development is fully controlled by the company, anyone can come and contribute to Fedora and CentOS. A good example is the Xen virtualization project that has not been part of RHEL since version 5. But Xen still has an active community, and they are still contributing as a member of the CentOS virtualization SIG. "They still provide us new code, and we still build it. They control their release cycle for what they're putting out, and they're valued members of the community. Anybody can come and contribute. The code is not controlled by Red Hat," said Perrin.

CentOS creates an opportunity for enterprise developers to target RHEL. One might draw the conclusion that enterprise Linux developers should target CentOS instead of Fedora, but that would be a mistake.

"If someone wants to contribute code that's targeted at enterprise users of RHEL, they should look at both options because what happens in Fedora is what you see in the next version of RHEL. Fedora is the upstream for all the community," reminded Perrin.

If somebody is trying to target a specific version of enterprise Linux and looks at CentOS, they must consider what happens to that work when the next version of RHEL or CentOS comes out. If you skip Fedora, that code won't be in RHEL or CentOS. So as tempting as it may sound, developers should look at the larger picture. "If they're only focusing on what exists today, they're ignoring what's coming tomorrow."

But it's complicated. Targeting CentOS is justifiable in a case when someone is looking at those layered projects that focus on enterprise Linux. A good example is RDO, OpenStack customized for CentOS. The life cycle for Fedora and RDO is reasonably short. Running that code on Fedora is not wise. It makes more sense to target the code for CentOS, which has a longer life cycle; it's supported for years and not months.

He warned again that developers must keep an eye on what's coming next instead of focusing on the 3.10 kernel that exists in CentOS 7/RHEL 7 today.

Changing Gears: Going Fast with Project Atomic

Earlier this year, Red Hat acquired CoreOS, the company that pioneered the concept of atomic updates. Red Hat responded to CoreOS by releasing RHEL Atomic Host; SUSE came out with SUSE CaaS Platform, and Canonical introduced Ubuntu Core. As more and more workload is becoming containerized, is it possible that atomic distros will replace traditional distros? Is it an evolution of operating systems, just the way virtualization and containerization are the evolution of application development and delivery?

"I don't think you can equate those two [traditional distros and atomic distros]," said Perrin. It's the same code that powers Atomic, RHEL, CentOS, and Fedora. The difference lies in the delivery and update mechanism.

When we are talking about a certain scale of deployment across 40 to 50 thousand servers, running `yum update` or `dnf update` across a network of that size may lead to some failure. "That's where the atomic method makes sense, because if something fails, it rolls back to the previous version. It's automatic. You don't end up with a corrupted Yum database and other problems that keep sys admins awake at night,"

That also doesn't mean that the bare metal approach with existing CentOS or RHEL customers has any less value. "It's just a different method of delivering the same code to deal with how the world of technology moves forward in a containerized world with different things," he said. "In some case, atomic makes sense, whereas in many others cases, traditional approaches are more appropriate. Atomic doesn't mean RHEL, CentOS, or Fedora are any less important. It's just a different method of delivering the same code."

Making CentOS Smarter with Machine Learning

The teams at CentOS and Fedora are planning to play with machine learning on the new stack of hardware they have installed for the continuous integration infrastructure. "We don't have any hard and fast plans for how we anticipate doing that yet or what the workflow is going to be, but it is absolutely something that we are considering," Perrin said.

Perrin said that he would very much want to have capabilities that would at least conduct tests against what users are running. They need to know if the tooling works: if an update reintroduced regression or if an update broke the GPU for acceleration. It's going to have a direct impact on CentOS/Fedora users who are using machine learning.

"We have two distinct use cases for AI – one targeting the users of our software and the second around how to help us provide better code," said Perrin.

"If we see clusters of bugs around particular software, there may be the potential for us to train AI to tell us about bugs or outdated packages. We could use a bot or we could use a piece of machine learning code to automatically fix a particular type of bug, increment that builds, re-spin the code, test it again, see if it passes, and then spit the update out where a person doesn't have to touch it. Ordinarily, it is a six-step process for a person to bump the spec file, build the software, test the software, and put the software out. If we could automate that process via some form of machine learning, fantastic." ■■■

ioBroker integrates different smart home protocols into a single easy interface

Many in One

The smart home market includes a number of different vendor environments. You'll need a capable broker to fit them under one roof.

By Stefan Heinle

The story of the Tower of Babel teaches us that stagnation and confusion can ensue when the community does not have a common language. In today's world, your own smart home project can fail if the components in your Internet of Things (IoT) community do not understand each other. So far, the IoT industry has not settled on a single, uniform language for home automation. Instead, various IoT vendors have their own competing communication systems. If you purchase all your IoT gadgets from the same vendor, the whole network will chatter harmoniously, but if you don't want to be locked into a single vendor environment, you might be wondering how you could ever manage these babbling devices from a single interface.

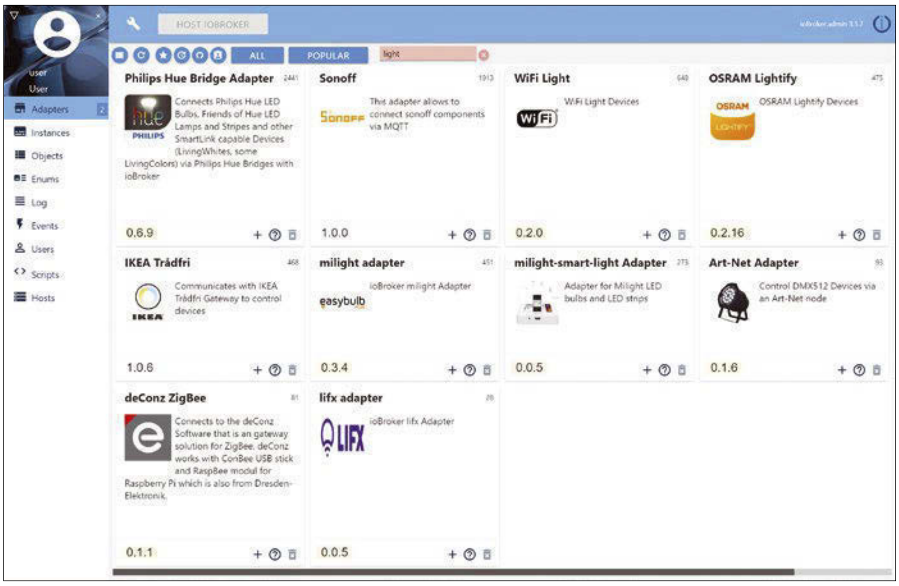
One interesting solution for the IoT Tower of Babel dilemma is ioBroker [1], an extremely capable broker and interpreter for home automation projects. The ioBroker project was launched in 2014, and it continues on with strong support from a dedicated developer community.

Author

Stefan Heinle is an electrical engineer and an enthusiastic practitioner of home automation. He is a certified KNX partner and author of the book *Home Automation with KNX, DALI, 1-Wire and Co.*, published by Rheinwerk. He offers supplementary information for the book, as well as professional smart home planning services, on the website *www.heimautomation-buch.de*.

What is ioBroker?

ioBroker offers a single user interface for managing IoT devices from several different vendor environments (Figure 1). The ioBroker system comes with a collection of software components called *adapters* that interface with the various hardware environments. The number of ioBroker adapters has increased from 100 to more than 180, and the count is still increasing. The adapters are installed individually, so you will never have more than you need.



Philips Hue Bridge Adapter 2441
Connects Philips Hue LED Bulbs, Friends of Hue LED Lamps and Stripes and other SmartLink capable Devices (LivingWhites, some LivingColors) via Philips Hue Bridges with ioBroker
0.6.9

Sonoff 1913
This adapter allows to connect sonoff components via MQTT
1.0.0

WiFi Light 640
WiFi Light Devices
0.2.0

OSRAM Lightify 475
OSRAM Lightify Devices
0.2.16

IKEA Trådfri 468
Communicates with IKEA Trådfri Gateway to control devices
1.0.6

milight adapter 451
ioBroker milight Adapter
0.3.4

milight-smart-light Adapter 273
Adapter for Milight LED bulbs and LED strips
0.0.5

Art-Net Adapter 93
Control DMX512 Devices via an Art-Net node
0.1.6

deConz ZigBee 81
Connects to the deConz Software that is an gateway solution for ZigBee. deConz works with ConBee USB stick and RaspBee modul for Raspberry Pi which is also from Dresden-Elektronik.
0.1.1

lifx adapter 26
ioBroker lifx Adapter
0.0.5

Figure 1: The ioBroker admin interface.

The view in Figure 1 is filtered for lighting applications. ioBroker currently supports 10 different lighting control systems alone, including the popular Osram Lightify devices, Philips Hue products, and Ikea Trådfri components.

Getting Started

You can set up a smart home universal gateway for just under 60 Euros. A Raspberry Pi (preferably model 3), a suitable power supply, a microSD card (with 16GB), a case, and a patch cable are all you need. If the Rasp Pi has to control several USB interfaces, an external USB hub is also recommended to avoid stability problems due to the weak USB power supply.

SD card images [2] are available for a variety of small board computers (SBC).

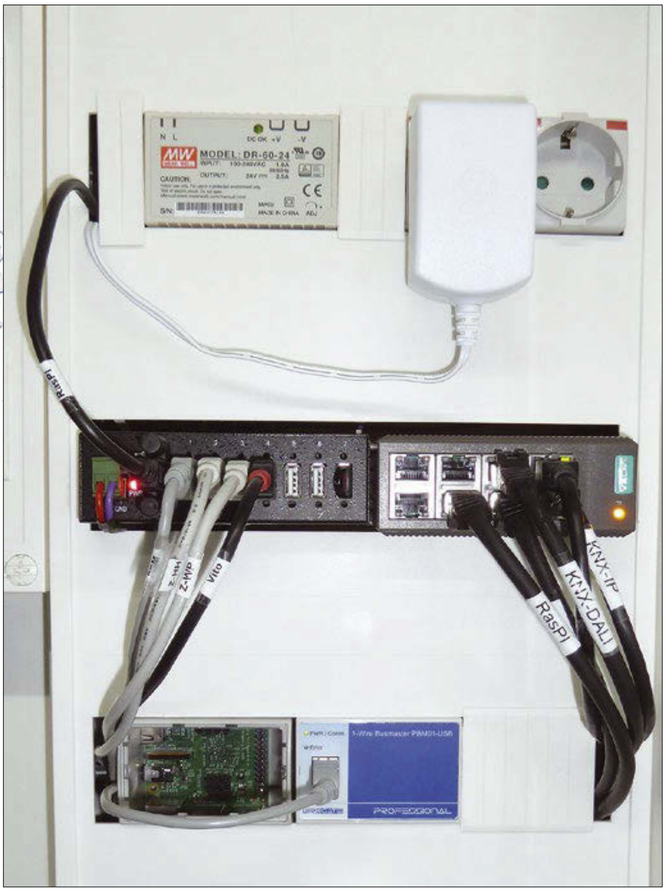


Figure 2: Rasp Pi as a top-hat rail installation.

In addition to various Rasp Pi builds, you will find images for Banana Pi, CubieTruck, Pine64, CuBox, Tinker Board, Orange Pi, Rock64, and Windows. You might want to consider creating an ioBroker gateway with a special DIN rail housing for the Rasp Pi (Figure 2, bottom left), along with a USB hub (middle left), and LAN switch (middle right).

Auto Discovery

For your first steps, you might want to try out ioBroker on a conventional computer before moving to an SBC. The ioBroker application runs on Linux, Windows, and OS X Intel-equivalent systems, in addition to the supported ARM-based platforms.

You can configure ioBroker in the browser. After installing and entering some geolocation data, the setup wizard automatically searches for available devices on the local network. On the test network, ioBroker found 21 devices and 11 services, creating adapter instances directly for the detected FRITZ!Boxes and an InfluxDB database server.

Cross-System Switching

Figure 3 shows an example that demonstrates how different smart home systems can exchange data using ioBroker. In this example, the ioBroker system supports three systems:

- Harmony Hub – an add-on hub for a Logitech universal remote control
- KNX – a standards-based building automation system
- Philips Hue – a lighting device, or more precisely, the Hue bridge, which manages the lighting device.

Although all three of these systems speak different languages, ioBroker lets you manage all three collections through a single interface.

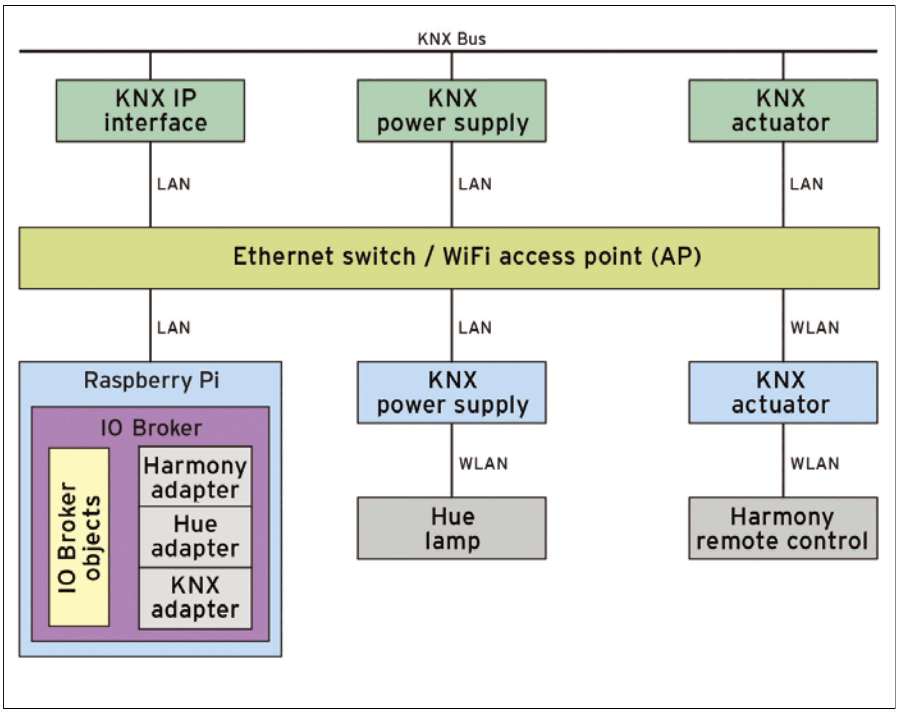


Figure 3: Sample setup with ioBroker as the central agent.

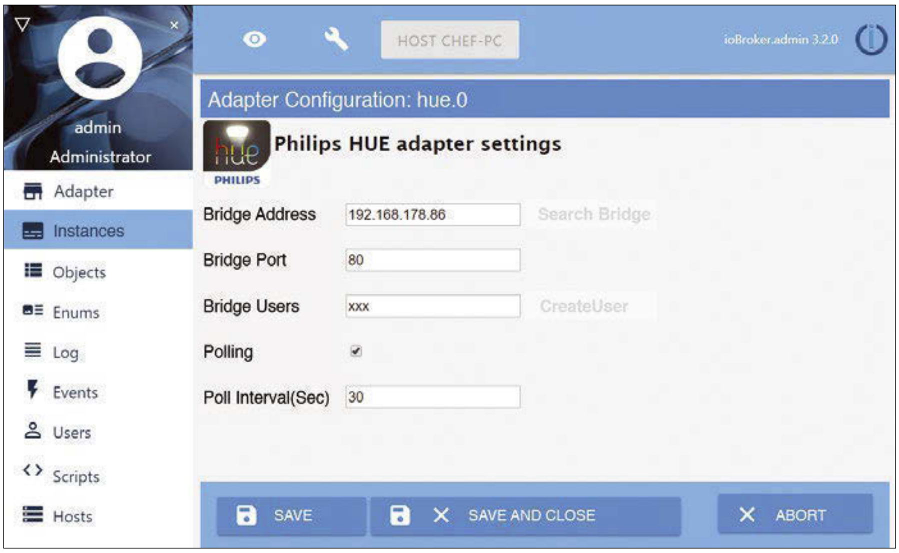
Before you can integrate these three environments through ioBroker, you must configure each of the environments individually. For this example, I did the following:

- Created a `Television` activity via the Harmony Hub.
- Created a KNX `Television` scene in the KNX system.
- Taught all lamps via the Hue bridge.

The automation part now consists of the user selecting the KNX `Television` scene by pressing the "Television" button on a Harmony remote control. The result of this scene is that the television switched on and the television lighting turned on via the Hue bridge.

The scene could also trigger other actions through the KNX bus, such as closing the blinds or dimming other light sources (outside the Hue system).

However, these transactions defined within the individual systems still have not achieved the integration possible through ioBroker. The ioBroker configuration will focus on building bridges between the Harmony Hub and KNX on the one hand and between KNX and the Hue system on the other.



Adapter Configuration: hue.0

Philips HUE adapter settings

Bridge Address 192.168.178.86

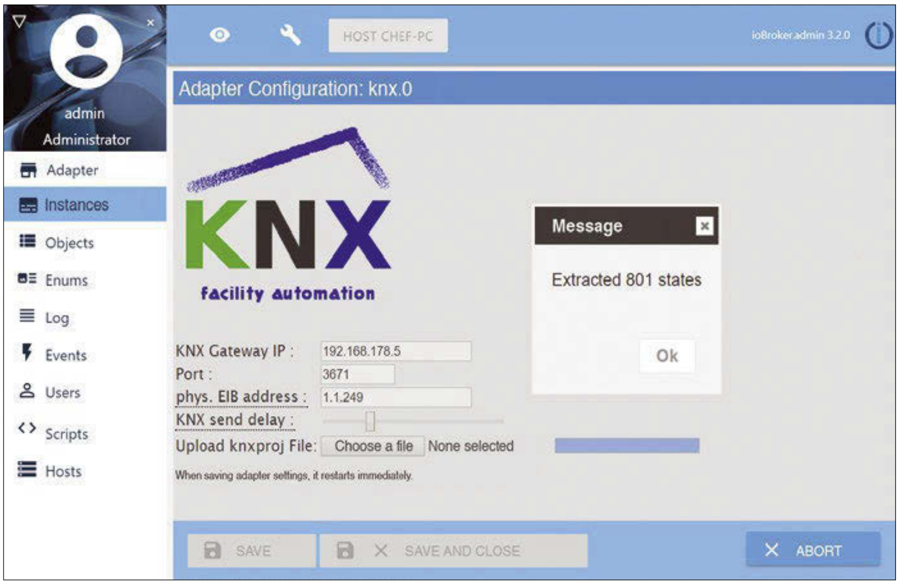
Bridge Port 80

Bridge Users xxx

Polling ☑

Poll Interval(Sec) 30

Figure 4: Configuring the Hue Adapter.



Adapter Configuration: knx.0

KNX Gateway IP : 192.168.178.5
Port : 3671
phys. EIB address : 1.1.249
KNX send delay :
Upload knxproj File: Choose a file None selected

When saving adapter settings, it restarts immediately.

Message

Extracted 801 states

Ok

Figure 5: Setting up the KNX adapter.

Adapter Integration

In order for ioBroker to communicate with these subsystems, a total of four adapters are required:

- JavaScript: `javascript` (already exists after auto-discovery)
- KNX system: `knx`
- Logitech Harmony Hub: `harmony`
- Philips Hue system/bridge: `hue`

When installing the Philips Hue adapter, only the IP address of the Hue bridge is important (*Bridge Address* field); you either create a new user or enter an existing user in the *Bridge Users* field (Figure 4).

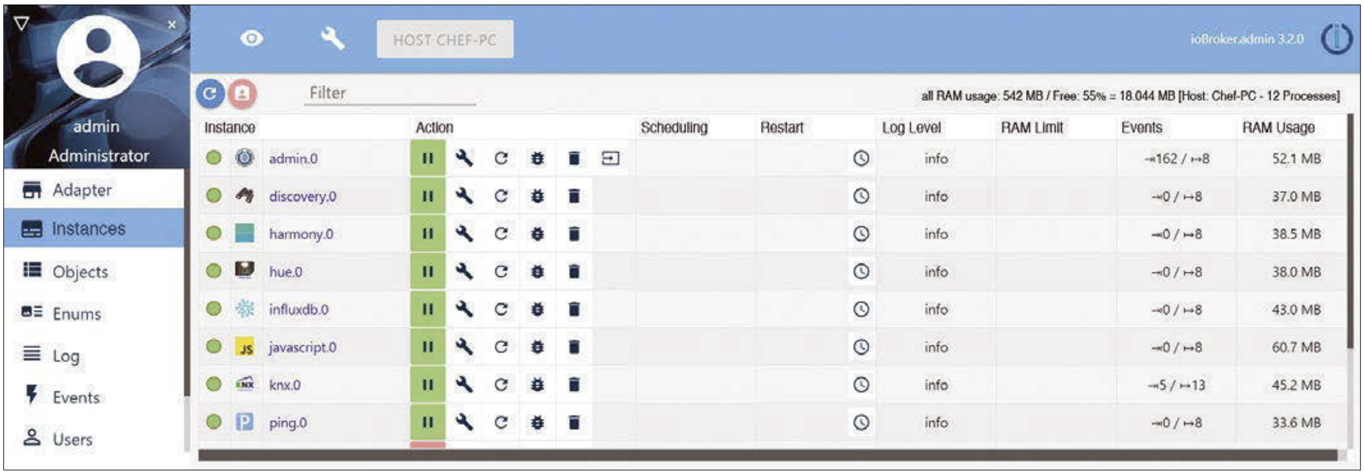
The *Instances* option in the sidebar menu should list the newly added adapter named `hue.0` and the status LED should turn green to indicate that the connection to the host (the Hue bridge) has been established. The Harmony adapter instance is created just as easily. The dialog box only requires two entries: the hub user and the hub password.

Slightly more configuration overhead is required for the KNX adapter. The wizard expects the IP address of the KNX IP interface or the KNX IP router, the corresponding port (3671), and the physical KNX address of the interface (say, 1.1.249). The wizard also offers to include an existing ETS project file (format `.knxproj`). You can create the project file with the ETS software [3] via the export function. Importing the ETS file can take a few minutes on a small system such as a Rasp Pi.

The information in the KNX adapter configuration view (Figure 5) represents the KNX group settings contained in the ETS project file. The ioBroker objects that are created automatically after installing the adapters facilitate communication among the devices. Writing to one of these objects triggers an action on the connected system (such as retrieving a KNX scene). If the connected system generates information, the information is also reflected in an object within ioBroker.

The JavaScript adapter is usually installed during the auto-discovery process. For more debugging when programming in JavaScript, it is a good idea to activate the *Expert* view in the ioBroker *Instances* tab. Once the *Expert* view is activated, you can raise the log level from *Info* to *Debug*, which provides better console output during JavaScript development.

If all adapters are configured correctly, the *Instances* view displays the green status LED for each adapter (Figure 6). By the way, you can also create multiple instances for an adapter. For example, a second existing Hue bridge could be included using the same procedure, but it would then have an instance name of `hue.1` instead of `hue.0`.



all RAM usage: 542 MB / Free: 55% = 18.044 MB [Host: Chef-PC - 12 Processes]

Instance	Action	Scheduling	Restart	Log Level	RAM Limit	Events	RAM Usage
admin.0				info		→162 / ↦8	52.1 MB
discovery.0				info		→0 / ↦8	37.0 MB
harmony.0				info		→0 / ↦8	38.5 MB
hue.0				info		→0 / ↦8	38.0 MB
influxdb.0				info		→0 / ↦8	43.0 MB
javascript.0				info		→0 / ↦8	60.7 MB
knx.0				info		→5 / ↦13	45.2 MB
ping.0				info		→0 / ↦8	33.6 MB

Figure 6: All the required adapters are successfully integrated.

Communication via Objects

ioBroker automatically creates a large number of objects during the adapter installation. In the *Object* tab (Figure 7), you can open, modify, and inspect all objects in a hierarchical view. This central view of object data is one of the great strengths of ioBroker. In principle, the software handles the communication between the individual smart home systems. The view is refreshed automatically when objects change. It is also very easy to trigger an action in a connected system by manually overwriting the object value in the `Value` column.

Communication with an external device is easy. The following example is intended to switch on a Hue light. In the object view, expand the hierarchical list until you find the objects you are looking for. The hierarchy is structured differently depending on the adapter. For example, the path for a Hue light would look like this:

```
hue.0 | Philips_hue | Liv | pendant_lamp_front
```

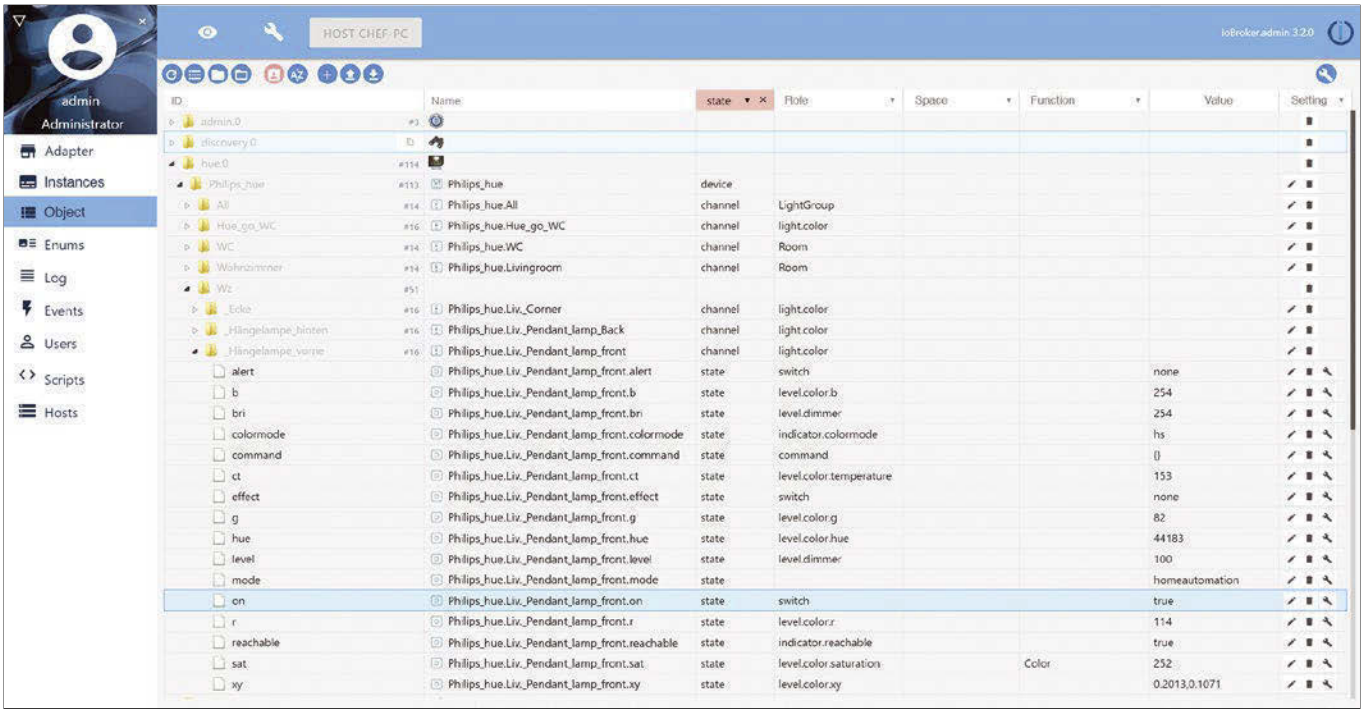
The designations `Liv` (for living room) and `pendant_lamp_front` come from the Hue bridge configuration and reflect the user settings at the time of Hue system commissioning.

When the last hierarchy level is expanded, ioBroker displays the usable objects. At this point, it is sufficient to change the value for the object in the `Value` column from `false` to `true` to switch on the `pendant_lamp_front` (Figure 7). In response to this, the `level` object is automatically updated to `100` and `bri` (brightness) to `254`.

Since the example is a multicolored LED luminaire, it makes sense to bring some color into play. If the `effect` object is described by the `colorloop` value, the Hue light permanently changes color. The value `none` stops the show. More information about the Hue color model is available at the Phillips Hue developers website [4].

Automation with JavaScript

You can read and write ioBroker objects manually, but you can also automate the process. ioBroker has three strategies



ID		Name	state	Role	Space	Function	Value
admin.0	#3						
discovery.0							
hue.0	#114						
Philips_hue	#113	Philips_hue	device				
All	#14	Philips_hue.All	channel	LightGroup			
Hue_go_WC	#16	Philips_hue.Hue_go_WC	channel	light.color			
WC	#14	Philips_hue.WC	channel	Room			
Wohnzimmer	#14	Philips_hue.Livingroom	channel	Room			
Wz	#51						
_Ecke	#16	Philips_hue.Liv._Corner	channel	light.color			
_Hängelampe_hinten	#16	Philips_hue.Liv._Pendant_lamp_Back	channel	light.color			
_Hängelampe_vorne	#16	Philips_hue.Liv._Pendant_lamp_front	channel	light.color			
alert		Philips_hue.Liv._Pendant_lamp_front.alert	state	switch			none
b		Philips_hue.Liv._Pendant_lamp_front.b	state	level.color.b			254
bri		Philips_hue.Liv._Pendant_lamp_front.bri	state	level.dimmer			254
colormode		Philips_hue.Liv._Pendant_lamp_front.colormode	state	indicator.colormode			hs
command		Philips_hue.Liv._Pendant_lamp_front.command	state	command			{}
ct		Philips_hue.Liv._Pendant_lamp_front.ct	state	level.color.temperature			153
effect		Philips_hue.Liv._Pendant_lamp_front.effect	state	switch			none
g		Philips_hue.Liv._Pendant_lamp_front.g	state	level.color.g			82
hue		Philips_hue.Liv._Pendant_lamp_front.hue	state	level.color.hue			44183
level		Philips_hue.Liv._Pendant_lamp_front.level	state	level.dimmer			100
mode		Philips_hue.Liv._Pendant_lamp_front.mode	state				homeautomation
on		Philips_hue.Liv._Pendant_lamp_front.on	state	switch			true
r		Philips_hue.Liv._Pendant_lamp_front.r	state	level.color.r			114
reachable		Philips_hue.Liv._Pendant_lamp_front.reachable	state	indicator.reachable			true
sat		Philips_hue.Liv._Pendant_lamp_front.sat	state	level.color.saturation		Color	252
xy		Philips_hue.Liv._Pendant_lamp_front.xy	state	level.color.xy			0.2013,0.1071

Figure 7: Device control via the *Object* view.

Listing 1: KNX Television Scene

```
01 on('harmony.0.Harmony-Hub.activities.currentActivity', function (obj) {
02  var newActivity = "";
03  var oldActivity = "";
04
05  console.log ("Harmony Hub Activity Event detected:");
06  newActivity = obj.state.val;
07  oldActivity = obj.oldState.val;
08
09 console.log ("new activity : "+ newActivity) ;
10 console.log ("old activity was: " + oldActivity);
11
12  switch (newActivity) {
13  case "Television":
14 console.log ("activity Television detected.");
15 // Triggering the Television KNX scene
16 setState('knx.0.central.scenes.scenes.television', 0, false /*no ack*/);
17  break;
18  }
19 });
20
21 on('knx.0.central.scenes.scenes_Television', function (obj) {
22  var scenesNo = toInt(obj.state.val) + 1;
23 console.log ("Retrieving KNX scene, no.:" + scenesNo);
24
25 switch (scenesNo) {
26  case 1:
27 console.log ("Switch on Hue lamp.");
28  setState('hue.0.Philips_hue.Wz._pendant_lamp_front.on', 1, false /*no ack*/);
29  break;
30  }
31 });
```

for automation: Scenes (please do not confuse these with KNX scenes), Flows in Node-RED [5], and JavaScript. I'll focus on the most powerful of the three alternatives: writing your own small JavaScript programs.

The *Scripts* tab takes you to the built-in JavaScript development environment in ioBroker. Pressing the *New Script* icon creates an empty text file named `Script1` in the `common directory`. You can rename this file as desired. Before saving the empty file, check the *Help output* box. In the built-in editor, you can start automating. To familiarize yourself with the JavaScript's ioBroker-specific extension, take a look at the documentation on GitHub [6].

The implementation of the previous automation example (TV button on remote control triggers the KNX `Television` scene, which in turn switches an additional Hue light) is shown in Listing 1. Line 1 contains the most important component: The `on()` statement defines a callback function that is used whenever the state of the `harmony.0.Harmony-Hub.activities.currentActivity` object changes. The object's complete path specification results from the hierarchical arrangement in the object tree. The specified string can be obtained in the *Object* view using the clipboard symbol above an object name.

The function then determines the value of the `currentActivity` object and displays it on the console together with its previous value for demonstration purposes. The `switch()` statement in Line 12 considers different values of the activity, and Line 16 finally triggers the KNX `Television` scene by writing to the corresponding ioBroker object using the `setState()` function. The value `0` represents the KNX scene with the number 1 (the value is the scene number minus one). For the adapter to actually set the value, you first need to set a third parameter to `false`. This sounds a bit illogical at first, but you'll find a detailed explanation on the ioBroker Wiki [7].

In the group monitor of the ETS software, you can see how the TV scene was triggered on the KNX side. The scene object with the KNX group address 15/4/8 receives the value `0`, which means that the program retrieves scene 1.

The second `on()` statement in Line 21 also switches on the Hue `pendant_lamp_front` when someone activates the KNX `Television` scene. Of course, you could have packed the `setState()` function call from Line 28 directly under the first `setState()` call from the Harmony callback function. However, a separate callback function (as in Listing 1) has the advantage that the Hue lamp is switched on even if the KNX scene was not triggered via ioBroker.

Conclusions

With even a little knowledge of JavaScript, almost any smart home subsystem can be integrated into a larger whole via the ioBroker platform. This integration is possible because the ioBroker system provides a large number of available adapters and an ingenious object database. ioBroker is a top-ranking solution for cost-effective smart home networking within your own four walls. ■■■

Info

[1] Project website: *http://www.iobroker.net*

[2] ioBroker SD images: *http://www.iobroker.net/docu/?page_id=2563&lang=de*

[3] KNX ETS software: *https://www.knx.org/knx-de/software/uebersicht/index.php*

[4] Philips Hue developers website: *https://developers.meethue.com/documentation/core-concepts*

[5] Node-RED homepage: *http://www.nodered.org*

[6] ioBroker JavaScript: *https://github.com/IO-Broker/IO-Broker.javascript/blob/master/doc/en/javascript.md*

[7] ioBroker status: *https://github.com/IO-Broker/IO-Broker/wiki/Adapter-Development-Documentation#commands-and-statuses*

Eclipse development tools for the Internet of Things

Gadget Talk

Eclipse IoT is a collection of 26 open source projects with the common goal of building an open development environment for the Internet of Things. *By Tim Schürmann*

The Eclipse project is a vast and diverse effort to produce a universal open source collection of software development tools. Separate projects within the Eclipse umbrella target specific standards and use cases, creating helpful libraries and software components that (at least theoretically) plug seamlessly into the universal Eclipse development environment.

The recent emergence of the Internet of Things (IoT) has not gone unnoticed within the Eclipse community. IoT programs and protocols connect household appliances, light bulbs, temperature sensors, and motorized devices, requiring a new crop of programming standards and technologies. The Eclipse IoT working group [1] is a collaboration of individuals and organizations dedicated to "development, promotion, and adoption of IoT technology." The group encompasses a wide range of projects that extend the Eclipse vision to a new generation of open source IoT development tools.

Table 1 shows a summary of Eclipse IoT projects. Each project has a different emphasis. For example, the Paho library (see Table 1) handles data exchange via the MQ Telemetry Transport (MQTT), a common messaging protocol for open IoT. All Eclipse Foundation projects are under open source licenses and are therefore available free of charge.

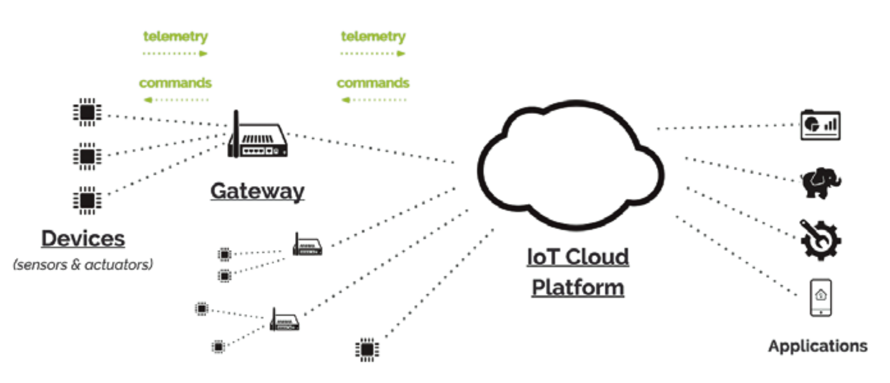


Figure 1: Eclipse organizes the IoT space into devices, gateways, and cloud platforms.

Stacked High

To keep track of the project flood, the Eclipse Foundation divides a typical IoT application into three central components (Figure 1). Sensors and actuators always form the basis. This category includes temperature sensors, as well as small motors and intelligent light sources. The Eclipse Foundation groups these devices under the term constrained devices.

The raw data is then collected by a gateway. For example, a sensor could pick up the temperature and humidity values from the greenhouse, and a gateway would then push the data into the IoT cloud.

The services running in the cloud process or analyze the data. To stick with the example of the greenhouse: A cloud application could enter the supplied temperatures in a chart and determine the probable water requirements of the plants.

A client application can then access the cloud and obtain the cloud data. This application could be a smartphone app that alerts its users if the temperature in the greenhouse rises dangerously. The whole procedure also works in the opposite direction: The cloud could send instructions to the gateway, causing it to open the greenhouse windows. The gateway then sends the corresponding control data to the appropriate actuators. Gateways thus also coordinate the cooperation of individual IoT devices.

Table 1: Eclipse's IoT Projects

Name	Task	URL
Eclipse 4diac	Industrial process control according to IEC 61499	*http://www.eclipse.org/4diac/*
Eclipse Agail	Modular software and hardware gateway framework	*http://agile-iot.eu*
Eclipse Californium	Implementation of the constrained application protocol (CoAP) in Java	*https://www.eclipse.org/californium/*
Eclipse Concierge	Implementation of the OSGi Core Specification R5	*https://www.eclipse.org/concierge/*
Eclipse Cyclone DDS	Implementation of the OMG Data Distribution Service (DDS)	*http://projects.eclipse.org/projects/iot.cyclonedds*
Eclipse Ditto	Management of digital twins	*https://www.eclipse.org/ditto/*
Eclipse Edje	High-level Java API that provides access to the hardware of a microcontroller and thus functions as a HAL	*http://projects.eclipse.org/projects/iot.edje*
Eclipse hawkBit	Management tools for rolling out software updates	*http://projects.eclipse.org/projects/iot.hawkbit*
Eclipse Hono	Provides a unified interface for communication with numerous IoT devices	*https://www.eclipse.org/hono/*
Eclipse IO-Fog	Universal run-time environment for microservices	*http://projects.eclipse.org/projects/iot.iofog*
Eclipse Kapua	Modular IoT cloud platform, forms the basis for cloud services, which then evaluate the delivered data	*https://www.eclipse.org/kapua/*
Eclipse Kura	Middleware and application container according to OSGi for gateway services, transforms a Raspberry Pi into a gateway	*https://www.eclipse.org/kura/*
Eclipse Leshan	OMA lightweight M2M Server and client	*https://eclipse.org/leshan/*
Eclipse Milo	Implementation of the OPC Unified Architecture	*http://projects.eclipse.org/projects/iot.milo*
Eclipse Mosquitto	MQTT broker	*https://projects.eclipse.org/projects/technology.mosquitto*
Eclipse NeoSCADA	Framework for generating industrial control systems (ICS)	*http://www.eclipse.org/eclipsescada/index.html*
Eclipse OM2M	IoT cloud platform for the telecommunications industry based on the One M2M specification	*http://www.eclipse.org/om2m/*
Eclipse Paho	Implementation of the MQTT protocol	*http://www.eclipse.org/paho/*
Eclipse Paho Incubator	Incubator project for Paho, where Paho developers can try out new ideas	*http://projects.eclipse.org/projects/iot.paho.incubator*
Eclipse Ponte	Ponte mediates and converts between different protocols and data formats – for example, data can be received via MQTT and then forwarded via CoAP	*http://www.eclipse.org/ponte/*
Eclipse SmartHome	Middleware and application container as per OSGi, focus on home automation	*https://www.eclipse.org/smarthome/*
Eclipse tinydtls	Implementation of the DTLS protocol	*http://projects.eclipse.org/projects/iot.tinydtls*
Eclipse Unide	Implementation of a lean production performance management protocol	*http://www.eclipse.org/unide/*
Eclipse Wakaama	Implementation of the OMA LWM2M standard	*http://projects.eclipse.org/projects/iot.wakaama*
Eclipse Vorto	Generation and Exchange of device information models	*http://www.eclipse.org/vorto/*
Eclipse Whiskers	Implementation of the OGC SensorThings API in the form of a JavaScript client and a lean server	*http://projects.eclipse.org/projects/iot.whiskers*

The constrained devices, gateways, and IoT cloud each consist of components that support or complement each other. For example, a constrained device might run a small real-time operating system with a mini-application that controls the actual hardware.

The Eclipse Foundation's terminology refers to a *stack* of software components serving each of the categories: constrained devices, gateways, and the IoT cloud.

Of Gatherers …

The sensors and actuators interact with the real world and usually take on a very special but clearly defined task. Examples include battery-powered temperature sensors, light switches, and light bulbs that require very little power. In addition, the intelligent devices often contain microcontrollers with limited computing power.

If the constrained device requires an embedded or real-time operating system, the Eclipse Foundation recommends established open source systems such as Contiki-NG [2], RIOT [3], FreeRTOS [4], Zephyr [5], or Apache Mynewt [6]. Access to the actual hardware is simplified by a hardware abstraction layer (HAL). Such a HAL is provided by the Eclipse Edje project. Edje offers an API that allows applications to directly access the hardware features of microcontrollers – such as a Raspberry Pi's GPIO contacts.

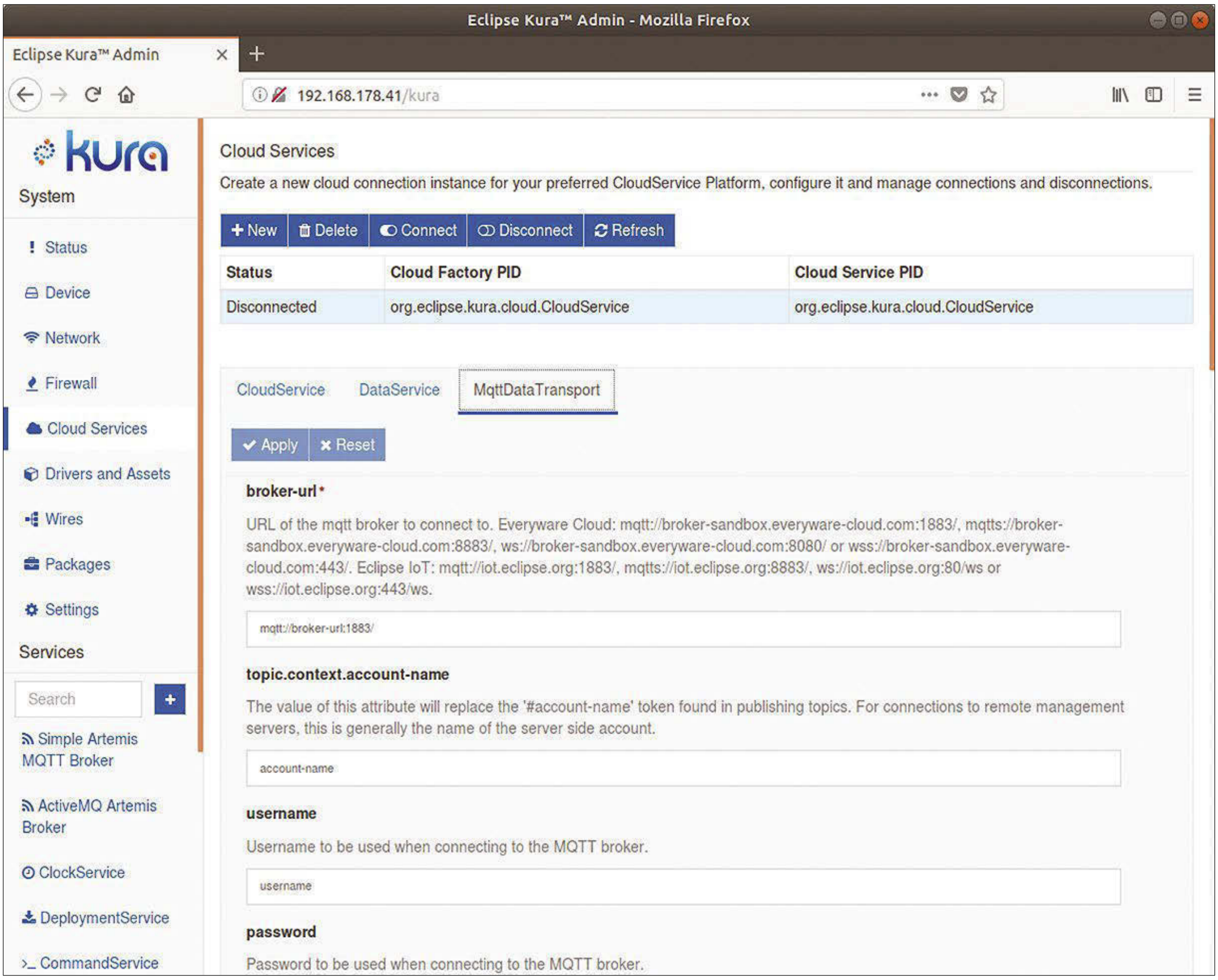
Another component forwards the read sensor data over the network. This transmission usually occurs wirelessly using Bluetooth, Z-Wave, MQTT, CoAP, or a similar protocol. Developers receive support through the Eclipse Wakaama and Paho projects. Paho provides a library with which developers can conveniently send data over the network using the MQTT protocol. Similarly, Eclipse Wakaama implements the OMA LWM2M standard.

A fourth component on the constrained device is responsible for remote management. This component lets you upgrade the firmware or check the battery status. (Eclipse Wakaama plays a role with this component as well.)

… and Hunters

Gateways occur in two variants: One variant is a hardware device specially designed for this task, such as a suitably equipped Raspberry Pi. The other variant is a gateway integrated into a larger system, such as a weather station, a large production plant, or an intelligent car.

In practice, several gateways are often used for separate components. For example, one gateway could monitor the greenhouse,



Cloud Services

Create a new cloud connection instance for your preferred CloudService Platform, configure it and manage connections and disconnections.

Status	Cloud Factory PID	Cloud Service PID
Disconnected	org.eclipse.kura.cloud.CloudService	org.eclipse.kura.cloud.CloudService

broker-url*

URL of the mqtt broker to connect to. Everyware Cloud: mqtt://broker-sandbox.everyware-cloud.com:1883/, mqtts://broker-sandbox.everyware-cloud.com:8883/, ws://broker-sandbox.everyware-cloud.com:8080/ or wss://broker-sandbox.everyware-cloud.com:443/. Eclipse IoT: mqtt://iot.eclipse.org:1883/, mqtts://iot.eclipse.org:8883/, ws://iot.eclipse.org:80/ws or wss://iot.eclipse.org:443/ws.

mqtt://broker-url:1883/

topic.context.account-name

The value of this attribute will replace the '#account-name' token found in publishing topics. For connections to remote management servers, this is generally the name of the server side account.

account-name

username

Username to be used when connecting to the MQTT broker.

username

password

Password to be used when connecting to the MQTT broker.

Figure 2: Eclipse Kura is easy to manage and monitor using the included web application.

and a second gateway could be dedicated to lighting and heating control.

Smart Doorkeeper

Gateways are smarter and more powerful than constrained devices. The substructure is usually a well-known operating system such as Linux or Windows. The gateway has its own component for communication with IoT devices and the cloud. It must be able to handle numerous protocols, such as Bluetooth and Ethernet. Another component allows the gateway to be (remotely) serviced and switched off in a controlled manner.

To compensate for network failures or faults, the gateways need to (temporarily) store the data collected by the sensors. This approach also enables direct analysis of sensor data on the gateway.

The Eclipse Foundation supports three projects, each of which contains all of the previously described components. Kura (Figure 2), which specializes in single-board computers such as the Raspberry Pi, is particularly well known. The modular structure of Kura allows Java developers to quickly adapt it to their own needs. For instance, they can program a small service that reads out the temperature sensors and forwards the data to Kura.

At its core, Kura provides an application container for Java developers that follows the OSGi standard. Kura also offers numerous APIs that make communication with the hardware and the network easier for the developer. Kura accesses RS-485 or GPIO interfaces, communicates via the CAN bus, and establishes a connection via WLAN or Ethernet. Integrated MQTT support enables applications running on the gateway to communicate transparently with the cloud.

Based on the MQTT protocol, Eclipse Kura also enables remote management. Among other things, the software running on the gateway can be controlled remotely.

Eclipse SmartHome, also intended for gateways, runs on Linux, Windows, and OS X. Like Kura, the SmartHome project provides an application container according to the OSGi specification. However, as the project name suggests, SmartHome focuses on intelligent home scenarios.

The middleware supports many home automation devices, including intelligent Hue lamps by Philips or Sonos devices. The main aim is to set up and operate an “Intranet of Things.” SmartHome offers an internal event bus to which external systems can also dock via SSE or MQTT. Predefined rules can be used to automate processes. SmartHome is configured via APIs, and the system includes an infrastructure for transferring firmware updates to the connected devices.

The third gateway project in the Eclipse universe is aimed at the corporate environment: Eclipse 4diac provides an open source infrastructure for distributed industrial process measurement and control systems. 4diac follows the well-known IEC 61499 standard.

Very Cloudy

The data collected by the gateway is uploaded to a cloud that uses OpenShift, AWS, or another data center solution. In the cloud are services and applications that process the data delivered by the gateways and forward the results to other applications as needed.

The basic building block for an IoT cloud is Eclipse Kapua: This modular platform offers all services necessary for managing the gateways and Smart Edge devices.

Eclipse OM2M, which follows the oneM2M standard, is the project of choice for the telecommunications industry. The OM2M project offers a horizontal Common Service Entity (CSE) that can act as an M2M server, gateway, or device.

The services running in the cloud need to be able to process the information delivered via a variety of protocols and in different data formats, and they need to normalize the information for other applications as needed. This is where Eclipse Hono jumps into the breach: The Hono project provides an API that can address devices in many protocols. In addition, Eclipse Mosquitto supplies a lightweight MQTT server, which is particularly suitable for embedded devices and systems with low energy consumption.

Please Register!

The Eclipse Foundation envisions a device registry, to which all currently connected sensors, actuators, and gateways would register. Eclipse Kapua comes with a registry as standard. This directory also helps to roll out software updates. Updates occur via the software tools from the Eclipse hawkBit project. To supplement this, Eclipse Leshan provides an implementation of the OMA LWM2M device management protocol.

Eclipse Ditto helps you manage digital twins of IoT devices. These digital twins are virtual images of real devices built in software. This virtual version is particularly useful in industrial applications, where the virtual version and its physical counterpart often must exchange data.

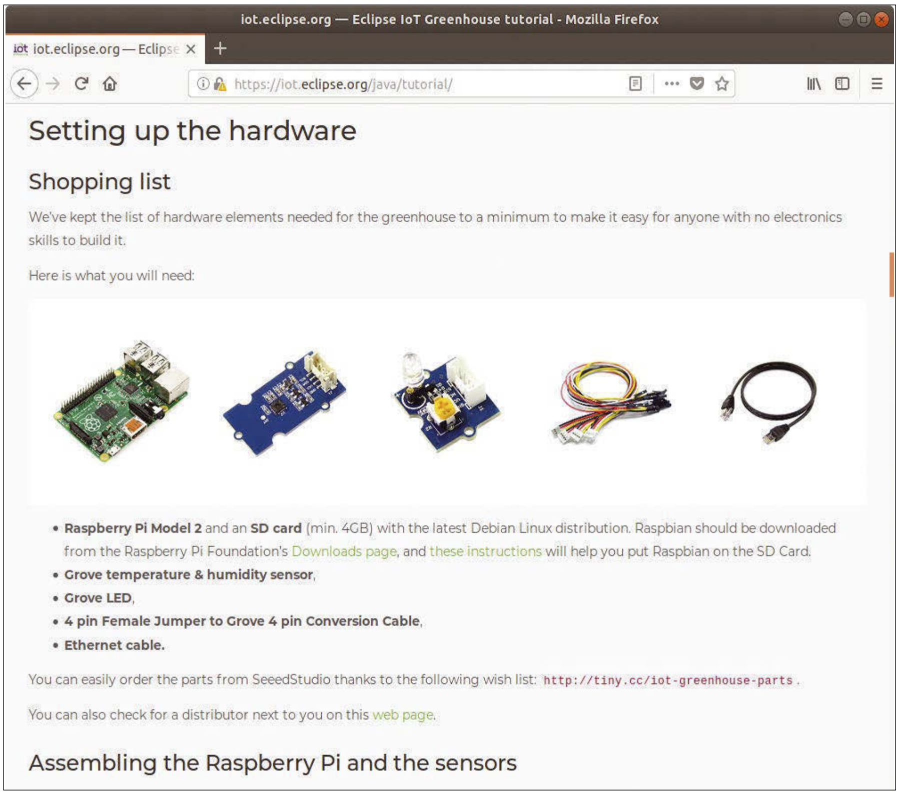
The delivered data must end up in a correspondingly flexible and large data repository. The information parked there is then analyzed and consolidated by the cloud application. The Foundation draws on established external Apache projects for data analysis, such as Hadoop, Spark, and Storm. For data visualization, the Eclipse Foundation recommends its own Eclipse BIRT [7] project, which is not part of Eclipse IoT.

Other Projects

Sensors, gateways, and the cloud need to communicate securely at all times. For example, only authorized gateways can tap a sensor, and attackers are not allowed to eavesdrop on the gateway’s communication with the cloud. Eclipse tinydtls helps: This implementation of the DTLS protocol gives developers an easy option for an encrypted data exchange.

Smooth collaboration between the components is only guaranteed if they interpret and analyze the data correctly. This requires valid ontologies and metadata. Two Eclipse projects follow on from this: Eclipse Whiskers implements the OGC SensorThings API, which allows devices and their data to be coupled. Eclipse Unide offers a protocol and a suitable implementation for Production Performance Management (PPM) in industrial production.

Finally, Eclipse Vorto provides several tools and repositories that developers can use to create and manage device information models. These models describe the features and capabilities of IoT devices. Eclipse Vorto can then, among other things, semi-automatically generate code from the models, which serves to integrate the devices into different platforms.



iot.eclipse.org — Eclipse IoT Greenhouse tutorial - Mozilla Firefox
iot.eclipse.org — Eclipse
https://iot.eclipse.org/java/tutorial/

Setting up the hardware

Shopping list

We've kept the list of hardware elements needed for the greenhouse to a minimum to make it easy for anyone with no electronics skills to build it.

Here is what you will need:

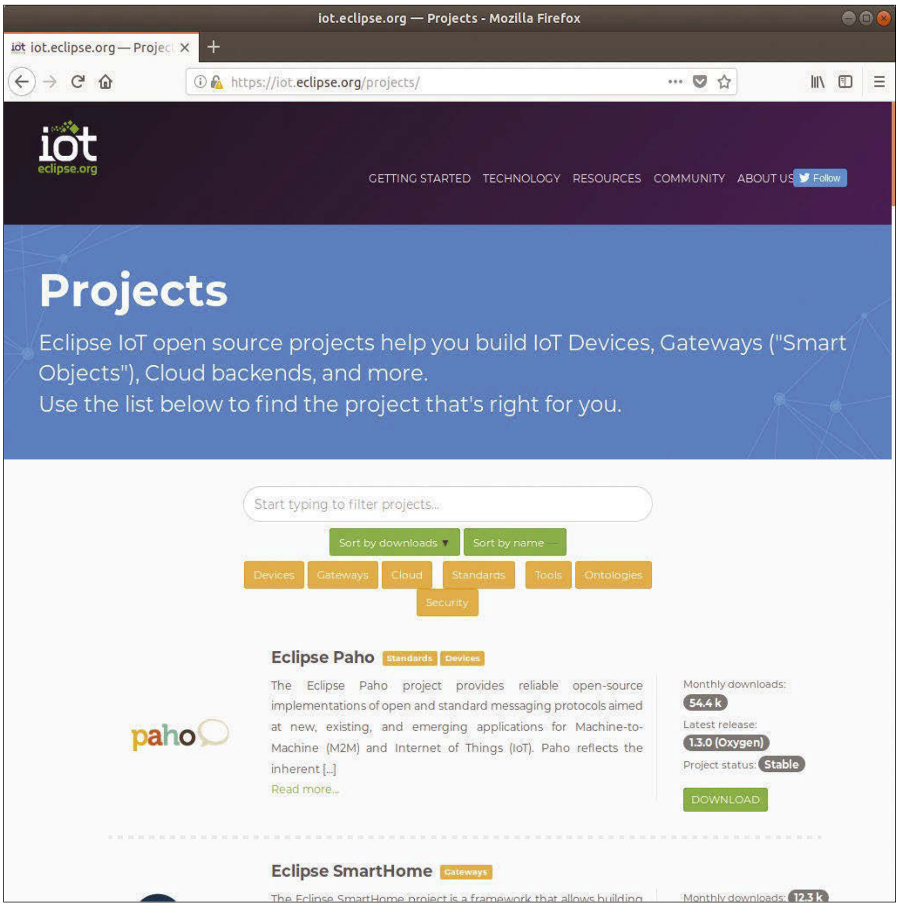
- **Raspberry Pi Model 2** and an **SD card** (min. 4GB) with the latest Debian Linux distribution. Raspbian should be downloaded from the Raspberry Pi Foundation's Downloads page, and these instructions will help you put Raspbian on the SD Card.
- **Grove temperature & humidity sensor**,
- **Grove LED**,
- **4 pin Female Jumper to Grove 4 pin Conversion Cable**,
- **Ethernet cable.**

You can easily order the parts from SeeedStudio thanks to the following wish list: `http://tiny.cc/iot-greenhouse-parts` .

You can also check for a distributor next to you on this web page.

Assembling the Raspberry Pi and the sensors

Figure 3: The tutorial on the Eclipse homepage uses a sensor and an LED module from Grove; unfortunately, the software mentioned in the tutorial is completely outdated.



Projects

Eclipse IoT open source projects help you build IoT Devices, Gateways ("Smart Objects"), Cloud backends, and more.
Use the list below to find the project that's right for you.

Eclipse Paho Standards Devices

The Eclipse Paho project provides reliable open-source implementations of open and standard messaging protocols aimed at new, existing, and emerging applications for Machine-to-Machine (M2M) and Internet of Things (IoT). Paho reflects the inherent [...]
Read more...

Monthly downloads: 54.4 k
Latest release: 1.3.0 (Oxygen)
Project status: Stable
DOWNLOAD

Eclipse SmartHome Gateways

The Eclipse SmartHome project is a framework that allows building

Monthly downloads: 12.3 k

Figure 4: The Eclipse IoT cosmos includes so many subprojects that it even has its own search function.

Initial Barriers

If you are interested in using Eclipse to develop for the IoT, you need to find the right projects for your purposes, but you also have to familiarize yourself with each project separately. The MQTT, CoAP, and LWM2M servers help. Developers can use the servers to quickly test the function of the corresponding clients.

The Eclipse IoT website features videos, presentations, and a solitary tutorial. The tutorial shows step-by-step how developers with a Raspberry Pi, a sensor, and the Eclipse projects Kura, Paho, and Californium monitor a small greenhouse (Figure 3). However, the tutorial uses hopelessly outdated software components that no longer run on current Raspbians, such as Kura 1.3.0, which is over two years old. (Kura 3.1.1 is the current version.)

The videos and presentations provided at the Eclipse IoT website focus on individual aspects of the Eclipse IoT cosmos and give an overview of the existing projects. Nevertheless, Eclipse IoT novices may stumble over the partly outdated and incomplete documentation of the projects, which are often more reminiscent of a reference. You should therefore plan a longer training period from the outset. Since many of the projects use Java, knowledge of the Java language and build tools is helpful.

Last but not least, the development of some of the projects is only making slow progress. For example, Eclipse Milo has had only one 0.1.0 release in 2016. The majority of projects are in incubator status and are not yet suitable for practical use.

If you are interested in an Eclipse IoT project, take a look at the development status, given in the project directory at the project website (Figure 4, [8]). ■■■

Info

[1] Eclipse IoT: *https://iot.eclipse.org*

[2] Contiki-NG: *https://github.com/contiki-ng*

[3] RIOT: *https://github.com/RIOT-OS/RIOT*

[4] FreeRTOS: *https://www.freertos.org*

[5] Zephyr: *http://zephyrplugins.github.io/*

[6] Apache Mynewt project: *https://mynewt.apache.org*

[7] Eclipse BIRT: *http://www.eclipse.org/birt/*

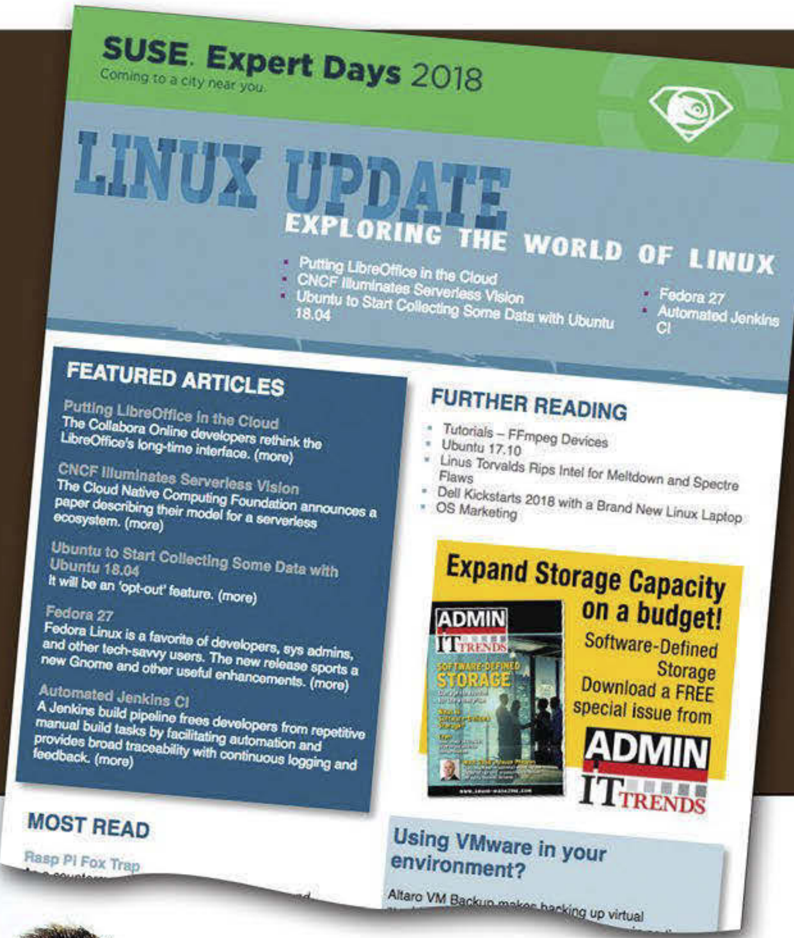
[8] Eclipse IoT projects at a glance: *https://iot.eclipse.org/projects/*

LINUX UPDATE

Need more Linux?

Our free Linux Update newsletter delivers insightful articles and tech tips to your mailbox twice a month. You'll discover:

- **Original articles on real-world Linux**
- **Linux news**
- **Tips on Bash scripting and other advanced techniques**
- **Discounts and special offers available only to newsletter subscribers**



SUSE. Expert Days 2018

Coming to a city near you.

LINUX UPDATE

EXPLORING THE WORLD OF LINUX

- Putting LibreOffice in the Cloud
- CNCF Illuminates Serverless Vision
- Ubuntu to Start Collecting Some Data with Ubuntu 18.04
- Fedora 27
- Automated Jenkins CI

FEATURED ARTICLES

Putting LibreOffice in the Cloud
The Collabora Online developers rethink the LibreOffice's long-time interface. (more)

CNCF Illuminates Serverless Vision
The Cloud Native Computing Foundation announces a paper describing their model for a serverless ecosystem. (more)

Ubuntu to Start Collecting Some Data with Ubuntu 18.04
It will be an 'opt-out' feature. (more)

Fedora 27
Fedora Linux is a favorite of developers, sys admins, and other tech-savvy users. The new release sports a new Gnome and other useful enhancements. (more)

Automated Jenkins CI
A Jenkins build pipeline frees developers from repetitive manual build tasks by facilitating automation and provides broad traceability with continuous logging and feedback. (more)

FURTHER READING

- Tutorials – FFmpeg Devices
- Ubuntu 17.10
- Linus Torvalds Rips Intel for Meltdown and Spectre Flaws
- Dell Kickstarts 2018 with a Brand New Linux Laptop
- OS Marketing

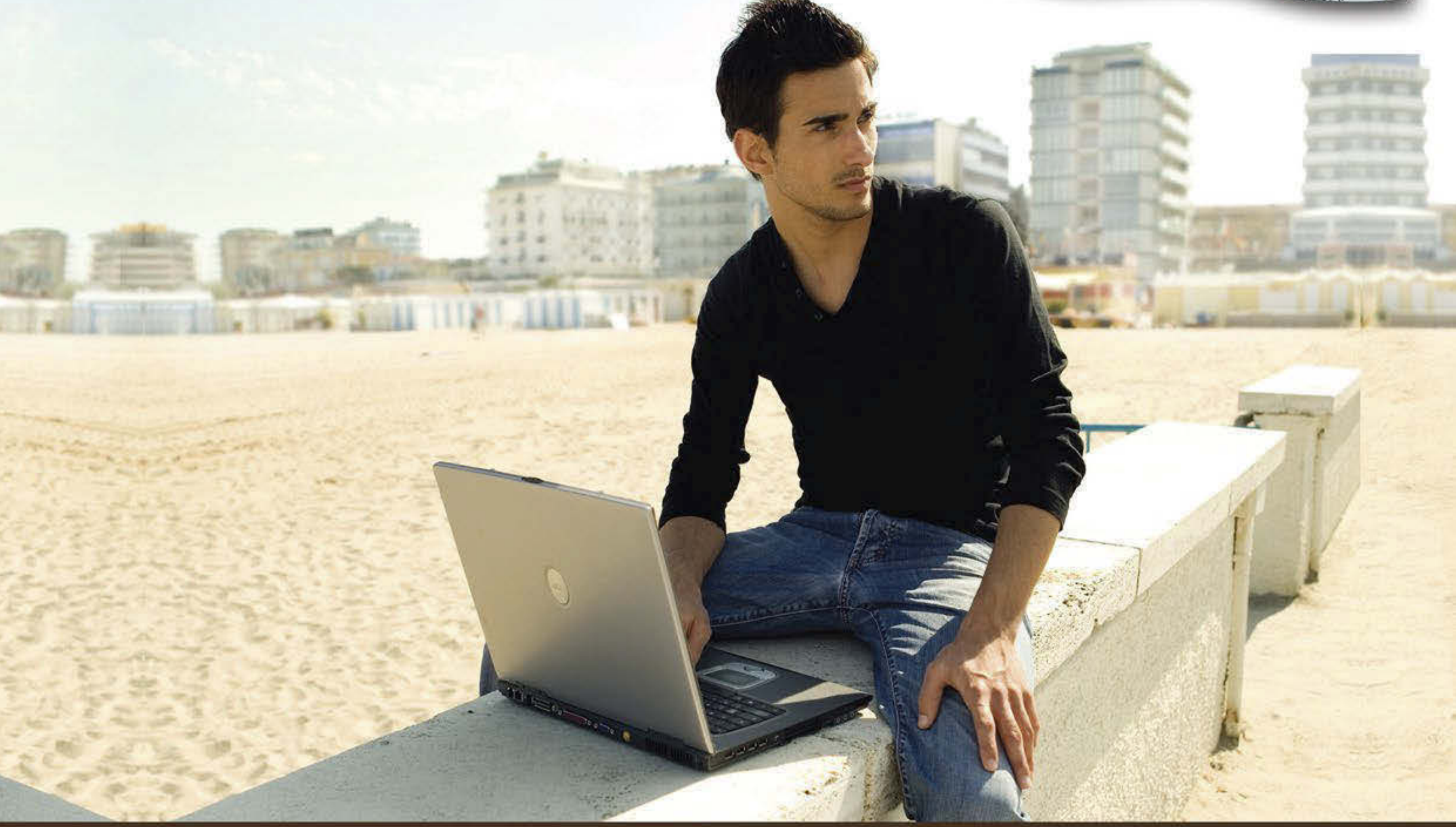
Expand Storage Capacity on a budget!
Software-Defined Storage
Download a FREE special issue from
ADMIN IT TRENDS

MOST READ

Rasp Pi Fox Trap

Using VMware in your environment?

Altaro VM Backup makes backing up virtual



www.linux-magazine.com/newsletter

Ft Photography, Fotolia

Using Bluetooth for in-home positioning services

Helping Spirits

The GPS tracking service isn't precise enough to provide positioning information within a home. Home automation expert Gunnar Beutner decided to harness the invisible spirits of Bluetooth for a home-grown indoor positioning system.

By Gunnar Beutner

Imagine coming home on a cold winter's day; it's already cosily warm, the lights are switched on automatically, and a favorite song is playing through the speakers. Everyday users can already implement this scenario without great difficulty. Software tools such as the open source Home Assistant [1] offer various components for determining the presence of home residents. GPS location capabilities, combined with automation rules, result in some interesting applications for heating and lighting control.

In most cases, however, GPS is not a suitable solution for automating light switches for the rooms within a house. The signal quality of GPS inside buildings is often too poor to allow fast detection, and it is not precise enough to allow location down to the room level.

A few commercial systems offer in-home positioning services. These systems usually rely on a series of sensors that receive signals from portable radio beacons. They then calculate their position based on the signal strength or the time difference of the received signals. However, most systems are proprietary and not intended for private users. One promising tool within this category is the DecaWave DWM1000 [2], which claims to have a detection accuracy of up to 10 centimeters indoors. Unfortunately, I couldn't find any manufacturer-independent information on its suitability for everyday use.

But what about Bluetooth?

Bluetooth as an Alternative

My next attempt was to explore the possibility of room location using Bluetooth beacons and Happy Bubbles [3] Bluetooth detectors, with one sensor in each room. The detectors have open source firmware and include the software necessary to collate the values measured by the individual detectors and determine the spatial position of the Bluetooth beacons.

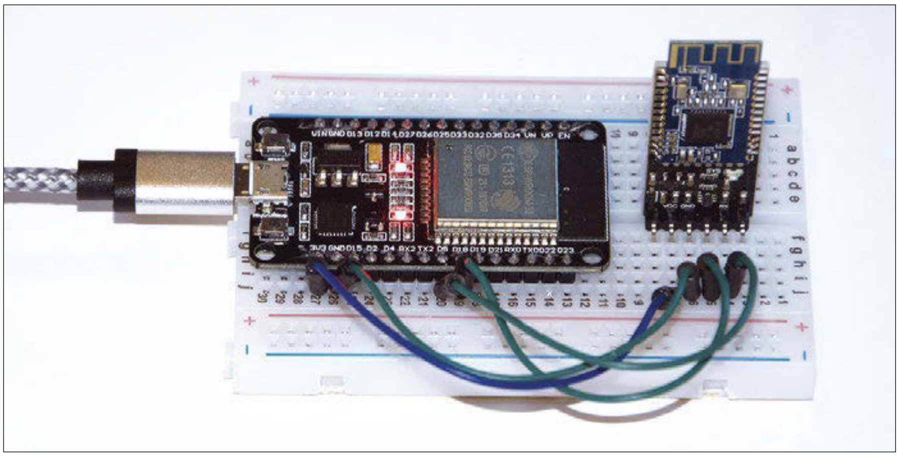


Figure 1: The ESP32 microprocessor (left) is connected to an HM-10 Bluetooth module on a plugin board for test purposes.

Unfortunately, real-world application of this technique comes with some complications. For example, the firmware on the Bluetooth module used by the sensors had a bug that caused the Happy Bubbles detector to crash if it received too many Bluetooth signals within a certain period of time. Although the latest hardware revision fixes the problem, older modules are difficult to update.

What is even more problematic is the fact that the signal strength of the Bluetooth signals generally fluctuates, even if the user does not move at all. In fact, it can happen that the Bluetooth beacon is close to the sensor, but the sensor reports that it is very far

away, thus causing the software to detect a transition to another room and switch off the light.

This effect is amplified when the owner sits in an unfavorable position relative to the sensor, thus shielding the signal. The human body mainly consists of water, which absorbs radio waves at a frequency of 2.4GHz very well. If you sit in the way, you can expect the room to get dark again.

My Own Sensor Platform

Given the complications of commercial alternatives, I decided to try my own approach. As a starting point, I decided on the ESP32 [4] as the hardware platform. For just EUR10, this microcontroller offers two integrated 240MHz processor cores and 512KB RAM. It also has a 2.4GHz wireless module, which it uses to communicate with other WLAN and Bluetooth devices.

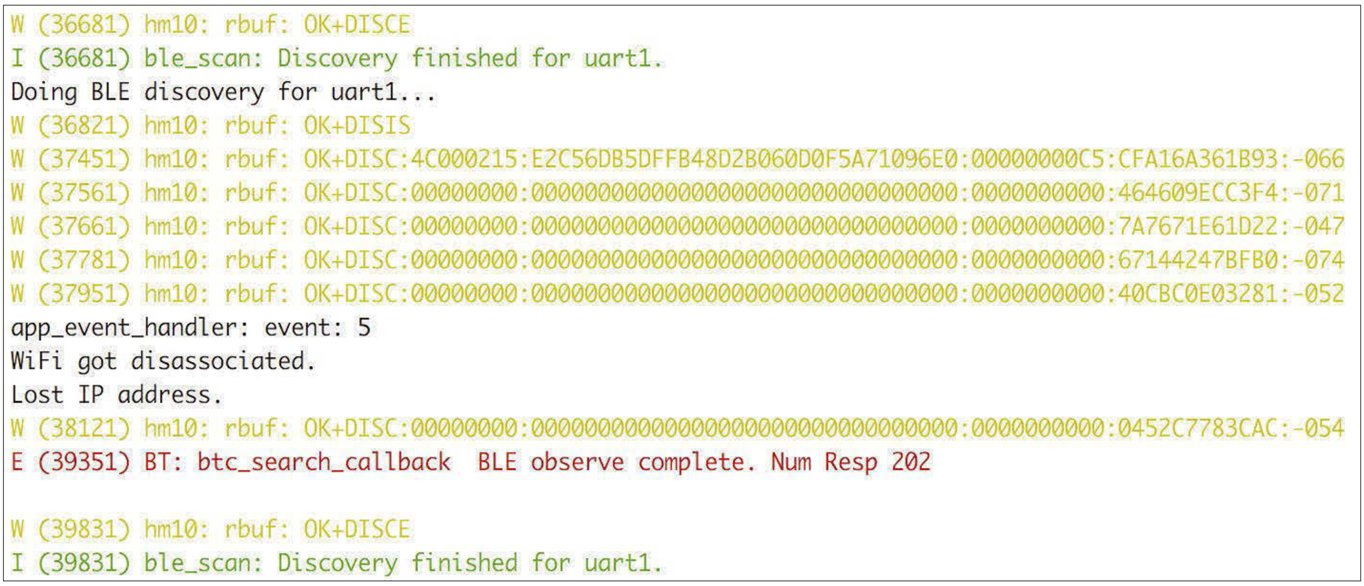
The power is supplied via a micro-USB port, which also serves as a virtual serial interface for writing your own programs to the flash memory of the microcontroller and tracking the output.

You cannot install Linux on the ESP32 chip. You can, however, use a Linux-ready open source SDK called the IoT Development Framework (ESP-IDF,[5]) from the manufacturer Espressif [6]. The SDK is based on the real-time operating system FreeRTOS [7], as well as a number of other open source projects, and it provides the experienced hobby developer with easy access to the built-in hardware. Using the SDK means you won't have to deal with too many platform-specific details.

You can connect external components like temperature sensors, LEDs or – as necessary for this project – additional Bluetooth modules via various GPIO pins. My goal was to install an ESP32 in each room and use its integrated Bluetooth module to measure the signal strength of the received Bluetooth beacons. In addition, I wanted to connect two more HM-10 Bluetooth modules to each ESP32 (Figure 1) and strategically distribute them in the rooms to enable ideal reception of the Bluetooth signals.

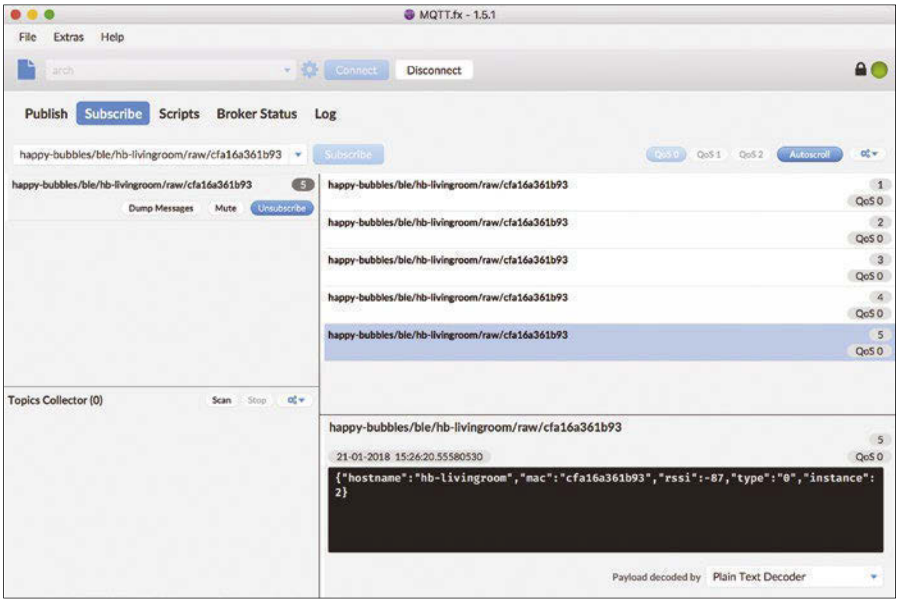
Connecting the Bluetooth modules requires four connections: Ground, supply voltage (3.3 volts provided by a voltage converter integrated on the ESP32), and one transmit and one receive line each for bidirectional communication.

The firmware [8] developed for the ESP32 communicates with the HM-10 modules via a serial interface. The modules offer all their functionality via AT commands, which some



```
W (36681) hm10: rbuf: OK+DISCE
I (36681) ble_scan: Discovery finished for uart1.
Doing BLE discovery for uart1...
W (36821) hm10: rbuf: OK+DISIS
W (37451) hm10: rbuf: OK+DISC:4C000215:E2C56DB5DFFB48D2B060D0F5A71096E0:00000000C5:CFA16A361B93:-066
W (37561) hm10: rbuf: OK+DISC:00000000:00000000000000000000000000000000:0000000000:464609ECC3F4:-071
W (37661) hm10: rbuf: OK+DISC:00000000:00000000000000000000000000000000:0000000000:7A7671E61D22:-047
W (37781) hm10: rbuf: OK+DISC:00000000:00000000000000000000000000000000:0000000000:67144247BFB0:-074
W (37951) hm10: rbuf: OK+DISC:00000000:00000000000000000000000000000000:0000000000:40CBC0E03281:-052
app_event_handler: event: 5
WiFi got disassociated.
Lost IP address.
W (38121) hm10: rbuf: OK+DISC:00000000:00000000000000000000000000000000:0000000000:0452C7783CAC:-054
E (39351) BT: btc_search_callback  BLE observe complete. Num Resp 202

W (39831) hm10: rbuf: OK+DISCE
I (39831) ble_scan: Discovery finished for uart1.
```

Figure 2: The firmware outputs log entries on the serial console: This excerpt shows the scan results for one of the Bluetooth modules.



```
happy-bubbles/ble/hb-livingroom/raw/cfa16a361b93   5

happy-bubbles/ble/hb-livingroom/raw/cfa16a361b93   1  QoS 0
happy-bubbles/ble/hb-livingroom/raw/cfa16a361b93   2  QoS 0
happy-bubbles/ble/hb-livingroom/raw/cfa16a361b93   3  QoS 0
happy-bubbles/ble/hb-livingroom/raw/cfa16a361b93   4  QoS 0
happy-bubbles/ble/hb-livingroom/raw/cfa16a361b93   5  QoS 0

happy-bubbles/ble/hb-livingroom/raw/cfa16a361b93   5
21-01-2018 15:26:20.55580530                       QoS 0
{"hostname":"hb-livingroom","mac":"cfa16a361b93","rssi":-87,"type":"0","instance":
2}
```

Figure 3: MQTT.fx displays the messages that the MQTT broker receives from the sensors.

people may still be familiar with from analog modems (Figure 2). Thus, the `AT+DISI?` command enables receive mode, whereupon the HM-10 module returns a list of detected Bluetooth beacons, including their signal strength.

The scan results initiate an `OK+ DISIS`. Each individual result starts with `OK+DISC` and contains the Bluetooth MAC address (for example cfa16a361b93) and the signal strength (about -66dBm). When all results are transmitted, the ESP32 receives a line with the contents `OK+DISCE`. This line marks the end of reception and the beginning of the next scan process.

An application must collect the sensor data in order to process it. The MQTT message bus protocol, which enjoys great popularity in the IoT sphere, does this job. Any number of MQTT clients log on to the central MQTT broker and exchange information via publish-subscribe messages. Each message has a topic that determines which clients receive the message. The ESP32 modules publish their sensor results every second with the topic `happy-bubbles/ble/Hostname/raw/MAC`. `Hostname` stands for the name of the ESP32 sensor, and `MAC` stands for the Bluetooth MAC address of the beacon.

The name of the topic has a historical meaning in my case, because I gradually switched my sensor network from Happy Bubbles sensors to my own hardware. An MQTT client like MQTT.fx (Figure 3, [9]), which is based on Eclipse Paho [10], finds out which messages are routed through the MQTT broker. RPM and Deb packages for MQTT.fx v1.6.0 are available online [11].

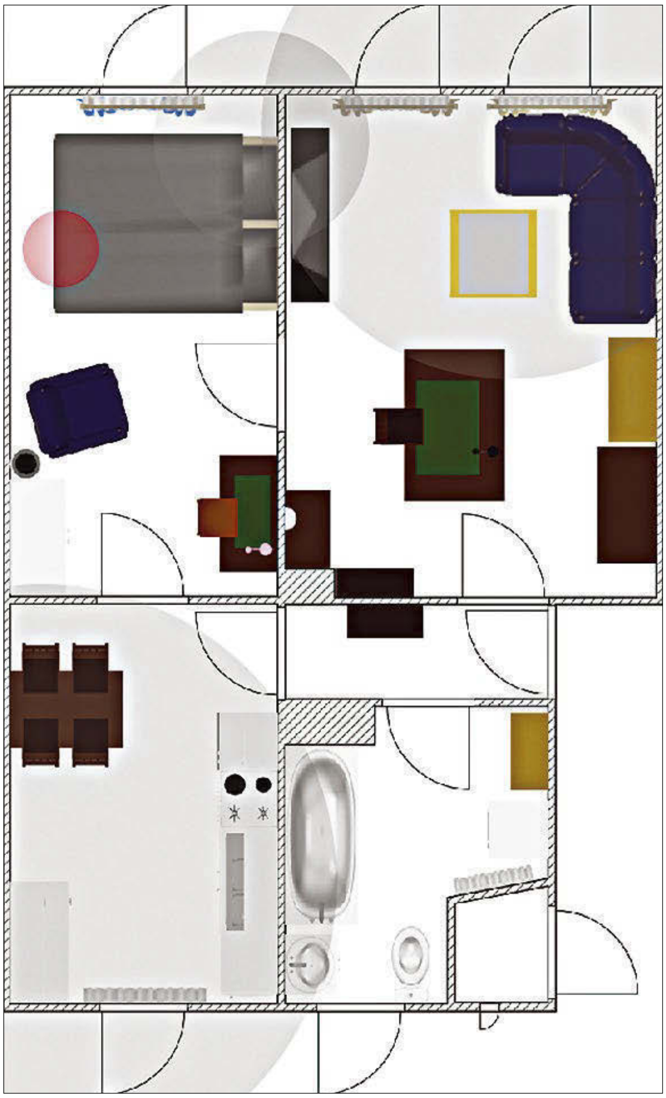


Figure 4: The room map shows the location of the Bluetooth beacon. The smaller the circle around the sensor, the closer the beacon is to the viewer.

Listing 1: Anybody Home?

```
sensor:
  - platform: template
    sensors:
      presence_kitchen:
        entity_id:
          - sensor.location_gunnar
          - sensor.location_lisa
        friendly_name: Presence Kitchen
        value_template: >-
          {%- if is_state("sensor.location_lisa",
           "Kitchen") or is_state("sensor.location_
           gunnar", "Kitchen") %}
            Here
          {% elif not is_state("sensor.location_lisa",
           "Away") or not is_state("sensor.location_
           gunnar", "Away") %}
            Home
          {% else %}
            Away
          {%- endif %}
```

Listing 2: Heating Depending on Location

```
automation:
  - alias: Heating Kitchen
    trigger:
      platform: time
      minutes: '/5'
      seconds: 00
    action:
      service: climate.set_temperature
      entity_id: climate.kitchen
      data_template:
        temperature: >-
          {%- if is_state("sensor.presence_kitchen",
            "Here") %}
            21
          {% elif is_state("sensor.presence_kitchen",
            "Home") %}
            19
          {% else %}
            17
          {%- endif %}
```

Listing 3: Lighting Control

```
automation Kitchen:
  - alias: Kitchen Light (on)
    trigger:
       platform: state
       entity_id: sensor.presence_kitchen
       to: Here
    action:
      - service: homeassistant.turn_on
        entity_id: light.kitchen
      - service: timer.cancel
        entity_id: timer.kitchen_light
  - alias: Kitchen Light (left room)
    trigger:
       platform: state
       entity_id: sensor.presence_kitchen
    condition:
      condition: template
      value_template: '{{ not is_state ("sensor.presence_
                       kitchen", "Here") }}'
    action:
       service: timer.start
       entity_id: timer.kitchen_light
  - alias: Kitchen Light (off)
    trigger:
       platform: event
       event_type: timer.finished
       event_data:
          entity_id: timer.kitchen_light
    action:
       service: homeassistant.turn_off
       entity_id: light.kitchen
```

Once the MQTT broker has collected all sensor results, another software component evaluates them. The Bluetooth Beacon Presence Detection Server (Presence, [12]) which is written in Go, logs on to the MQTT broker as an MQTT client and receives the sensor data from it.

Based on signal strength, Presence decides on the room in which each user's beacon is most likely found (Figure 4) and publishes its own results via MQTT.

The MQTT broker then sends the exact room position of the residents in real time to the home automation tool Home Assistant [1]. A template sensor calculates one of the following states for each room:

- At least one person is in the room.
- Nobody is in the room (but in one of the others).
- No one is home.

The Home Assistant rule would look like Listing 1.

Other automation rules now access this state and control the heating, for example, so that the temperature in the kitchen rises to 21 degrees Celsius when someone is in the room. If someone else is in the apartment, the Home Assistant reduces the temperature to 19 degrees (Listing 2).

For lighting control, Home Assistant starts a timer on leaving a room that switches off the light after five minutes. If someone enters the room during this time, the timer stops (Listing 3).

Conclusions

Commercial Bluetooth sensors have proven their value in practical applications. However, even a tiny error can cause complications. For example, if the light goes out due to an inaccurate reading, the user is left suddenly sitting in the dark.

The solution described in this article, which is based on the ESP32 hardware platform and the Home Assistant open source automation framework, offers an alternative approach that could improve your chances for keeping the room lit, and along the way, you just might learn a little more about Bluetooth, MQTT, and home automation. ■■■

Info

[1] Home Assistant: *https://home-assistant.io*

[2] DWM1000 modules: *https://www.decawave.com/products/dwm1000-module*

[3] Happy Bubbles detectors: *https://www.happybubbles.tech/presence/detector*

[4] ESP32: *https://en.wikipedia.org/wiki/ESP32*

[5] ESP-IDF: *https://github.com/espressif/esp-idf*

[6] ESP-IDF SDK on Linux: *https://esp-idf.readthedocs.io/en/latest/get-started/linux-setup.html*

[7] FreeRTOS: *https://freertos.org*

[8] BLE detector: *https://github.com/gunnarbeutner/ble-detector*

[9] MQTT.fx: *http://mqttfx.org*

[10] Eclipse Paho: *https://www.eclipse.org/paho/*

[11] MQTT.fx 1.6.0 for Linux: *http://www.jensd.de/apps/mqttfx/1.6.0/*

[12] Presence server: *https://github.com/happy-bubbles/presence*

Author

Gunnar Beutner has been developing software since 1994. At Netways, he discovered Icinga 2, which is now his favorite project.

Searchable Linux Archive:
17,000
Pages of Practical Know-How
16 YEARS OF LINUX!
Issues 1-191
► All the Tricks
► All the Hacks
SOLD OUT!
THE COMPLETE LINUX MAGAZINE ARCHIVE

ADMIN
Network & Security
COMPLETE ARCHIVE DVD
ISSUES 00-39
OVER
3,000
PAGES OF UP-CLOSE SYSTEM ADMINISTRATION
ISSUE 40/2017
DVD ROM
Smart and Practical Information for Network Professionals

THE COMPLETE
UBUNTU
user
ARCHIVE
DVD ROM
Over 3,000 PAGES! 7 GREAT YEARS OF UBUNTU USER
Searchable DVD
Content Available in Both HTML and PDF Formats
LINUXVOICE
ARCHIVE DVD: ALL 32 ISSUES!
ISSUE 200 JULY 2017
LINUX MAGAZINE
DVD ROM
3,400 PAGES OF LINUX AND OPEN SOURCE LOVE!
raspberry pi
GEEK
ISSUE 23/2017
SOLD OUT!
RASPBERRY PI GEEK ARCHIVE DVD!
More than 1,800 PAGES of information, projects, and fun!

Old hardware and Linux

Oldies but Goodies

Corporations and organizations don't need to buy new computers every two or three years because the old ones are no longer serviceable. We look at which tasks these hardware seniors can handle. *By Erik Bärwaldt*

Over the years, many corporations – and home users – accumulate discarded computers, printers, and laptops. If nobody disposes of them immediately as electronic waste, they often stand around unused in storage rooms. A larger market for used equipment has established itself, and used systems often find buyers or lessors quickly. Because of tax regulations and leasing contract terms, the market is fed by company inventories with equipment only a few years old.

Prices drop extremely quickly, especially in the case of professional computer systems. Moreover, new owners might be able to write these devices off immediately for tax purposes as low-value assets for an instant benefit. The new owner also contributes to a positive ecological balance, because the production of new computer systems is resource intensive.

Unlike new versions of Windows, which might not provide drivers for older hardware, Linux distributions potentially support older hardware better than brand-new equipment. Some special Linux distributions with the right drivers concentrate specifically on such older hardware, but which used systems can admins best buy and run with Linux? Which desktops support production use with the lowest possible administration overhead?

Hardware Criteria

When you are choosing used hardware to run Linux, you need to consider several things up front to prevent imminent failure or costly upgrades of the systems after a short service life.

For example, it is a good idea to look primarily for older high-performance systems from known sources: Conventional desktop PCs for all-around use in offices quickly reach their performance limits or require more RAM or larger hard disks. Because most contain mid-range components, they are also not suitable for computationally intensive tasks you can find in CAD or multimedia environments.

Professional workstations or servers are far better suited candidates for repurposed systems, and they usually have well-equipped basic configurations. In the mobile segment, you should look for the business series in used laptops. In these cases, manufacturers usually offer replacement parts such as rechargeable batteries and optical drives for a considerably longer period of time. Additionally, business devices are generally built with far more valuable components than consumer goods, and because they are more solidly constructed, manufacturers often offer longer warranty periods.

Some pricy portable computers offering workstation performance are consistently tuned by manufacturers for performance. In addition to powerful processors and large memory capacities, they also come with high-performance graphics units, allowing the computers to cope easily with graphically intensive applications. IT buyers purchasing used mobile systems need to differentiate between business all-arounders and mobile workstations, depending on the application scenario.

Software Criteria

A suitable Linux distribution, in terms of available hardware capacity and expected performance potential, will ideally allow you to manage systems easily and give you access to as extensive a software repository as possible, which ensures that users have a sufficient number of programs from which to choose.

Lead Image © Marc Chesneau, Fotolia.com

In the enterprise environment, the systems should also be easy to use. Because of the many working environments under Linux, you will have to make your selection carefully: On computers with relatively little RAM, resource-consuming desktop environments such as KDE Plasma or Gnome are of little use, and even Xfce can be out of place, because desktop environments can occupy more than 500MB of RAM from the outset.

If you run several large standard applications (e.g., LibreOffice, Gimp, Firefox) simultaneously, you will stress the swap partition. A slow hard disk by today's standards will make every task a test of patience.

Significantly fewer resources are consumed by lean desktops like Mate, LXDE, and LXQt. Although they lack detailed configuration options, they are generally based on operating concepts that have been around for decades and require little training. These desktops usually leave enough free memory for heavyweight applications, even on lower powered computer systems, so they rarely or never use the slow swap partition.

Test Scenario

For the practical tests, several older computer systems from different generations were used. In addition to a 2005 HP Compaq NC6220 laptop with a single-core Dothan series processor, I tested a more modern Fujitsu Siemens notebook with a Core 2 Duo processor from 2008 and a high-performance HP Z600 workstation with two quad-core Xeon processors from 2010.

The latest system to be tested was a 2013 HP Elitebook 2570p with a third-generation Core i5 CPU. With the exception of the two oldest notebooks, all devices were equipped with SATA SSD mass storage. In the older systems, I also upgraded main memory to a mid-range configuration; the oldest HP Compaq notebook had a minimal 2GB of RAM.

Out of competition, but for comparison, I also tested the third-generation Raspberry Pi microcomputer (built in 2016). Above all, I wanted the small-board computer (SBC) to give me a feeling for the hardware performance of the veteran computers and show whether it makes sense as an inexpensive desktop replacement for simple office workstations.

The lean test distribution was the Q4OS Debian derivative [1]. Because this operating system is available in both 32- and 64-bit versions, the test results are easy to compare. After a fresh installation, the test team measured the startup behavior on all computers using the appropriate system tools and then determined the memory requirements without any applications running. Under Q4OS, especially, the various desktop environments were tested because the Debian descendant can be installed and used with just a few mouse clicks.

Another test checked processor performance for demanding tasks (e.g., transcoding a DVD with HandBrake [2]), always launching the software with the same settings.

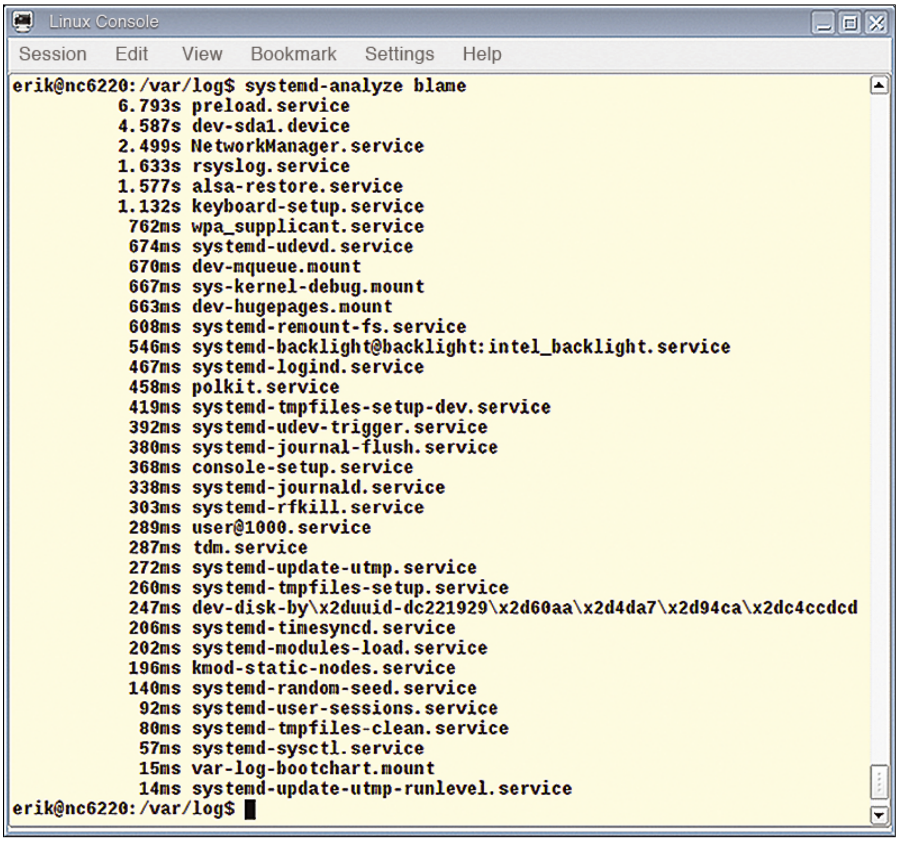
Q4OS was also installed on the Raspberry Pi so it could be compared directly with the full-grown systems. With the exception of video transcoding, it solved the same tasks as Intel-based systems with the same parameters, making the differences in computing power clearly visible and throwing light on the tasks for which the SBC is better suited in everyday office life and how close its performance comes to Intel-based systems.

The idea behind the test was not to collect synthetic standard benchmark results. Benchmarks usually only measure the performance of individual components. When it comes to the real application performance of a complete system, benchmarks have little informative value.

HP NC6220 Laptop

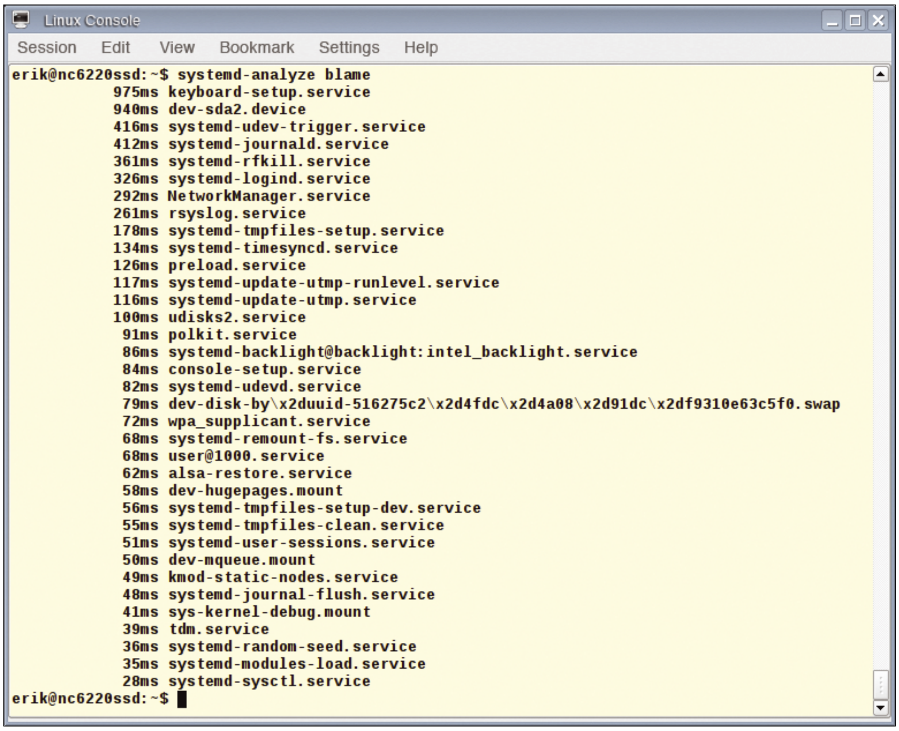
The 14-inch Hewlett-Packard NC6220 laptop [3] was the dinosaur in the test field at almost 13 years of age. Today it would cost between $50 and $100 dollars. The business series device has 2GB of RAM and an 80GB hard disk with an EIDE connector. In addition to Gigabit Ethernet, it has an Intel 2200BG WiFi card, which already supports modern encryption methods according to the WPA-2 standard and achieves a transfer rate of 54Mbps.

The laptop had the 32-bit version of Q4OS installed, which by default uses the KDE 3.x-based Trinity Desktop Environment [4] that impresses with its low resource requirements. Because Q4OS allows you to import other desktop environments at the click of a mouse, the test team also integrated the Mate environment [5], which is based on a similarly aged Gnome 2.x.



```
erik@nc6220:/var/log$ systemd-analyze blame
          6.793s preload.service
          4.587s dev-sda1.device
          2.499s NetworkManager.service
          1.633s rsyslog.service
          1.577s alsa-restore.service
          1.132s keyboard-setup.service
           762ms wpa_supplicant.service
           674ms systemd-udevd.service
           670ms dev-mqueue.mount
           667ms sys-kernel-debug.mount
           663ms dev-hugepages.mount
           608ms systemd-remount-fs.service
           546ms systemd-backlight@backlight:intel_backlight.service
           467ms systemd-logind.service
           458ms polkit.service
           419ms systemd-tmpfiles-setup-dev.service
           392ms systemd-udev-trigger.service
           380ms systemd-journal-flush.service
           368ms console-setup.service
           338ms systemd-journald.service
           303ms systemd-rfkill.service
           289ms user@1000.service
           287ms tdm.service
           272ms systemd-update-utmp.service
           260ms systemd-tmpfiles-setup.service
           247ms dev-disk-by\x2duuid-dc221929\x2d60aa\x2d4da7\x2d94ca\x2dc4ccdcd
           206ms systemd-timesyncd.service
           202ms systemd-modules-load.service
           196ms kmod-static-nodes.service
           140ms systemd-random-seed.service
            92ms systemd-user-sessions.service
            80ms systemd-tmpfiles-clean.service
            57ms systemd-sysctl.service
            15ms var-log-bootchart.mount
            14ms systemd-update-utmp-runlevel.service
erik@nc6220:/var/log$ █
```

Figure 1: The HP NC6220 startup behavior under Q4OS needs improvement.



```
erik@nc6220ssd:~$ systemd-analyze blame
           975ms keyboard-setup.service
           940ms dev-sda2.device
           416ms systemd-udev-trigger.service
           412ms systemd-journald.service
           361ms systemd-rfkill.service
           326ms systemd-logind.service
           292ms NetworkManager.service
           261ms rsyslog.service
           178ms systemd-tmpfiles-setup.service
           134ms systemd-timesyncd.service
           126ms preload.service
           117ms systemd-update-utmp-runlevel.service
           116ms systemd-update-utmp.service
           100ms udisks2.service
            91ms polkit.service
            86ms systemd-backlight@backlight:intel_backlight.service
            84ms console-setup.service
            82ms systemd-udevd.service
            79ms dev-disk-by\x2duuid-516275c2\x2d4fdc\x2d4a08\x2d91dc\x2df9310e63c5f0.swap
            72ms wpa_supplicant.service
            68ms systemd-remount-fs.service
            68ms user@1000.service
            62ms alsa-restore.service
            58ms dev-hugepages.mount
            56ms systemd-tmpfiles-setup-dev.service
            55ms systemd-tmpfiles-clean.service
            51ms systemd-user-sessions.service
            50ms dev-mqueue.mount
            49ms kmod-static-nodes.service
            48ms systemd-journal-flush.service
            41ms sys-kernel-debug.mount
            39ms tdm.service
            36ms systemd-random-seed.service
            35ms systemd-modules-load.service
            28ms systemd-sysctl.service
erik@nc6220ssd:~$ █
```

Figure 2: Even on the NC6220 with an EIDE interface, an SSD really speeds things up.

Amazingly Agile

After installation, the computer was amazingly agile despite its Stone Age single-core Pentium M series processor and technically outdated hard drive: Trinity was up and running after about 25 seconds. Starting the kernel took about 4 seconds and the userspace environment 20 seconds.

Analysis showed that preload and network services and hard disk mounting were slowing down the boot process. The slow mounting of the hard disk can be traced to the filesystem check the system performs for safety reasons each time it is started (Figure 1). The picture was not much different with the Mate desktop. The kernel took 4 seconds, and the userspace environment was ready for use after 20 seconds.

Small differences in RAM showed up between the desktops: Whereas Trinity used 384MB of RAM, or 392MB when including shared memory for the graphics card and the cache, the Mate environment was satisfied with 350 and 360MB, respectively. Thus, Mate proved to be even more economical in resource consumption.

To test the negative influence of slow EIDE mass storage on the performance of the system during many accesses, the Toshiba hard disk was replaced with an OWC SSD [6]. The boot time of the system with Trinity was reduced to about 15 seconds, of which the kernel took less than 4 seconds and the userspace environment a good 10 seconds.

Even the most time-consuming services changed and were processed in less than a second. Despite a filesystem check that was also preset in the SSD, the system mounted the disk significantly faster (Figure 2). A similar picture emerged with the retrofitted Mate desktop: The boot times until the system was fully operational were about 15 seconds.

Next, the system transcoded video to check processor performance. The free HandBrake software makes intensive use of special instruction set extensions in modern processors and is capable of multithreading, resulting in significant speed advantages on multicore processors. In this test, as expected, the aged Dothan processor had to admit defeat.

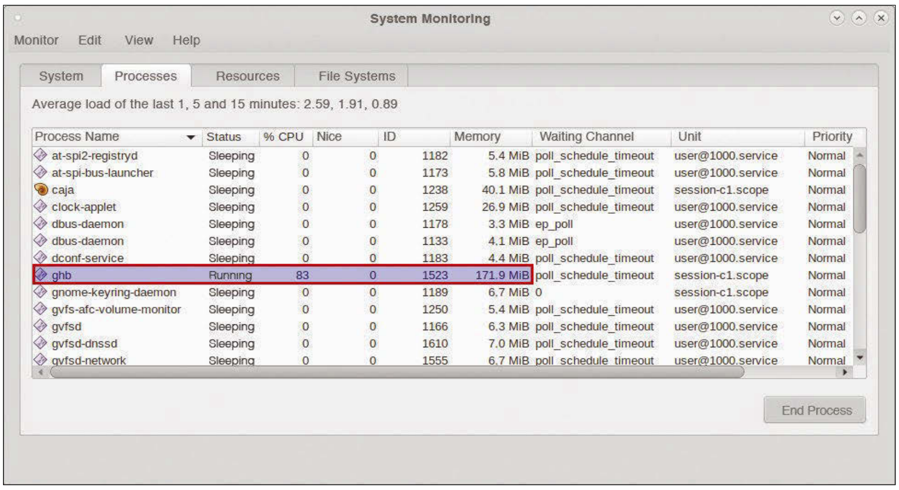
When transcoding the source medium into a Matroska container with PAL resolution using the H.264 video format and an audio track in AAC format, the CPU worked continuously at full load, averaging just 8 to 10 frames per second. One reason is that the processor only supports the MMX2 and SSE instruction set extensions, not modern technologies like SSE4 [7]. Machines with processors older than the Core 2 Penryn generation are generally not suitable for demanding multimedia tasks (Figure 3).

Esprimo Mobile D9510 Laptop

The D9510 [8] laptop produced in 2008 by Japanese-German joint venture Fujitsu Siemens is available for $80 to $140. It comes with a Core 2 generation dual-core processor and 4GB of RAM. The system can use the 64-bit variant of Q4OS to access the largest possible amount of RAM.

The D9510 has far more modern mass storage than the HP NC6220: Instead of the slow EIDE interface, the disk subsystem uses a SATA II bus [9], which promises significantly more agile system behavior. The system sported a conventional 2.5-inch hard disk.

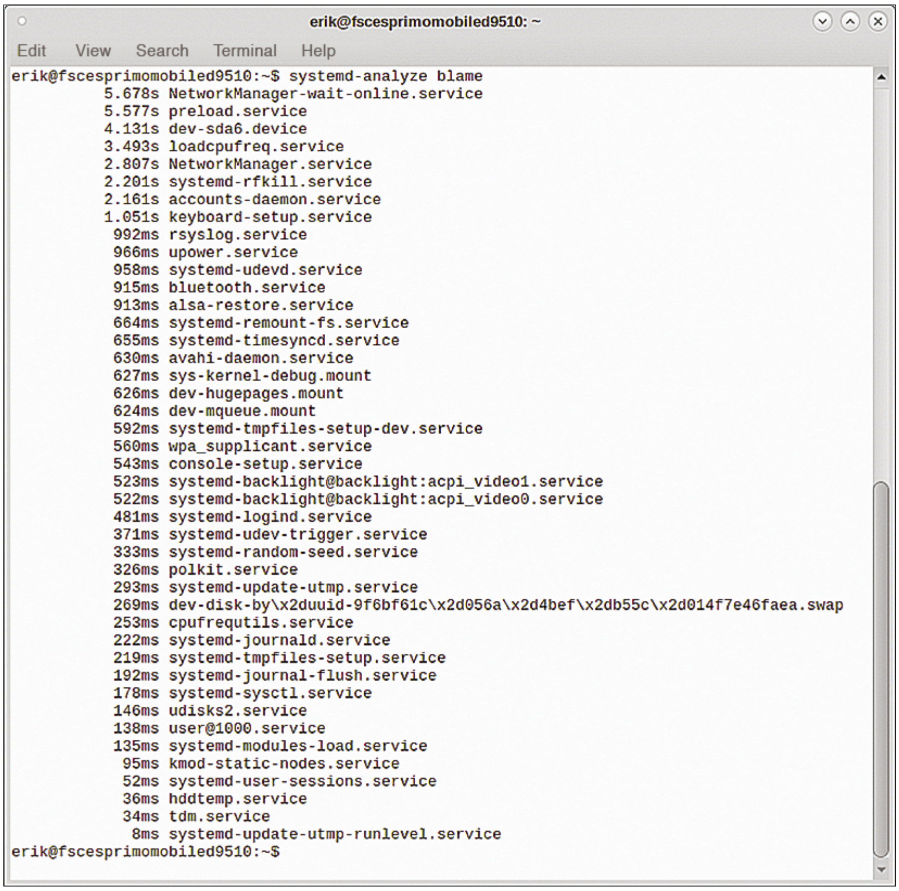
The Q4OS operating system installed on the D9510 without any problems; as expected, it showed far better startup



Average load of the last 1, 5 and 15 minutes: 2.59, 1.91, 0.89

Process Name	Status	% CPU	Nice	ID	Memory	Waiting Channel	Unit	Priority
at-spi2-registryd	Sleeping	0	0	1182	5.4 MiB	poll_schedule_timeout	user@1000.service	Normal
at-spi-bus-launcher	Sleeping	0	0	1173	5.8 MiB	poll_schedule_timeout	user@1000.service	Normal
caja	Sleeping	0	0	1238	40.1 MiB	poll_schedule_timeout	session-c1.scope	Normal
clock-applet	Sleeping	0	0	1259	26.9 MiB	poll_schedule_timeout	user@1000.service	Normal
dbus-daemon	Sleeping	0	0	1178	3.3 MiB	ep_poll	user@1000.service	Normal
dbus-daemon	Sleeping	0	0	1133	4.1 MiB	ep_poll	user@1000.service	Normal
dconf-service	Sleeping	0	0	1183	4.4 MiB	poll_schedule_timeout	user@1000.service	Normal
ghb	Running	83	0	1523	171.9 MiB	poll_schedule_timeout	session-c1.scope	Normal
gnome-keyring-daemon	Sleeping	0	0	1189	6.7 MiB	0	session-c1.scope	Normal
gvfs-afc-volume-monitor	Sleeping	0	0	1250	5.4 MiB	poll_schedule_timeout	user@1000.service	Normal
gvfsd	Sleeping	0	0	1166	6.3 MiB	poll_schedule_timeout	user@1000.service	Normal
gvfsd-dnssd	Sleeping	0	0	1610	7.0 MiB	poll_schedule_timeout	user@1000.service	Normal
gvfsd-network	Sleeping	0	0	1555	6.7 MiB	poll_schedule_timeout	user@1000.service	Normal

Figure 3: For demanding tasks, the Dothan CPU has to admit defeat.



```
erik@fscesprimomobiled9510:~$ systemd-analyze blame
          5.678s NetworkManager-wait-online.service
          5.577s preload.service
          4.131s dev-sda6.device
          3.493s loadcpufreq.service
          2.807s NetworkManager.service
          2.201s systemd-rfkill.service
          2.161s accounts-daemon.service
          1.051s keyboard-setup.service
           992ms rsyslog.service
           966ms upower.service
           958ms systemd-udevd.service
           915ms bluetooth.service
           913ms alsa-restore.service
           664ms systemd-remount-fs.service
           655ms systemd-timesyncd.service
           630ms avahi-daemon.service
           627ms sys-kernel-debug.mount
           626ms dev-hugepages.mount
           624ms dev-mqueue.mount
           592ms systemd-tmpfiles-setup-dev.service
           560ms wpa_supplicant.service
           543ms console-setup.service
           523ms systemd-backlight@backlight:acpi_video1.service
           522ms systemd-backlight@backlight:acpi_video0.service
           481ms systemd-logind.service
           371ms systemd-udev-trigger.service
           333ms systemd-random-seed.service
           326ms polkit.service
           293ms systemd-update-utmp.service
           269ms dev-disk-by\x2duuid-9f6bf61c\x2d056a\x2d4bef\x2db55c\x2d014f7e46faea.swap
           253ms cpufrequtils.service
           222ms systemd-journald.service
           219ms systemd-tmpfiles-setup.service
           192ms systemd-journal-flush.service
           178ms systemd-sysctl.service
           146ms udisks2.service
           138ms user@1000.service
           135ms systemd-modules-load.service
            95ms kmod-static-nodes.service
            52ms systemd-user-sessions.service
            36ms hddtemp.service
            34ms tdm.service
             8ms systemd-update-utmp-runlevel.service
erik@fscesprimomobiled9510:~$
```

Figure 4: Dawdlers on the Fujitsu Siemens laptop.

times than the NC6220. Here, too, network initialization, hard disk mounting, and the preload service applied the brakes at boot time (Figure 4).

Between two physical cores and the processing of individual system services in parallel, the overall latencies were lower on the D9510 than they would be on single-core processor devices. The Fujitsu notebook only takes around 21 seconds with the Mate desktop and a good 19 seconds with Trinity before it is ready for use. However, memory usage was significantly higher than with the HP Compaq notebook: Trinity and Mate grabbed between 520 and 530MB of RAM.

The Penryn processor was far better at transcoding a video DVD than the Dothan processor, thanks to the integrated instruction set extensions and the two processor cores. Although HandBrake used both processor cores almost to capacity and memory requirements increased significantly, the computer achieved rates of between 45 and 70 frames per second (Figure 5), which makes this setup suitable for applications that require a little more computing power.

Faster with SSD

A significant increase in speed was expected after replacing the SATA II hard disk with an SSD, because additional performance potential was certainly available. The SATA II bus supports maximum physical transfer rates of about 250MBps (the EIDE interface in the HP Compaq NC6220 reached a maximum of 100MBps). The access times of an SSD when retrieving and storing data are significantly shorter than with conventional disks, as well.

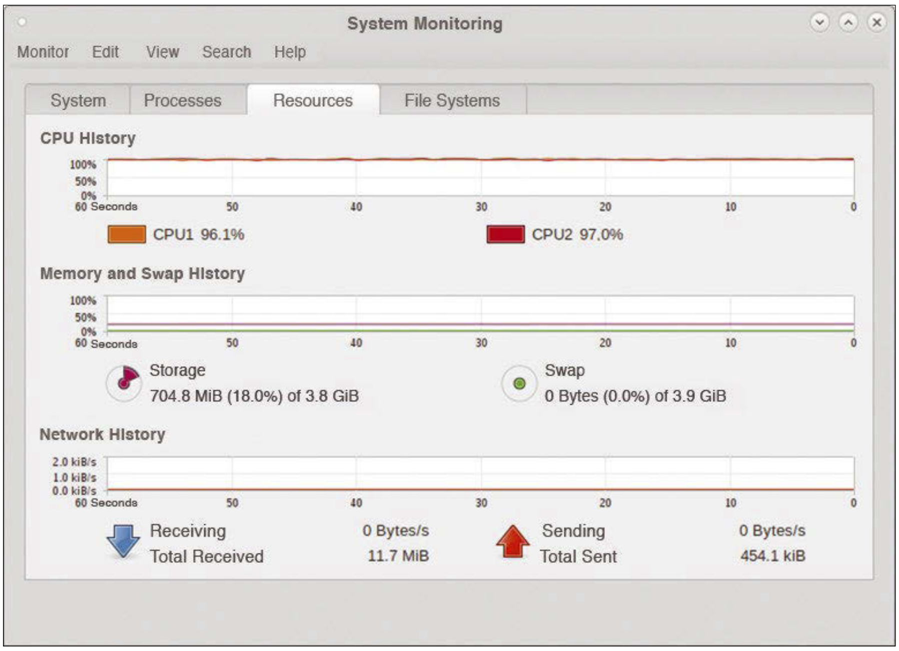
If you plan to run very memory intensive applications such as image processing or even large databases, you can upgrade the disk, but bear in mind that the Core 2 chipsets in notebooks only support RAM expansion to a maximum of 8GB.

HP Z600 Workstation

The 2010 HP Z600 [10] professional workstation by Hewlett-Packard is a real powerhouse. With an exterior designed by BMW Designworks, the tower weighs approximately 19kg and is equipped with two Intel Xeon processors [11]. The 16GB of RAM in a triple-channel configuration means only one third of the maximum capacity is available. The two Nehalem generation Xeon processors each have four physical and four logical cores onboard.

A 240GB SSD by British manufacturer Integral was used as mass storage in the test device, but the workstation has additional slots for hard disks or SSDs. In addition to SATA, disks can use a SAS interface, and you can run the Z600 with hardware RAID configurations.

The workstation can be further upgraded with even more powerful Hexacore



CPU1 96.1%
CPU2 97.0%

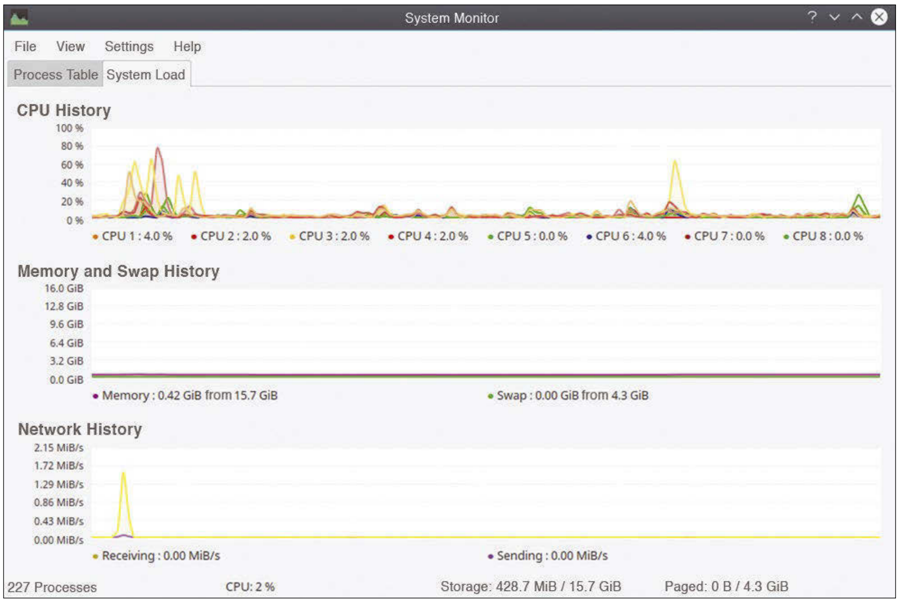
Storage
704.8 MiB (18.0%) of 3.8 GiB

Swap
0 Bytes (0.0%) of 3.9 GiB

Receiving 0 Bytes/s
Total Received 11.7 MiB

Sending 0 Bytes/s
Total Sent 454.1 kiB

Figure 5: The Fujitsu Siemens laptop was running at full capacity when asked to transcode video.



CPU 1 : 4.0 % · CPU 2 : 2.0 % · CPU 3 : 2.0 % · CPU 4 : 2.0 % · CPU 5 : 0.0 % · CPU 6 : 4.0 % · CPU 7 : 0.0 % · CPU 8 : 0.0 %

Memory : 0.42 GiB from 15.7 GiB · Swap : 0.00 GiB from 4.3 GiB

Receiving : 0.00 MiB/s · Sending : 0.00 MiB/s

227 Processes · CPU: 2 % · Storage: 428.7 MiB / 15.7 GiB · Paged: 0 B / 4.3 GiB

Figure 6: The KDE Plasma 5 desktop on the Z600 is surprisingly frugal.

processors, each with six physical and six logical cores, and with PCIe SSDs, which accelerate the system enormously and provide more storage capacity, especially for write-heavy applications. Depending on the equipment, this setup would cost between $150 and $800.

Also note that systems in the HP Z series were designed for use 24/7; therefore, they do not use conventional RAM devices, but RAM based on the DDR3 standard with error-correcting code (ECC) memory. Because most workstations of the Z600 series also have a powerful dedicated graphics card, generally designed for two- or three-monitor operation, the machines are suitable for graphically demanding applications such as CAD.

Because some graphics adapters only have DMS-59 output [12], operating a monitor requires an adapter cable with two monitor connections. Such cables are available with VGA, DVI-I, display port, or HDMI connections.

Very Fast

Unsurprisingly, the Z600 Workstation demonstrates its brute speed during the Q4OS installation routine, which prompted the installation of a heavyweight desktop environment such as Plasma 5 and Gnome. After installation, the usual performance specifications were checked.

As expected, the Z600 boot process was far faster than for the previous test candidates: After the install, the system was ready for use in just 12 seconds. The heavyweight desktops, especially, showed almost no latency after logging in compared with the two lightweights Trinity and Mate.

This time, the WiFi connection came courtesy of a PCI card, which took several seconds to establish a connection, and slowed down the system at the start. Latencies caused by the preload service or SSD mounting no longer occurred with the Z600.

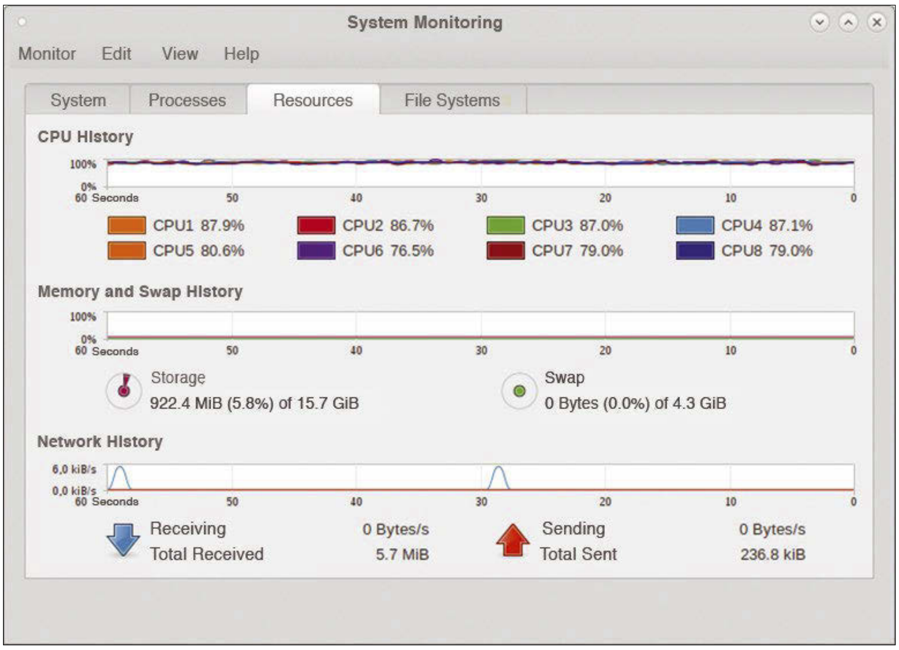
Surprisingly, the resource check revealed that the supposedly massive KDE Plasma 5 desktop used less memory than Mate or Trinity: Plasma 5 managed with around 380 to 430MB, and the desktop environments usually considered leaner required between 600 and 630MB of RAM (Figure 6).

The Z600 also shined during video transcoding by playing to the strengths of the two Xeon CPUs. The process utilized 75 to 88 percent of the cores on average and, for individual cores, at times slightly more than 90 percent. The Z600 thus completed the DVD video transcoding in a few minutes. The frame rate rose to well over 300 frames per second at times (Figure 7).

The eight-year-old professional workstation is still in the upper performance segment and, in some respects, can compete with current computers with four- and eight-core processors. In view of the otherwise good equipment, the Z600 proves to be a genuine alternative for admins looking for inexpensive but powerful used equipment.

HP Elitebook 2570p Notebook

The HP Elitebook 2570p [13] by Hewlett-Packard came onto the market in 2013 for demanding road warriors. It is now available for $150 to $250. The 12.5-inch notebook is consistently designed for mobile use and is designed and manufactured to military standards – which makes it correspondingly robust. A device with an Ivy

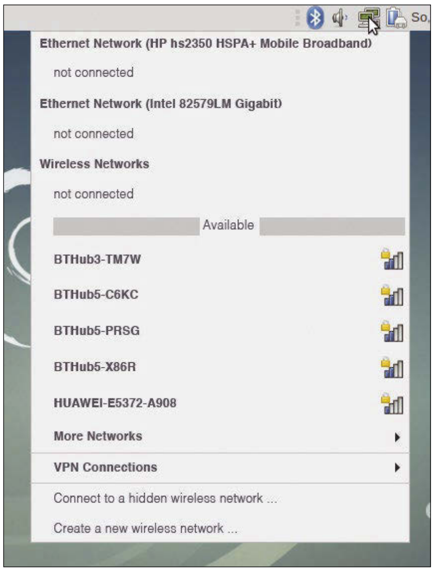


CPU1 87.9% · CPU2 86.7% · CPU3 87.0% · CPU4 87.1%
CPU5 80.6% · CPU6 76.5% · CPU7 79.0% · CPU8 79.0%

Storage: 922.4 MiB (5.8%) of 15.7 GiB
Swap: 0 Bytes (0.0%) of 4.3 GiB

Receiving 0 Bytes/s · Total Received 5.7 MiB
Sending 0 Bytes/s · Total Sent 236.8 kiB

Figure 7: Even the most demanding tasks did not push the Z600 to its limits.



Ethernet Network (HP hs2350 HSPA+ Mobile Broadband)
not connected
Ethernet Network (Intel 82579LM Gigabit)
not connected
Wireless Networks
not connected
Available
BTHub3-TM7W
BTHub5-C6KC
BTHub5-PRSG
BTHub5-X86R
HUAWEI-E5372-A908
More Networks
VPN Connections
Connect to a hidden wireless network ...
Create a new wireless network ...

Figure 8: Q4OS boasts excellent driver features – even for otherwise stubborn WWAN cards.

Bridge generation Core i5 processor, 8GB of RAM, and a 960GB SSD by Mushkin was used in the test.

Q4OS initially showed no weaknesses in hardware detection. Unlike Debian, it includes proprietary firmware components, so even the Qualcomm WiFi WWAN card built into the Elitebook was easily detected, and it was usable after inserting a SIM card (Figure 8).

However, the Q4OS test system did have problems with input devices: The use of integrated mouse buttons was tricky at times, and clicked windows disappeared immediately after opening and only remained open permanently after the third or fourth click.

Apart from that, the Elitebook was fairly inconspicuous: Booting took 13 seconds – again the WiFi network connection turned out to be the biggest slowdown factor (Figure 9). Because it supports the SATA III standard, the Elitebook 2570p has significantly improved performance, especially for read and write operations on hard disks. Although the SSD is quite old, it achieves very attractive data transfer rates thanks to the very fast SandForce controller, which achieves good data compression and buffering. It almost fully utilized the SATA III bus.

The built-in Core i5 CPU with a normal maximum clock frequency of 2.5GHz and a turbo frequency of 3.1GHz achieved a video transcoding rate of 120 to 150 frames per second on the two physical and logical cores, which brings the midrange CPU up to the values of the four-years-older Xeon CPU of the HP workstation. However, the older processor has a lower clock speed. The mobile processor in the notebook performs very well in this performance segment because of some multimedia instruction set extensions that the older processor does not support.

The Elitebook was on track when it came to memory requirements, too: The Mate desktop grabbed around 530 to 550MB, and Trinity claimed slightly more resources, at 560 to 580 MB.

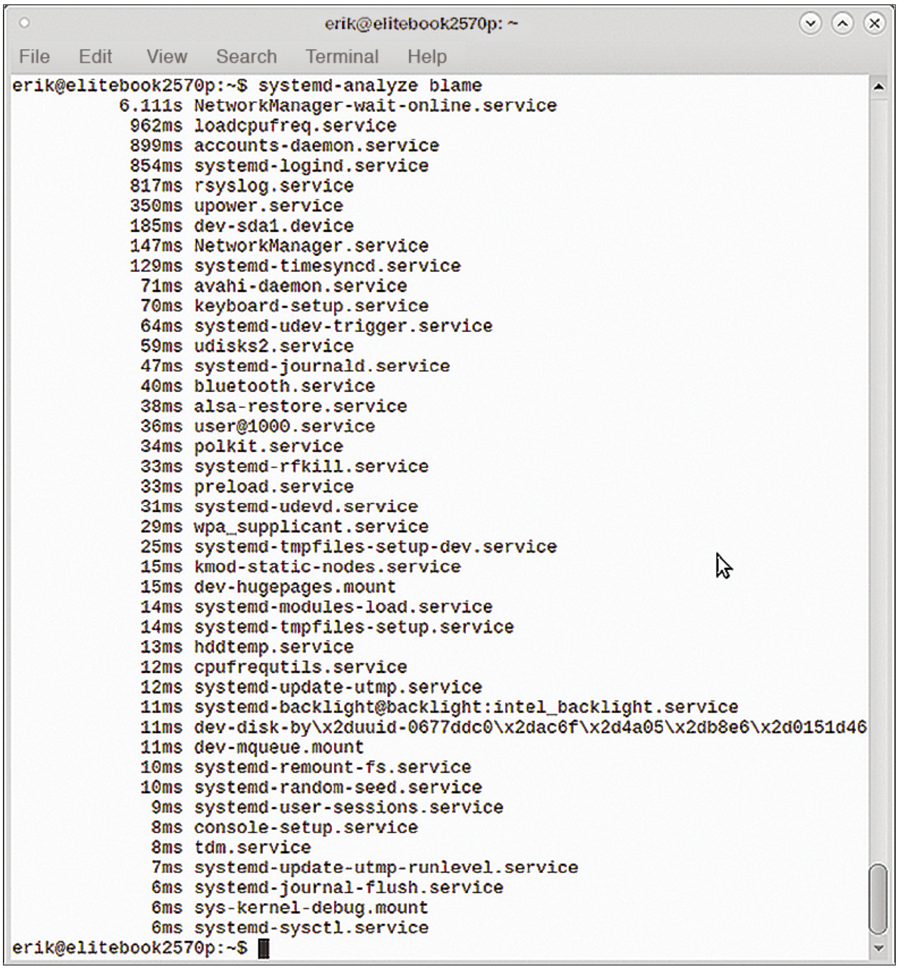
Raspberry Pi 3 Model B

The third generation of the British SBC costs about $35 and comes with some improvements compared with its predecessor: The Broadcom BCM2837 quad-core processor is the first 64-bit CPU in a Raspberry Pi [14]. It clocks at 1.2GHz but lacks hyperthreading and turboboost. However, the CPU can be overclocked with a few tricks.

The built-in WiFi hardware makes it possible to establish a wireless network connection without an additional WiFi USB dongle. Thanks to microSD card support, sufficient memory is also available. Another advantage of the SBC is the HDMI connection for the monitor, which allows the device to cooperate with most modern TV sets. Audio signals also travel over this bus.

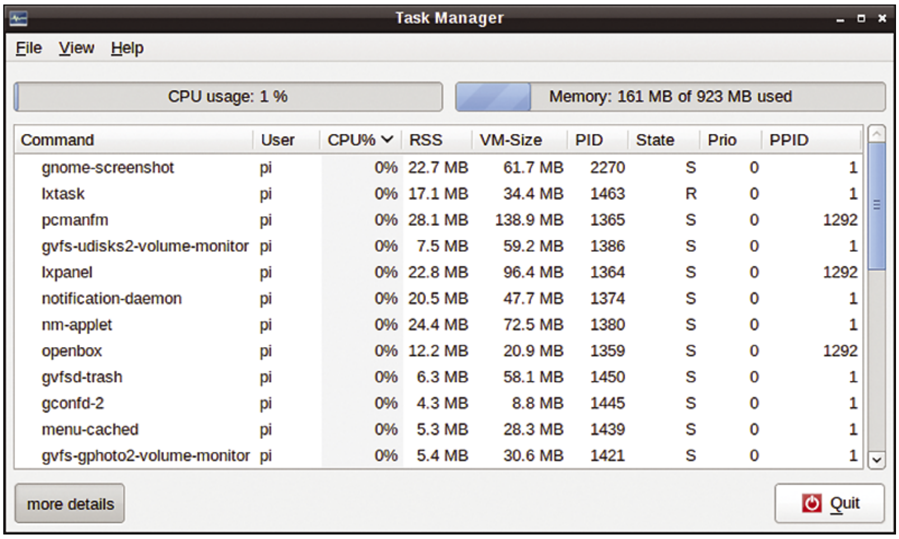
The ready-made Q4OS image [1] for the Raspberry Pi was still version 1.8 (Orion) at the time of testing, although Scorpion (v2.4) has since been released. After downloading and unpacking the ZIP archive, I transferred the 350MB image to a 4GB microSD card, which then served as the boot medium.

The operating system does not work in Live mode but is installed, so you log on with the *pi* username and *raspberry* password. Afterward, you need to complete some basic configurations, possibly adjust the locale, change the password, and perform similar tasks. In the end, Q4OS comes up with its familiar desktop.



```
erik@elitebook2570p:~$ systemd-analyze blame
          6.111s NetworkManager-wait-online.service
           962ms loadcpufreq.service
           899ms accounts-daemon.service
           854ms systemd-logind.service
           817ms rsyslog.service
           350ms upower.service
           185ms dev-sda1.device
           147ms NetworkManager.service
           129ms systemd-timesyncd.service
            71ms avahi-daemon.service
            70ms keyboard-setup.service
            64ms systemd-udev-trigger.service
            59ms udisks2.service
            47ms systemd-journald.service
            40ms bluetooth.service
            38ms alsa-restore.service
            36ms user@1000.service
            34ms polkit.service
            33ms systemd-rfkill.service
            33ms preload.service
            31ms systemd-udevd.service
            29ms wpa_supplicant.service
            25ms systemd-tmpfiles-setup-dev.service
            15ms kmod-static-nodes.service
            15ms dev-hugepages.mount
            14ms systemd-modules-load.service
            14ms systemd-tmpfiles-setup.service
            13ms hddtemp.service
            12ms cpufrequtils.service
            12ms systemd-update-utmp.service
            11ms systemd-backlight@backlight:intel_backlight.service
            11ms dev-disk-by\x2duuid-0677ddc0\x2dac6f\x2d4a05\x2db8e6\x2d0151d46
            11ms dev-mqueue.mount
            10ms systemd-remount-fs.service
            10ms systemd-random-seed.service
             9ms systemd-user-sessions.service
             8ms console-setup.service
             8ms tdm.service
             7ms systemd-update-utmp-runlevel.service
             6ms systemd-journal-flush.service
             6ms sys-kernel-debug.mount
             6ms systemd-sysctl.service
erik@elitebook2570p:~$ 
```

Figure 9: Thanks to the SATA III SSD, the small Elitebook 2570p is very fast.



CPU usage: 1 %

Memory: 161 MB of 923 MB used

Command	User	CPU%	RSS	VM-Size	PID	State	Prio	PPID
gnome-screenshot	pi	0%	22.7 MB	61.7 MB	2270	S	0	1
lxtask	pi	0%	17.1 MB	34.4 MB	1463	R	0	1
pcmanfm	pi	0%	28.1 MB	138.9 MB	1365	S	0	1292
gvfs-udisks2-volume-monitor	pi	0%	7.5 MB	59.2 MB	1386	S	0	1
lxpanel	pi	0%	22.8 MB	96.4 MB	1364	S	0	1292
notification-daemon	pi	0%	20.5 MB	47.7 MB	1374	S	0	1
nm-applet	pi	0%	24.4 MB	72.5 MB	1380	S	0	1
openbox	pi	0%	12.2 MB	20.9 MB	1359	S	0	1292
gvfsd-trash	pi	0%	6.3 MB	58.1 MB	1450	S	0	1
gconfd-2	pi	0%	4.3 MB	8.8 MB	1445	S	0	1
menu-cached	pi	0%	5.3 MB	28.3 MB	1439	S	0	1
gvfs-gphoto2-volume-monitor	pi	0%	5.4 MB	30.6 MB	1421	S	0	1

Figure 10: The Raspberry Pi quickly reached its RAM limit in the test.

Ready to Go

As with Q4OS PC versions, you install additional software or alternative desktops with the Desktop Profiler. The version for the Raspberry Pi only offers KDE SC 4.14.2 or the two lean desktop environments Xfce and LXDE, in addition to the Trinity desktop. Gnome and Mate are missing.

System performance with the default Trinity desktop is much like that of Intel-based hardware with two cores, but it is not necessarily suitable for use with the KDE SC desktop. The LXDE desktop was used for comparison.

The system boots as quickly as a full-grown computer system: During the test, Q4OS took about 14 seconds to start from a fast 8GB microSD card until the LXDE desktop was ready for use. However, it took about 40 seconds for the heavyweight KDE SC 4.14 environment to launch.

That said, the operating system memory requirements are likely to be far more important on the Raspberry Pi than on a desktop computer. The SBC only has 1GB of RAM and cannot be upgraded any further. Even without active applications, the KDE SC desktop occupied a total of around 800MB of RAM, a large part of which was used as temporary storage. The lean LXDE environment only needed about 140MB, and the Trinity interface about 400MB (Figure 10).

After installing some standard applications, the system ran far more slowly, even under LXDE, than the Intel systems: LibreOffice and Gimp had relatively long latency times at startup; Firefox 52 ESR took about 10 seconds to build the screen completely. High memory utilization by these typical office applications was noticeable, as well. Under the Trinity desktop, after calling LibreOffice, only about 150MB of RAM was free; in Firefox without any other open applications, this went down to just 80MB.

With the LXDE desktop, the system was more agile. After calling LibreOffice, around 500MB of RAM remained free, and Firefox noticeably reduced RAM resources in this scenario with more than 620MB.

Conclusions

The test series in this article shows that legacy hardware is not necessarily too slow for modern Linux systems. With a lean Linux distribution, older desktops and notebooks with single-core processors, as represented by the HP Compaq NC6220, are still suitable for simple office tasks or as file or proxy servers in a small intranet. The prerequisite is that the system in question has sufficient RAM and corresponding mass storage capacities.

Systems built in 2007, 2008, or later that use dual-core processors continue to be suitable for everyday use for simple and medium-duty tasks. Because these computers usually come with 2 or 4GB of RAM by default, they usually do not need to be upgraded.

Formerly very expensive professional workstations like the HP Z600, which the company exclusively equipped with premium components at the time, still perform demanding tasks today. These devices are even suitable for video transcoding and are sufficiently future-proof.

Older Xeon workstations keep up with current upper mid-range devices, not only because of their high processor performance, but also because of their versatile interfaces. Moreover, such systems are generally more fail-safe than entry-level consumer or business computers.

A modern and energy-saving Raspberry Pi is more suitable for simple office tasks on a very small budget. Sufficient processor power cannot compensate for the small amount of memory and the relatively slow mass storage subsystem. The SBC would only become serious competition for Intel systems if the manufacturer were to increase the RAM and provide mass storage with a fast SATA interface. ■■■

Info

[1] Q4OS: *https://q4os.org*

[2] HandBrake: *https://handbrake.fr*

[3] HP NC6220 laptop: *https://h10057.www1.hp.com/ecomcat/hpcatalog/specs/provisioner/05/ED664UC.htm*

[4] Trinity desktop: *http://www.trinitydesktop.org*

[5] Mate: *https://mate-desktop.org/*

[6] OWC SSD: *https://eshop.macsales.com/shop/SSD/OWC/Mercury_Legacy_Pro*

[7] SSE4: *https://en.wikipedia.org/wiki/Streaming_SIMD_Extensions*

[8] Fujitsu Siemens D9510 laptop: *http://support.harlander.com/uploads/treiberdatenbank/notebooks/fujitsu_siemens/esprimo_mobile_d9510/datenblatt.pdf*

[9] SATA standard: *https://en.wikipedia.org/wiki/Serial_ATA*

[10] HP Z600: *https://support.hp.com/us-en/product/hp-z600-workstation/3718663/manuals*

[11] Intel Xeon: *https://ark.intel.com/products/37103/Intel-Xeon-Processor-E5530-8M-Cache-2_40-GHz-5_86-GTs-Intel-QPI*

[12] DMS-59 graphics adapter: *https://en.wikipedia.org/wiki/DMS-59*

[13] HP Elitebook 2570p: *https://support.hp.com/us-en/product/hp-elitebook-2570p-notebook-pc/5259393/product-info*

[14] Raspberry Pi 3: *https://www.raspberrypi.org/magpi/raspberry-pi-3-specs-benchmarks/*

POWER MANAGEMENT
LINUX MAGAZINE
IoT AT HOME
VMware vs VirtualBox
BITKEY
SAFER BOOT
INSIDE A COMPILER
TERMINAL TUNING
IS FIREFOX BACK
FREE BOOT!
MYCROFT
LINUXVOICE

Configuring graphics tablets on Linux

xsetwacom

With a little extra effort, on-the-fly adjustment of your graphics tablet on Linux is possible with this command-line tool. *By Bruce Byfield*

At first glance, Linux support for graphics tablets looks like a throwback to 15 years ago, when hardware support meant adding kernel modules and manual file configuration. However, a little digging shows that support for graphics tablets is actually strong in modern Linux, although unorthodox compared with support on Windows or OS X. A major part of that support is the command-line application `xsetwacom` – although it has a quirk or two of its own to overcome [1].

Graphics tablets generally consist of a tablet and a stylus, which commonly has an eraser on one end. The tablet is a representation of the computer screen, and the stylus – or in some models, fingers – are a replacement for a mouse. This arrangement allows graphic artists to work online with more precision than a mouse permits. Additionally, the stylus places far less stress on the hand, which makes it an ergonomic alternative to a mouse. (Figure 1).

Part of the trouble with using a graphics tablet under Linux is that manufacturers, including Wacom [2], the most popular choice among users, do not directly support Linux. Instead, with Wacom's encouragement, support is provided by the Linux Wacom Project [3]. The Project has done heroic work over the years, but its site is geared to developers, making it easy for end users to overlook basic information such as the table of supported tablets [4]. Instead, search results are likely to lead users first to outdated instructions that describe how to download the necessary kernel modules [5], and, if required, add the necessary lines to `xorg.conf` [6] to enable the tablet, stylus, and mouse.

This information may still be needed in some cases, but its prominence in

Author

Bruce Byfield is a computer journalist and a freelance writer and editor specializing in free and open source software. In his addition to his writing projects, he also teaches live and e-learning courses. In his spare time, Bruce writes about Northwest coast art. You can read more of his work at *http://brucebyfield.wordpress.com*

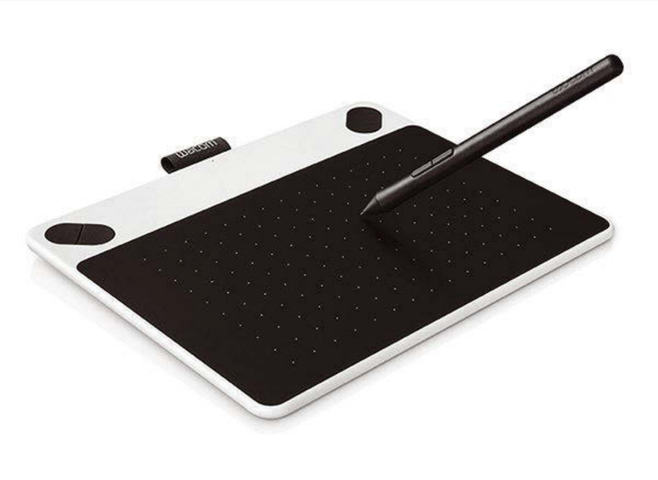
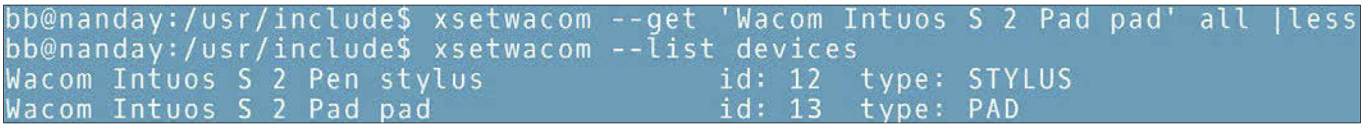


Figure 1: Much like all graphics tablets, the Intuos Draw, an entry-level graphics tablet, is useful for both onscreen drawing and as an ergonomic mouse replacement. Copyright © 2017-2018 Wacom

Lead Image © Vasyl Nesterov, 123RF.com



```
bb@nanday:/usr/include$ xsetwacom --get 'Wacom Intuos S 2 Pad pad' all |less
bb@nanday:/usr/include$ xsetwacom --list devices
Wacom Intuos S 2 Pen stylus             id: 12  type: STYLUS
Wacom Intuos S 2 Pad pad                id: 13  type: PAD
```

Figure 2: Supported tablets will list at least a tablet and a stylus; some will also include an eraser.

search engines obscures the fact that support for dozens of graphics tablets is compiled in the kernels of many major distributions, which means that, these days, installation is generally no more complicated than plugging in the tablet. Users do not even need to install drivers; yet, new users who follow the first search results that come up can spend hours doing unnecessary setup.

Compounding the problem are two other peculiarities of Linux. First, graphics applications like Gimp and Krita have tools that enhance working with tablets, but which to the uninitiated often appear to be required and somewhat arcane. Second, utilities that give a control dialog similar to the one that Wacom ships for Windows are not keeping pace with the Linux Wacom Project. Since new tablet models are added every few kernel releases, new users are often puzzled by the fact that the `lsusb` command clearly shows that their operating system supports their graphics tablets, yet the tools in Gnome and KDE fail to detect the tablet that they are using to test the tools.

Enter `xsetwacom`, a command for adjusting on-the-fly how a Wacom tablet or a compatible tablet interacts with the X Window System (so far, the command does not run on Wayland). With dozens of options, it offers a similar degree of configurability to that found in Wacom's desktop tools for Windows and OS X, but in a less well organized way.

Oh, and one more small detail to note: Options set with `xsetwacom` last only until you log out or unplug the tablet. The next time you log in or reconnect, `xsetwacom` options have to be entered again. This limitation is less of a problem than it first appears, but it is still a nuisance to unsuspecting users who simply want to get their work done.

Fine-Tuning a Tablet

To begin using `xsetwacom`, you should familiarize yourself with its `list` commands. To modify many settings, you will need the name of the tablet, stylus, and – if present – the eraser. Start with the command:

```
xsetwacom --list devices
```

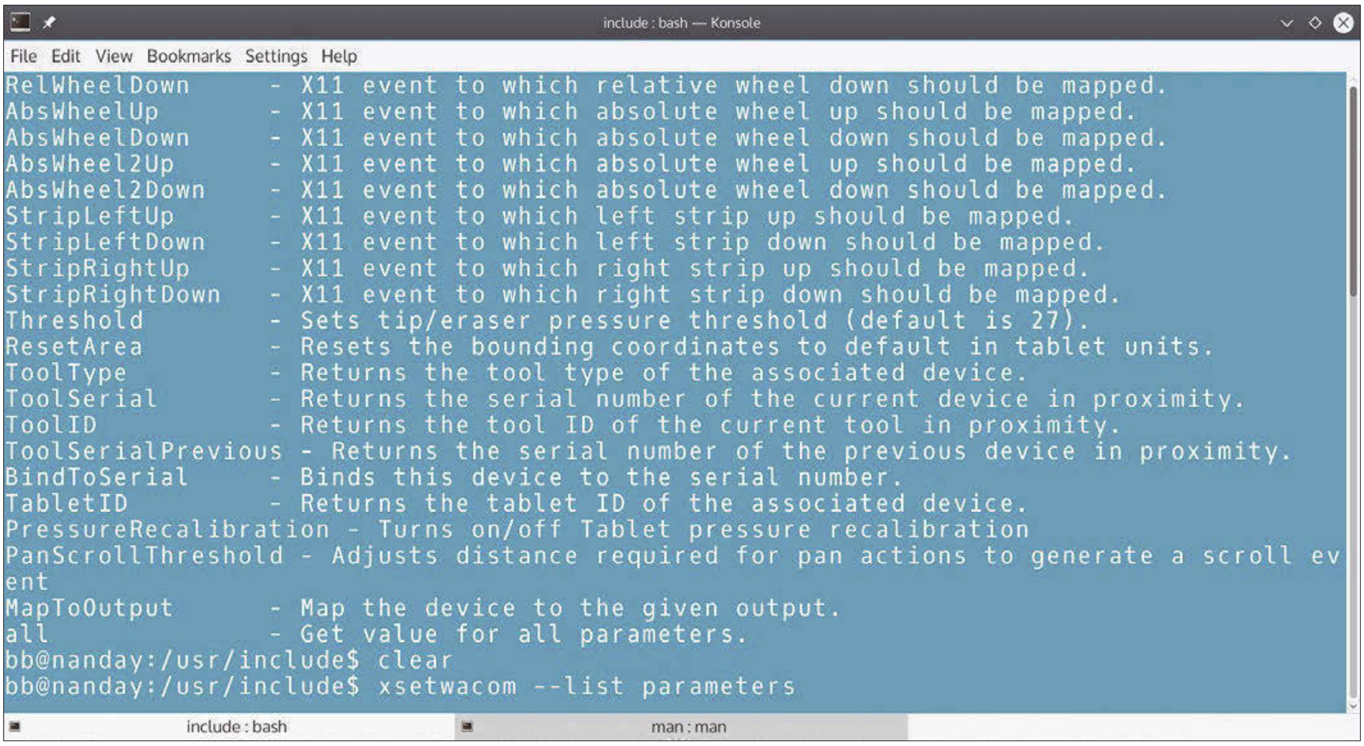
If the command detects the devices, then you are ready to use the tablet. This will also give you the device's proper names, which are needed to set options. Notice, too, that the tablet and stylus are separate devices. If the stylus includes an eraser, it will also be listed (Figure 2).

To see which parameters you can set, enter:

```
xsetwacom --list parameters
```

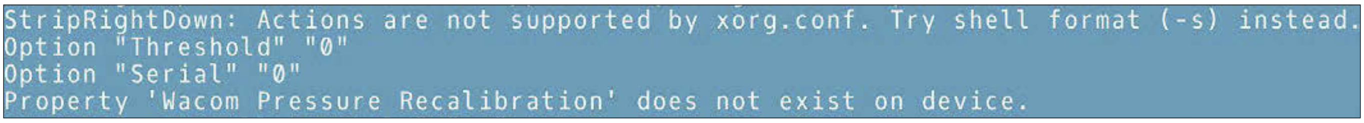
Parameters can be entered either using `xsetwacom`'s `--list option` or, at the cost of a loss of flexibility, by adding them to `/etc/X11/xorg.conf`. Depending on the device, some parameters may not be supported if entered in `xorg.conf`, but you can check how to input a command by adding the `-s` option for a list of parameters supported only from the shell or `-x` for a list of those that can be added to `xorg.conf`, as well as the proper format for them (Figure 3).

Generally, you will want to know what parameters can be added to your tablet in particular, so you will want to enter



```
RelWheelDown       - X11 event to which relative wheel down should be mapped.
AbsWheelUp         - X11 event to which absolute wheel up should be mapped.
AbsWheelDown       - X11 event to which absolute wheel down should be mapped.
AbsWheel2Up        - X11 event to which absolute wheel up should be mapped.
AbsWheel2Down      - X11 event to which absolute wheel down should be mapped.
StripLeftUp        - X11 event to which left strip up should be mapped.
StripLeftDown      - X11 event to which left strip down should be mapped.
StripRightUp       - X11 event to which right strip up should be mapped.
StripRightDown     - X11 event to which right strip down should be mapped.
Threshold          - Sets tip/eraser pressure threshold (default is 27).
ResetArea          - Resets the bounding coordinates to default in tablet units.
ToolType           - Returns the tool type of the associated device.
ToolSerial         - Returns the serial number of the current device in proximity.
ToolID             - Returns the tool ID of the current tool in proximity.
ToolSerialPrevious - Returns the serial number of the previous device in proximity.
BindToSerial       - Binds this device to the serial number.
TabletID           - Returns the tablet ID of the associated device.
PressureRecalibration - Turns on/off Tablet pressure recalibration
PanScrollThreshold - Adjusts distance required for pan actions to generate a scroll ev
ent
MapToOutput        - Map the device to the given output.
all                - Get value for all parameters.
bb@nanday:/usr/include$ clear
bb@nanday:/usr/include$ xsetwacom --list parameters
```

Figure 3: `xsetwacom` **supports dozens of parameters.**



```
StripRightDown: Actions are not supported by xorg.conf. Try shell format (-s) instead.
Option "Threshold" "0"
Option "Serial" "0"
Property 'Wacom Pressure Recalibration' does not exist on device.
```

Figure 4: Note how some parameters supported by `xsetwacom` are not available for this device, and some cannot be used in `xorg.conf`.

```
xsetwacom --get 'DEVICE-NAME' all
```

which can also take the `-s` or `-x` option. All of these options return dozens of lines of results, so pipe through `less` so you can read them easily. You will find more details about individual parameters in the command's man pages (Figure 4).

Study the parameters available for your device. When you are ready to add them, use the following command structure:

```
xsetwacom set 'DEVICE-NAME' ↵
  PARAMETER VALUE
```

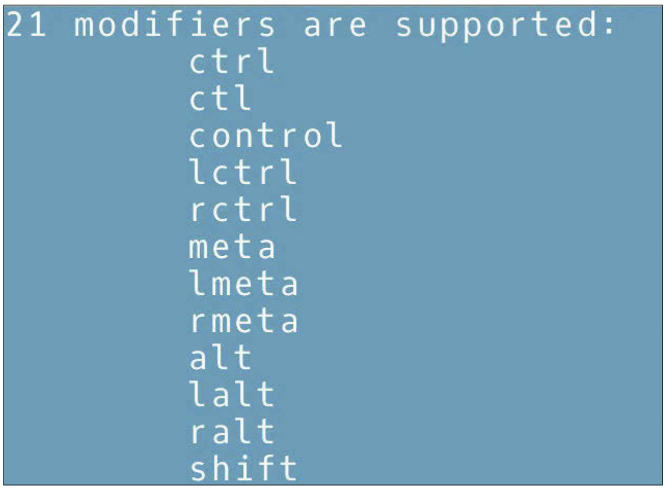
Except for errors, no feedback is provided unless you add the `-v` option.

Detailing all available parameters for all devices would require a small book. However, the most basic ones are those that affect general behavior. Almost all graphics tablets support switching the `Mode` parameter, toggling between `Absolute`, in which clicking with the stylus on the tablet automatically moves the cursor on the screen to a different position, and `Relative`, in which it does not. Every tablet I have tried is set by default to `Absolute`, where a point on the tablet corresponds to a point on the screen, which is easier for beginners to learn. However, more experienced uses often prefer `Relative`.

Another frequently used parameter is `Area`, which maps the tablet to the screen, using the values `x1`, `y1`, `x2`, and `y2` to define the active area of the tablet by its corners. Some users may choose to shrink the active area of the tablet so that the cursor moves further with the same movement. Those with a square monitor, or some other unusually shaped screen, might also want to redefine `Area`, especially for use in `Absolute` mode. If your monitor rotates, an alternative to `Area` may be `Rotate`. Values to complete Rotate are `none`, `half`, `cw` (clockwise), and `ccw` (counterclockwise).

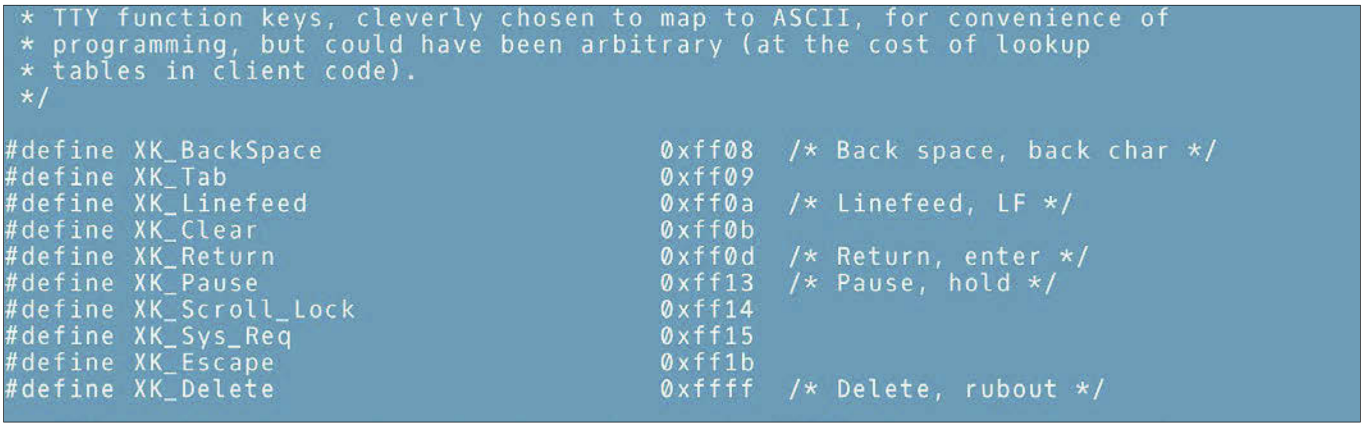
You might also want to toggle other functionality. For example, `Touch` can turn support for using fingers as a stylus on and off, and `Gestures` can do the same for controlling the position of the cursor on the screen with a small array of movements, rather like those that were used over a decade ago on Palm devices.

Other parameters set finer details of how the tablet operates. `ScrollDistance`, for instance, sets how far the stylus moves up or down when dragging on a window's scrollbar. The pressure needed to activate both the stylus and any eraser can be set with `Threshold`. Similarly, `TapTime` sets the fastest speed for activating a link or hotspot and is usually set to 250 milliseconds. And, because a stylus can be used without actually touching the screen, its `CursorProximity` (which is presumably set in millimeters, although the man page does not specify) can also affect performance. The Intuos line of Wacom tablets, whose lower end includes entry-level models, can operate as high a 105mm above the tablet, whereas the slightly more expensive Graphire line has a default of 42mm. Presumably,



```
21 modifiers are supported:
        ctrl
        ctl
        control
        lctrl
        rctrl
        meta
        lmeta
        rmeta
        alt
        lalt
        ralt
        shift
```

Figure 5: A partial list of supported modifiers for buttons.



```
 * TTY function keys, cleverly chosen to map to ASCII, for convenience of
 * programming, but could have been arbitrary (at the cost of lookup
 * tables in client code).
 */

#define XK_BackSpace                     0xff08  /* Back space, back char */
#define XK_Tab                           0xff09
#define XK_Linefeed                      0xff0a  /* Linefeed, LF */
#define XK_Clear                         0xff0b
#define XK_Return                        0xff0d  /* Return, enter */
#define XK_Pause                         0xff13  /* Pause, hold */
#define XK_Scroll_Lock                   0xff14
#define XK_Sys_Req                       0xff15
#define XK_Escape                        0xff1b
#define XK_Delete                        0xffff  /* Delete, rubout */
```

Figure 6: Part of the complete list of key symbols that can be used for modifying tablet buttons is available in `/usr/include/X11/keysymdef.h`.

there is an upper limit to the setting, although it is undocumented.

However, by far the most popular parameter is the settings for the tablet or stylus buttons. These are used to write macros, using a set of modifiers that include command keys such as Ctrl and Alt, function and navigation tools, and special characters (Figure 5).

A list of the more common modifiers is given when you enter the command:

```
xsetwacom --list modifiers
```

You can also add modifiers not listed by `--list modifiers` using the key symbols in `/usr/include/X11/keysymdef.h`. The `XK_` prefix to each key symbol listed in `keysymdef.h` should be removed when entering the symbol in `xsetwacom` (Figure 6).

Some tablet and stylus buttons may be programmed by default to emulate left and right mouse clicks or a scroll wheel. All these functions can be remapped to other buttons. In particular, stylus buttons can be awkward to use, so their default functionality might be shifted to the tablet buttons. Using modifiers, you can also program buttons for basic macros in commonly used applications, such as Cut, Copy, Paste, Save, Save as, Undo, and Redo, especially on higher end tablets, which usually have more buttons. Another common use for the buttons is adding special characters such as the pound or euro symbol, or for setting up an AltGr key for typing a whole array of diacritical marks. Some of these same shortcuts can be added to a keyboard, especially a programmable one, so how the tablet and stylus buttons are used can require some planning. In my experience, determining what to do with the array of available options is something to decide over time, as you gain experience and confidence with the tablet.

In all events, the command structure for programming buttons is:

```
xsetwacom --set 'DEVICE' BUTTON-ID ➘
  'key MODIFIER+MODIFIER'
```

On an Intuos Draw tablet, for example, to assign the Undo function to the first button on the tablet, the modifier at the end of the command would be:

```
xsetwacom ➘
  --set 'Wacom Intuos S 2 Pad pad' ➘
  Button 1 'key Undo'
```

Pursuing Permanence

The fact that `xsetwacom` settings are temporary is often less of a concern than they might seem. Most of its options are not needed to use a graphics tablet as a mouse, and settings that affect drawing capabilities are retained in graphics applications like Gimp, Inkscape, or Krita – although settings do have to be enabled separately in each application.

Moreover, an obvious solution exists. Users can create a script that contains all their commonly used `xsetwacom` settings and either run it after they plug in their tablet, or, better yet, set it to run as they log in to their desktop environments. Alternatively, an `.xsessionrc` file can be created, and the settings placed in it. If users have other groups of options that they only need occasionally, they can create separate scripts to start those as well.

If someone asks if Linux supports graphics tablets, you can answer that basic support is built-in. Advanced options, though, require some extra effort, although usually not the module compiling and installation that an Internet search might suggest. True, those efforts are more than a modern Linux user expects, but if you are tempted to complain, just remember: Not too many years ago, this amount of extra effort or worse was required for printers, scanners, and even the mouse. If the current state of support is often less than convenient, it is still considerably better than nothing. ■■■

Info

[1] xsetwacom: *https://linux.die.net/man/1/xsetwacom*

[2] Wacom: *https://wacom.com/en-ca*

[3] Linux Wacom Project: *https://linuxwacom.github.io/*

[4] Supported tablets: *https://github.com/linuxwacom/input-wacom/wiki/Device-IDs*

[5] Installing kernel modules: *https://github.com/linuxwacom/xf86-input-wacom/wiki/Building-The-Driver*

[6] xorg.conf configuration: *https://github.com/linuxwacom/xf86-input-wacom/blob/master/conf/70-wacom.conf*

Python generators simulate gambling

Strange Coincidence

Can 10 heads in a row really occur in a coin toss? Or, can the lucky numbers in the lottery be 1, 2, 3, 4, 5, 6? We investigate the law of large numbers. By Mike Schilli

If you follow the stock market, you are likely to hear people boasting that their buy and sell strategies allegedly only generate profits and never losses. Anyone who has read *A Random Walk down Wall Street* [1] knows that wild stock market speculation only leads to player bankruptcy in the long term. But what about the few fund managers who have managed their deposits for 20 years in such a way that they actually score profits year after year?

According to *A Random Walk down Wall Street*, even in a coin toss competition with enough players, there would be a handful of "masterly" throwers who, to the sheer amazement of all other players, seemingly throw heads 20 times in a row without tails ever showing up. But how likely is that in reality?

Heads or Tails?

It is easy to calculate that the probability of the same side of the coin turning up twice in succession is 0.5x0.5 (i.e., 0.25), since the probability of a specific side showing once is 0.5 and the likelihood of two independent events coinciding works out to be the multiplication of the two individual probabilities. The odds on a hat trick (three successes in a row) are therefore 0.5^3 = 0.125, and a series of 20 is quite unlikely at 0.5^20 = 0.000000954. But still, if a program only tries often enough, it will happen sometime, and this is what I am testing for today. The short Python `coin_throw()` generator in Listing 1 [2] simulates coin tosses with a 50 percent probability of heads or tails coming up.

Listing 1 is divided into the `coin_throw()` coin toss generator and the experiment evaluation in the `experiment()` function. In a `for` loop as embedded in line 16, it looks like `coin_throw()` would return a long list of result strings consisting of `heads` or `tails`, but in reality, the function is a dynamic generator that returns a new value whenever an iteration pump like the `for` loop asks if there is more coming.

In the Generator House

How does the generator, whose output is shown in Figure 1, work? The key is the `yield` command in line 8 of Listing 1. Python interrupts the execution of the current function for a `yield`, remembers its internal state, and returns the string (`heads` or `tails`) to the calling program in line 8. The `randint(0,1)` method returns the integer value `0` or 1 with 50 percent probability, and the script picks one of the two entries in the `sides` tuple.

The next time the `coin_throw()`function is called, the program flow returns to the previous location within the function to continue where `yield` left off. In this case, this is the endless loop with `while True` in line 7, whose condition is still true, whereupon another `yield` command again returns a random value. Thus the generator produces an endless sequence of values in `coin_throw()` and always performs a new coin toss if the `for` loop in the main program wants more.

The `experiment()` function's remaining code stores the number of identical coin toss outcomes in sequence so far in the variable `run`, the maximum longest sequence so far in `max_run`, the previous coin toss in `prev` (`heads` or `tails`), and the total number of tosses so far in `count`.

Fatal Double Down

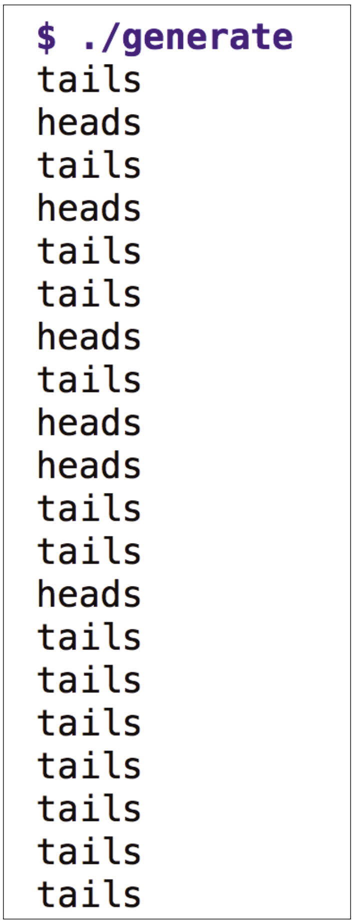
The output from the script in Figure 2 reveals that after 21 million passes, a sequence of 23 tails in succession actually occurred. If a player had doubled the bet for the next game each time they lost, a strategy called "doubling down," they would have needed a total of

Author

Mike Schilli works as a software engineer in the San Francisco Bay area, California. Each month in his column, which has been running since 1997, he researches practical applications of various programming languages. If you email him at *mschilli@perlmeister.com* he will gladly answer any questions.

Listing 1: cointhrow

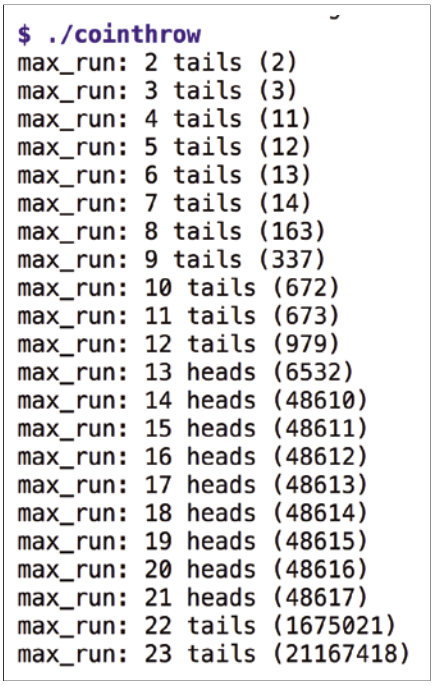
```
#!/usr/bin/env python3
import random

def coin_throw():
  sides=('heads', 'tails')

  while True:
    yield sides[random.randint(0, 1)]

def experiment():
  run     = 1
  max_run = 0
  prev    = ""
  count   = 0

  for side in coin_throw():
    count += 1

    if prev == side:
      run += 1
      if run > max_run:
        max_run = run
        print("max_run: %d %s (%d)" %
              (max_run, side, count))
    else:
      prev = side
      run  = 1

experiment()
```



```
$ ./generate
tails
heads
tails
heads
tails
tails
heads
tails
heads
heads
tails
tails
heads
tails
tails
tails
tails
tails
tails
tails
```

Figure 1: Python generator coin toss.

2^23 – 1 = $8,388,607 of betting capital, assuming an initial bet of $1, not to go bankrupt if they had initially bet on heads, only to see 23 tails in a row with growing frustration.



```
$ ./cointhrow
max_run: 2 tails (2)
max_run: 3 tails (3)
max_run: 4 tails (11)
max_run: 5 tails (12)
max_run: 6 tails (13)
max_run: 7 tails (14)
max_run: 8 tails (163)
max_run: 9 tails (337)
max_run: 10 tails (672)
max_run: 11 tails (673)
max_run: 12 tails (979)
max_run: 13 heads (6532)
max_run: 14 heads (48610)
max_run: 15 heads (48611)
max_run: 16 heads (48612)
max_run: 17 heads (48613)
max_run: 18 heads (48614)
max_run: 19 heads (48615)
max_run: 20 heads (48616)
max_run: 21 heads (48617)
max_run: 22 tails (1675021)
max_run: 23 tails (21167418)
```

Figure 2: Longest sequences of identical coin tosses.

Great Stuff

Python not only offers generators with the `yield` keyword; classes can also implement generators as iterators. For this the Pythonista defines two methods: `__iter__()` and `__next__()`. The quadruple underscores ("dunders") mark the official entry points for Python's standard library. If the Python interpreter sees a loop head like `for n in Roulette()`, it instantiates an object of the `Roulette` class, uses `__iter__()` to access its iterator, and then retrieves new values from it with calls to `__next__()` until the iterator throws an exception.

In the class defined in Listing 2, `__iter__()` conveniently returns the `Roulette` instance itself, because the class does not need a separate iterator, since it implements the iterator itself with `__next__()`. The latter always returns a new random number in the range 0 to 36 and never throws an exception, so that the flow of the `for` loop in the main program never stops.

The `Roulette` class defines two class variables: `slots` as the highest number on the `roulette` wheel and `numbers` as a sequence of numbers from 0 to 36 inclusive. It makes sense to define the variables once only for the class and not to rebuild them for each instance or even every time the iterator is called.

Class or Instance?

Python's class variables differ from instance variables in that they are not accessed with `self.variable`, but with `__class__.variable` or `self.__class__.variable`. For read-only access, you could even reference the class variable using the `self.variable` instance path.

But if you modify the latter, you may be in for a surprise, because Python creates a new instance variable behind your back. The instance variable will be decoupled from the class variable so that each object will modify its own, instead of propagating changes to the class level. Also, methods do not find the class variable simply by its name; if you simply reference `slots` in `__iter__()`, you can

Listing 2: roulette.py

```
#!/usr/bin/env python3
import random

class Roulette:
  slots   = 36
  numbers = range(0,slots+1)

  def __iter__(self):
    return self

  def __next__(self):
    return self.__class__.numbers[random.randint(0, self.__class__.slots)]
```

expect the syntax checker to blow up in your face.

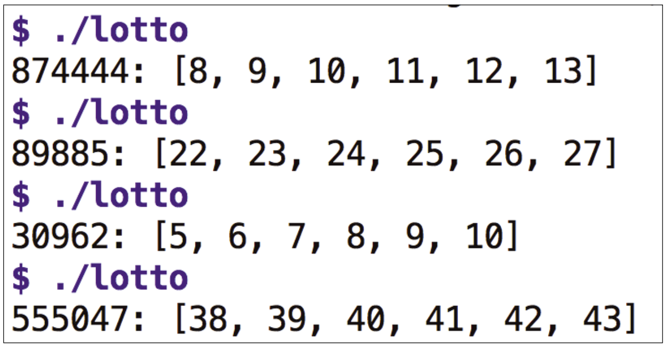
As so often in the Python world, there is a small but subtle difference between versions 2 and 3: The iterator entry into the generator class goes by the name of `next` in Python 2 and not `__next__` as in Python 3; programmers who want to use both versions thus typically simply define another `next()` method, which passes the parameters fed to it to `__next__()` without modification. Python 3 does not use `next()`, so the compatibility trick does no harm there.

Listing 3: roulette-run

```
$ ./roulette
max_run: 2 11 (27)
max_run: 3 15 (6249)
max_run: 4 34 (57393)
max_run: 5 1 (3363284)
max_run: 6 0 (95846456)
max_run: 7 26 (357289507)
```

Listing 4: lotto

```
01 #!/usr/bin/env python3
02 import random
03
04 def lotto_draw():
05   total   = 49
06   draws   = 6
07   numbers = list(range(1,total+1))
08   size    = total
09   result  = []
10
11   for _ in range(draws):
12     idx = random.randrange(size)
13     result.append(numbers[idx])
14     numbers[idx] = numbers[size-1]
15     size -= 1
16
17   return sorted(result)
18
19 def is_consecutive(draw):
20   prev    = ""
21   for number in draw:
22     if prev < 0:
23       prev=number
24     elif prev + 1 == number:
25       prev = number
26     else:
27       return False
28   return True
29
30 count = 0
31 while True:
32   count += 1
33   draw=lotto_draw()
34   if is_consecutive(draw):
35       print("%d: %s" % (
              count, str(draw)))
36       break
```



```
$ ./lotto
874444: [8, 9, 10, 11, 12, 13]
$ ./lotto
89885: [22, 23, 24, 25, 26, 27]
$ ./lotto
30962: [5, 6, 7, 8, 9, 10]
$ ./lotto
555047: [38, 39, 40, 41, 42, 43]
```

Figure 3: The lottery generator determines the number of draws until it finds a curious outcome.

The output of the statistical evaluation of the roulette generator is shown in Listing 3. After 27 rounds, a doublet appeared for the first time: the number 11 occurred twice in a row. After 6,249 rounds of *Faites vos jeux*, 15 occurred three times in a row; after 57,393 games, 34 occurred four times in a row, and so on.

Tumultuous Scenes

What would happen in Las Vegas at a roulette table if zero came up come six times in a row as in Listing 3 after 95 million rounds? Tumultuous scenes would probably take place in the casino before the pit boss appeared and sent the croupier home for the day, because every player at the table would immediately suspect that something fishy was going on. But, seen statistically, everything is above board; it's an inevitable fact that even random values will repeat at some time.

Lottery Winner

What would TV viewers think if the lottery lady announced that the numbers drawn from a bucket with 49 balls were 1, 2, 3, 4, 5, and 6? Since the probability of getting all six numbers right is about 1:14 million and there are 44 combinations of consecutive lottery number combinations (1, 2, 3, 4, 5, 6 through 44, 45, 46, 47, 48, 49), the chance of a straight in the lottery is about 1:318,000 – this means that the incredible event would occur relatively quickly with a fast draw generator.

Listing 4 shows an automatic drawing machine in the `lotto_draw()` function. From 49 numbered balls in the `numbers` list, it draws six random numbers and then removes them to prevent double draws. Since it takes a significant amount of compute time to remove an element from a Python list and move up the remaining elements to close the gap, the function swaps the value of the selected element with the last element in the list and reduces the list length size by one – much faster!

Following this algorithm, `lotto_draw()` returns a sorted list of six randomly selected balls. The main program starting in line 30 uses `is_consecutive()` to check whether the drawn numbers each differ only by one from their predecessor. If this is the case, line 35 prints the number of draws in `count` and the lucky numbers that led to the termination. Figure 3 shows that this sometimes occurs after 30,000 passes; sometimes, however, it takes more than 800,000 – purely random, but within the calculated probability.

Python Tricks [3] by Dan Bader is recommended for implementing this and other cool Python tricks. It shows a multitude of everyday programming tasks with elegant Python solutions. It is perfectly suited for users of other programming languages (like Perl!) who are mainly interested in converting typical idioms into clean Python and don't want to start with Adam and Eve and "Hello World." ■■■

Info

[1] Malkiel, Burton G. *A Random Walk down Wall Street*. Norton & Company, 2016: *https://www.amazon.com/Random-Walk-Down-Wall-Street-ebook/dp/B00QH9NTSI*

[2] Listings for this article: *ftp://ftp.linux-magazine.com/pub/listings/linux-magazine.com/212/*

[3] Bader, Dan. *Python Tricks*. Dan Bader, 2017: *https://dbader.org/products/python-tricks-book/*

7:05 AM
ADMIN
iPad
Backup and replicate VMs on VMware ESXi
Clone Freedom
Setting Up the Source
Target Host and S
Installation
Tech News
8
9
10
11
12
13
14
15

FREE DVD
Detecting Threats with Apache Spot
CLOUD CAPACITY PLANNING
ADMIN
Network & Security
FREE DVD INSIDE
Cloud Capacity Planning
How much cloud do you really need?
DOCKER SECURITY
Why a container isn't as safe as a VM
Best Practices for Securing Active Directory
Jenkins
Automated orchestration with continuous logging
Scout2
Security auditing for AWS environments
Jekyll
Fast and ligh HTML engine
Group Polic
Integrating s with AD logo
WWW.ADMIN-MAGAZINE.COM

FREE DVD
PIRATEBOX
Handy anonymous file server for parties and meetups
MYCROFT
An open source personal assistant
LINUX
MAGAZINE
Double-Sided DVD INSIDE!
MYCROFT
Can an open source personal assistant compete with Alexa?
Scanners and Linux
Peppermint OS
Peppy distro built for the cloud
Meltdown & Spectre
Kernel developers take swift action to protect Linux
Knight's Tour
Classic chess puzzle gets a Python twist
Node-RED
Control a Rasp Pi using text messages
LINUXVOICE
WWW.LINUX-MAGAZINE.COM

The sys admin's daily grind: systemd-analyze

Startup Scenes

In sys admin columnist Charly's case, no fumbled system startup goes undetected. This was already the case with SysVinit and has not changed with systemd. In terms of the his analysis tools, no stopwatch goes unturned. By Charly Kühnast

If a distribution shows its splash screen on startup – and almost all of them do – then I turn it off as of the second boot. I want to see the kernel messages rushing by when booting. Because more often than you might think, there's a problem with booting. When I see something questionable scurry by, I search with a tool like Bootchart [1] to find and fix a hanging process or similar issues quickly.

On systems with systemd, this is also possible in principle, but the tool is different: `systemd-analyze`. When called without parameters, `systemd-analyze` calculates the elapsed time until system startup is completed, broken down into kernel and userspace processes:

```
Startup finished in 3.507s (kernel) + ⮐
  16.334s (userspace) = 19.842s
```

Of course, I want to know more about those processes now. The command `systemd-analyze blame` gives me a list of all processes started at boot time. They are sorted by the elapsed time before the process has fully launched. I usually only see the 10 biggest time hogs, as shown in Listing 1.

Listing 1: systemd-analyze blame | head

```
1min 3.753s apt-daily.service
    40.702s apt-daily-upgrade.service
     8.649s fail2ban.service
     4.246s networking.service
     3.327s smokeping.service
     2.622s apache2.service
     2.303s mysql.service
     2.206s postfix.service
     1.835s dev-sdb1.device
      993ms munin-node.service
```

In this particular case, I can ignore the candidates in lines 1 and 2 – they appear if the system performs unattended updates or upgrades on startup. Fortunately, they do not block any other processes and simply wriggle around a bit in the background. `systemd-analyze` does not calculate them for this reason, as can already be seen from the sums.

Serial Plot

With the `plot` parameter, I can create a kind of timeline in SVG format, which shows the boot process in a very clear graphical way. For this to work, the *graphwiz* package must be installed. The command is then:

```
systemd-analyze plot > systemd-boot.svg
```

Figure 1 shows a small section of the results. The start time of the individual processes is symbolized here by a red bar. Not only the start order and times, but all the dependencies between the services, can be summarized by the tool in the form of an SVG graphic:

```
systemd-analyze dot | dot -Tsvg > ⮐
  dependencies.svg
```

Caution: The result is really huge and probably only makes life easier for owners of A1-sized plotters. ■■■

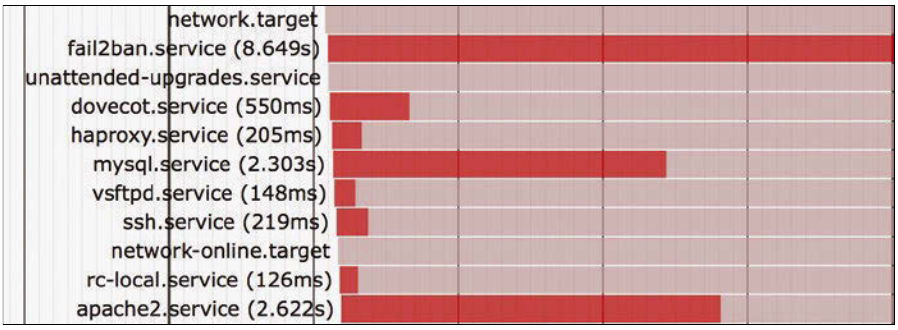


Figure 1: An SVG format timeline showing each service's system start times.

Info

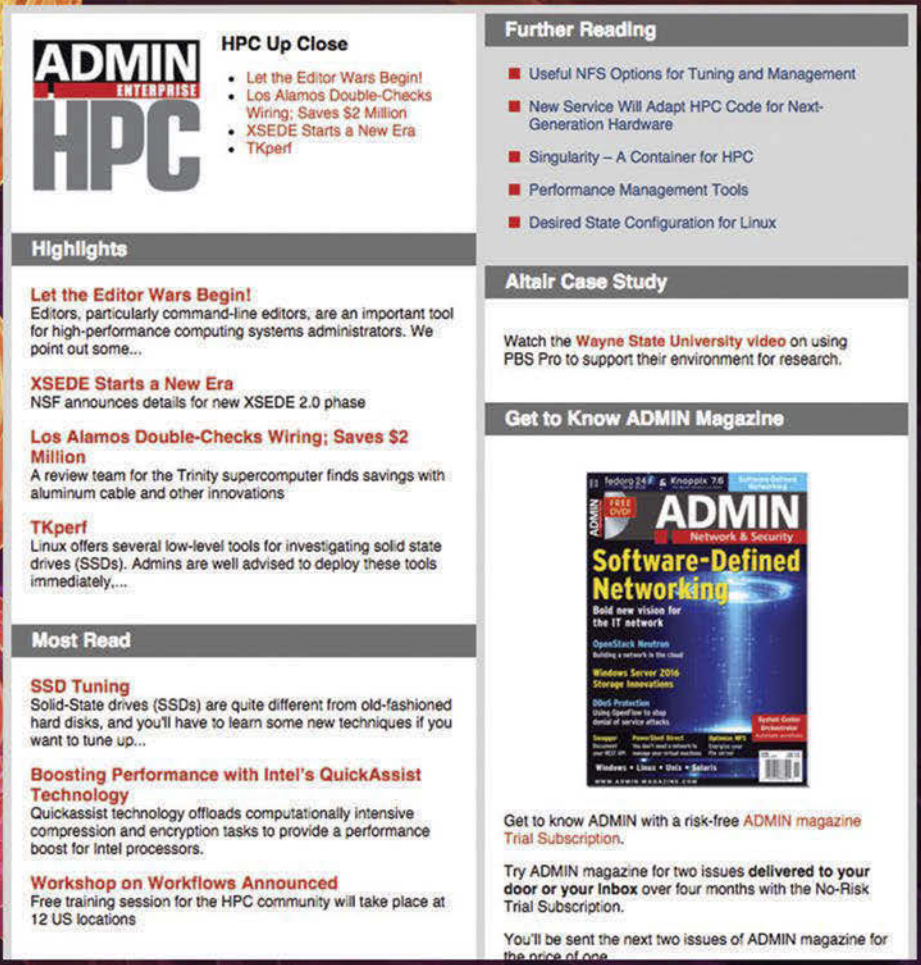
[1] Bootchart: *http://www.bootchart.org*

Author

Charly Kühnast manages Unix systems in the data center in the Lower Rhine region of Germany. His responsibilities include ensuring the security and availability of firewalls and the DMZ.

GOT CLUSTER?

Tune in to the HPC Update newsletter for news, views, and real-world technical articles on high-performance computing.



HPC Up Close

- Let the Editor Wars Begin!
- Los Alamos Double-Checks Wiring; Saves $2 Million
- XSEDE Starts a New Era
- TKperf

Further Reading

- Useful NFS Options for Tuning and Management
- New Service Will Adapt HPC Code for Next-Generation Hardware
- Singularity – A Container for HPC
- Performance Management Tools
- Desired State Configuration for Linux

Highlights

Let the Editor Wars Begin!
Editors, particularly command-line editors, are an important tool for high-performance computing systems administrators. We point out some...

XSEDE Starts a New Era
NSF announces details for new XSEDE 2.0 phase

Los Alamos Double-Checks Wiring; Saves $2 Million
A review team for the Trinity supercomputer finds savings with aluminum cable and other innovations

TKperf
Linux offers several low-level tools for investigating solid state drives (SSDs). Admins are well advised to deploy these tools immediately,...

Altair Case Study

Watch the Wayne State University video on using PBS Pro to support their environment for research.

Get to Know ADMIN Magazine

fedora 24
Knoppix 7.6
FREE DVD!
ADMIN
Network & Security
Software-Defined Networking
Bold new vision for the IT network
OpenStack Neutron
Windows Server 2016 Storage Innovations
DDoS Protection
Using OpenFlow to stop denial of service attacks
Windows • Linux • Unix • Solaris

Get to know ADMIN with a risk-free ADMIN magazine Trial Subscription.

Try ADMIN magazine for two issues **delivered to your door or your Inbox** over four months with the No-Risk Trial Subscription.

You'll be sent the next two issues of ADMIN magazine for the price of one.

Most Read

SSD Tuning
Solid-State drives (SSDs) are quite different from old-fashioned hard disks, and you'll have to learn some new techniques if you want to tune up...

Boosting Performance with Intel's QuickAssist Technology
Quickassist technology offloads computationally intensive compression and encryption tasks to provide a performance boost for Intel processors.

Workshop on Workflows Announced
Free training session for the HPC community will take place at 12 US locations

hpc.admin-magazine.com/Newsletter

MakerSpace

Raspberry Pi 3 Model B+ overview

Powerhouse

The new Raspberry Pi 3 B+ stands out from its predecessor in many respects. Our tests show what the new model can do.

By Maximilian Batz

Since its launch 2012, the Raspberry Pi has sold more than 19 million units, 9 million of them as the Raspberry Pi 3 (RPi3). On Pi Day, March 14, 2018, the Raspberry Pi Foundation (the Foundation) took the opportunity to upgrade the popular model in the form of the RPi3 Model B+ (RPi3B+; Figure 1), thus finally fulfilling some of the community's long-cherished wishes: Gigabit Ethernet with a maximum of 300Mbps throughput, power over Ethernet (PoE) support, and an improved preboot execution environment (PXE) boot. The real highlight, however, is the new wireless 802.11ac module (see the "Precertified" box).

Precertified

For small-board computer (SBC) hobbyists, the RPi3B+ precertified wireless module is a highly interesting innovation to enterprises, startups, and the business world for integration into Compute Module 3 projects. Effective shielding has made it possible to certify the element as a modular solution. Thanks to this certification by the Foundation, which invested an enormous amount of time and money in the process, corporations can now launch wireless- and Bluetooth-capable Rasp Pi-based products for the networked Internet of Things (IoT) world, and it will be 10 times cheaper and faster than before, according to Eben Upton.

The new RPi3B+ [1], like the Pi Zero W (wireless + Bluetooth) [2], was designed by Roger Thornton. The new model uses the same CPU as its predecessor, but now has a clock speed of 1.4GHz and features significantly improved heat management.

The form factor has also remained the same: All connections, sockets, and LEDs are in the usual place. The RAM is still 1GB, because the VideoCore IV processor – the GPU and video core of the Raspberry Pi in all previous models starting with the RPi1B – only supports a maximum 1GB of RAM.

The new Rasp Pi appears to be compatible with existing cases (I tested it successfully with a Tek-Berry3), HATs, and other accessories.

BCM2837B0

The heart of the RPi3B+, the BCM2837B0 system-on-a-chip (SoC), has a metal "heat spreader" that dissipates heat better than in previous models. Internally, the BCM2837 did not change sufficiently to justify a new model number: The core is a 64-bit quad-core ARMv8-A Cortex-A53. However, the integrated heat spreader and some other optimizations allow the clock frequency of the cores to be increased to 1.4GHz – around 16 percent

Lead Image © Shamain, 123RF.com

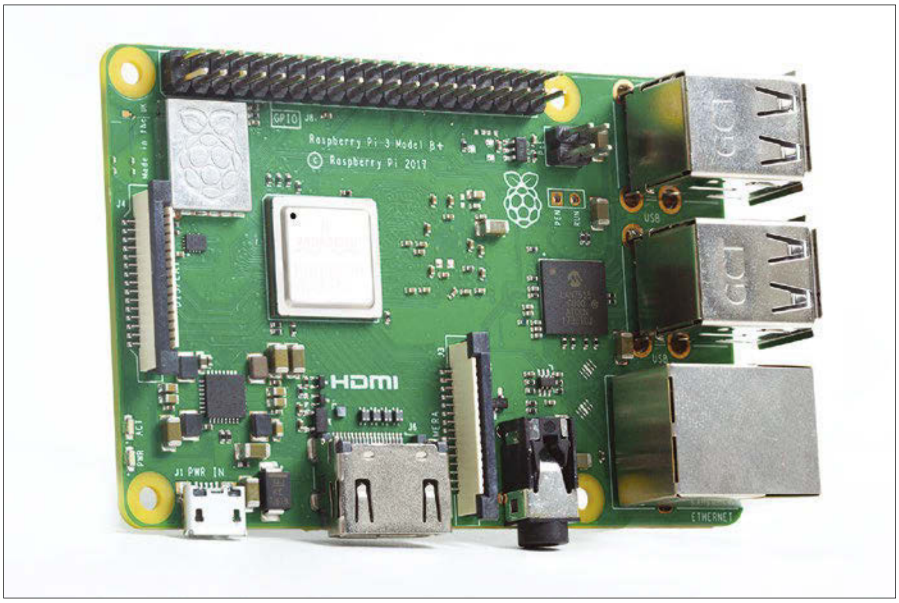


Figure 1: The new RPi3B+ meets some of the community's long-cherished needs, including faster wireless and Gigabit Ethernet (figure courtesy of the Raspberry Pi Foundation).

more than the RPi3B. The real highlight is that the hardware can consistently maintain this performance under load significantly longer than in earlier models. The Foundation has completely redesigned the computer's power supply.

Thanks to the MxL7704 power management integrated circuit (PMIC), the board looks tidier by combining a number of previously discrete power supplies. The MxL7704 provides the various voltages required by different components, with significantly higher stability and the possibility of more modular adjustments. For example, it can supply the core with slightly less or slightly more voltage to influence performance and heat dissipation.

Below 70°C internal temperature, the BCM2837B0 raises the CPU frequency to 1.4GHz; above this temperature, it drops to 1.2GHz. The MxL7704 reduces the core voltage in parallel, thus running the CPU at high performance for as long as possible, up to a maximum of 80°C. However, the CPU never typically reaches this threshold, so it will never slow down.

Of course, all of these features come with a price: The RPi3B+ draws significantly more power than its predecessor. The designers therefore strongly recommend a high-quality 2.5A power supply. The official Foundation power supply [3] keeps you on the safe side.

Gigabit Ethernet

The well-known bottleneck, the single USB 2.0 bus with a 480Mbps gross data rate, continues to exist in the RPi3B+. However, the Microchip Technology LAN7515 extracts the maximum benefit. In addition to the four USB ports, you get a GigE LAN connection with 300Mbps real throughput, which is three times more than its predecessor. In everyday life (e.g., when streaming video from a USB hard drive), the data rate is lower because all data runs through the SoC and the single USB bus.

At least PXE boot works better with the new LAN port. All known problems of the RPi3B with this remote boot procedure were fixed by developer Gordon Hollingworth in the boot ROM of the BCM2837B0.

A forthcoming PoE Hardware Attached on Top (HAT) expansion board (Figure 2) lowers the 48V applied to PoE, in addition to data on the wire, to the 5V required by the Rasp Pi. The RPi3B+ routes the wires from the LAN socket for HATs via a four-pin header.

In the future, thanks to PoE and PXE, an RPi3B+ can be connected by LAN cable and started without an additional SD card or power supply. A fan on the PoE HAT ensures that the SBC does not overheat.

Wireless and Bluetooth

Hardware offloads AES and TKIP encryption, relieving the load on the Cypress CYW43455 single-chip 802.11ac (formerly a BCM43455) WiFi chipset when encrypting wireless network traffic. The CYW43455 is fully backward compatible with IEEE 802.11a/b/g/n and transmits in the 5 and 2.4GHz bands.

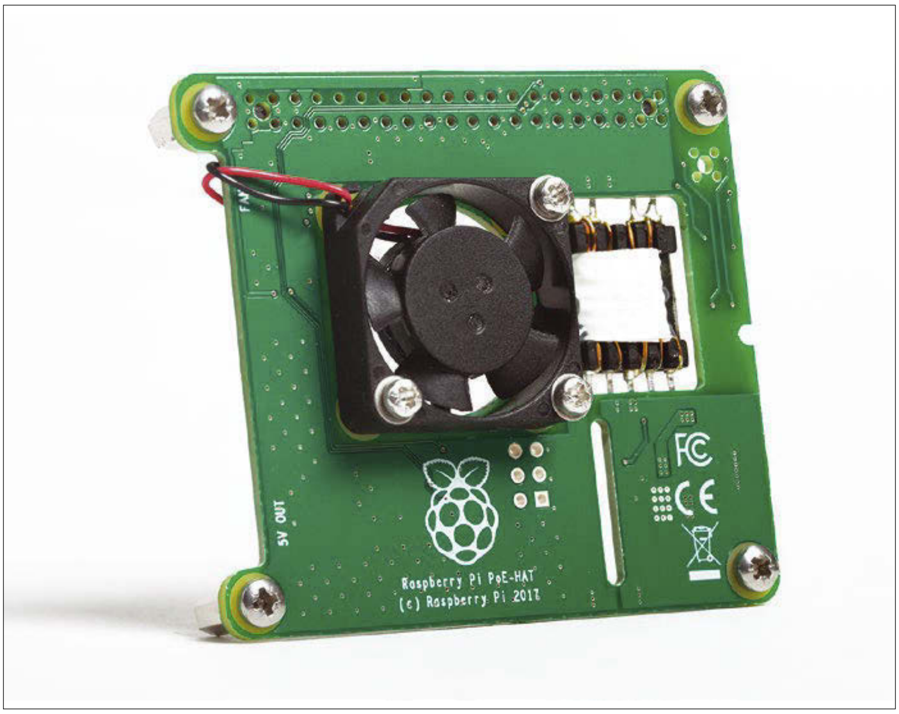


Figure 2: Using a HAT, you can supply the RPi3B+ with data and additional power, if required, via Ethernet (figure courtesy of the Raspberry Pi Foundation).

Table 1: RPi3B+ vs. RPi3B Test Results

	RPi3B	RPi3B+	Comment
Specifications			
SoC	Broadcom BCM2837	Broadcom BCM2837B0	Quad-core Cortex-A53
CPU	Cortex-A53 at 1.2GHz	Cortex-A53 at 1.4GHz	RPi3B+ tested at 1.2GHz
RAM	1024MB LPDDR2	1024MB LPDDR2	–
LAN	10/100Mbps	1,000Mbps	USB 2.0
Power	5V, 2.5A	5V, 2.5A	Micro-USB
System			
SysBench (seconds) (1)	93.05	92.86	Slower is better
SysBench (seconds) (2)	372.09	371.34	Slower is better
RAM			
`mbw`, MEMCPY method (MiBps)	694.80	785.59	Higher is better
`mbw`, DUMB method (MiBps)	721.43	807.20	Higher is better
`mbw`, MCBLOCK method (MiBps)	1,086.91	1,287.24	Higher is better
Network (LAN)			
`iperf` (Mbps) (3)	94	319	Higher is better
SD Card			
`hdparm`, cached (MBps)	601.8	600.0	Higher is better
`hdparm`, disk reads (MBps)	21.6	21.7	Higher is better
Power			
Idle	4.2W	7.6W	–
Under load	11.7W	14.0W	During SysBench (4)

(1) CPU – total time, 3 repeats with 4 threads

(2) CPU – total time by event execution, 3 repeats with 4 threads

(3) 3 repeats

(4) 3 repeats with 4 threads

The chipset supports a single spatial stream (1x1) for a data rate of up to 433.3Mbps – PHY (physical layer) gross data rate – over 802.11ac. It can handle 20/40/80MHz channels with optional short guard interval (SGI); the 80MHz channels were added by the 802.11ac standard, which means, the broader the channel, the more throughput. Theoretically, 802.11ac supports channel width up to 160MHz. However, hardly any devices are on the market for this, and the Rasp Pi would not benefit from it anyway.

The CYW43455 is connected to the RPi3B+ SoC via SDIO (SD input/output; the Arasan eMMC memory controller built into the SoC). The Broadcom SD controller itself does not support SDIO and therefore only serves the SD card. The Arasan controller supports the SDIO v3.0 interface according to the specifications; however, it seems unlikely that it also supports the DDR50 (dual data rate at 50MHz with up to 400Mbps throughput) and SDR104 (up to 432Mbps throughput) transmission formats offered by the CYW43455.

The Arasan device normally runs on the RPi3B+ at a clock speed of 41.6MHz. When transmitting 4 bits simultaneously, you can achieve 166.4Mbps and, with support for DDR, even 332.8Mbps. Apparently, various circumstances prevent clocking the controller at 50MHz. In practice, the limit will therefore be 166.4Mbps throughput. In any case, compared with the RPi3B, this is already double the data rate and could be better exploited in the less densely populated 5GHz band.

Connecting via the SDIO means that wireless traffic does not depend on the USB port, providing more bandwidth for do-it-yourself routers. In everyday use, up to 102Mbps throughput in the 5GHz band should be realistic. All this indicates that the Arasan SDIO port is the bottleneck.

As with the Pi Zero W, the antenna is a Proant PCB, so the RPi3B+ should achieve slightly better performance (throughput, range) in the 2.4GHz range than does the RPi3.

Bluetooth 4.2

The RPi3B supported Bluetooth 4.1; thanks to the CYW43455, the RPi3B+ now offers Bluetooth 4.2 – that is, "Bluetooth Classic" and Bluetooth LE (Low Energy). Version 4.2 improves speed, privacy, and data security for Bluetooth LE. Compared with the previous version, the data packets are two and a half times faster and are capable of containing up to 10 times more data.

Bluetooth 4.1 already provides for easier connection of devices without a hub, but Bluetooth 4.2 takes this to a whole new level: The IPSP-supported 6LoWPAN (Internet Protocol Support Profile/IPv6 over low-power wireless personal area networks) takes IPv6 to small, low-performance devices for integration into the Internet of Things.

The Generic Attribute Profile (GATT) provides a smart Bluetooth gateway to the Internet. For example, a BLE 4.2 sensor can send and receive messages through the RPi3B+ as a gateway. Bluetooth beacons that attempt to track your device require explicit permission as of v4.2. The LE Secure Connections pairing option is also available as of Bluetooth 4.2.

Performance

I was only able test the RPi3B+ for this article with a pre-release image with wireless not yet enabled and the CPU still clocked at 1.2GHz. A test with the new software image is coming soon, and the performance values for the RPi3B+ should improve significantly. As Table 1 shows, though, they are already impressive.

Both the RPi3B and RPi3B+ devices were tested with a SanDisk SD card (class 10) and an EM 231 energy monitor by MK Electronics. A keyboard and HDMI and Ethernet cables were connected. The remote station in the LAN was a GigE port on a laptop connected with a gigabit switch by TP Link.

Interesting Facts

To run all functions of the new RPi3B+, you definitely need new software: It does not boot with NOOBS before version 2.7.0 dated March 14, 2018. The developers are also working on a firmware update for Raspbian Jessie. Other distributions, like LibreELEC and RetroPie, will follow suit in time.

The 40-pin GPIO header stayed the same, so all extension HATs fit. As with the RPi3B and Pi Zero W, Bluetooth is connected via the UART (PL011). If you use the UART through the GPIO header, you should note the known limitations of its mini UART. Alternatively, swap the two UARTs via overlay or disable Bluetooth [4].

The RUN header previously had two pins, of which one was ground suitable for mounting a reset button. However, the power supply remained switched on. With the RUN pin continuously grounded, the CPU was off for all intents and purposes, but the hardware still consumed power.

In the RPi3B+, the ground pin was removed from this header and a new PEN (power enable) pin was added. If you connect it to ground, it completely drains the power from the board. The board then transitions to the lowest consumption state possible, making it useful for projects that operate with a battery and wake up using a microcontroller.

Conclusions

The brand new Raspberry Pi 3 B+ is, as the name suggests, not a Raspberry Pi 4. However, in almost every aspect of an SBC, it offers slightly more than its predecessor – and at an identical price of $35 (excluding tax and shipping).

The RPi3B+ fulfills some of the long-desired wishes of the community. More CPU power, Gigabit Ethernet, 802.11ac, and PoE are only the tip of the iceberg. Like the RPi1B+, the RPi3B+ is an incremental development, not a quantum leap. The VideoCore IV, which is far more deeply hardwired to the BCM283x than the CPU cores, restricts the new Rasp Pi to a RAM limit of 1GB. However, experience with the RPi3B shows that even H.265 videos can run smoothly through hardware acceleration.

Production of the RPi3B+ will continue until at least January 2023, and the Foundation assures the community that the other models will also be continued as long as they see demand. ■■■

Info

[1] Raspberry Pi 3 Model B+: *https://www.raspberrypi.org/blog/raspberry-pi-3-model-bplus-sale-now-35/*

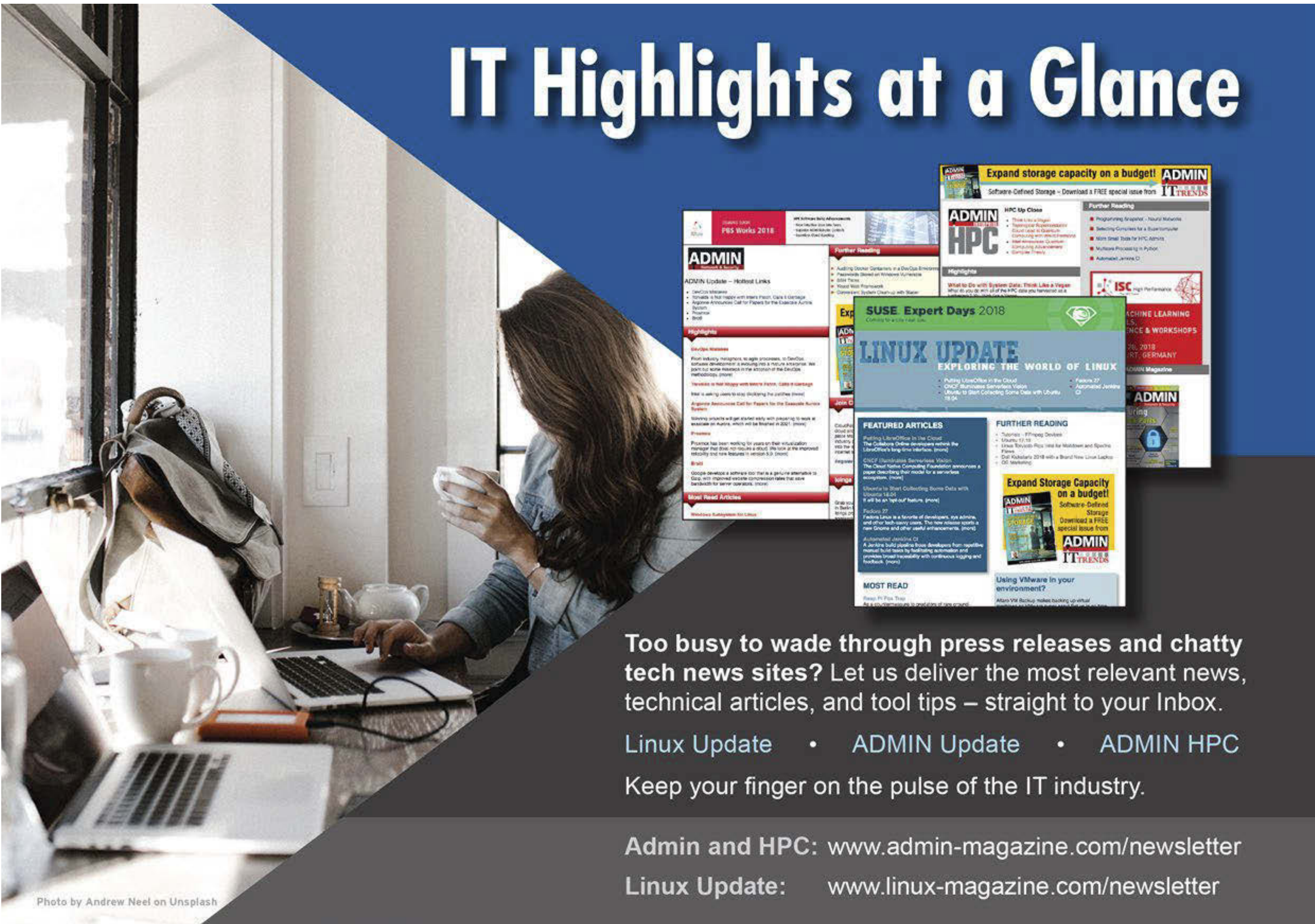
[2] Raspberry Pi Zero W: *https://www.raspberrypi.org/products/raspberry-pi-zero-w/*

[3] Official Foundation power supply: *https://www.raspberrypi.org/products/raspberry-pi-universal-power-supply/*

[4] Raspberry Pi UARTs: *https://www.raspberrypi.org/documentation/configuration/uart.md*

Author

Maximilian Batz is CEO of the Rasp Pi Internet Shop Pi3g (*http://www.pi3g.com*), which specializes in complete, preconfigured Raspberry Pi sets with custom software solutions and matching accessories. His goal is to make the innovative miniature computer accessible to everyone.



MakerSpace

Simulate digital circuits with Logisim

Circuit Sense

Throw out the cumbersome pencil and paper; you can design, draw, and test digital circuits with Logisim.

By Anzela Minosi

Processors are at the heart of every computer – from PCs to tablets to smartphones. Therefore, computer architecture is part of every computer science curriculum. During your studies, you get to peek under the hood of the computer and learn how to design digital circuits, which is exactly where Logisim [1] comes into play. Logisim is a useful tool for learning how to use digital circuits to create electronic designs and simulate small circuits – or even entire microprocessors.

Ubuntu and Fedora offer Logisim as a package of the same name in their repositories, so you can install it through the distribution's package manager and call it up from the menu. For other distributions, you can download the generic version of the software from SourceForge [2] and use the command:

```
$ java -jar logisim-generic-<Version>.jar
```

The generic Logisim version is a Java application that requires a working Java Runtime Environment (JRE), which can be obtained either from the repositories of the distribution you use or directly from the Oracle website [3].

Logisim lets you implement simple logical functions like XOR and is capable of creating multiplexers and decoders. Additionally, you can create components that store data, such as flip-flops, memory, and registers [4].

E Is for Entry Level

The basics of Logisim can be better explained using a simple example: an aptitude test with three tasks, at least two of which must be solved to pass the test.

To make a digital circuit of this problem, you first need to identify the inputs and outputs. Here, inputs *a1* to *a3* represent the three tasks. Two outputs *b* and *d* stand for the states passed or failed.

If you were to create the circuit manually, you would determine the functions for the outputs and set up a truth table to define the outputs. For inputs of two or three *1*s, you would add a *1* under *b* and a *0* otherwise. Column *d* it is exactly the opposite: It is true if *b* is false and vice versa.

The functions for the two outputs can then be read from the truth table. To set up the respective function correctly, look only at those lines of the inputs in which a *1* results for *b* or *d* (Listing 1).

In Boolean algebra, an addition stands for an OR gate and a multiplication for

Listing 1: Truth Table Functions

```
f_b(a1,a2,a3) = (¬a1a2a3)+(a1¬a2a3)+(a1a2¬a3)+(a1a2a3)
f_d(a1,a2,a3) = (a1¬a2¬a3)+(¬a1a2¬a3)+(¬a1¬a2a3)+(¬a1¬a2¬b3)
```

Lead Image © donatas1205, 123RF.com

an AND gate. Because the previous circuit would require three OR gates and eight AND gates for each function, you should check to see whether the function can be minimized according to the laws of Boolean algebra. Fortunately, you can assign this task to Logisim.

Less Is More

To begin, start Logisim and save a new project by clicking *File* | *Save As*; then, enter the variables mentioned above in the *Inputs* and *Outputs* tabs under *Window* | *Combinational Analysis* (Figure 1). In digital circuits, combinational logic does not depend on prior inputs but depends only on the current input.

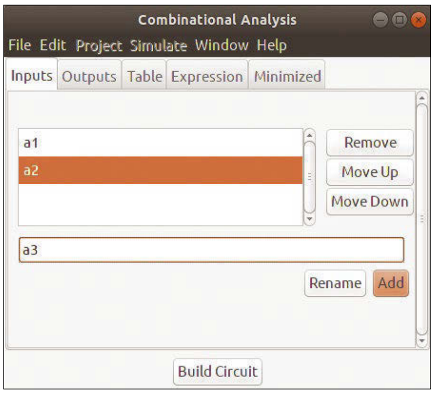
Next, switch to the *Table* tab, where you determine under which conditions the outputs are true or false by clicking on the placeholders in the output columns until the desired value appears (Figure 2).

Finally, you decide how Logisim should minimize the function in the *Minimized* tab (Figure 3) – you have the choice between *Sum of products* (a circuit specified in disjunctive normal form, so variables are connected by OR symbols) or *Product of sums* (a circuit specified in conjunctive normal form, so variables are connected by AND symbols). In this example, I use *Sum of products*. Once you have determined the minimization format, switch to the *Expression* tab. The minimized versions of the two functions that implement the circuit appear there (Figure 4).

Before leaving the Combinational Analysis window, press the *Build Circuit* button and assign a name to the new circuit (Figure 5). Logisim then draws the circuit on the basis of the truth table. On the left side of the overview window you can access the circuits you create by double-clicking on them if needed.

More Memory

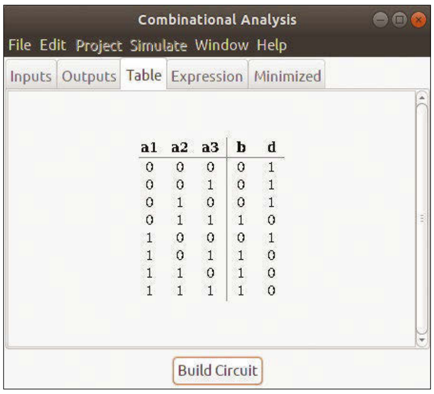
The Combinational Analysis window does not conjure up a circuit for flip-flops, so you must invest a little manual labor. As an example, you can create an edge-triggered D flip-flop. To assign a name to the circuit, right-click on the project name on the left side and select *Add Circuit*. Logisim then creates an empty canvas onto which you can drag the components.


Combinational Analysis
File Edit Project Simulate Window Help
Inputs Outputs Table Expression Minimized
Remove
Move Up
Move Down
Rename
Add
Build Circuit

a1
a2

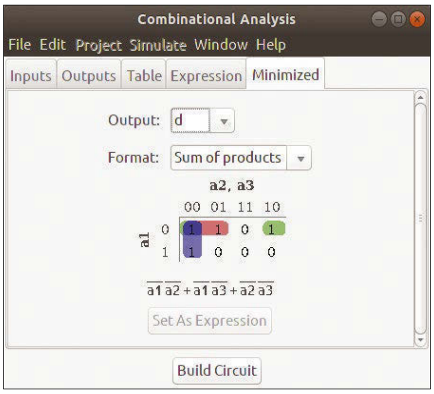
a3

Figure 1: To build a circuit with Logisim, you first need inputs.


Combinational Analysis
File Edit Project Simulate Window Help
Inputs Outputs Table Expression Minimized
Build Circuit

a1	a2	a3	b	d
0	0	0	0	1
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

Figure 2: Logisim creates the entries for the inputs automatically. All you have to do is look at the results of the outputs.


Combinational Analysis
File Edit Project Simulate Window Help
Inputs Outputs Table Expression Minimized
Set As Expression
Build Circuit

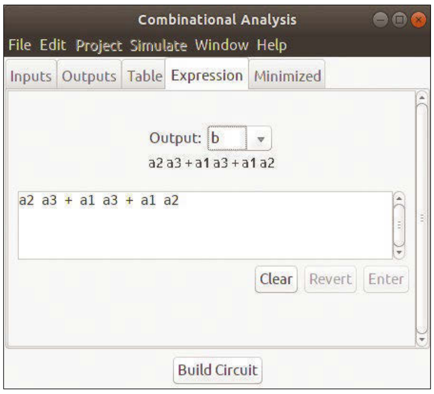
Output: d

Format: Sum of products

a1 \ a2, a3	00	01	11	10
0	1	1	0	1
1	1	0	0	0

$\overline{a1}\,\overline{a2} + \overline{a1}\,\overline{a3} + \overline{a2}\,\overline{a3}$

Figure 3: Define the normal form of the function in the *Minimized* tab.

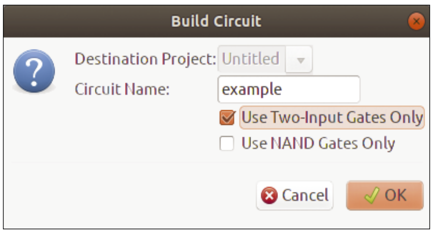

Combinational Analysis
File Edit Project Simulate Window Help
Inputs Outputs Table Expression Minimized
Clear
Revert
Enter
Build Circuit

Output: b

a2 a3 + a1 a3 + a1 a2

```
a2 a3 + a1 a3 + a1 a2
```

Figure 4: Logisim simplifies the function for the example's output *b*.



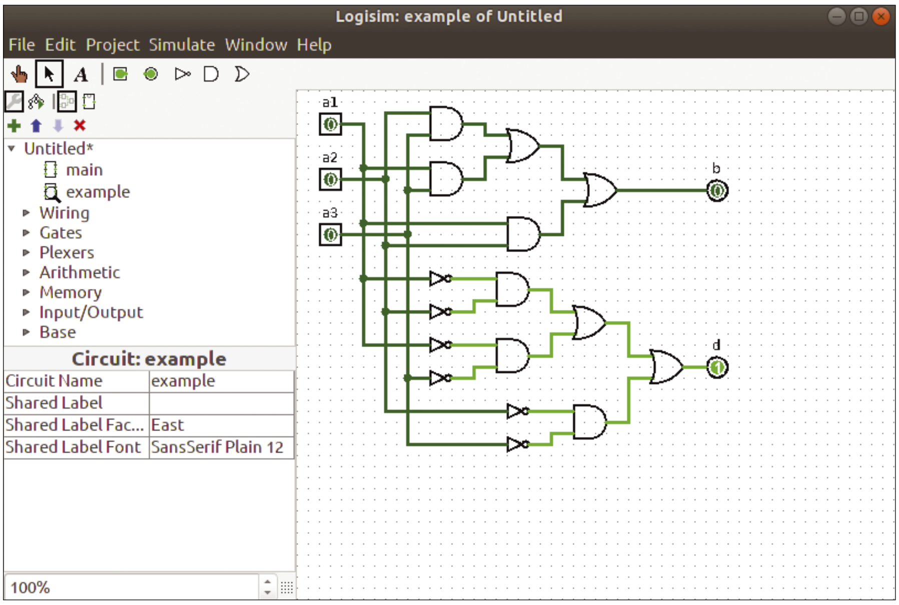
Destination Project: Untitled

Circuit Name: example

☑ Use Two-Input Gates Only

□ Use NAND Gates Only

Figure 5: Logisim can draw the circuit either with two-input gates or NAND gates only.



Circuit: example

Circuit Name	example
Shared Label	
Shared Label Fac...	East
Shared Label Font	SansSerif Plain 12

Figure 6: The component libraries appear under the project folder that holds your circuits. When you select a component in the canvas on the right side, the corresponding attribute table appears at the bottom left.

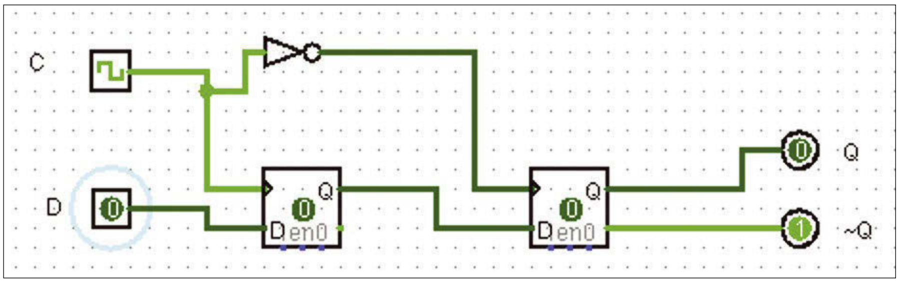


Figure 7: This D flip-flop has a time control: It only changes its state when the clock switches to *1*.

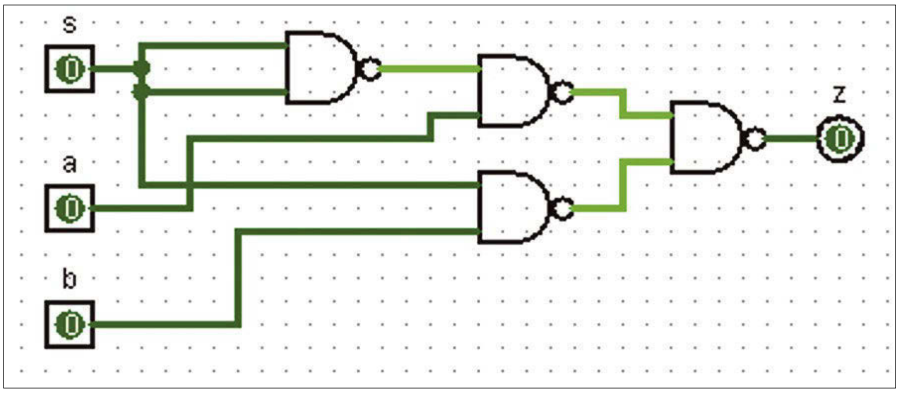


Figure 8: The 1-bit multiplexer logic circuit can only be implemented with the aid of NAND gates.

In the navigation pane on the left, you will find the libraries with components provided by the software (Figure 6). Flip-flops can be found in the *Memory* library. After opening the library, drag a D flip-flop onto the canvas.

Logisim also provides a place to assign component attributes in a corresponding attribute table that shows up in the lower left section when you select a component. Your D flip-flop should react when the clock has a value of *1*, so you need to set the *Trigger* attribute to *rising edge*.

Next, select the component and create a duplicate by copying the element with Ctrl + C and pasting it with Ctrl + V. Logisim copies the attributes as well, so you do not need to enter them again.

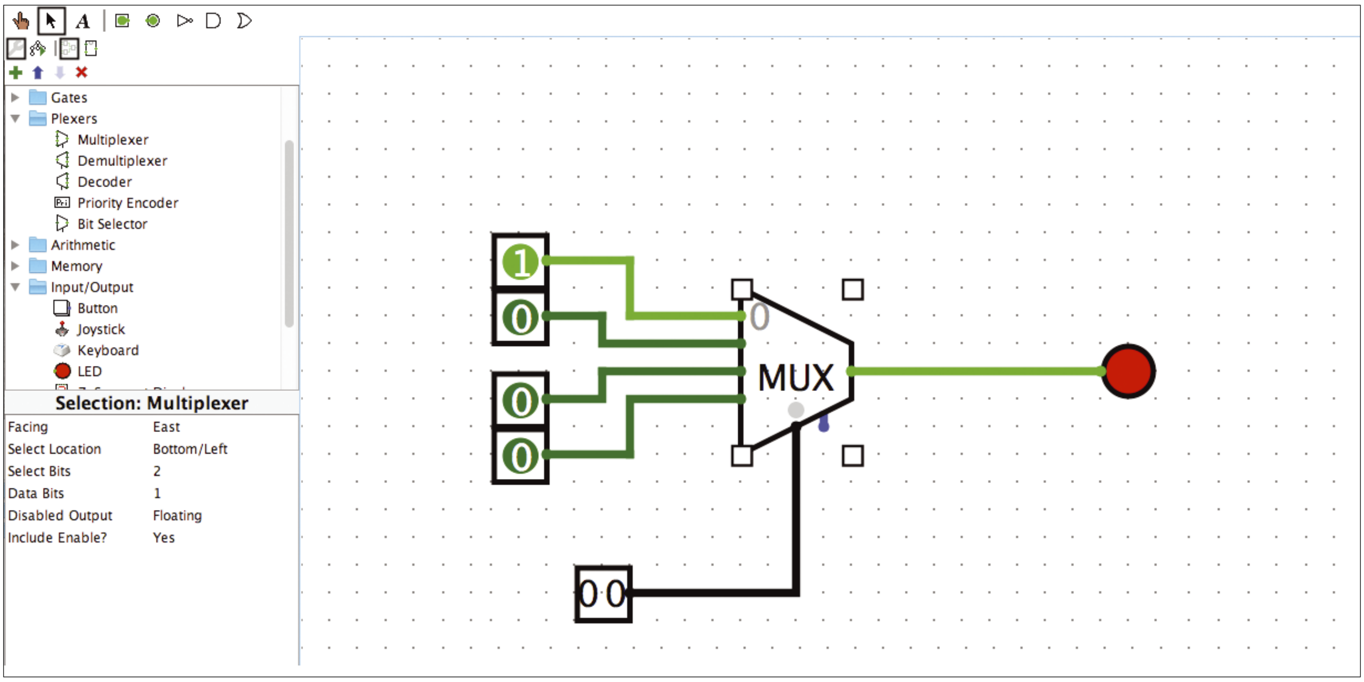
Your circuit needs input and output pins, as well as a *Clock*, which can be found in the *Wiring* library, and a *Controlled Inverter* (a gate that implements the NOT function and turns a *0* into a *1* and vice versa), from the *Gates* library. After you have dragged the components onto the canvas, you can change their orientation. Usually the input and output pins point in opposite directions (Figure 7). You should also set the *Output?* attribute of the output pins to logical *Yes*.

The truth table for a flip-flop does not need manually created circuits. Once all the drawing elements are on the sheet, you just need to connect them. To do so, mouse over the inputs or outputs of a component, click, and drag a line to the next input or output. To draw a line with corners, first create a straight line, then release the mouse button, and continue drawing in a different direction.

Labeling the individual components is also recommended so other users of the circuit can identify the elements: Click the letter icon in the menubar and place the cursor on the canvas to enter a label.

Multiplex Connection

In multiplex cinemas, people walk in at a common entrance and take a seat in



Selection: Multiplexer

Facing	East
Select Location	Bottom/Left
Select Bits	2
Data Bits	1
Disabled Output	Floating
Include Enable?	Yes

Figure 9: In this 2-bit multiplexer, the inputs are on the left side and the control pin is at the bottom. The output is an LED located to the right of the multiplexer.

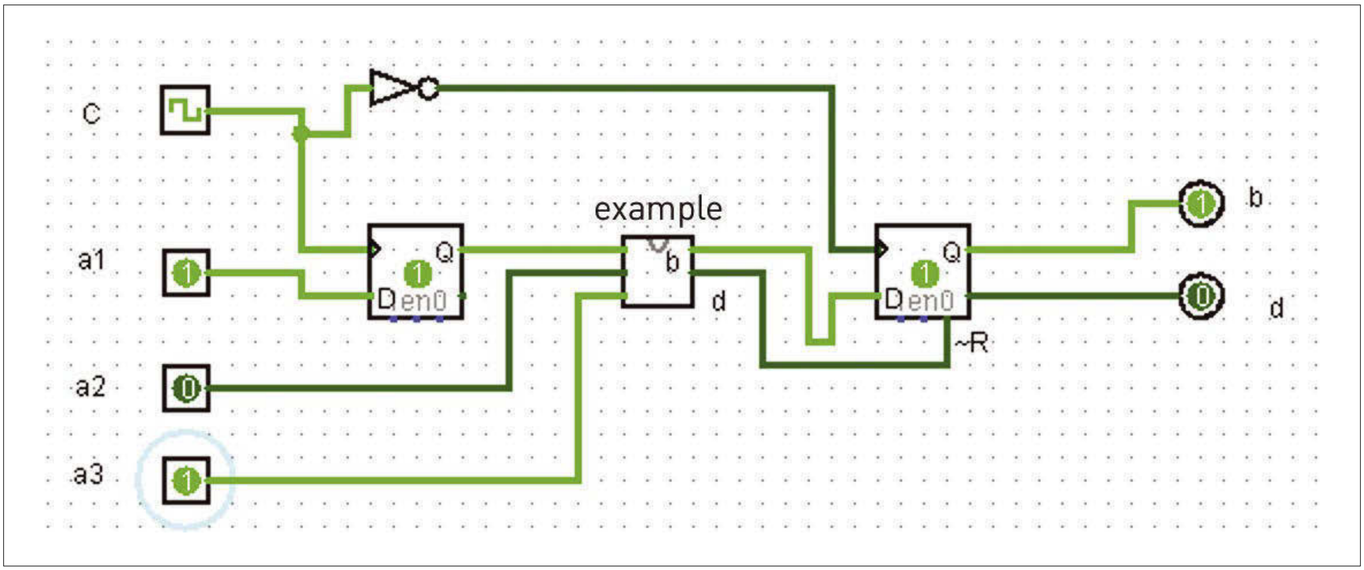


Figure 10: Logisim displays a subcircuit as a block, to which you can add inputs and outputs.

the desired theater. An *n*-bit multiplexer works similarly. However, encoders, which simulate the end of a movie night, wherein visitors come out of the individual cinemas and leave the building from a common exit, need a closer look.

The design of a parallel 1-bit multiplexer (Figure 8) begins in the Combinational Analysis window. To start, create the appropriate truth table and then click *Build Circuit*. In the window that opens, select the *NAND* gate and assign a name to the circuit. A NAND operator corresponds to an inverted AND operator. NAND gates are considered universal, because any logical function can be implemented by this type alone.

To create a multiplexer manually, open the *Plexers* library and drag a *Multiplexer* onto the canvas. For a 2-bit multiplexer, you need two data lines and one control bit, which you set in the *Select Bits* and *Data Bits* drop-downs, respectively, in the attribute table (Figure 9). This combination gives you four inputs (00, 01, 10, 11) numbered 0 through 3, another input for the selector (control) pin, and one output. After connecting pins to the inputs of the multiplexer, set the control signal – found on the bottom side of the multiplexer – to *2* bits in the attribute window, and then add an LED for the output, which is found under *Input/Output* | *LED*.

Divide and Conquer

When programming, you reuse code by creating functions (subroutines, methods), and you do the same thing in Logisim by creating subcircuits.

To create a timed circuit, you can integrate the aptitude test, which in this case represents the logic, between two D flip-flops. To start, create a new circuit by calling up the context menu of the project folder with a right-click, choose *Add Circuit*, and give it a name. Drag two flip-flops to the canvas.

To insert the example subcircuit between the two D flip-flops, click once on the name of the subcircuit in the tree structure on the left. The mouse pointer changes to a block that represents the subcircuit, which you now add to the canvas and connect to the flip-flops (Figure 10).

The D flip-flop has only one output, which also needs to go to an input on the subcircuit, so you need to add two more input pins to the subcircuit. The subcircuit has two outputs, one of which you connect to the input of the second D flip-flop. The other output goes to the reset input (~*R*) of the second D flip-flop.

Testing

The most beautiful circuit is of little use if it doesn’t work, so you should test your circuit before using it. To test logical circuits such as this aptitude test, choose the Poke (pointing finger) tool in the menubar. By clicking an input pin several times, you can change its value and observe the behavior of the outputs. Logisim color highlights active lines (i.e., those through which data bits are currently flowing) during the simulation.

Logisim has a special feature for time-controlled circuits: The *Simulate* | *Ticks Enabled* menu item automatically starts ticking the clock, so you only have to change the data inputs to observe the circuit's behavior.

Logisim also offers an export function that creates a drawing of the circuit under *File* | *Export Image*. In the resulting window, you select the desired image format and scaling.

Conclusions

Logisim handles the graphical elements of circuits, such as rectangles, trapezia, lines, and so on, so you can focus on the essentials of electronics design and become familiar with circuits. ■■■

Info

[1] Logisim: *http://www.cburch.com/logisim/docs/2.5.0/en/guide/index.html*

[2] Download Logisim: *https://sourceforge.net/projects/circuit/files/*

[3] Oracle JRE: *https://java.com/en/download/linux_manual.jsp*

[4] “The Basics of Logic Design”: *http://www.csbio.unc.edu/mcmillan/Media/appendixC.pdf*

Author

Anzela Minosi uses the Sabayon Linux distribution on her PC. When she’s not running or programming her own PC support, you are likely to find her sunbathing.

MakerSpace

Ergonomic Computing and Open Source Occupational Safety

Finding ergonomic open source software and hardware can be difficult, but it's becoming easier.

By Bruce Byfield

Ergonomic computing is a subject that only interests most people when they start having injuries. However, there is much to be said about preventative measures, and, with an aging population, ergonomic devices are becoming increasingly common. However, if you prefer to use ergonomic devices that also use open source software and hardware, the options range from limited to nonexistent.

Choosing ergonomic hardware can be difficult, because there are no standards for using the term. Too often, "ergonomics" is little more than a buzzword. Just as gluten-free can be seen in the grocery store on oranges, nuts, and countless other products that do not have gluten in them to start with, so many devices that are labelled ergonomic are of dubious value for those who wish to minimize the results of repetitive stress or prolonged inactivity. For example, although standing desks have been widely touted as healthier, and even as making users smarter and more productive, such claims are typical of untested devices and should be regarded skeptically. Moreover, some studies suggest that standing desks are no healthier than sitting, and that the problem in both cases is being still for long periods of time [1].

With the verdict still out on such allegedly ergonomic devices, discussing them seems pointless. For that reason, I will focus on three general categories in which ergonomics has been studied more thoroughly: chairs, keyboards, and pointing devices.

Chair Ergonomics

I am unaware of any open source designs for chairs. Rather, the reverse is true: Ergonomic chairs are regarded as masterpieces of industrial design and are prestige products that are priced accordingly. Depending on the chair and the selected options, an ergonomic chair can cost from $500-$3,000. Leading manufacturers of ergonomic desk chairs include Humanfactor [2] and Steelcase [3], but the best known is probably Herman Miller [4], a furniture company that is now as well known for its chairs as its modern furniture. Herman Miller's line of chairs include the Aeron (once a popular perk at tech startups), Celle, Embody, Mirra 2, Sayl (Figure 1), and Setu. One reason for Herman Miller's predominance is that each of its product lines is described in depth on its site, which gives buyers a sense of what they should look for in an ergonomic chair.

Well-padded chairs are often thought of as executive chairs, but are generally the last ones you should consider when buying an ergonomic chair. Instead, look at the suspension on the chair's back, and, to a lesser extent, the seat. Usually, an ergonomic chair looks much thinner, if not spindly, than the traditional executive chair, but it is actually far healthier.

Lead Image © Volodymyr Horbovyy, 123RF.com

In addition to the suspension, look at the available features. The seat height should be adjustable, so that it can be set to your height, and so should the armrests – not only vertically, but horizontally as well. Depending on how you sit when typing, you may also want the tilt of the seat to be adjustable. If you spend eight hours or more at a keyboard each day, you should also check whether lumbar support is available on the back.

Whenever possible, you should also try possible chairs before buying. Is the chair stable? Can you lean back in it? If you like to slide across the floor to colleagues' workstations, is the chair light enough for you to continue to indulge your habit? These are only questions that you can answer in person. Very likely, your final choice will be idiosyncratic, if only because some chairs are intended for different weights.

Keyboard Options

Any keyboard can give you relief from repetitive stress if you use keyboard shortcuts rather than the mouse and menus or toolbars. Of course, keyboard shortcuts take a while to learn, but Linux users have an advantage that Vim key bindings are available for a wide variety of desktop applications, including Firefox, Chrome, and LibreOffice, as well as many desktop editors [5]. Other tools, such as Bash history and tab completions and even arrow keys, can also keep your hands on the keyboard and making minimal motions.

Some relief can also be had from programmable keyboards, such as many models that are sold for gaming, such as the Razer [6] line of products. These keyboards can be programmed with the macros of your choice to cut down on typing. Many of these gaming keyboards also have backlit keys, which can reduce eye strain in dim lighting, as well as mechanical keyswitches that generally require less pressure when typing. However, check that such keyboards have a web-based configurator, or else you may be unable to configure them easily on Linux. A safer option is to consider Input Club's open source programmable mechanical keyboards, some of which ship assembled, and others which are aimed at the do-it-yourselfers [7].

Figure 1: The Sayl chair is one of Herman Miller's most popular ergonomic chairs.

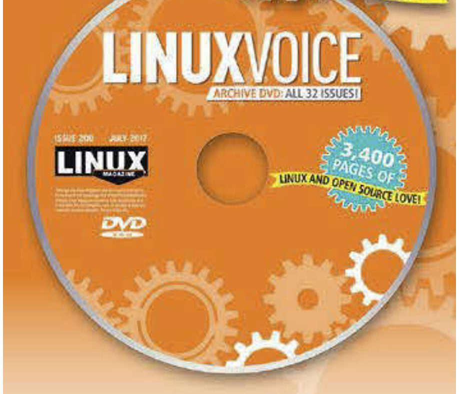
True ergonomic keyboards have other options to make typing easier. Many are divided into halves that can be positioned as close or as far apart, or even horizontally. The halves can also be tilted in several different ways. On each half, keys are arranged in curves to reduce the strain on reaching fingers and arranged in a slight bowl so that fingers move more naturally. Perhaps the most important feature is that the banks of keys are arranged in a diagonal line, instead of each higher row of keys being staggered to the left, as on most keyboards – a feature that reduces the distance between them.

All the features of both programmable mechanical keyboards and ergonomic keyboards are available in least three open source keyboards: the ErgoDox EZ [8] (Figure 2), Input Club's Infinity ErgoDox (which comes in a kit) [9], and the Keyboardio Model 01 [10]. Mounted on solid maple, the Model 01 also offers aesthetics far in advance of any keyboard on the market, whether proprietary or open source. Like other programmable mechanical keyboards or ergonomic keyboards, these open source alternatives sell for $300-$325.

The Mouse and Other Pointing Devices

Despite its widespread popularity, the standard mouse can cause arthritis, tendonitis, and carpal tunnel syndrome, because it places the hand in an unnatural position for prolonged periods [11]. Many so-called ergonomic mouses have been manufactured, but, even with programmable buttons, their improvement is minimal. The best-regarded traditional mouses are those like the ones manufactured by Anker and Anthrodesk; these mouses are on a vertical angle [12].

Instead, many look for an alternative in other pointing devices. One alterna-



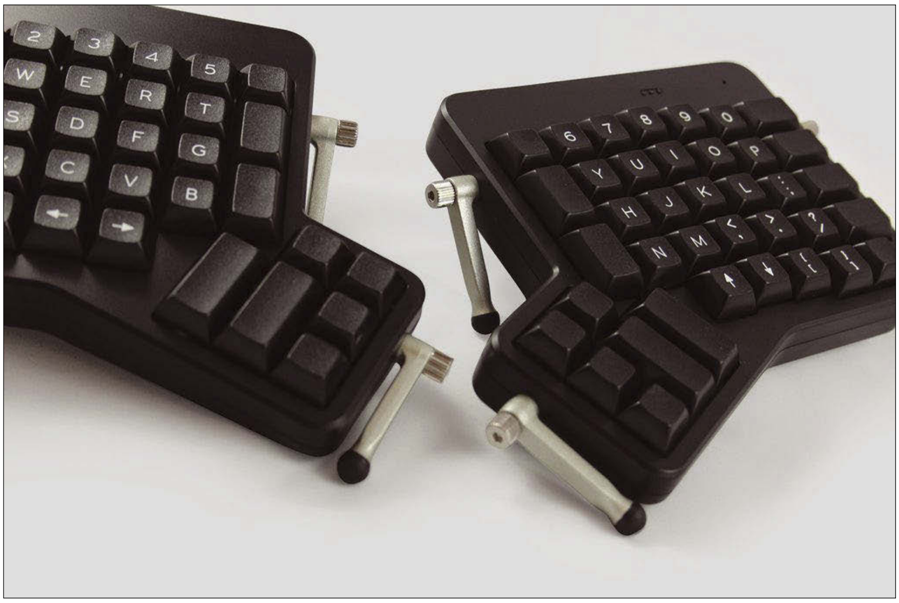


Figure 2: The ErgoDox EZ was one of the first open source ergonomic keyboards.

tive is a trackball, which moves the cursor using a roller. However, while the hand is not clenched when using a trackball the way it is with a mouse, it remains outstretched. In my experience, trackballs take some practice to use well and, while an improvement on the mouse, can still cause injuries when used over a period of hours.

The most ergonomic alternative is a drawing tablet. If you are using one primarily as a pointing device, the cost is about $90-$140 for an entry-level tablet. The tablet itself is a map of your monitor screen, and the cursor is moved by a stylus, with a button for right-clicking. For scrolling, the stylus can use the windows' scrollbars, or else you can use the keyboard's arrow keys.

The stylus is gripped like a pen and can be put down or balanced with a couple of fingers when typing, a change of position that helps to reduce repetitive stress. In addition, tapping the stylus on the tablet to activate a link is a mild gesture compared to moving a mouse and clicking a mouse button. With most tablets, you can scroll when the stylus is slightly above the tablet. Add a wrist pad to place your hand a bit higher than the tablet, and the hand is poised in as comfortable a position as possible.

Figure 3: An entry-level Wacom tablet makes an ideal replacement for a mouse. It is not open hardware, but it is supported by recent Linux Kernels.

Best of all, as detailed in "Command Line – xsetwacom" in this issue, support for many tablets from Wacom [13] is compiled directly into modern Linux kernels (Figure 3). Using `xsetwacom`, users can configure tablets by issuing a series of commands and compiling them into a script to run before using the tablet. While this support is quirky, in the long term, the health of your hands and wrists make the effort worthwhile.

Devices Plus Habits

All these solutions combined do not completely eliminate the injuries caused by spending too much time in front of the computer. However, since I now use an ergonomic chair, keyboard, and pointing device, I can say that the combined effect reduces the injuries of a full day at the computer by at least 80 percent in my case. Even when I have twinges at the end of a day, they are gone by the next morning.

The effect is even higher when I use software that forces me to take a regular break, like Take a Break [14]. Such software can break my concentration – usually at the worst possible time – but, together with ergonomic hardware, it can keep me working long hours for days at a time.

Going completely ergonomic is hard, and using open source whenever you can is even harder. However, the combination is not impossible in many cases, especially when combined with some sensible work habits, like using keyboard shortcuts or taking regular breaks. While some gaps in functionality remain, the combination is at least possible to a degree not previously available as recently as five years ago. ■■■

Info

[1] Standing desks: *http://www.gwinnettdailypost.com/local/health/prolonged-standing-at-work-can-cause-discomfort-throughout-body/article_4945560c-b3b4-55f6-a078-83a34ced7cbc.html*

[2] Humanfactor: *http://www.humanfactorseating.co.uk/*

[3] Steelcase: *http://web.steelcase.com/wellness/index.html#/physical/*

[4] Herman Miller: *http://store.hermanmiller.com/office/office-chairs?lang=en_US*

[5] Vim keybindings: *https://www.reddit.com/r/vim/comments/3tluqr/my_list_of_applications_with_vi_keybindings/*

[6] Razer keyboards: *https://duckduckgo.com/?q=razer+keyboard&t=ffnt&atb=v59-2aa&ia=products*

[7] Input Club: *https://input.club/*

[8] ErgoDox EZ: *https://ergodox-ez.com/*

[9] Input Club Infinity ErgoDox: *https://input.club/devices/infinity-ergodox/*

[10] Keyboardio Model 01: *https://shop.keyboard.io/*

[11] Mouse design: *https://www.alexfergus.com/blog/the-best-ergonomic-computer-mouse-and-why-your-wrist-needs-it*

[12] Ergonomic mouse designs: *https://duckduckgo.com/?q=ergonomic+mouse&t=ffnt&atb=v59-2aa&ia=products*

[13] Wacom: *https://www.wacom.com/en-us*

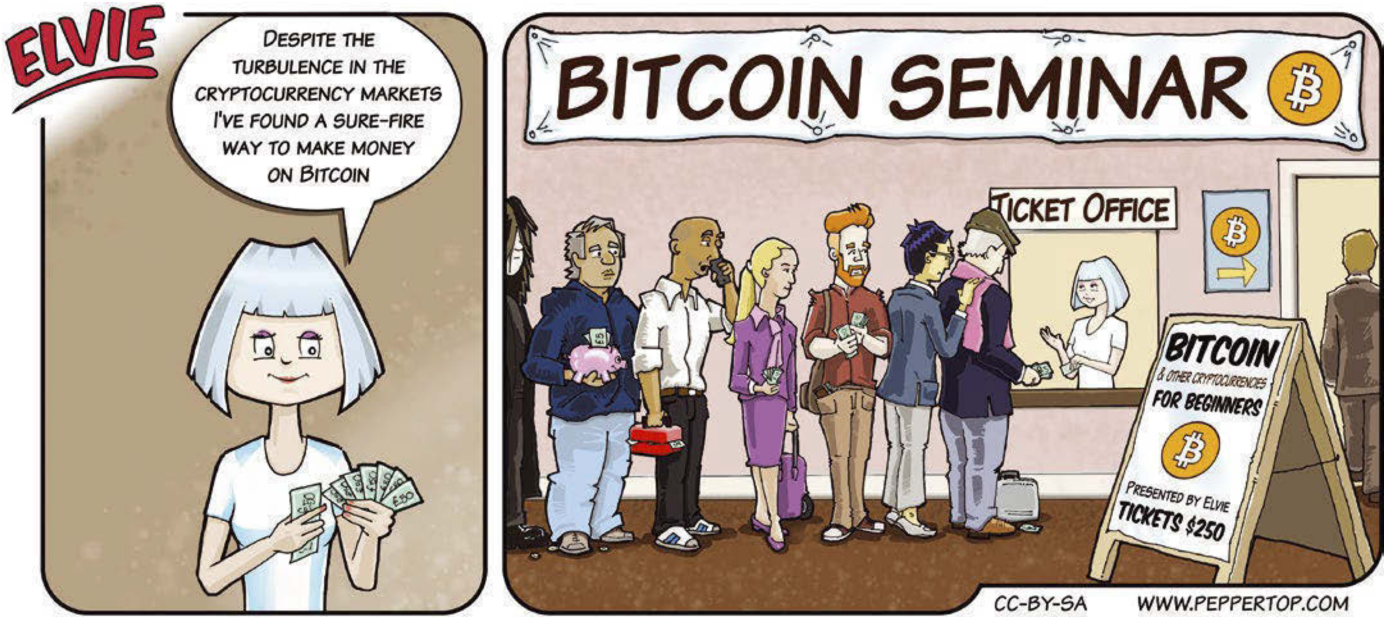
[14] Take a Break: *https://linoxide.com/linux-how-to/tool-force-users-periodical-breaks/*

If the only applications you needed were Firefox and LibreOffice, you probably wouldn't need a Linux magazine. Part of the fun of reading (and writing) a computer magazine is discovering new tools you don't already know about. This month we introduce you to Mermaid – a cool app that lets you draw charts and diagrams using text-based commands. Linux has several tools for ripping audio CDs, and this month we present two of the best: soundKonverter and free:ac.

We also bring you a tutorial on how to look for text in local files and messages using the powerful Recoll search tool, and we help you get started with exploring the popular Minetest gaming engine.

Image © Olexandr Moroz, 123RF.com

LINUXVOICE


ELVIE
DESPITE THE TURBULENCE IN THE CRYPTOCURRENCY MARKETS I'VE FOUND A SURE-FIRE WAY TO MAKE MONEY ON BITCOIN
BITCOIN SEMINAR
TICKET OFFICE
BITCOIN & OTHER CRYPTOCURRENCIES FOR BEGINNERS
PRESENTED BY ELVIE
TICKETS $250

CC-BY-SA WWW.PEPPERTOP.COM

MADDOG'S DOGHOUSE

Government paranoia regarding encryption can result in unreasonable search and seizure.

Jon "maddog" Hall is an author, educator, computer scientist, and free software pioneer who has been a passionate advocate for Linux since 1994 when he first met Linus Torvalds and facilitated the port of Linux to a 64-bit system. He serves as president of Linux International®.

Government surveillance

This week a chilling event happened. The Russian courts declared that Telegram, the messaging application that a lot of my friends, business associates, and I use, had to create a back door, so they could decrypt the messages sent in order to "fight terrorism."

About 200,000,000 monthly active users (not ALL of them are my intimate friends) use Telegram for a variety of reasons. One of the reasons I use Telegram is that they have a full-feature desktop for GNU/Linux. Another reason is that they guarantee a robust, secure, end-to-end encryption of my messages.

In an 18-minute court case, at which Telegram was not represented, the Russian courts decided that Telegram had to provide this access to people's data.

Telegram CEO , Pavel Durov, refused to do this, and the Russian government started blocking Telegram inside of Russia, where about 19 percent of the smartphone users use the application.

As the Russian government started reaching out to ISPs to block Telegram's ports, many Russians simply turned to Virtual Private Networks (VPNs) to get around the blockages, much like they do to watch blocked television shows or other types of censorship.

Russia has also called for Telegram to be removed from Apple App Store and Google Play, a tactic that may only slow, but not stop, Telegram's use.

The really sad part is that the Russian government does not understand the problem. There is no "master key" for Telegram's encryption, and it would not matter if there was.

I have been dealing with governmental paranoia around encryption for over 30 years. Not at the same level as the Electronic Frontier Foundation (EFF) or other advocacy groups, but more at the individual product level.

The pattern is always the same. You bring out a product that needs authentication. One of the most important parts of authentication is encryption. Without strong encryption, the "bad guys" will see your clear-text passwords, whether they be "sniffed" at public WiFi access points or hacked at various services' large servers. Without good choice of passphrases and good encryption of those passphrases, many of your accounts could be compromised at one time.

Then the government asks the magic words: "Does your product use encryption?" You then have to do a huge song and dance to prove to the government that your encryption cannot be used to encrypt messages or data, so the government is capable of seeing what you store.

The sad part of all this is that every bad person on the face of the earth already knows how to encrypt their messages. Encryption techniques have been published in books, printed on T-shirts, and taught in classes; even simple compression could be viewed as an encryption technique. If you do not know which compression technique was used, the stream of bits looks like it was "encrypted" (which, in effect, it was).

Yes, some encryption techniques are stronger and "harder" than others to decrypt, but those could easily be applied at the end-user level, and applications like Telegram would be incapable of helping the government decrypt them.

What the government really wants to prevent is everything being encrypted all the time, resulting in them having to expend huge amounts of CPU to decrypt all of the messages of people they think are the "bad guys." At the same time, however, they would be decrypting a lot of messages of people that are sending legitimate and harmless messages.

In April 2015, a cable TV show called *Last Week Tonight with John Oliver* showed an interview with Edward Snowden, the person who first alerted the citizens of the USA to how the US National Security Agency (NSA) was spying on both US and foreign citizens. Edward Snowden was branded a criminal and has sanctuary in Moscow at this time. The host, John Oliver, traveled to Moscow and together with Snowden tried to tell people exactly how prevalent this spying was. In the course of the interview, Snowden revealed that some NSA employees would pass around sexually-oriented pictures for their own pleasure. Oliver revealed that while many people could not remember who Snowden was, or what he did, or what he revealed, they were very upset that someone at the NSA could see their "private parts" or those of their loved ones.

I am sure that 99.9 percent of the people at the NSA are professional, hardworking people, but it only take one person to use personal information for theft, blackmail, or to "lose" the information to others who would use it poorly.

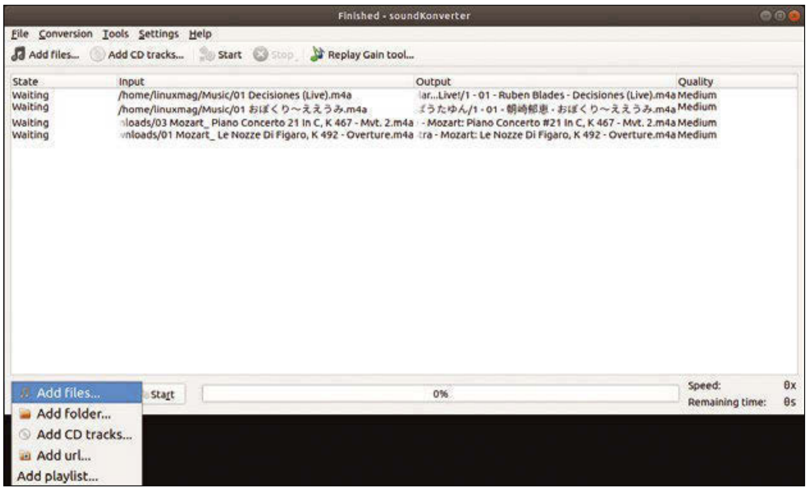
The founding fathers of the United States objected to the king's soldiers barging into their homes and searching them without reason, so our Constitution was written to particularly forbid this. Authorities HAD to obtain a search warrant. Now we are letting the government search our "homes" again. ■■■

Refound Sound

Transfer the contents of audio CDs to your PC and convert the contents of music folders with soundKonverter and fre:ac.

BY ERIK BÄRWALDT AND THOMAS LEICHTENSTERN

Figure 1: The first soundKonverter window you see is simple and straightforward.



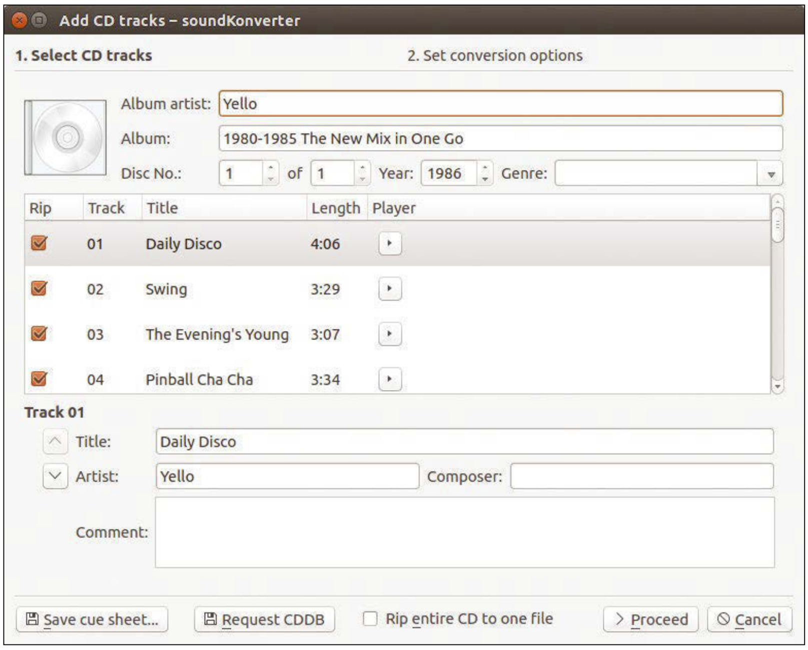
Finished - soundKonverter

State	Input	Output	Quality
Waiting	/home/linuxmag/Music/01 Decisiones (Live).m4a	lar...Live!/1 - 01 - Ruben Blades - Decisiones (Live).m4a	Medium
Waiting	/home/linuxmag/Music/01 おぼくり〜ええうみ.m4a	ようたゆん/1 - 01 - 朝崎郁恵 - おぼくり〜ええうみ.m4a	Medium
Waiting	nloads/03 Mozart_ Piano Concerto 21 In C, K 467 - Mvt. 2.m4a	- Mozart: Piano Concerto #21 In C, K 467 - Mvt. 2.m4a	Medium
Waiting	wnloads/01 Mozart_ Le Nozze Di Figaro, K 492 - Overture.m4a	tra - Mozart: Le Nozze Di Figaro, K 492 - Overture.m4a	Medium

0%

Speed: 0x
Remaining time: 0s

Figure 2: soundKonverter has checkboxes for selecting the tracks to be ripped. If metadata is missing, you can enter the information in the appropriate fields.



Add CD tracks – soundKonverter

1. Select CD tracks 2. Set conversion options

Album artist: Yello
Album: 1980-1985 The New Mix in One Go
Disc No.: 1 of 1 Year: 1986 Genre:

Rip	Track	Title	Length	Player
☑	01	Daily Disco	4:06	▸
☑	02	Swing	3:29	▸
☑	03	The Evening's Young	3:07	▸
☑	04	Pinball Cha Cha	3:34	▸

Track 01
Title: Daily Disco
Artist: Yello Composer:
Comment:

☐ Rip entire CD to one file

The Linux world plays host to several tools for ripping music CDs, but demanding audiophile users are often dissatisfied with the functionality of conventional rippers: Many of the tools lack important additional functions, such as automatic volume adjustment.

With free programs like soundKonverter [1] and fre:ac [2], you can do everything from ripping and transcoding in a single interface. In some cases, the programs do not provide the functions directly, but instead act as graphical front ends for other audio utilities.

soundKonverter

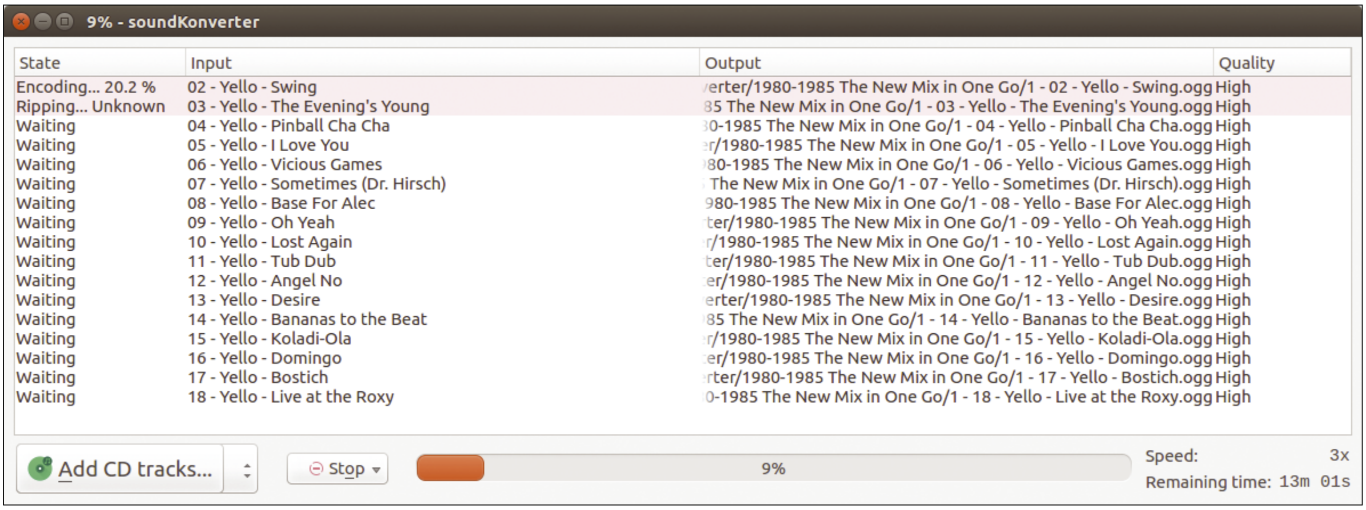
soundKonverter has been in development for more than 10 years and is continually maintained by the project developers. soundKonverter resides in the repositories of most major Linux distributions. See the documentation for your own Linux distro for more on obtaining and installing soundKonverter in package form.

Once you have installed soundKonverter, you will find the icon in the *Multimedia* (or *Sound & Video*) section of your application menu.

The software comes up with a simple window featuring a large list area with information about individual tracks, a menubar at the top, a progress bar at the bottom, and a button for adding media or files. Pressing the button opens a drop-down that lets you select the source of the content you want to edit (Figure 1). You can enter a directory, a single file, an audio CD, or a URL, and you can call up a playlist in M3U format.

Selecting an audio CD as the source medium shows the contents of the CD in a pop-up window in the form of a track list. Check the boxes under the *Rip* column to select the tracks you want to rip and transcode (Figure 2). A click on the *Play* icon beside each title starts playback. At the top of the window, enter the general metadata, such as the artist and album, and at the bottom, specify the details for each individual track.

After you select the desired tracks, click *Proceed*. The next dialog lets you define the output format, and the *Detailed* tab defines the quality or compression rates.



9% - soundKonverter

State	Input	Output	Quality
Encoding... 20.2 %	02 - Yello - Swing	erter/1980-1985 The New Mix in One Go/1 - 02 - Yello - Swing.ogg	High
Ripping... Unknown	03 - Yello - The Evening's Young	85 The New Mix in One Go/1 - 03 - Yello - The Evening's Young.ogg	High
Waiting	04 - Yello - Pinball Cha Cha	0-1985 The New Mix in One Go/1 - 04 - Yello - Pinball Cha Cha.ogg	High
Waiting	05 - Yello - I Love You	r/1980-1985 The New Mix in One Go/1 - 05 - Yello - I Love You.ogg	High
Waiting	06 - Yello - Vicious Games	80-1985 The New Mix in One Go/1 - 06 - Yello - Vicious Games.ogg	High
Waiting	07 - Yello - Sometimes (Dr. Hirsch)	The New Mix in One Go/1 - 07 - Yello - Sometimes (Dr. Hirsch).ogg	High
Waiting	08 - Yello - Base For Alec	980-1985 The New Mix in One Go/1 - 08 - Yello - Base For Alec.ogg	High
Waiting	09 - Yello - Oh Yeah	ter/1980-1985 The New Mix in One Go/1 - 09 - Yello - Oh Yeah.ogg	High
Waiting	10 - Yello - Lost Again	r/1980-1985 The New Mix in One Go/1 - 10 - Yello - Lost Again.ogg	High
Waiting	11 - Yello - Tub Dub	ter/1980-1985 The New Mix in One Go/1 - 11 - Yello - Tub Dub.ogg	High
Waiting	12 - Yello - Angel No	er/1980-1985 The New Mix in One Go/1 - 12 - Yello - Angel No.ogg	High
Waiting	13 - Yello - Desire	erter/1980-1985 The New Mix in One Go/1 - 13 - Yello - Desire.ogg	High
Waiting	14 - Yello - Bananas to the Beat	85 The New Mix in One Go/1 - 14 - Yello - Bananas to the Beat.ogg	High
Waiting	15 - Yello - Koladi-Ola	r/1980-1985 The New Mix in One Go/1 - 15 - Yello - Koladi-Ola.ogg	High
Waiting	16 - Yello - Domingo	er/1980-1985 The New Mix in One Go/1 - 16 - Yello - Domingo.ogg	High
Waiting	17 - Yello - Bostich	rter/1980-1985 The New Mix in One Go/1 - 17 - Yello - Bostich.ogg	High
Waiting	18 - Yello - Live at the Roxy	0-1985 The New Mix in One Go/1 - 18 - Yello - Live at the Roxy.ogg	High

Speed: 3x
Remaining time: 13m 01s

Figure 3: Thanks to multithreading support, soundKonverter transcodes in parallel.

You can also specify how and where to save the transcoded files and whether the software calculates a replay gain. If a replay gain tag is activated, soundKonverter automatically adjusts the volume to ensure that songs from different sources are played at the same volume.

If you want to play back a mix of digital and analog recordings, replay gain reliably compensates for the considerable dynamic differences between the two technologies. After completing the configuration, a click on *OK* transfers the desired tracks to the primary program window.

You can start the transcoder with the *Start* button at the bottom of the window. The progress bar and a percent activity indicator under the *State* column indicate the progress of the conversion (Figure 3). soundKonverter supports multithreading on state-of-the-art processors and can process several files simultaneously.

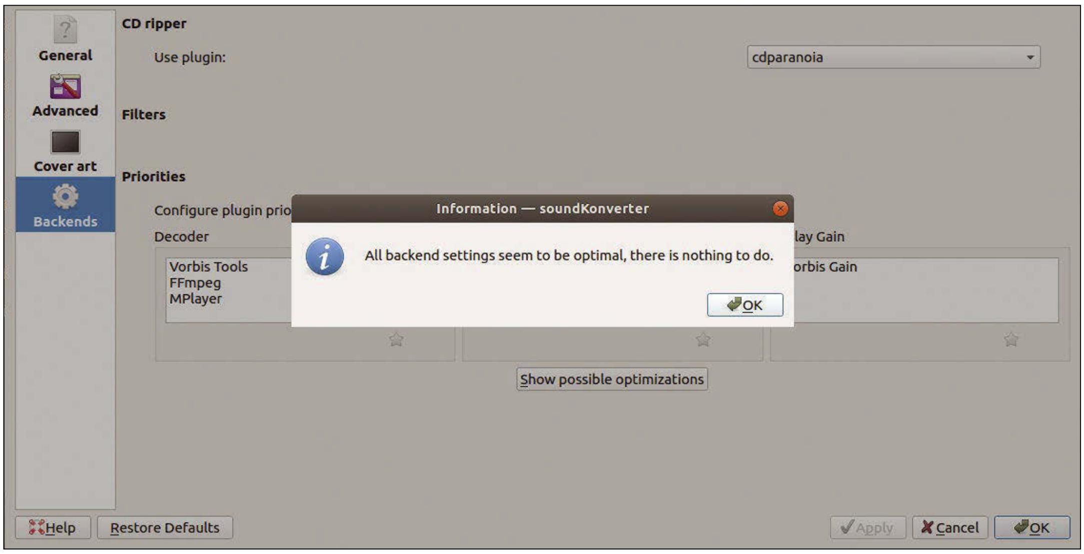
If you load a file directory instead of ripping an audio CD, soundKonverter skips the dialog for defining the readable data and immediately transfers the available audio files into the specified file directory in the main window. After setting the target format and volume adjustment, begin the conversion by clicking *Start*.

Profiles

Internally, the program works with profiles that make ripping and transcoding far easier, especially for large music collections. You can access these profiles in the menu from *Settings | Configure soundKonverter*. The *General* section has some basic options for the user interface, conversion, and replay gain tool. In the *User Interface* area, one of the predefined profiles shows in the *Default profile* selection box.

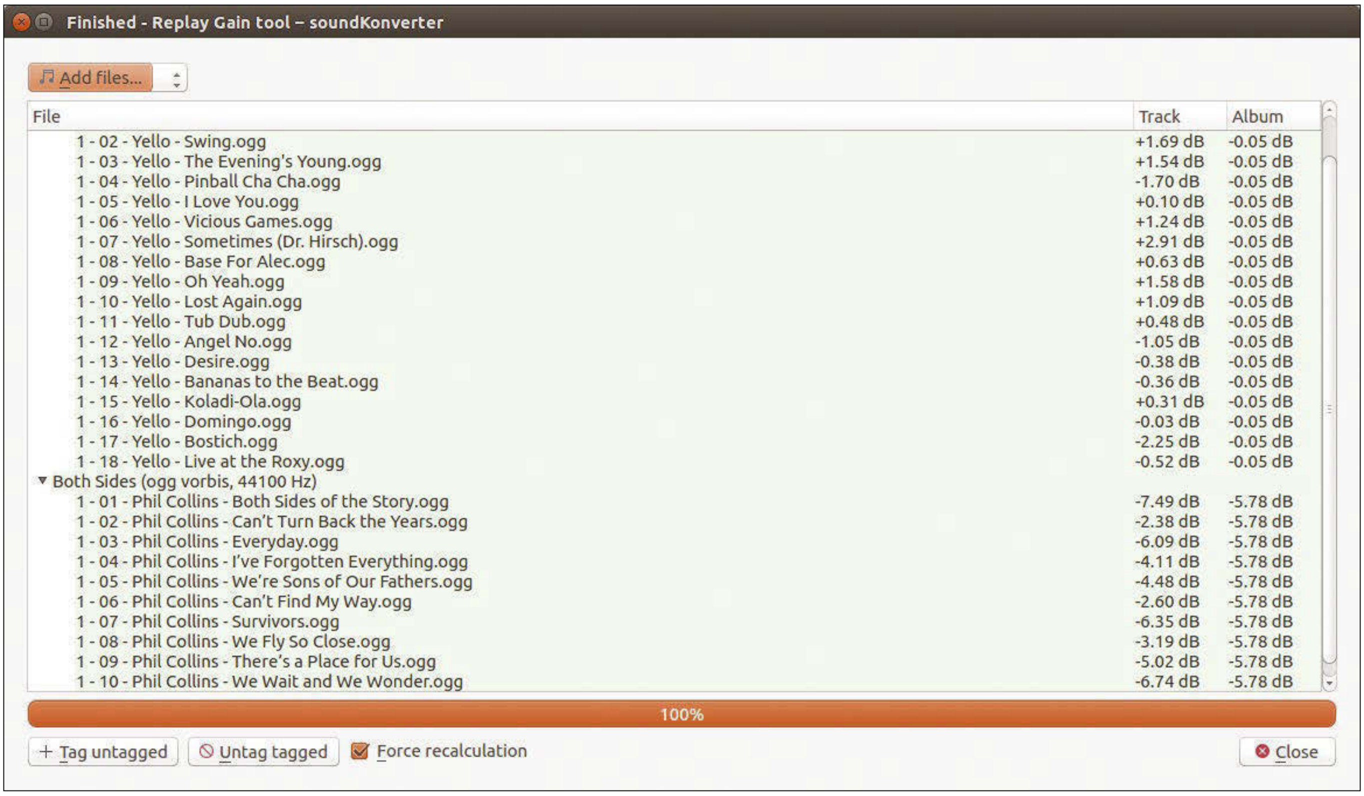
The settings in the *Cover Art* group allow the inclusion of cover images. Under *Advanced*, you can enter metadata for Ogg-Vorbis and FLAC files. In the *Backends* section, you can define which encoders and decoders are used, which program

Figure 4: Checking the back-end settings in the soundKonverter Settings dialogs.



CD ripper
Use plugin: cdparanoia
Filters
Priorities
Configure plugin prio
Decoder
Vorbis Tools
FFmpeg
MPlayer
lay Gain
orbis Gain

Information — soundKonverter
All backend settings seem to be optimal, there is nothing to do.



File	Track	Album
1 - 02 - Yello - Swing.ogg	+1.69 dB	-0.05 dB
1 - 03 - Yello - The Evening's Young.ogg	+1.54 dB	-0.05 dB
1 - 04 - Yello - Pinball Cha Cha.ogg	-1.70 dB	-0.05 dB
1 - 05 - Yello - I Love You.ogg	+0.10 dB	-0.05 dB
1 - 06 - Yello - Vicious Games.ogg	+1.24 dB	-0.05 dB
1 - 07 - Yello - Sometimes (Dr. Hirsch).ogg	+2.91 dB	-0.05 dB
1 - 08 - Yello - Base For Alec.ogg	+0.63 dB	-0.05 dB
1 - 09 - Yello - Oh Yeah.ogg	+1.58 dB	-0.05 dB
1 - 10 - Yello - Lost Again.ogg	+1.09 dB	-0.05 dB
1 - 11 - Yello - Tub Dub.ogg	+0.48 dB	-0.05 dB
1 - 12 - Yello - Angel No.ogg	-1.05 dB	-0.05 dB
1 - 13 - Yello - Desire.ogg	-0.38 dB	-0.05 dB
1 - 14 - Yello - Bananas to the Beat.ogg	-0.36 dB	-0.05 dB
1 - 15 - Yello - Koladi-Ola.ogg	+0.31 dB	-0.05 dB
1 - 16 - Yello - Domingo.ogg	-0.03 dB	-0.05 dB
1 - 17 - Yello - Bostich.ogg	-2.25 dB	-0.05 dB
1 - 18 - Yello - Live at the Roxy.ogg	-0.52 dB	-0.05 dB
▾ Both Sides (ogg vorbis, 44100 Hz)		
1 - 01 - Phil Collins - Both Sides of the Story.ogg	-7.49 dB	-5.78 dB
1 - 02 - Phil Collins - Can't Turn Back the Years.ogg	-2.38 dB	-5.78 dB
1 - 03 - Phil Collins - Everyday.ogg	-6.09 dB	-5.78 dB
1 - 04 - Phil Collins - I've Forgotten Everything.ogg	-4.11 dB	-5.78 dB
1 - 05 - Phil Collins - We're Sons of Our Fathers.ogg	-4.48 dB	-5.78 dB
1 - 06 - Phil Collins - Can't Find My Way.ogg	-2.60 dB	-5.78 dB
1 - 07 - Phil Collins - Survivors.ogg	-6.35 dB	-5.78 dB
1 - 08 - Phil Collins - We Fly So Close.ogg	-3.19 dB	-5.78 dB
1 - 09 - Phil Collins - There's a Place for Us.ogg	-5.02 dB	-5.78 dB
1 - 10 - Phil Collins - We Wait and We Wonder.ogg	-6.74 dB	-5.78 dB

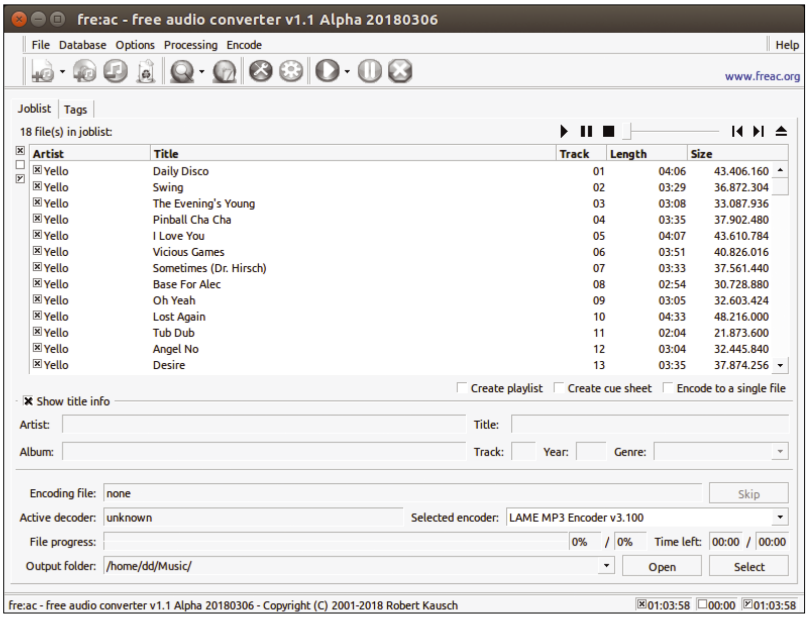
Figure 5: The replay gain tool in soundKonverter lets you adjust the volume of different tracks to match.

rips audio CDs, and which plugins you want to activate for volume adjustment (replay gain).

Linux has a nearly unmanageable variety of command-line encoders and decoders, which can sometimes lead to incompatibilities. soundKonverter therefore allows you to check your selection of back ends for optimal interaction. Choose the *Backends* tab and click *Show possible optimizations*; the software checks the combination of back ends you have selected and then displays the results in a small pop-up window (Figure 4).

If the software is not functioning as expected, call up the logfile under *Conversion | View logs*. Missing plugins or back ends are highlighted in red.

Figure 6: The numerous functions in the fre:ac program window indicate a higher level of complexity than is found in soundKonverter.


fre:ac - free audio converter v1.1 Alpha 20180306
File Database Options Processing Encode
Help
www.freac.org
Joblist
Tags
Skip
Open
Select

18 file(s) in joblist:

Artist	Title	Track	Length	Size
☒ Yello	Daily Disco	01	04:06	43.406.160
☒ Yello	Swing	02	03:29	36.872.304
☒ Yello	The Evening's Young	03	03:08	33.087.936
☒ Yello	Pinball Cha Cha	04	03:35	37.902.480
☒ Yello	I Love You	05	04:07	43.610.784
☒ Yello	Vicious Games	06	03:51	40.826.016
☒ Yello	Sometimes (Dr. Hirsch)	07	03:33	37.561.440
☒ Yello	Base For Alec	08	02:54	30.728.880
☒ Yello	Oh Yeah	09	03:05	32.603.424
☒ Yello	Lost Again	10	04:33	48.216.000
☒ Yello	Tub Dub	11	02:04	21.873.600
☒ Yello	Angel No	12	03:04	32.445.840
☒ Yello	Desire	13	03:35	37.874.256

□ Create playlist □ Create cue sheet □ Encode to a single file

☒ Show title info

Artist: Title:

Album: Track: Year: Genre:

Encoding file: none

Active decoder: unknown Selected encoder: LAME MP3 Encoder v3.100

File progress: 0% / 0% Time left: 00:00 / 00:00

Output folder: /home/dd/Music/

fre:ac - free audio converter v1.1 Alpha 20180306 - Copyright (C) 2001-2018 Robert Kausch ☒01:03:58 □00:00 ☑01:03:58

Volume

soundKonverter also lets you adjust the volume of existing audio files in different formats. To adjust the volume, choose *Tools | Replay Gain tool*. In the new window, select the directory where the music files are located.

soundKonverter then queries the output formats, and the module enables the formats it supports. All files then appear in a tree view in the title list. You can start the volume adjustment by clicking *Force recalculation* (Figure 5).

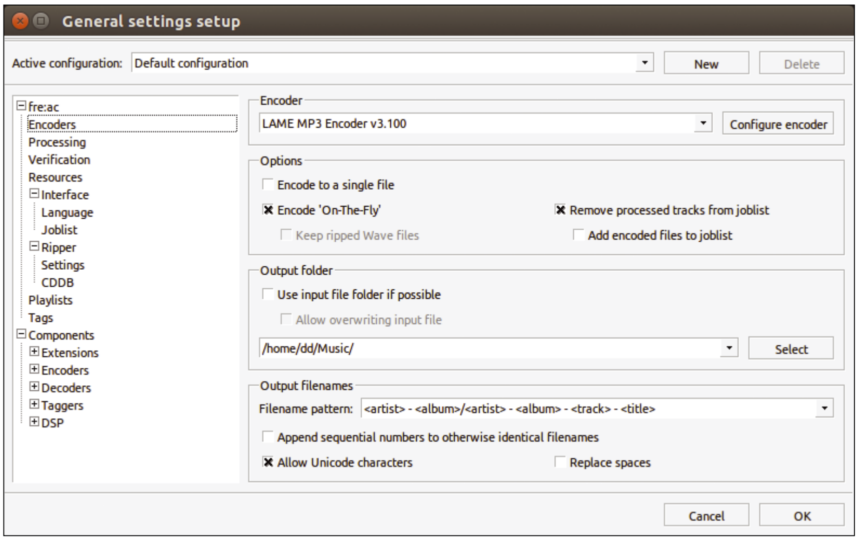
fre:ac

fre:ac, developed by Robert Kausch, has been in development for more than 15 years. Despite a low version number (1.0.31a), the cross-platform audio tool is up to date, with support for many formats, multithreading, and a CDDB/freedb connection.

fre:ac provides good performance and an impressive range of features, including support for cue sheets and the ability to extract an audio file from a video recording.

Getting Started

You can obtain fre:ac for 32- and 64-bit architectures as a tarball from the developer's website [2].



Active configuration: Default configuration

Encoder: LAME MP3 Encoder v3.100

Options

- ☐ Encode to a single file
- ☒ Encode 'On-The-Fly'
- ☐ Keep ripped Wave files
- ☒ Remove processed tracks from joblist
- ☐ Add encoded files to joblist

Output folder

- ☐ Use input file folder if possible
- ☐ Allow overwriting input file

/home/dd/Music/

Output filenames

Filename pattern: <artist> - <album>/<artist> - <album> - <track> - <title>

- ☐ Append sequential numbers to otherwise identical filenames
- ☒ Allow Unicode characters
- ☐ Replace spaces

Figure 7: The fre:ac configuration window offers a variety of possibilities to adapt the converter to your own needs.

The project specifies at least a Pentium 4 or Athlon 64 processor and 512 MB of RAM. The minimum available hard disk space is just 100MB.

After unpacking the archive, move the newly created program directory to a folder of your choice and optionally create a launcher in the menu hierarchy. To start fre:ac from the program directory, click on the `freac` file.

At first glance, you can see that fre:ac is much more complex than soundKonverter. In addition to the usual menubar and buttonbar, the main window has a list area at the top for the selected audio tracks and an info area at the bottom that displays information on the current track, the selected encoder and decoder, and the output path (Figure 6).

To configure fre:ac, select *Options | General Settings*. The extensive configuration dialog lets you make granular adjustments. For example, you can specify a software encoder in the *Encoders* section (Figure 7). Unlike soundKonverter, fre:ac includes its own converters and is not dependent on other converters installed on the system. You can access the encoder settings by clicking on the *Configure encoder* button to the right of the Encoder selection box.

In the *Ripper | CDDB* dialog, you can define whether or not to query a CDDB/freedb server for access to cover images, album track lists, and sometimes lyrics, which the software correctly tags to tracks.

Use the *Playlists* tab to create playlists in the standard M3U format that most audio players can read. As a special feature, the software also generates a cue sheet (see box "No Interruptions").

Pressing *OK* applies the changes. After loading an audio CD, fre:ac reads it and lists the contents in the main window. By default, it checks the boxes to enable all tracks for ripping. Clicking on a track displays the metadata associated with the track in the information fields at the bottom of the window; you can edit or add to the data as needed.

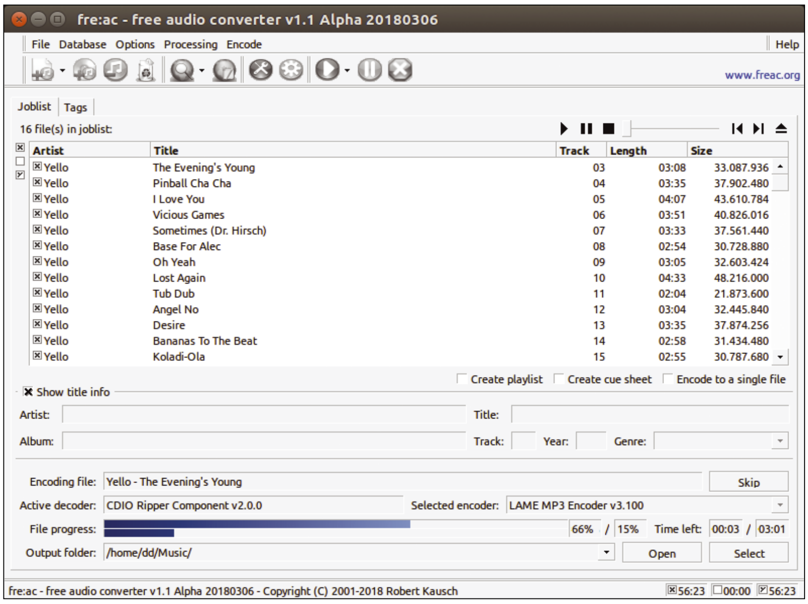
To create a personal playlist, uncheck some of the individual titles in the list. A click on the *Play* button in the top toolbar starts reading and converting to the selected target format. Alternatively, you can play back individual tracks by clicking the *Play* button directly above the list area. Use the buttons to the right to pause, stop, and jump to the next or previous track. If you mouse over one of the entries, a flyout appears with information about the album and the title, including the sample rate and resolution, track length, and bit rate.

After selecting tracks, start the process of reading the CD. In the lower area, the software displays a two-line horizontal progress bar; the upper half symbolizes the progress in ripping and

No Interruptions

Especially for classical music, live recordings, or concept albums, gaps between tracks in compressed formats like MP3 are annoying. fre:ac lets you handle this problem in an elegant way. Slot in the desired CD and enable the checkboxes next to *Create cue sheets* and *Create only a single playlist and/or cue sheet file per conversion*. The software then converts the entire CD to a single file and writes track boundaries and additional information to a corresponding cue sheet.

Players like the VLC media player process such cue sheets as playlists and support jumping between titles. If you want to burn the file to a CD, the cue sheet will also help: For example, it can be used as the source in the K3b disk burning program instead of audio files. K3b would break down the audio track on the CD into individual tracks once again. By the way, you can revert this with fre:ac, which also reads cue sheets and splits the file back into its original components, if desired.



16 file(s) in joblist:

Artist	Title	Track	Length	Size
☒ Yello	The Evening's Young	03	03:08	33.087.936
☒ Yello	Pinball Cha Cha	04	03:35	37.902.480
☒ Yello	I Love You	05	04:07	43.610.784
☒ Yello	Vicious Games	06	03:51	40.826.016
☒ Yello	Sometimes (Dr. Hirsch)	07	03:33	37.561.440
☒ Yello	Base For Alec	08	02:54	30.728.880
☒ Yello	Oh Yeah	09	03:05	32.603.424
☒ Yello	Lost Again	10	04:33	48.216.000
☒ Yello	Tub Dub	11	02:04	21.873.600
☒ Yello	Angel No	12	03:04	32.445.840
☒ Yello	Desire	13	03:35	37.874.256
☒ Yello	Bananas To The Beat	14	02:58	31.434.480
☒ Yello	Koladi-Ola	15	02:55	30.787.680

□ Create playlist □ Create cue sheet □ Encode to a single file

☒ Show title info

Artist: Title:

Album: Track: Year: Genre:

Encoding file: Yello - The Evening's Young

Active decoder: CDIO Ripper Component v2.0.0 Selected encoder: LAME MP3 Encoder v3.100

File progress: 66% / 15% Time left: 00:03 / 03:01

Output folder: /home/dd/Music/

fre:ac - free audio converter v1.1 Alpha 20180306 - Copyright (C) 2001-2018 Robert Kausch ☒56:23 □00:00 ☑56:23

Figure 8: The double bars in the fre:ac info section show the progress in processing the current job. The upper half shows the individual track, the lower half the overall status.

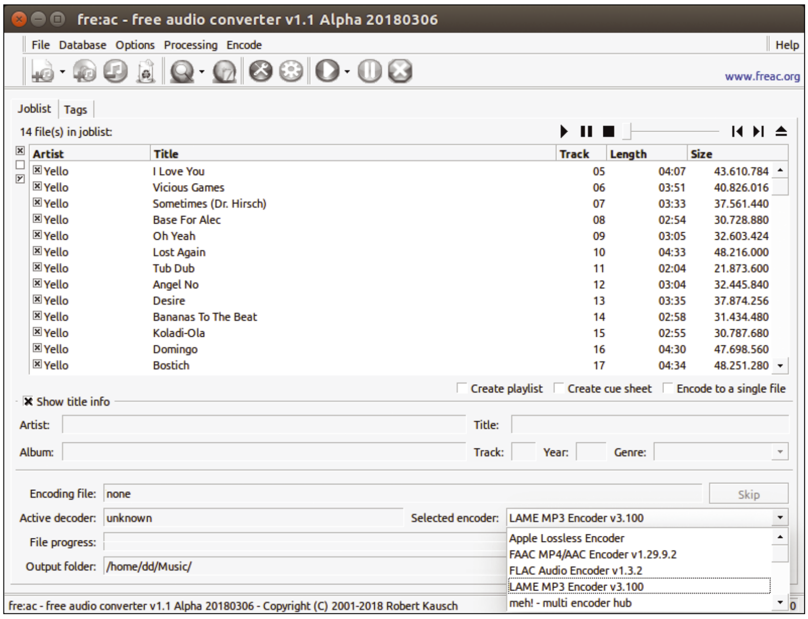
transcoding the current title; the lower half shows the overall progress (Figure 8).

When transcoding, fre:ac creates a separate subfolder for the audio files in the target directory. If you want to transcode different tracks with different encoders or different qualities, highlight them individually and make a choice in the *Selected encoder* field. If necessary, you can then adjust the encoding quality with *Options | Configure selected encoder* menu.

Only one track can be transcoded at a time with this procedure; fre:ac then stores the tracks in the subfolder for the audio CD. The software deletes the ripped tracks from the list after completion.

By default, fre:ac uses the existing replay gain settings in the metadata of each track to determine the volume. At the moment, the software lacks a function to adjust the volume itself. When asked, the developer said this feature will probably be available in an upcoming version.

Figure 9: fre:ac offers 14 target formats.



14 file(s) in joblist:

Artist	Title	Track	Length	Size
☒ Yello	I Love You	05	04:07	43.610.784
☒ Yello	Vicious Games	06	03:51	40.826.016
☒ Yello	Sometimes (Dr. Hirsch)	07	03:33	37.561.440
☒ Yello	Base For Alec	08	02:54	30.728.880
☒ Yello	Oh Yeah	09	03:05	32.603.424
☒ Yello	Lost Again	10	04:33	48.216.000
☒ Yello	Tub Dub	11	02:04	21.873.600
☒ Yello	Angel No	12	03:04	32.445.840
☒ Yello	Desire	13	03:35	37.874.256
☒ Yello	Bananas To The Beat	14	02:58	31.434.480
☒ Yello	Koladi-Ola	15	02:55	30.787.680
☒ Yello	Domingo	16	04:30	47.698.560
☒ Yello	Bostich	17	04:34	48.251.280

□ Create playlist □ Create cue sheet □ Encode to a single file

☒ Show title info

Artist: Title:

Album: Track: Year: Genre:

Encoding file: none

Active decoder: unknown Selected encoder: LAME MP3 Encoder v3.100

File progress:

Output folder: /home/dd/Music/

fre:ac - free audio converter v1.1 Alpha 20180306 - Copyright (C) 2001-2018 Robert Kausch

Converting Audio Files

If you want to convert music files from the lossless but memory-intensive FLAC format to Ogg-Vorbis, which is lossy but produces significantly smaller files, you can load them into the list view with the *Add audio file(s)* button in the upper-left corner of the buttonbar.

Select the target format in the *Selected encoder* selection box (Figure 9) and adjust the output quality, if necessary; then, click on *Start the encoding process*. fre:ac stores the converted files in the target directory.

If you select a lossless target format for a lossy source format, fre:ac points out that no quality improvement is expected for the target format, but the file size is expected to increase. The dialog asks if you would like to select a different target format.

fre:ac completes conversions between different file formats very quickly; a single track in the target format appears after just a few seconds.

Playlists

fre:ac creates playlists of the transcoded titles in the standardized M3U format, which most players can process. To generate playlists automatically, activate the *Create playlist* option. fre:ac creates a playlist for the tracks in the subdirectories corresponding to the albums.

Conclusions

Both soundKonverter and fre:ac are powerful programs for ripping music CDs and converting audio files. Speedy operation and very good results make these two tools stand out. soundKonverter is more appealing for occasional users, but fre:ac offers a significantly greater range of functions – mainly with the large number of configuration options for encoders and decoders.

fre:ac also offers excellent multithreading on modern computers, with configurable parallel processing. As a unique feature, soundKonverter has automatic volume adjustment, a feature that is currently under development in fre:ac. The fre:ac tool offers other unique advantages, such as the ability to create cue sheets or extract audio tracks from video files. ■■■

Info

[1] soundKonverter: *https://github.com/dfaust/soundkonverter*

[2] fre:ac: *https://www.freac.org/*

Celebrating 25 Years of Linux!

ORDER NOW and SAVE 25% on 7 years of *Ubuntu User*!

THE COMPLETE

UBUNTU user

ARCHIVE

Over 3,000 PAGES! 7 GREAT YEARS OF UBUNTU USER

Searchable DVD!

All Content Available in Both HTML and PDF Formats

ENTER CODE: SAVE25

shop.linuxnewmedia.com

Drawing on Command

Mermaid lets you create diagrams from simple text-based statements.

BY FRANK HOFMANN & MANDY NEUMEYER

If you want to simplify your documentation and avoid using complex tools like Visio, Dia, or Inkscape for displaying charts and diagrams, Mermaid [1] might be just right for you.

Mermaid, which is based on JavaScript, is similar to open source tools such as Graphviz [2] and JS Sequence Diagrams [3]. These tools generate flow charts and other useful diagrams based on simple text-based commands. Mermaid focuses on UML sequence diagrams, and it supports different output styles that are reminiscent of manual drawings.

Mermaid follows the minimalist concept of Markdown and AsciiDoc formats. This approach opens the door to automated document generation and website integration. The Mermaid project is still relatively unpolished. As you'll learn in this article, the local version has some issues that make it hard to depend on for real production work, but you can use the online Mermaid Live Editor at the project website [4] to generate diagrams and explore the Mermaid command syntax.

The Mermaid project is developed and maintained by Scandinavian Knut Sveidqvist and is licensed under the MIT license. The development is a community effort anchored by the project page on GitHub [1].

In Detail

Mermaid is available as a standalone tool for the command line and as a JavaScript library for website integration. Using the Live Editor on the Mermaid website, you can test the Mermaid functions before you install to learn what the description language can do. Listing 1 contains the flowchart with which Figure 1 was created.

Listing 1: Building a Mermaid Flowchart

```
graph TD;
A[Move] -->|Define Date| B(Rent a van from the moving company);
B --> C{Pack boxes};
C -->|15 boxes| D[Livingroom];
C -->|5 boxes| E[Kitchen/Bath];
C -->|4 boxes| F[Bedroom];
```

Mermaid is based on the Markdown text formatting language, but it goes one step further and transfers the Markdown concept to flow, sequence, and Gantt diagrams. The Mermaid syntax is more economical than Graphviz without compromising functionality.

As Listing 1 shows, you can use simple instructions to describe each element in a diagram. You need to terminate each line of the description with a semicolon, but the interpreter is forgiving in case of errors. After declaring the diagram type, the individual nodes and the *edges* follow; the edges are the references that show how the nodes relate to each other and what they look like.

Additional text and brackets define the basic representation of the nodes, edges, and labels. With the help of Cascading Style Sheets (CSS), you can then put the finishing touches on the image; Mermaid comes with three pre-built templates.

Chart Types

Mermaid offers a visualization feature for graphs or trees, sequences, Gantt and class diagrams,

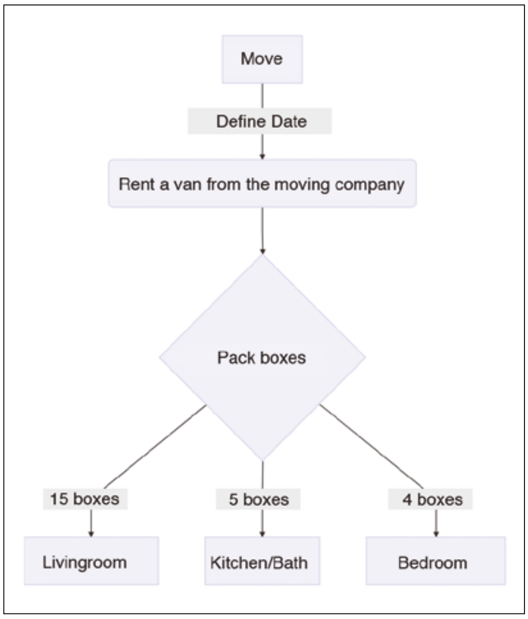
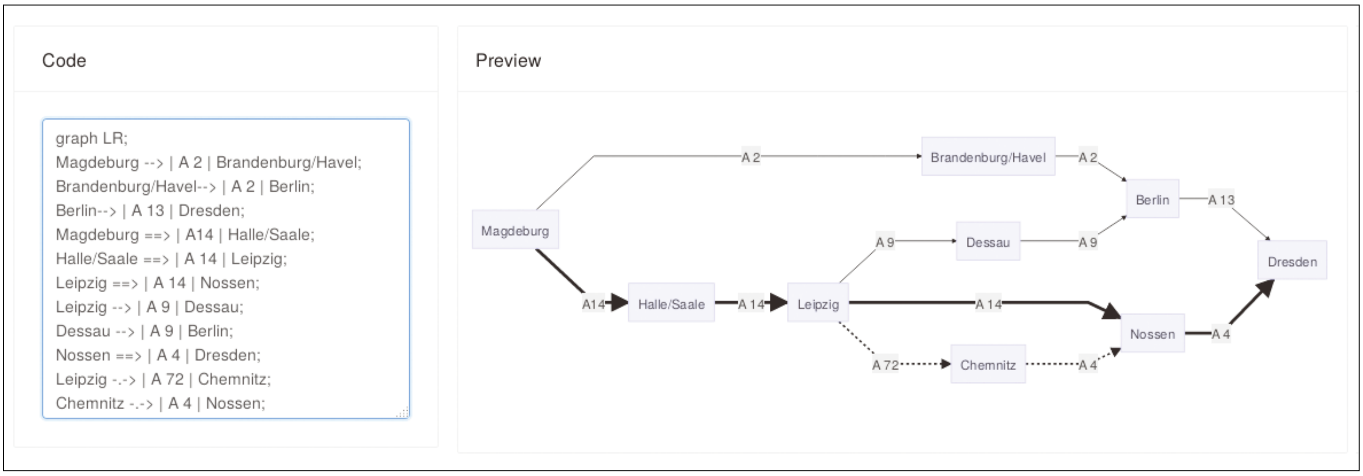


Figure 1: Mermaid only needs a few instructions to create a simple diagram for moving your home.



Code

```
graph LR;
Magdeburg --> | A 2 | Brandenburg/Havel;
Brandenburg/Havel--> | A 2 | Berlin;
Berlin--> | A 13 | Dresden;
Magdeburg ==> | A14 | Halle/Saale;
Halle/Saale ==> | A 14 | Leipzig;
Leipzig ==> | A 14 | Nossen;
Leipzig --> | A 9 | Dessau;
Dessau --> | A 9 | Berlin;
Nossen ==> | A 4 | Dresden;
Leipzig -.-> | A 72 | Chemnitz;
Chemnitz -.-> | A 4 | Nossen;
```

Figure 2: A route generated as a flowchart with Mermaid.

and Git graphs. Each chart is introduced with a keyword, such as `graph`, `sequenceDiagram`, or `gantt`.

For a graph, you also need to specify the reading direction. `TD` stands for "top down," `TB` for "top bottom," and `BT` for "bottom top." `RL` means "right left," and `LR` means "left right" (from left to right). Listing 1 describes a graph for a home moving project that you read from top to bottom and that thus has a `TD` label.

The following examples show how you can use Mermaid to implement various graphical images. All images are taken from the Live Editor on the project's website.

Enter the description in the text field on the left, and the tool generates an image in real time on the right. To use this image in other documents, you can download it as a file in SVG format or take a screenshot of the image.

Graphs and Trees

Figure 2 shows a graph for possible routes from the German city Magdeburg to Dresden. You can read this graph from left to right (`LR`).

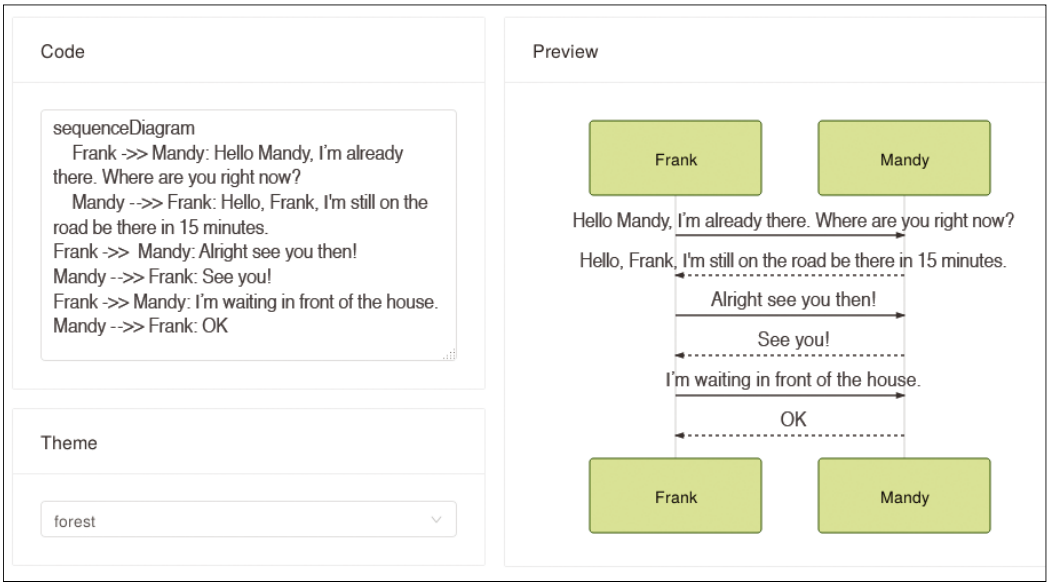
You need to consider several factors when formulating a graph. For example, an edge with an arrowhead has the form `-->` for the normal line width, `==>` produces a thicker line, and `-.-` produces a dotted line. You do not need to add any spaces between the node text and the beginning or the end of the edge, but spaces do make the description more legible. Enclose the description text for the edge either in pipe characters (`Berlin --> | A 13 | Dresden`) or write it directly into the edge (`Berlin --> A 13 Dresden`).

Node identifiers with special characters need to be in quotes. If you enter an identifier in the form `D[living room]`, Mermaid remembers the abbreviation (`D` in this case), which you can then use as a reference later on. The type of parentheses determines how the node is displayed: Square brackets create rectangles, curly brackets create rhombuses, and round brackets create rectangles with rounded corners.

Sequence Diagram

A sequence diagram describes how processes, programs, instances, or even people interact with each other (Figure 3). The communication sequence follows the `sequenceDiagram` keyword. The sender and receiver correspond to the nodes in the graphs. Each message is in a single line. You use the arrowhead to indicate who is transmitting something to whom. Place a colon between the recipient and the message, and you can add additional space to improve readability.

Figure 3: In the sequence diagram, Mermaid traces the course of a conversation typical of instant messengers.



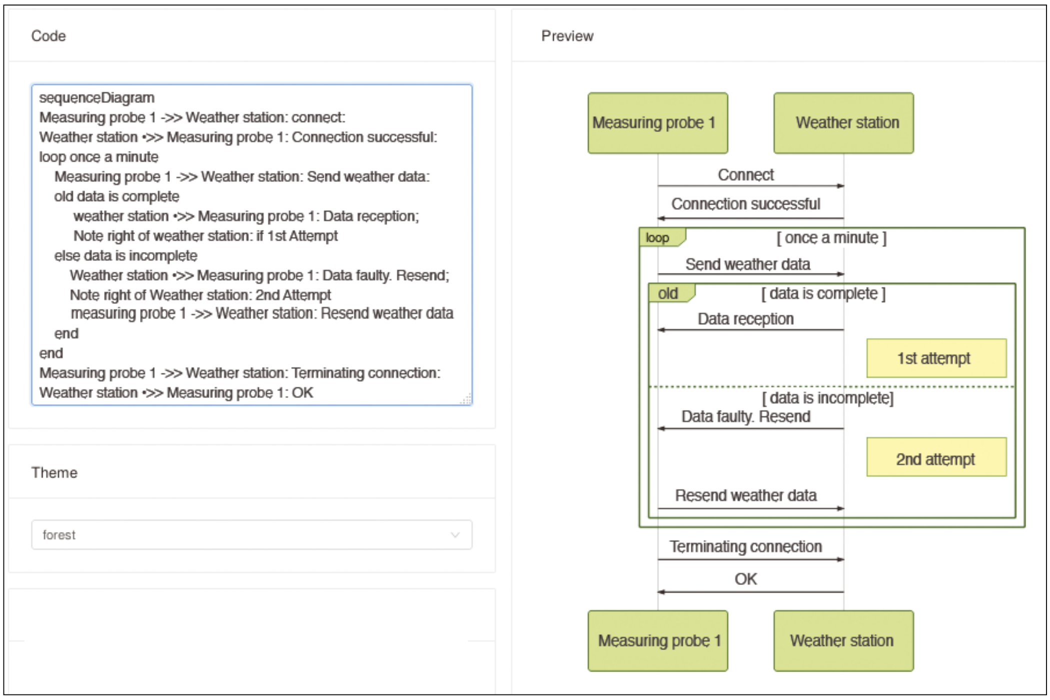
Code

```
sequenceDiagram
   Frank ->> Mandy: Hello Mandy, I’m already
there. Where are you right now?
   Mandy -->> Frank: Hello, Frank, I'm still on the
road be there in 15 minutes.
Frank ->> Mandy: Alright see you then!
Mandy -->> Frank: See you!
Frank ->> Mandy: I’m waiting in front of the house.
Mandy -->> Frank: OK
```

Theme

forest

Preview
Frank
Mandy
Hello Mandy, I’m already there. Where are you right now?
Hello, Frank, I'm still on the road be there in 15 minutes.
Alright see you then!
See you!
I’m waiting in front of the house.
OK
Frank
Mandy



```
sequenceDiagram
Measuring probe 1 ->> Weather station: connect:
Weather station ->> Measuring probe 1: Connection successful:
loop once a minute
   Measuring probe 1 ->> Weather station: Send weather data:
   old data is complete
       weather station ->> Measuring probe 1: Data reception;
       Note right of weather station: if 1st Attempt
   else data is incomplete
      Weather station ->> Measuring probe 1: Data faulty. Resend;
      Note right of Weather station: 2nd Attempt
      measuring probe 1 ->> Weather station: Resend weather data
   end
end
Measuring probe 1 ->> Weather station: Terminating connection:
Weather station ->> Measuring probe 1: OK
```

Figure 4: The sequence diagram shows a simple log for a weather station with a loop and an if condition.

Mermaid goes beyond simple sequence diagrams. Other possibilities include notes, activations, loops, conditional blocks, and optional entries. In Figure 4, the weather station sensor used as an example collects data and sends it to the station once a minute, provided it is active and there is a connection to the weather station. If problems occur during transfer, the sensor repeats the data transmission.

A loop for repetition consists of a `loop-end` block. You formulate a condition with `alt-else-end`. You can add notes with the `Note` keyword and a position specification. This tells Mermaid whether to place the note in the image to the right or left of the node.

Gantt Diagram

A Gantt bar chart shows the chronological sequence of events and project steps (Figure 5).

After the `gantt` keyword, you assign a program `title` and specify the `dateformat`. For each subproject, define a heading using the `section` keyword; in Figure 5, the three areas are named `Weather Data Evaluation`, `Hardware Procurement`, and `Documentation`.

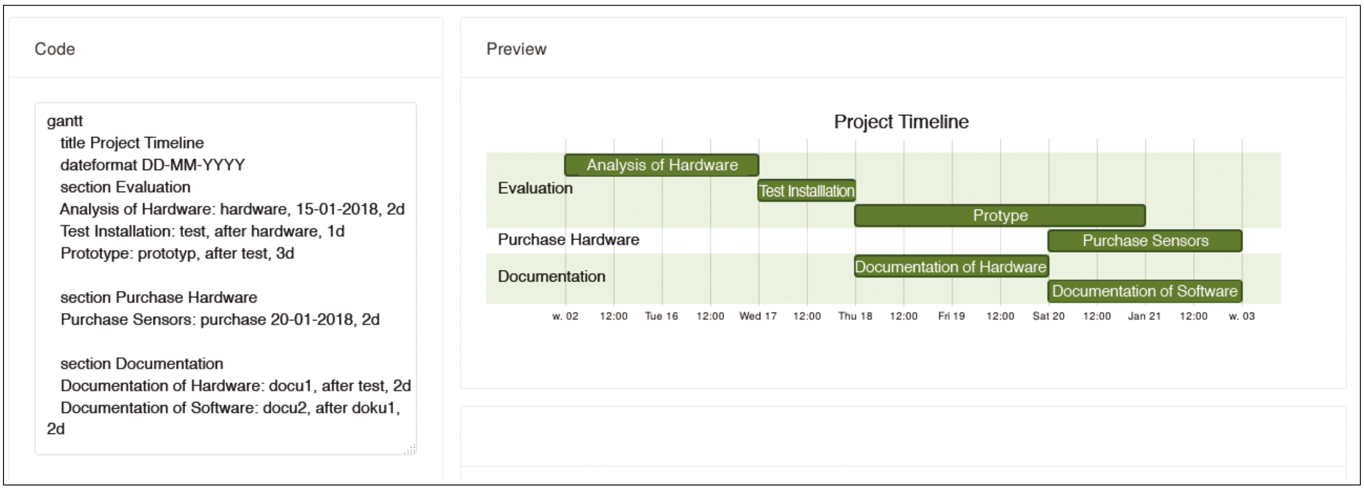
For each section of the subproject, assign a title that appears inside the box. If the text does not fit in the box, it will be displayed next to the box. Each section defines an identifier – i.e., a unique name – as well as a position and a time specification. You can use either an exact time or a value of the form `after section_name` as position specifiers.

Setting Up Mermaid

Mermaid packages are available for some Linux distributions, but other distros haven't yet added the packages to their repositories. See your Linux distro's package manager to see if a package is available.

If you can't find Mermaid in your distro's package repository, you can still install it using the Node Package Manager (npm) [5] from the JavaScript framework Node.js [6]. To set up Node.js 9 and Mermaid under Debian, Ubuntu, or Linux Mint, run the five commands in Listing 2 as an administrative user.

The first two steps install Node.js. The third command loads the `build-essential` package, which contains important tools for building



```
gantt
   title Project Timeline
   dateformat DD-MM-YYYY
   section Evaluation
   Analysis of Hardware: hardware, 15-01-2018, 2d
   Test Installation: test, after hardware, 1d
   Prototype: prototyp, after test, 3d

   section Purchase Hardware
   Purchase Sensors: purchase 20-01-2018, 2d

   section Documentation
   Documentation of Hardware: docu1, after test, 2d
   Documentation of Software: docu2, after doku1,
2d
```

Figure 5: Representing a project timeline as a Gantt diagram.

Listing 2: Installing Mermaid

```
# curl -sL https://deb.nodesource.com/setup_9.x | sudo -E bash -
# apt-get install nodejs
# apt-get install build-essential
# npm install -g mermaid
# npm install -g mermaid-cli
```

software. The last two steps set up Mermaid with npm, both as a JavaScript framework and as a command-line tool.

After the installation, you can use the `mermaid` command, to which you pass the desired output format of the graphic and the file with the diagram description when calling. For example, the following command creates a graphic in PNG format from the source code file `flowchart.txt`:

```
$ mermaid --png flowchart.txt
```

To generate an SVG file instead, specify `-s` or `--svg` as the output format parameters. In the case of this example, the generated image file goes by the name of `flowchart.png` or `flowchart.svg`.

In our lab, we discovered that the command-line tool offers only a subset of the options offered by the online version. In addition, the generated images differ from the output in the online editor: For example, edge labels and arrowheads are missing. The local Mermaid also constantly trips up over non-standard characters and brackets in descriptions. Another weak point is the design of the diagrams. The command-line tool does not currently let you include your own style sheets. Thus, the output is limited to the three styles provided.

The add-on module `mermaid-filter` [7], which uses pandoc to help Markdown understand Mermaid statements directly while translating from Markdown to HTML and thus builds the diagrams in the background, could also do with a revision. The description sounds promising, but the installation failed on Debian 8/9. The reason for this remains unclear.

To work around these problems, you can only use the SVG format images generated by the online editor. Automation only partially succeeds; your only option is to prepare the images or turn to Graphviz. If you want to process the output of Mermaid elsewhere in your documents, you must therefore use the basically less suitable PNG format.

Both the online version and the installable version produced faulty vector graphics: Common web browsers display them reasonably correctly, but programs like Gimp, LibreOffice Draw, Inkscape, or Adobe Illustrator can't do anything with them. However, since the less buggy online version only offers SVG as an export format, you have no choice but to continue working with screenshots.

Conclusions

The idea behind Mermaid seems captivating: Graphs and diagrams are compiled in the form of a text description, provided with a style sheet, and then translated into an image. The text format ensures easy composition, and it also enables uncomplicated archiving of the data in a version control system.

Currently, the local version has several problems that cloud the picture, including installation issues, incorrect rendering of special characters, and incorrect edge labels. The online edition also had some issues with unusable vector graphics. The Mermaid project is on the right track, and if you're interested in an easy tool for generating simple charts and diagrams, you'll enjoy exploring and experimenting with the online editor: In its current state, however, Mermaid is not really reliably enough for steady production use.

The Authors

Frank Hofmann works from Berlin, Geneva, and Cape Town as a developer, trainer, and author. He is the coauthor of the *Debian Package Management* book (*https://www.dpmb.org/*).

Mandy Neumeyer works in the tourism industry and has lived in South Africa for nine years. She is currently building up additional income as a digital nomad.

Info

[1] Mermaid: *https://mermaidjs.github.io/*

[2] Graphviz: *https://www.graphviz.org/*

[3] JS Sequence Diagrams: *https://bramp.github.io/js-sequence-diagrams/*

[4] Mermaid Live Editor: *https://mermaidjs.github.io/mermaid-live-editor/*

[5] npm: *https://www.npmjs.com/*

[6] Node.js: *https://nodejs.org/*

[7] mermaid-filter: *https://github.com/raghur/mermaid-filter*

FOSSPicks

Sparkling gems and new releases from the world of Free and Open Source Software

Graham tears himself away from updating Arch Linux to search for the best new free software. BY GRAHAM MORRISON

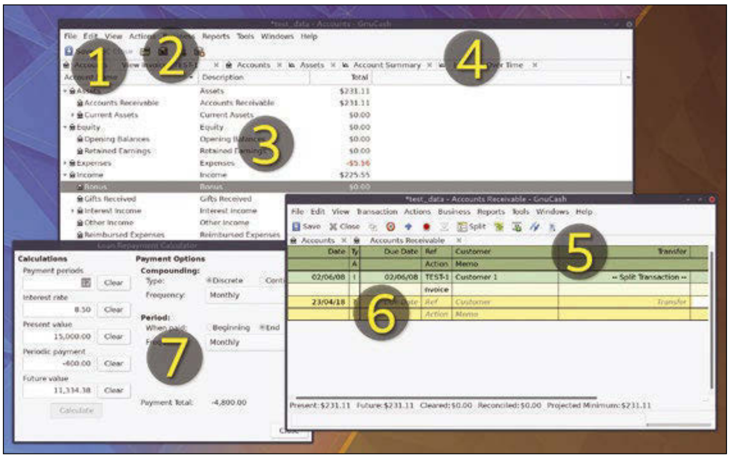
Money manager

GnuCash 3

It's not and never has been the most exciting subject for most of us, but keeping on top of your personal finances is an essential life skill. Perhaps the most important part of that skill is looking into the box of your own (or your business') finances, and one great way of making that more interesting is to use some clever software to help take both the monotony and the complexity out of the problem. That all this has to do with money is perhaps why there isn't a great deal of open source software from which to choose, while there is a great deal of proprietary software and services that will offer to help for a fee. However, for many years, there has been GnuCash. While it's never been the easiest or most intuitive point of entry for taking back financial control, it's always been very capable, powerful, and free.

GnuCash uses something called double-entry bookkeeping to handle your finances. This approach doesn't have the immediacy or the simplicity of a modern commercial application. It basically means that you need to balance every transaction as you enter them into a ledger. Money comes in from one place and leaves from another, and two transactions (the double entry) are listed to show this flow of money. It has the advantage of acting a little like a checksum value as it stops you from making a mistake in one entry, because it won't balance with the second. It gives you great insight into how credits and debits flow through your accounts. However, double entry doesn't have the same autonomous approach taken by modern software or services, but that is definitely a good thing if you like control. GnuCash has always been comprehensive and capable of helping with tax returns and the complex mix of input and output, investments, and liabilities that most of us juggle. But until this release, it's always been consigned to the ancient era of Gtk+ user interfaces.

It's fitting then that this is a major milestone release with version number 3.0, because the main addition is the long-worked-on migration to GTK+ 3.0. This single update, brings all the niceties that we now take for granted in many modern Gnome applications, including its refined visuals and input control. But there are many other updates in addition to this. You can generate four new reports, for example, and the CSV importer that many of us rely on to sync GnuCash with our online bank statements now actually works. Regular expressions can be used to filter transactions, and prices can now have 18 characters of precision, if you need them. Small features like accounts being referred to internally using UUIDs rather than names and much improved chart rendering also help bring GnuCash into the 21st century. At the same time, GnuCash has received a rash of updates to keep the old application ticking over, so while there's no rush to perform the upgrade, if you've always been too cautious to trust your finances to an old Gnome application, now is the time to give the new version a go!



1 **Import data:** Use clever Bayesian filters to import your bank's exported files or your own spreadsheets. 2 **GTK+ 3:** Everything looks so much nicer now that GnuCash has upgraded its back end. 3 **Double entry:** As ever, the key to using GnuCash is balancing your inputs and your outputs. 4 **Tabbed views:** Open as many different reports or account lists as you need. 5 **Split transactions:** Input and output can be split according to their source and destination. 6 **Invoice generation:** Use GnuCash to send invoices. 7 **Calculators:** GnuCash can work out details such as loan amounts and even tax payments.

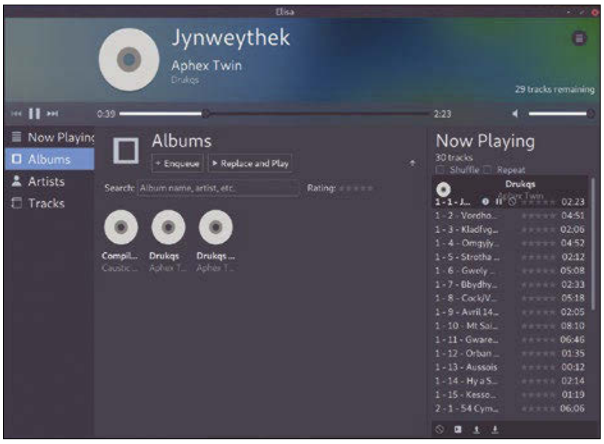
Project Website
https://github.com/Gnucash/

Music player

Elisa 0.1.0

It's perhaps a little surprising that there are still music playing applications being developed well into the 21st century. Ever since modest CPUs had the power to decode MP3 files in real time decades ago, this is a problem that should have been solved. And yet, like the infinite variation in music taste, no single application has yet to nail the perfect design for everyone or even for a significant majority. Amarok, the KDE music player, did get close for a while and was a popular alternative to proprietary alternatives, but its user interface became too bloated for some, leading to a fork back to an old version and the creation of the Clementine project; Amarok development subsequently slowed. All of which has left the KDE terrain, in particular, fertile for the cultivation of a new music player – which is what Elisa is, albeit at a very early stage of growth.

The best thing about Elisa, despite its nascent development state, is the user interface. It feels like a new-age KDE Plasma application, where the interface is well designed and doesn't get in the way of the core functionality. That functionality isn't reading lyrics, integrated research notes on artists from Wikipedia, or seeing a mood bar for the type of music – the functionality is the simple process of listening to music. The main view lists the music in your collection, using thumbnails of album covers if you have them. In fact, the only configuration option currently offered by Elisa is the location of your music, yet the rest of the player is surprisingly mature. In particular, the playlist pane features some excellent design and layout work and already allows you to rate your tracks for easy retrieval. If development continues at this quality and pace, we'll have a contender for the best Linux music player by the end of the year.



If you're looking for a KDE-friendly simple music player, the new Elisa is showing great promise.

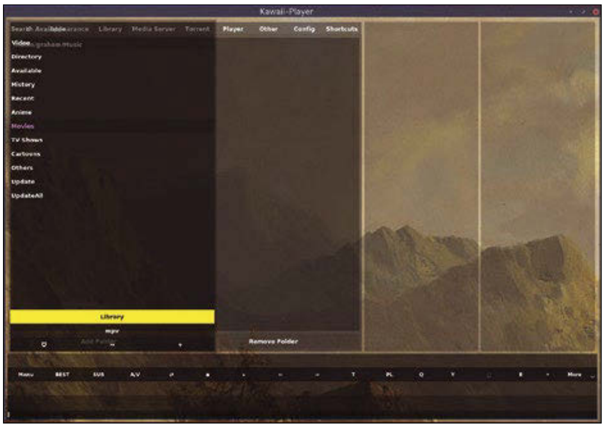
Project Website
https://community.kde.org/Elisa

Media player and server

Kawaii-Player 3.3.1

Unfortunately, this great little application has nothing to do with the synthesizer manufacturer that helped make Jan Hammer famous in the 1980s by producing possibly the best additive synth you could buy, certainly in hardware form. Instead, Kawaii-Player is a Linux video player with a built-in media server. It's built on top of the long standard and widely compatible mpv and MPlayer, which means the biggest problems are already solved – performance and media compatibility. These tools are used to handle playback while the application itself adds a huge number of additional features and playback control. The developer has stated that using these technologies to create a simple and lightweight audio and video player plus library was the primary motivation for creating Kawaii-Player, as they felt nothing fit the same remit – MPlayer is certainly complex and unwieldy if you're not used to its environment. This is all a little surprising when you first run Kawaii-Player or look at the features it supports, because it gets very close to being complex and unwieldy itself, mainly thanks to so many features.

The main user interface is a little like an old version of Kodi, albeit with a watercolor landscape for the background. You add files to your library and watch them, using the long control bar to manage playback. But behind this, there's a great deal of control. You can browse your collection, generate universal playlists, make bookmarks, download fan art, and even watch YouTube videos. You can endlessly customize the way subtitles appear or the video processing used for movie playback. Additionally, you can use BitTorrent to stream and play back content, serve files as a standard media server, use a web interface for remote control, and even run in a headless mode. If you're looking for a media player pitched somewhere between VLC and Kodi, this may be it.



There aren't many media players that have a landscape image for a background or that have so many features crammed into an austere interface.

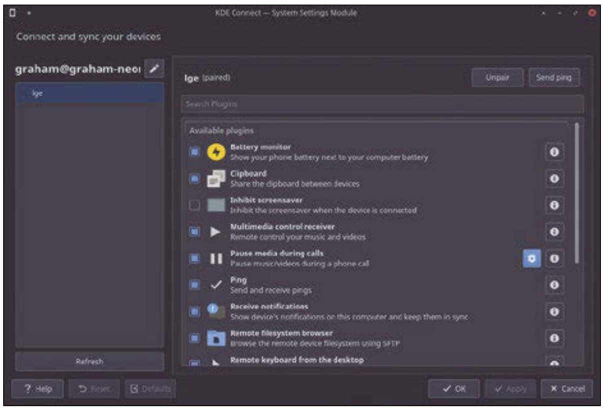
Project Website
https://github.com/kanishka-linux/kawaii-player

Phone integration

KDE Connect 1.3.0

KDE Connect is a utility I fell in love with at first sight. You install an app on your Android device and another on your Linux desktop, and your phone and computer become part of the same computing environment. Phone notifications appear on your desktop, your desktop notifications appear on your phone, your music player will stop playing if you get a call, and you can browse your phone's filesystem from the desktop file manager. There's even a command-line interface and keyboard and touchpad emulation from your phone. While these initially sound like gimmicks, they've actually saved me serious hassle when mouse batteries have failed – on more than one occasion! As most of us spend a considerable part of the day working on a computer with a phone nearby, it's amazing that something like this didn't exist sooner. It's a credit to the KDE team behind this utility for implementing it in such a way that actually works and remains useful.

When I first looked at KDE Connect in these pages, it had yet to cause a ripple. Despite already being brilliant, its requirements of a cutting-edge version of KDE and the installation of an Android app of dubious origin (for free!) possibly slowed down adoption. But over a year later, it's gone from strength to strength. There have been a constant stream of updates with new features, and there's now even a version that decouples itself from the majority of KDE Plasma bindings that make it work so well with KDE. There's even the promise of a version that uses Bluetooth rather than a common network connection for those times you need KDE Connect synergy without wanting to connect to a network. This release, version 1.3.0, is also a strong step forward, with better Nautilus support, working album art transfers, and wildcards for file transfers from the command line. If you're a Gnome user who has been sitting on the fence, now is the time to try it.



☑ Battery monitor – Show your phone battery next to your computer battery
☑ Clipboard – Share the clipboard between devices
☐ Inhibit screensaver – Inhibit the screensaver when the device is connected
☑ Multimedia control receiver – Remote control your music and videos
☑ Pause media during calls – Pause music/videos during a phone call
☑ Ping – Send and receive pings
☑ Receive notifications – Show device's notifications on this computer and keep them in sync
☑ Remote filesystem browser – Browse the remote device filesystem using SFTP
☑ Remote keyboard from the desktop

KDE Connect works well in lots of situations, but it's particularly good at remote control, such as with a Kodi-running PC.

Project Website
https://community.kde.org/KDEConnect

Selection filter

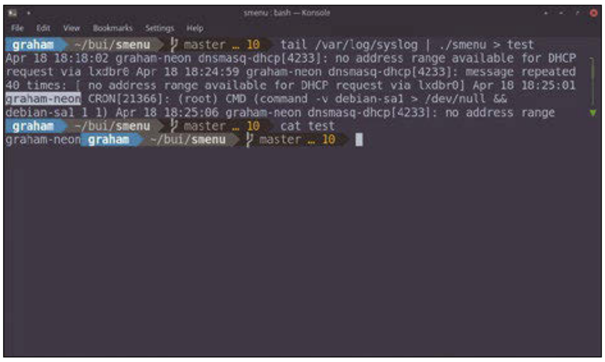
smenu

This is an excellent little utility that performs a simple job so comprehensively that you'll wonder how you ever used the command line without it. The easiest way to see what smenu does is to pipe some input into it, such as by running:

```
tail /var/log/syslog | smenu
```

You'll see exactly the same output you're used to – the last few lines of the system log – but one of those words in the output will be highlighted. Best of all, you can now use the up, down, left, and right cursor keys (or Vim navigation keys) to change which word is highlighted, just as you might in a word processor. When you press return, it looks like nothing further happens, but in the background the word you selected has been passed to the standard output. If you were to pipe the output from the command example, for instance, to a new file, that file would now contain the word you selected.

This saves you lots of messing around with a tool and is brilliant for things like variable assignment in Bash or grabbing IP addresses and other values for which it might be difficult to create a regular expression. But there are many more options than simple search, all of which can be triggered with switches or added to a configuration file. You can extract numbers from words, for example, or set a post-selection processing action. Words can be selected or not selected using regular expressions, and there are lots of controls for splitting the input into columns and making those columns fill your terminal's fill width. It's perfect when you're working with unpredictable data fields or you want to test out your own scripts without hard-coding the selection algorithms into the file.



```
graham ~/bui/smenu  master … 10  tail /var/log/syslog | ./smenu > test
Apr 18 18:18:02 graham-neon dnsmasq-dhcp[4233]: no address range available for DHCP
request via lxdbr0 Apr 18 18:24:59 graham-neon dnsmasq-dhcp[4233]: message repeated
40 times: [ no address range available for DHCP request via lxdbr0] Apr 18 18:25:01
graham-neon CRON[21366]: (root) CMD (command -v debian-sa1 > /dev/null &&
debian-sa1 1 1) Apr 18 18:25:06 graham-neon dnsmasq-dhcp[4233]: no address range
graham ~/bui/smenu  master … 10  cat test
graham-neon graham ~/bui/smenu  master … 10
```

The name of this project is a little misleading because it doesn't really implement a menu!

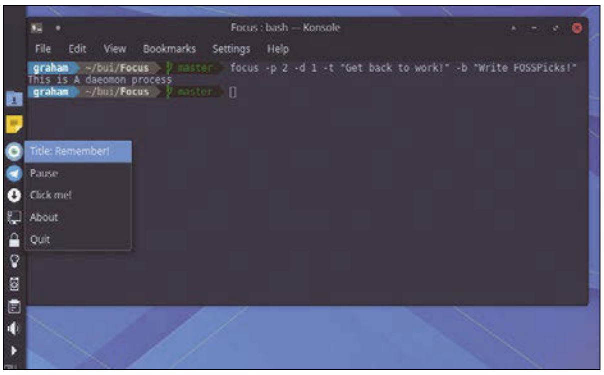
Project Website
https://github.com/p-gen/smenu

Concentration spell

Focus

This is another simple tool that's brilliantly executed and perfect for people for whom the Internet has simply become a huge distraction engine – people like me. Just writing those two sentences required a news refresh, a Bitcoin value check, a brief skim over my Reddit subscriptions, and an idle through some *Hacker News* comments (let's not mention the 20 minutes spent watching Prince on YouTube). It's almost at the point where the writing itself is the distraction while I ascend to the astral plane of infinitely networked knowledge. Of course, there are lots of solutions to this problem. Turning the Internet off isn't a bad idea, and I've previously looked at tools that will only let you access white-listed locations. But this tool, called Focus, is another option, and for me it's preferable to other more draconian measures.

Focus runs from the command line and allows you to set a reminder containing any text you choose. This reminder will then pop up after a certain period of time of your choice has passed. There are two particularly neat things about its implementation. The first is that it runs as a daemon, which means it takes very few resources and won't default to blocking you in the shell, and the second is that it uses native notifications to show you your reminder. This is particularly useful because you can use your desktop's notification system to manage these reminders more effectively, and they're more likely to be noticed while you're watching YouTube. Entering



```
graham  ~/bui/Focus  master  focus -p 2 -d 1 -t "Get back to work!" -b "Write FOSSPicks!"
This is A daeomon process
graham  ~/bui/Focus  master  []
```

Focus includes a tray applet for pausing notifications and for quitting the daemon.

```
focus -p 2 -d 1 -t "Get back to
  work!" -b "Write FOSSPicks!"
```

on the command line, for instance, will set a two-minute reminder that pops up for an hour reminding me to write some words. It could even be used to remind you to take a break. Enter `0` for the duration and Focus reminds you of your duties forever.

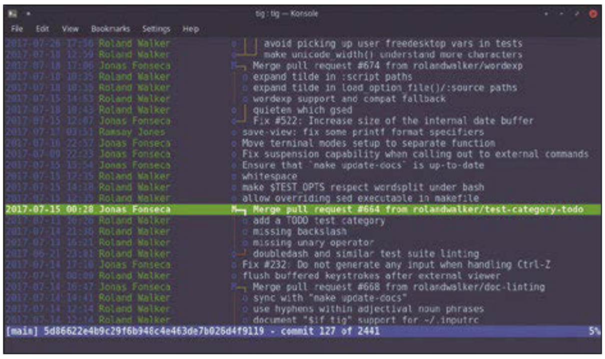
Project Website
https://github.com/nagyation/Focus

Git browser

tig

Many of us are spending more and more of our time with the `git` command. It's true that `git` is primarily useful for programmers, as it allows them to work as part of team in which everyone commits to the same codebase, where releases and bugs are tracked and coordinated. But it's also useful in lots of other cases. There are tools that use `git` to version control your `/etc/` configuration directory, for example, or serve as your own secret password repository to websites using a tool like `pass`. `git` is commonly used to build documentation, websites, and databases. If you need snapshots of something that's constantly changing and written in text, there's a good chance `git` could be a solution.

But the main problem with `git` is its complexity, and that's not just because it was built for programmers. It was initially built by Linus Torvalds, and as we know, he doesn't suffer fools gladly, nor nicefy a tool simply to appease those who don't understand its functionality. This approach often makes `git` difficult to understand, especially outside of programmer's circles and especially outside of its core feature set of pushing and pulling from a repository. Anything that can make this easier is going to be a huge help for us non-`git` masters, and that's exactly what `tig` does. It's a terminal-based UI to various different `git` views, such as the recent commits, diffs, the log, and the blame list, which is a great way to see who wrote each line of a file. It does all of this without you needing to remember any weird arguments or command structures – just press *h* to see the few keyboard shortcuts it does use and begin exploring your favorite projects and repositories.



```
2017-07-26 17:30 Roland Walker      o │ │  avoid picking up user freedesktop vars in tests
2017-07-18 12:59 Roland Walker      o─┴─┘  make unicode_width() understand more characters
2017-07-18 17:06 Jonas Fonseca      M─┐ Merge pull request #674 from rolandwalker/wordexp
2017-07-18 10:35 Roland Walker      │ o expand tilde in :script paths
2017-07-18 10:35 Roland Walker      │ o expand tilde in load_option_file()/:source paths
2017-07-15 14:53 Roland Walker      │ o wordexp support and compat fallback
2017-07-18 10:43 Roland Walker      o │ quieten which gsed
2017-07-15 12:07 Jonas Fonseca      o─┘ Fix #522: Increase size of the internal date buffer
2017-07-17 03:51 Ramsay Jones       o save-view: fix some printf format specifiers
2017-07-16 22:37 Jonas Fonseca      o Move terminal modes setup to separate function
2017-07-09 22:33 Jonas Fonseca      o Fix suspension capability when calling out to external commands
2017-07-15 15:54 Jonas Fonseca      o Ensure that `make update-docs` is up-to-date
2017-07-15 12:35 Roland Walker      o whitespace
2017-07-15 14:18 Roland Walker      o make $TEST_OPTS respect wordsplit under bash
2017-07-15 12:35 Roland Walker      o allow overriding sed executable in makefile
2017-07-15 00:28 Jonas Fonseca      M─┐ Merge pull request #664 from rolandwalker/test-category-todo
2017-07-11 20:26 Roland Walker      │ o add a TODO test category
2017-07-14 21:36 Roland Walker      │ o missing backslash
2017-07-13 16:21 Roland Walker      │ o missing unary operator
2017-06-21 23:01 Roland Walker      o─┘ doubledash and similar test suite linting
2017-07-14 17:10 Jonas Fonseca      o Fix #232: Do not generate any input when handling Ctrl-Z
2017-07-14 00:00 Roland Walker      o flush buffered keystrokes after external viewer
2017-07-14 16:47 Jonas Fonseca      M─┐ Merge pull request #668 from rolandwalker/doc-linting
2017-07-14 14:41 Roland Walker      │ o sync with "make update-docs"
2017-07-14 12:14 Roland Walker      │ o use hyphens within adjectival noun phrases
2017-07-14 12:14 Roland Walker      │ o document "$if tig" support for ~/.inputrc
[main] 5d86622e4b9c29f6b948c4e463de7b026d4f9119 - commit 127 of 2441                    5%
```

Even if you have no interest in `git`, the `tig` command opens up all kinds of details about your favorite projects – and even the Linux kernel!

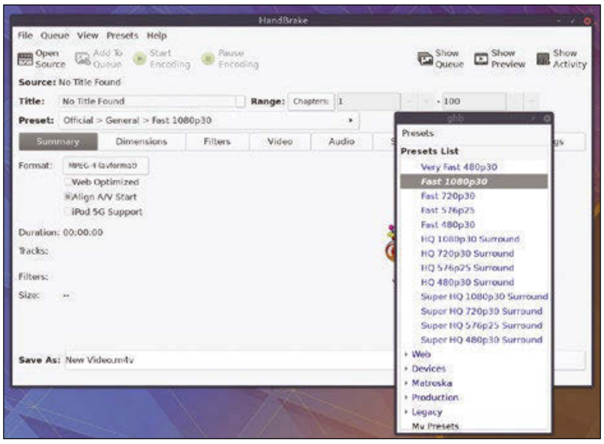
Project Website
https://jonas.github.io/tig/

Media transcoder

HandBrake 1.1.0

It took 13 years to get to the 1.0 release, so it looks like the HandBrake team are putting some serious effort into keeping up the pace of development. And while the world seems to continue spinning when the majority of its denizens care little for media transcoding, if you have any interest in running your own media center, it's still an essential skill. The performance of low-powered devices, such as a Raspberry Pi, connected to your entertainment center and screen isn't so much limited by CPU power or hardware acceleration as it is by the demands set by the media you're trying to play. Even the humblest of ARM devices can play back 4K video if the codec is finely tuned, and good compression can make the difference between your WiFi reaching capacity and still being able to host two Netflix sessions alongside your *Blade Runner* binge.

This major update adds 4K preset support for lots of devices, including 2160p/4K for Apple TV, Chromecast, Fire TV, and Roku, with an additional preset for the Matroska Multimedia Container. There's also experimental Linux support for Intel Quick Sync Video, which is Intel's encoding and decoding acceleration. The user interface has also had a much needed tweak, now with floating presets and a workflow that goes from top to bottom and left to right, rather than the random array of clicks you needed to make in previous versions. If you'd rather use the command line, that gets some updates too, with support



Source: No Title Found
Title: No Title Found Range: Chapters: 1 - 100
Preset: Official > General > Fast 1080p30
Format: MPEG-4 (avformat)
Web Optimized
Align A/V Start
iPod 5G Support
Duration: 00:00:00
Tracks:
Filters:
Size: --
Save As: New Video.m4v

HandBrake isn't necessarily as powerful as FFmpeg or MPlayer, but it doesn't need a hundred command-line options either.

for adaptive streaming and JSON formatting for output scan and status information. It's because of updates like these that HandBrake has become the de facto tool for many who need to deal with media transcoding, and this is another major release.

Project Website
https://handbrake.fr/

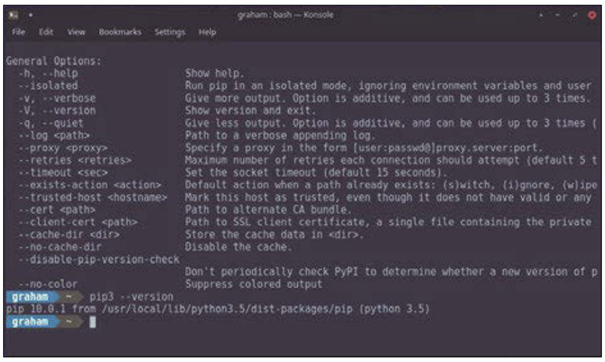
Python packager

pip 10

This is something I've not covered before in these pages, but it's something typically used every month to install a variety of dependencies that many of the things upon which we look depend. Pip is a Python package manager – or more accurately, *the* Python package manager. That a programming language like Python needs its own package manager says a lot about both Python's popularity and the way that many Python applications use modules and libraries from other packages. Pip is the way to handle all these distribution issues in a cross-platform and Python-friendly way, whether you're using Ubuntu or a Raspberry Pi. It should mean that installation is as simple as typing `pip install qutebrowser`.

Pip 10 is a big release, and it deals with one specific issue we've all known has been coming for years: Python 3 coexisting with Python 2.6. Python 3 has already been around for a long time, but because it does things so differently from the way previous versions of Python worked, it's meant that Python 2.x era tools, libraries, and applications have often remained current and even preferable. There's still Python 2.7, but deprecating 2.6 is a big step forward. This isn't really going to affect your use of Python, as you'll still be able to keep the same tools from the old era, and many installations have defaulted to calling pip either pip2 or pip3, depending on the overall dependencies, so you can still use pip2. The new version now includes a `config` command for creating, editing, adding to, and removing from a configuration, and the default upgrade strategy has become "only-if-needed," which will help people who want to stick to older versions of the software on which they rely.

> Pip 10 is a big release, and it deals with one specific issue we've all known has been coming for years: Python 3 coexisting with Python 2.6.



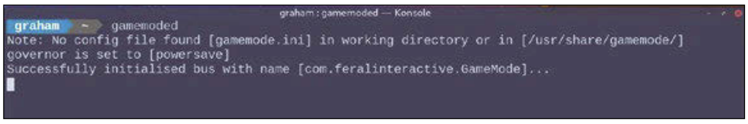
```
General Options:
  -h, --help                  Show help.
  --isolated                  Run pip in an isolated mode, ignoring environment variables and user
  -v, --verbose               Give more output. Option is additive, and can be used up to 3 times.
  -V, --version               Show version and exit.
  -q, --quiet                 Give less output. Option is additive, and can be used up to 3 times (
  --log <path>                Path to a verbose appending log.
  --proxy <proxy>             Specify a proxy in the form [user:passwd@]proxy.server:port.
  --retries <retries>         Maximum number of retries each connection should attempt (default 5 t
  --timeout <sec>             Set the socket timeout (default 15 seconds).
  --exists-action <action>    Default action when a path already exists: (s)witch, (i)gnore, (w)ipe
  --trusted-host <hostname>   Mark this host as trusted, even though it does not have valid or any
  --cert <path>               Path to alternate CA bundle.
  --client-cert <path>        Path to SSL client certificate, a single file containing the private
  --cache-dir <dir>           Store the cache data in <dir>.
  --no-cache-dir              Disable the cache.
  --disable-pip-version-check
                              Don't periodically check PyPI to determine whether a new version of p
  --no-color                  Suppress colored output
graham  ~  pip3 --version
pip 10.0.1 from /usr/local/lib/python3.5/dist-packages/pip (python 3.5)
graham  ~  
```

Pip is one of those lovely recursive acronyms that stands for pip Installs Packages.

Project Website
https://pypi.org/project/pip/

Gaming upgrade

GameMode



```
graham:gamemoded — Konsole
graham  ~  gamemoded
Note: No config file found [gamemode.ini] in working directory or in [/usr/share/gamemode/]
governor is set to [powersave]
Successfully initialised bus with name [com.feralinteractive.GameMode]...
```

There's not much to see, but GameMode can control the power characteristics of your kernel for better game performance.

Modern gaming places your system under a great deal of stress. Your CPU and GPU are usually tested to their limits, as are your system cooling and memory, as you push past normal desktop duties and into levels of clock cycle performance and data throughput typically reserved for render farms and crypto mining. One big difference between Linux and Windows is that many of us have put our machines together from our own hardware choices, and we seldom run the same combinations of distribution, graphics driver, and even kernel version. All of these elements have an effect on gaming performance. Windows users can usually rely on people with the same hardware and operating system setup sharing their configuration. But with only one third of one percent of Steam users using Linux (according to the March 2018 Steam Hardware and Software Survey), Linux users can't rely on the same thing, which means they can't copy configurations or share optimizations in the same way.

To address some of these issues, the Mac and Linux game publisher Feral Interactive (who has also just released *Rise of the Tomb Raider* on Linux) has created GameMode. GameMode is a neat idea that's a little like the game mode you get on some monitors, dropping extraneous features not required while running a game and also giving the games themselves more control over how Linux is running. It starts with the `powersave` or `ondemand` governors, but it could grow to include allowing a game to request a set of kernel optimizations and essential prelaunch optimizations, such as disabling desktop effects and OpenGL compositing. Like game mode on a monitor, the idea is to give as much power to one specific job, the game, as possible. For this project to be successful, it needs to be adopted by other game publishers, which is why Feral has made its technology open source, with the 3-clause BSD license (revised).

Project Website
https://github.com/FeralInteractive/gamemode

Old game engine

XLEngine

You may now be able to play Bethesda's classic Skyrim in virtual reality, but its ancient Daggerfall is difficult to beat on immersion.

This is a project that's definitely in an early development stage and, at the time of writing, can't quite play the games it's hoping to support. But there's lots of potential, and the small team behind it are working hard. XLEngine is a modern game engine that plays a few specific yet classic old games. In particular, it could play Daggerfall and Dark Forces, which are two ancient games that can always be replated for new fun and entertainment. The game engine has been around for a while, but the source code was closed. The code has recently been released in an unknown state of repair. While it builds and plays its two most famous games, many things remain broken. However, it is making rapid progress, and the fact that the Linux version builds and even runs when previously it didn't is a strong indication of the project's potential.

It's worth all this development because these games are such classics and won't easily run on modern hardware. DOSBox will work, but using XLEngine allows you to run an old game using modern technology such as OpenGL, which could mean bringing the game to platforms like the Raspberry Pi, a platform that is completely different from the original old x86 of those early PCs. Daggerfall is a wonderful open world RPG game that still has a lot to offer, as shown by the continued popularity of the nearly decade old Skyrim recently released for the Nintendo Switch and SteamVR virtual reality platform (sadly not for Linux, though). Daggerfall is also a free download, so there's nothing to lose by trying it if you haven't yet and tracking the development of this promising game engine.

Project Website
https://github.com/Mindwerks/XLEngine

Close Search

Even in the age of cloud computing, personal computers often hold thousands of files: text files, spreadsheets, word processing docs, configuration files, and HTML files, as well as email and other message formats. If it takes too long to find the file you need, chase it down with the Recoll local search engine.

BY MARCO FIORETTI

Recoll [1] is free software for Linux and Windows systems that adds a local search engine to your desktop or local network. And if you think that desktop search engines don't make sense in this age of cloud computing, I beg to disagree!

Look inside any school, NGO, small/medium enterprise, or individual computer used for more than a few years: Almost always, you will find big archives of mostly textual content that will never be uploaded in the cloud or otherwise exposed to an online search engine. Sometimes the reason is mere lack of time, bandwidth, or money. Sometimes it is privacy. Sometimes the reason is easier compliance with regulations like the European General Data Protection Regulation (GDPR) [2]. In all cases, deploying local search capability could make thousands and thousands of files much more useful for their owners.

Recoll is an excellent answer to the need for a local search engine. The Recoll search tool offers flexible interfaces, good documentation, and easy installation. Thanks to a relatively simple search language, Recoll can analyze and index text inside all the most common document formats, even when those documents are "hidden" inside other files (for example, an OpenDocument file zipped and attached to an email message). In most cases, you can preview or open the files found with your search by just clicking on them inside the Recoll window.

The first part of this tutorial explains how Recoll works and how to install it and configure its most critical functions. The second part describes the Recoll search syntax and offers a few tips to help you make the best use of Recoll.

Architecture and User Interfaces

Strictly speaking, Recoll is just a wrapper, albeit a great one, for the open source information retrieval library called Xapian [3]. It is Xapian, not Recoll, that performs all the high-level indexing and classification of your documents. Xapian is also directly usable via scripts in Perl, Java, and other languages. But it is Recoll that makes the desktop search really usable, by doing all the rest of the work, from overall configuration to obscure, low-level tasks like *stemming*. Stemming is the process of reducing similar words to their common root. It is thanks to stemming that you can search for a word like "hacker" and receive results for "hackerS" or "hacking" in addition to the original search term.

The other tasks that Recoll handles directly are extracting text from your files, decoding your queries and, of course, presenting their results in a format that makes it easy to browse and open them from a graphical interface.

With the right libraries and plugins, you can perform Recoll searches directly from Python and other languages, or from desktop environments like Unity or KDE. This article will focus on the native Recoll GUI, its web-based equivalent, and, of course, the evergreen command-line option.

Installation

Recent Recoll versions are available as binary packages for Windows and the most popular Linux distributions. On Ubuntu, for example, type the following commands at the prompt to add a Personal Package Archive (PPA) repository for `recoll` and install both the graphical and command-line interfaces:

```
#> sudo add-apt-repository ⮐
  ppa:recoll-backports/recoll-1.15-on
#> sudo apt-get update
#> sudo apt-get install recoll -y
```

(Don't be fooled by the `1.15` in the repository name: The command will install the current version of Recoll, whatever it is). After those commands, typing `recoll` in the desktop search bar will show you the icon that opens the Recoll native GUI. To search at the command prompt or in a shell script, use the command `recollq`. Use the `recollindex` command to generate an index.

You must install the Recoll web interface separately. Go to the Github page for the web interface [4], download the `master.zip` archive for your version of Recoll, and unzip it to expand a folder called `recoll-webui-master`. The file inside the folder called `webui-standalone.py` is a mini web server, which you can reach with your browser at the address *http://localhost:8080*. The mini web server is a bit slow, but it works right away for all the users of the local network, with one (well documented) caveat: You cannot directly open local files from the links in its listings unless you explicitly authorize Firefox to do so (see the box entitled "Authorizing Firefox").

If you plan to use Recoll on a regular basis, you might wish to configure your Linux system to start it automatically when the system boots. See your Linux distro's documentation for more on configuring an application to launch at system startup.

Indexing Configuration

No search engine is better than its index. Telling Recoll how to create and maintain the index is the most critical part of the configuration (Figure 1).

Recoll has a system-wide configuration file (`/usr/share/recoll/examples/recoll.conf` on Ubuntu), but each user also gets a personal configuration – with a higher priority. The personal configuration file is stored in `$HOME/.recoll/recoll.conf`. The first time you start it, the Recoll GUI will ask you how to configure the index and will save your choices in your personal file. Among other things, you may define which files types should be indexed and the default language.

By default, Recoll will only have one index for your whole home directory, but it may handle many, totally independent indexes. The only requirement is that each index has a dedicated configuration directory. The simplest way to make Recoll create a separate configuration and index seems to be to create an empty directory and then start the software from the command line with the `-c` option pointing to it:

```
#>mkdir $HOME/.recoll-customconfig
#>recoll -c $HOME/.recoll-customconfig
```

You can search in more indexes simultaneously by adding them in the *Preferences | External Index Dialog* of the GUI. Don't forget that, when you search on multiple indexes, Recoll will use all their data, but it will only use the configuration of one index: the default index, or the index explicitly set with the `RECOLL_CONFDIR` environment variable or the `-c` option.

Indexing Web Pages?

It is possible, and in fact quite easy, to use Recoll to index and search the web pages you visit. First, download and install the Firefox add-on called Recoll WE [5]. Second, enable web indexing from the *GUI Index* configuration panel, or by setting the `processwebqueue` variable to `1` in the configuration file. After that, one click on the *Recoll WE* button in Firefox will save a complete copy of the current page in a folder, where Recoll will later find, copy, and index it. The only limitation of this method is that Recoll stores all the copies in a cache of fixed size. No matter how big you make it, if that cache fills up, Recoll will automatically remove the older files it contains. You must explicitly archive any web page you want to keep indefinitely.

Authorizing Firefox

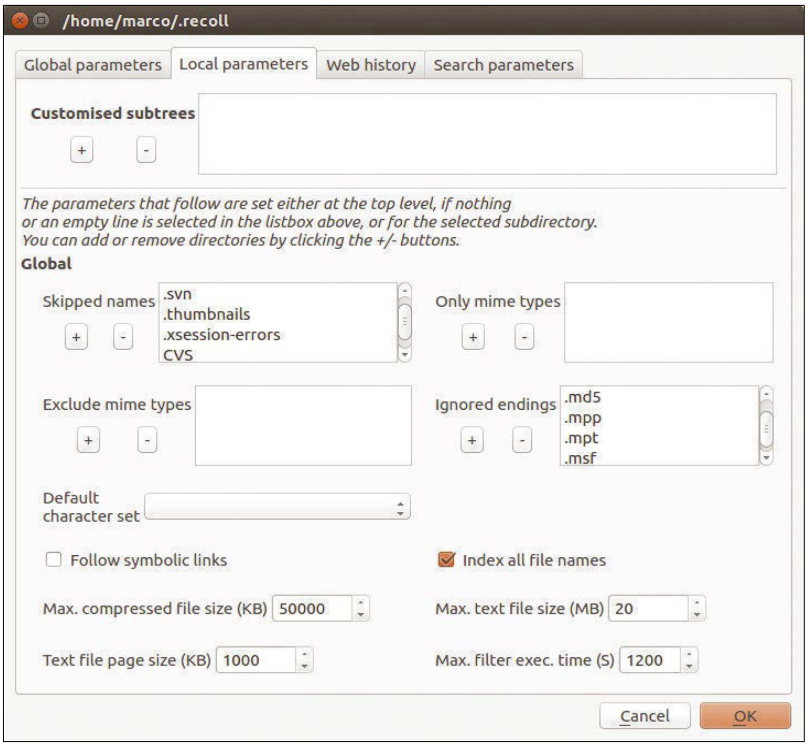
To authorize Firefox to let you open local files, add the contents of the file `examples/firefox-user.js` into `~/.mozilla/firefox/<profile>/user.js` and restart Firefox.

Index Maintenance

The command-line program that manages the actual indexing is called `recollindex`. You can launch `recollindex` from the File menu of the GUI, from the command line, or from a cron job. There are two ways, accessible from the *Preferences | Indexing schedule* panel, to keep an index up to date. One way is to make `recollindex` run all the time in the background, indexing and re-indexing files as soon as they are created or modified. This approach may slow down your computer, so avoid it unless you really, really need an index that is as up to date as possible.

In most cases, it is much better to only run `recollindex` at regular intervals, possibly every few

Figure 1: A detail of the most crucial, but not difficult, Recoll configuration phase: The detailed definition of which files should be indexed, and how.



Customised subtrees

The parameters that follow are set either at the top level, if nothing or an empty line is selected in the listbox above, or for the selected subdirectory. You can add or remove directories by clicking the +/- buttons.

Global

Skipped names: .svn, .thumbnails, .xsession-errors, CVS

Only mime types

Exclude mime types

Ignored endings: .md5, .mpp, .mpt, .msf

Default character set

☐ Follow symbolic links

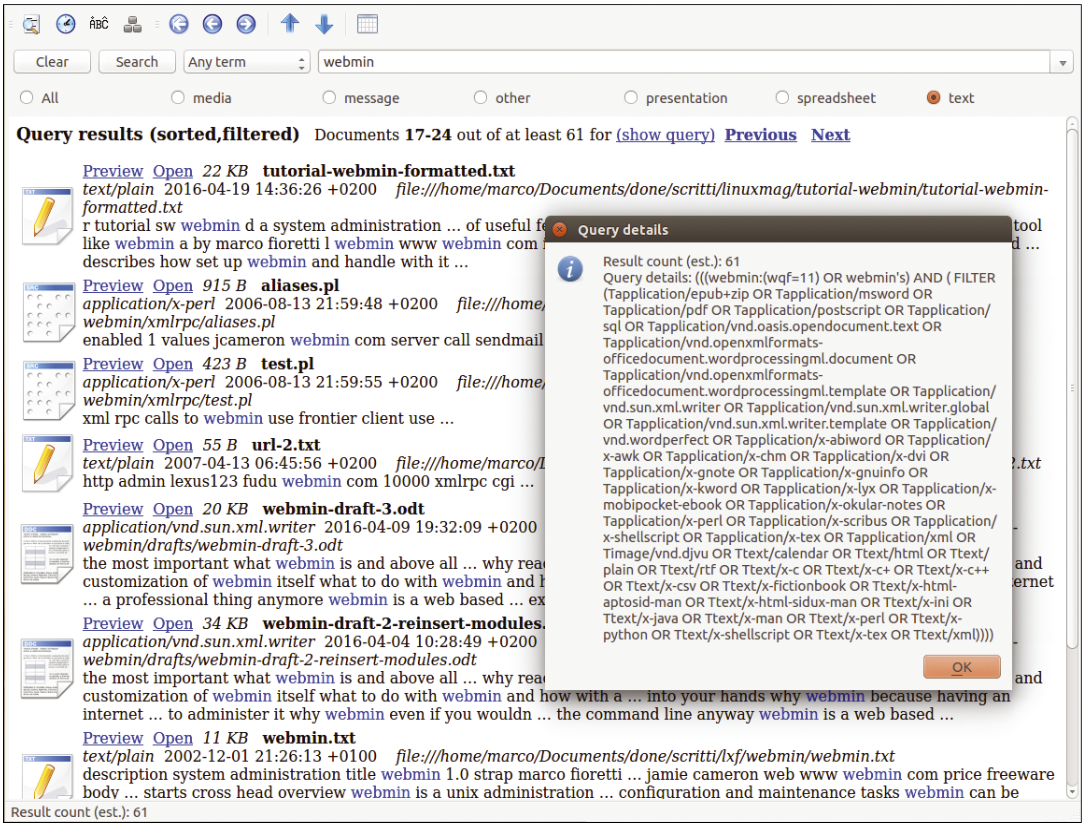
☑ Index all file names

Max. compressed file size (KB) 50000

Max. text file size (MB) 20

Text file page size (KB) 1000

Max. filter exec. time (S) 1200



Query results (sorted,filtered) Documents **17-24** out of at least 61 for (show query) **Previous** **Next**

Preview Open *22 KB* **tutorial-webmin-formatted.txt**
text/plain 2016-04-19 14:36:26 +0200 *file:///home/marco/Documents/done/scritti/linuxmag/tutorial-webmin/tutorial-webmin-formatted.txt*
r tutorial sw webmin d a system administration ... of useful fe … tool
like webmin a by marco fioretti l webmin www webmin com … d ...
describes how set up webmin and handle with it ...

Preview Open *915 B* **aliases.pl**
application/x-perl 2006-08-13 21:59:48 +0200 *file:///home/… webmin/xmlrpc/aliases.pl*
enabled 1 values jcameron webmin com server call sendmail

Preview Open *423 B* **test.pl**
application/x-perl 2006-08-13 21:59:55 +0200 *file:///home/… webmin/xmlrpc/test.pl*
xml rpc calls to webmin use frontier client use ...

Preview Open *55 B* **url-2.txt**
text/plain 2007-04-13 06:45:56 +0200 *file:///home/marco/l… 2.txt*
http admin lexus123 fudu webmin com 10000 xmlrpc cgi ...

Preview Open *20 KB* **webmin-draft-3.odt**
application/vnd.sun.xml.writer 2016-04-09 19:32:09 +0200 … *webmin/drafts/webmin-draft-3.odt*
the most important what webmin is and above all ... why rea… and
customization of webmin itself what to do with webmin and h… ernet
... a professional thing anymore webmin is a web based ... ex…

Preview Open *34 KB* **webmin-draft-2-reinsert-modules.**…
application/vnd.sun.xml.writer 2016-04-04 10:28:49 +0200 … *webmin/drafts/webmin-draft-2-reinsert-modules.odt*
the most important what webmin is and above all ... why rea… and
customization of webmin itself what to do with webmin and how with a ... into your hands why webmin because having an
internet ... to administer it why webmin even if you wouldn ... the command line anyway webmin is a web based ...

Preview Open *11 KB* **webmin.txt**
text/plain 2002-12-01 21:26:13 +0100 *file:///home/marco/Documents/done/scritti/lxf/webmin/webmin.txt*
description system administration title webmin 1.0 strap marco fioretti ... jamie cameron web www webmin com price freeware
body ... starts cross head overview webmin is a unix administration ... configuration and maintenance tasks webmin can be

Result count (est.): 61

Query details

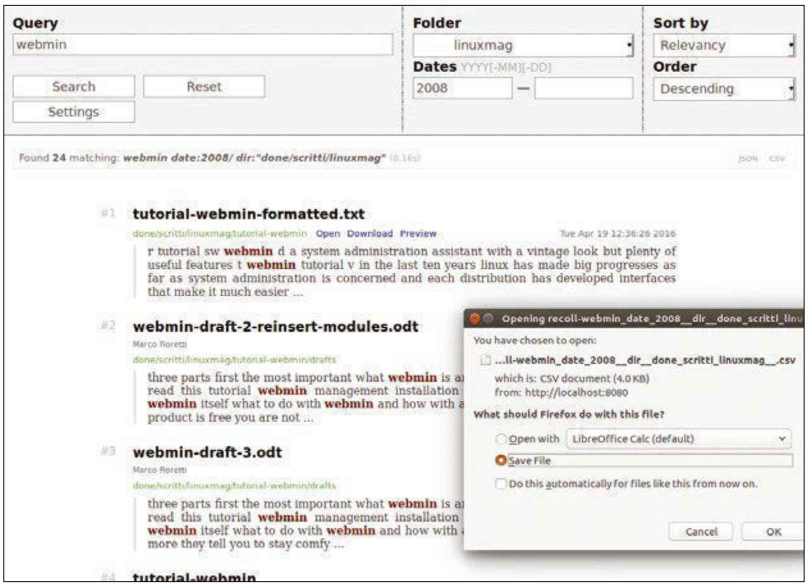
Result count (est.): 61
Query details: (((webmin:(wqf=11) OR webmin's) AND (FILTER (Tapplication/epub+zip OR Tapplication/msword OR Tapplication/pdf OR Tapplication/postscript OR Tapplication/sql OR Tapplication/vnd.oasis.opendocument.text OR Tapplication/vnd.openxmlformats-officedocument.wordprocessingml.document OR Tapplication/vnd.openxmlformats-officedocument.wordprocessingml.template OR Tapplication/vnd.sun.xml.writer OR Tapplication/vnd.sun.xml.writer.global OR Tapplication/vnd.sun.xml.writer.template OR Tapplication/vnd.wordperfect OR Tapplication/x-abiword OR Tapplication/x-awk OR Tapplication/x-chm OR Tapplication/x-dvi OR Tapplication/x-gnote OR Tapplication/x-gnuinfo OR Tapplication/x-kword OR Tapplication/x-lyx OR Tapplication/x-mobipocket-ebook OR Tapplication/x-okular-notes OR Tapplication/x-perl OR Tapplication/x-scribus OR Tapplication/x-shellscript OR Tapplication/x-tex OR Tapplication/xml OR Timage/vnd.djvu OR Ttext/calendar OR Ttext/html OR Ttext/plain OR Ttext/rtf OR Ttext/x-c OR Ttext/x-c+ OR Ttext/x-c++ OR Ttext/x-csv OR Ttext/x-fictionbook OR Ttext/x-html-aptosid-man OR Ttext/x-html-sidux-man OR Ttext/x-ini OR Ttext/x-java OR Ttext/x-man OR Ttext/x-perl OR Ttext/x-python OR Ttext/x-shellscript OR Ttext/x-tex OR Ttext/xml))))

Figure 2: The Recoll search result window: a clean listing, with buttons to preview or open every file discovered, as well as a *show query* link to check what Recoll did.

days. By default, this program will analyze and index only the files that change after its previous run. You can, however, rebuild the whole index in any moment – from the GUI or by passing the `-z` option to `recollindex`. To update a different index, use the `-c` option to show `recollindex` the path to the corresponding configuration directory.

Basic and Advanced Search

Running basic searches in Recoll could not be any simpler: Enter the search term in the top text box, click the *Search* button, and browse the results (Figure 2). The *Advanced Search* panel (reachable from the Tools menu) and the web interface (Figure 3) can filter results in many ways, based on keywords or the type, age, and size of the file. All options are quite easy to understand by trial and error, so I will focus on an even more powerful feature: the language by which Recoll understands and can perform very complex queries. Once you know how to write a query, you can pass it to the `recollq` command line utility or enter it in the text box of the desktop interface. You'll need to set the search mode to `Query language`.



Found **24** matching: ***webmin date:2008/ dir:"done/scritti/linuxmag"*** (0.16s)

#1 **tutorial-webmin-formatted.txt**
done/scritti/linuxmag/tutorial-webmin Open Download Preview Tue Apr 19 12:36:26 2016
r tutorial sw **webmin** d a system administration assistant with a vintage look but plenty of useful features t **webmin** tutorial v in the last ten years linux has made big progresses as far as system administration is concerned and each distribution has developed interfaces that make it much easier ...

#2 **webmin-draft-2-reinsert-modules.odt**
Marco Fioretti
done/scritti/linuxmag/tutorial-webmin/drafts
three parts first the most important what **webmin** is a…
read this tutorial **webmin** management installation…
webmin itself what to do with **webmin** and how with a…
product is free you are not ...

#3 **webmin-draft-3.odt**
Marco Fioretti
done/scritti/linuxmag/tutorial-webmin/drafts
three parts first the most important what **webmin** is a…
read this tutorial **webmin** management installation…
webmin itself what to do with **webmin** and how with…
more they tell you to stay comfy ...

#4 **tutorial-webmin**

Opening recoll-webmin_date_2008__dir__done_scritti_linu…
You have chosen to open:
…ll-webmin_date_2008__dir__done_scritti_linuxmag__.csv
which is: CSV document (4.0 KB)
from: http://localhost:8080
What should Firefox do with this file?
○ Open with LibreOffice Calc (default)
● Save File
□ Do this automatically for files like this from now on.

Figure 3: The Recoll web interface is almost as easy to use as the desktop one, and it provides the same option to save the results of any search as a spreadsheet.

Query Structure

Each Recoll query can contain several elements (Figure 4). Each element is composed of a value (e.g., "linux") and an optional field that describes where Recoll should look for that value. Field name and value are separated by a colon.

If a query contains several elements, Recoll will only return files that match all of them, unless you tell it otherwise using parentheses and logic operators. Consider the following examples:

```
Linux Stallman Torvalds
Linux AND Stallman OR Torvalds
(Linux AND Stallman) OR Torvalds
Linux Stallman -Torvalds
"Lord of the Rings"
```

The first query returns all documents that have a name or contents containing all the three terms listed. The second query only returns documents that contain `Linux` and either `Stallman` or `Torvalds`. The third query returns documents that contain the word `Torvalds` or both the words `Linux` and `Stallman`: This happens because the `OR` operator has higher priority than the `AND`, but you can set priorities as you want with parentheses. The fourth query, thanks to the negation operator (-) finds the documents that contain both the words `Linux` and `Stallman` but not the word `Torvalds`. The double quotes in the last query tell Recoll to search for the complete phrase `Lord of the Rings`.

The next things to know are that (even if they may seriously slow down Recoll) you can use wild cards (as in "*ter" to find computer, commuter, etc.), and stemming is used by default, except for all-uppercase words. For example, searching for `linux` will return documents containing words like `linuxian` or `linuxer`. To avoid that and only get the exact word you want, enclose it in double quotes.

Recoll queries become more powerful when you pair double quotes with modifiers to allow *proximity searches*. To understand the concept of a proximity search, compare the following queries:

```
"Linux rules"
"Linux rules"p
"Linux rules"po10
```

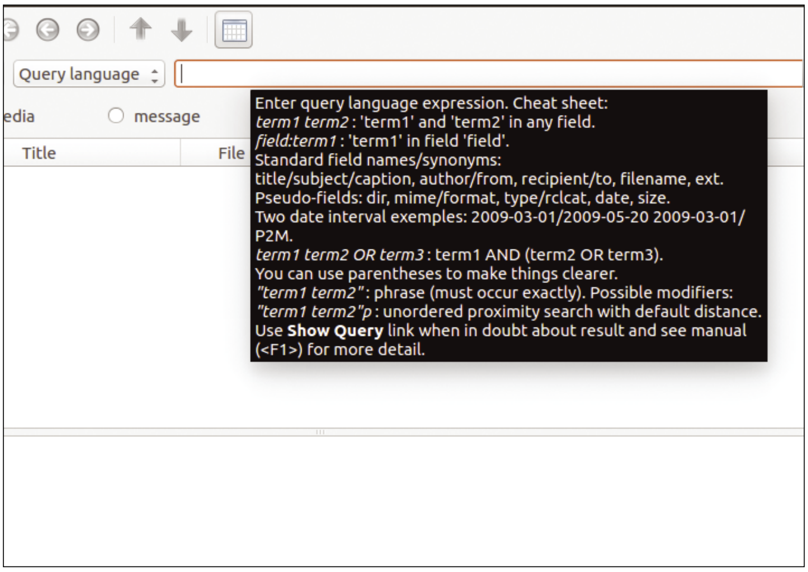
The first query, thanks to the quotes, finds only the documents containing the exact sentence `Linux rules`. The `p` modifier attached (without spaces!) to the second statement makes Recoll find any document that contains those two words next to each other, but in any order. The `o` modifier in the last query asks Recoll for all the documents in which those words appear in any order, but with up to 10 other words between them, as in `Linux is an operating system that rules` or `Linux really, really rules`!

Basic Query Filters

In addition to describing which terms you want and in which combinations, you can specify where, in a document, Recoll should look for the search string. For instance, you can look in `title`, `author`, `recipient`, `keyword`, `ext` (that is, file extension), `filename`, and `dirname`. To find all documents with the sentence `Linux rules` in the title, or whose author is `Marco`, just enter:

```
title:"Linux rules" OR ↵
  author:Marco
```

As this example shows, Recoll filters are easy to use – as long as you are aware of some basic properties of file formats and differences among them. Keywords, for example, are supported only in a few formats, like OpenDocument or markdown with front matter. The `author` field in Recoll is equivalent to the `From` address in email files and the `Author` in OpenDocument texts. In the same spirit, both the `Subject` of an email and the contents of the HTML `<title>` tag are the `title`, as far as Recoll is concerned.



Enter query language expression. Cheat sheet:
term1 term2 : 'term1' and 'term2' in any field.
field:term1 : 'term1' in field 'field'.
Standard field names/synonyms:
title/subject/caption, author/from, recipient/to, filename, ext.
Pseudo-fields: dir, mime/format, type/rclcat, date, size.
Two date interval exemples: 2009-03-01/2009-05-20 2009-03-01/P2M.
term1 term2 OR term3 : term1 AND (term2 OR term3).
You can use parentheses to make things clearer.
"term1 term2" : phrase (must occur exactly). Possible modifiers:
"term1 term2"p : unordered proximity search with default distance.
Use **Show Query** link when in doubt about result and see manual (<F1>) for more detail.

Figure 4: Another built-in way to learn how Recoll works: A handy cheat sheet appears showing information for the *Query language* search mode.

The `dir` filter restricts searches to specific directories or parts of the filesystem:

```
dir:/reports/drafts
```

makes Recoll look only in folders like `Documents/reports/drafts`, `Documents/archive/reports/drafts`, and so on.

Special Filters

Recoll has four more filters that all work in the same way, regardless of how you mix them with other filters by means of parentheses or boolean operators. These filters are `rclcat`, `mime`, `size`, and `date`. `rclcat` defines what file categories you are interested in. The available categories are those in the main GUI window: `text`, `spreadsheet`, `presentation`, `message`, `media`, and `other`. The `mime` filter indicates specific types of files:

```
"-mime:text/plain"
```

means find all files except plain text files.

Finding only files with sizes in a certain range is as easy as writing, for example:

```
size>100 size=<1000
```

to find only files of size between 101 and 1000 bytes, inclusive.

The Recoll query language also understands dates and periods – in other words, intervals of time.

Dates have the format `YYYY-MM-DD`, but only the year is mandatory. Periods are strings beginning with a capital `P` (as in "period") followed by any combination of number of years (`Y`), months (`M`), and days (`D`):

```
P2Y10D = 2 years and ten days
P2M5D = 2 months and 5 days
```

Combining dates and periods with a slash tells Recoll to find only documents created or modified within a specified date range:

```
date:2018-04-01/2018-04-30
date:2018-04-01/P30D
date:P3M/
date:/2017-12-25
```

The first filter indicates the whole month of April 2018. The second filter says exactly the same thing, but with a syntax much easier to implement in a shell script. The third filter means "anything that is no more than three months older than the current date" (assuming you have no files with dates in the future). The last filter means "anything dated before Christmas of 2017."

Recoll Tips

If you have read this far, you already know enough Recoll to use it productively, but why stop there? Following are some tips to help you enter "Recoll brainstorming mode" and go even further.

First, keep your files in good order. The most efficient search engine is one that can work in a well ordered directory tree, on files with good metadata and names that are meaningful, but also portable and future-proof: As the Recoll manual says, "non-ASCII UNIX file paths are an unending source of trouble and are best avoided." Basically, if Recoll works well, it means that your documents are also easy to back up or process with any other tool. In fact, Recoll can help you put your files in order: You may, for example, ask Recoll to list all the files about backpacking that are not in your `backpacking` folder, just so you can move them there.

Second, restrict the indexed area as much as you can, while taking advantage of multiple indexes. Ideally, documents that you need to keep for whatever reason, but that you will never need to search, should go in a separate folder, excluded from any Recoll indexing: Should you need to search these files some day, you can always generate a new index on the spot. If all the files you care about indexing are in your `$HOME/Documents/work` folder, set that folder as the top directory, instead of the default `$HOME`, in the *Preferences | Indexing configuration* panel. If you need to search your work files every day, but your personal files maybe once a month, create separate indexes for `$HOME/Documents/work` and `$HOME/Documents/personal`, with different update schedules for each index, and then use the `work` index as your default index, using or adding the other only when necessary.

Third, spend some time playing with search queries in the desktop GUI, both to save the best ones to reuse (via the File menu) and to learn how to write good ones for the command-line interface. Alas, the GUI saves searches (in the configuration folder's `savedqueries` subfolder) in an XML format not usable on the command line by `recollq`. However, you can learn how the queries are expanded and passed to Xapian by clicking on *Show Query* at the top of the results list. Another great way to master the Recoll query language is comparing the results obtained in the GUI with those returned by `recollq`.

Fourth, remember that you can use Recoll also as a semi-automatic, general-purpose analyzer of all of your documents, in different ways, and for many different purposes. As just one example, running `recollq` with the `-b` switch outputs only the names of the files matching the current query. Therefore, you can write a shell script that runs that command and, if it returns at least one line, notifies you by email that somebody added a document with that string inside. Other `recollq` options print out only the abstracts (`-A`) or metadata (`-m`) of the files they find. Also available are excellent options to import all the documents you want inside any database, as well as text analysis software. Even with the desktop GUI, you can click the button in the top bar that displays the search results as a table; the Results menu will let you save the same results as a spreadsheet in CSV format. ■■■

The Author

Marco Fioretti is a freelance author, trainer, and researcher based in Rome, Italy, who has been working with Free and Open Source software since 1995 and on open digital standards since 2005. Marco also is a board member of the Free Knowledge Institute (*http://freeknowledge.eu*).

Info

[1] Recoll: *www.lesbonscomptes.com/recoll*

[2] EU GDPR: *https://www.eugdpr.org/*

[3] Xapian: *https://xapian.org*

[4] Recoll web interface: *https://github.com/koniu/recoll-webui*

[5] Recoll WE: *https://addons.mozilla.org/en-US/firefox/addon/recoll-we/*

Building your own nodes and mods in Minetest

Minetest is much more than a clone of a certain popular proprietary game. It offers infinite customization that allows you to create blocks, objects, fun educational exercises, and even games within the game, dishing up features well beyond those of any other closed source alternative.

BY PAUL BROWN

Although I have written about Minetest [1] before, a four page article hardly does the game justice. In fact, the tag "game" hardly does Minetest justice. The November 2017 issue of *Linux Magazine* [2] we looked at gameplay, but for the regular readers, actually playing the game should be the least of their concerns.

Minetest allows you to create much more beyond what is served up in the standard package. In a way, Minetest is a blank canvas (Figure 1), and, in the grand tradition of free software, you are not restricted to the default palette, and in fact, you are given the tools to create your own paints and brushes.

All Mods

Minetest is made up of two parts: the main engine and the mods. It is the mods that make the game playable. If you have installed Minetest from your distro's repositories, look into your `/usr/share/minetest/games/minetest/` directory. You'll see that the brunt of bringing the game to life is born by the files in the `mods/` directory. A "world" in Minetest is, to all practical effects, its mods.

The default world that comes with Minetest is basic. You have a nice variety of materials and things you can craft, but, for example, there are no animals and no monsters. This is by design: The creators of Minetest assume you are going to want to tailor your experience, so they give you the bare minimum, and it is up to you to bring in (or create, as you shall see later) your own modifications.

However, if you want to get into some serious gameplay before tailoring, head over to the Minetest website [3] to see what's available.

Let's grab some animals, shall we? Notice that, apart from the `minetest/` directory in `/usr/share`, you also have a hidden `.minetest` directory in your own home directory. Change into the `mods/` subdirectory:

```
cd ~/.minetest/mods
```

And download the general mobs and farm animals mods made by Minetest Forums member TenPlus1:

```
git clone ↵
  https://github.com/tenplus1/mobs_redo.git
git clone ↵
  https://github.com/tenplus1/mobs_animal.git
```

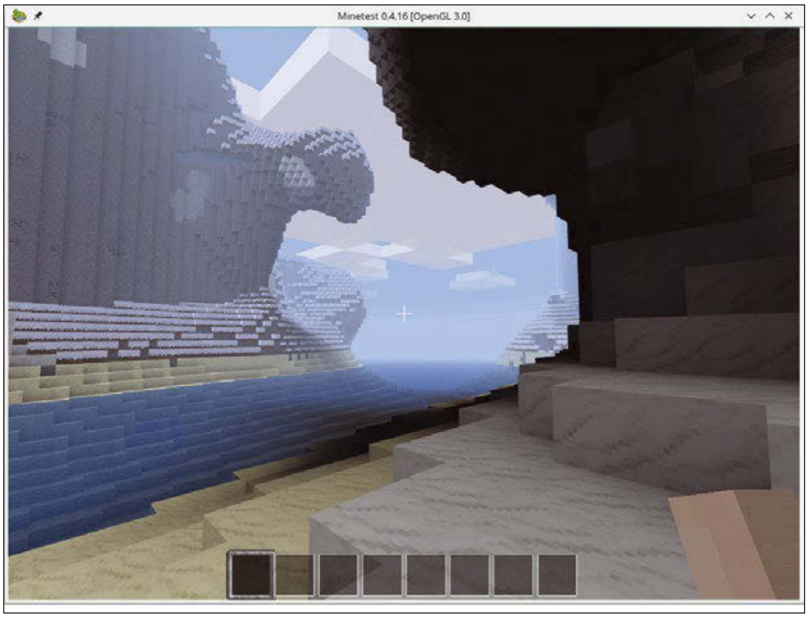
Rename the `mobs_redo/` folder to `mobs/`:

```
mv mobs_redo mobs
```

And run Minetest.

Create a new world or pick an old one, and click the *Configure* button. In the panel on the right, you

Figure 1: You can model the worlds of Minetest in more ways than one.



will see a list of available mods. Pick *mobs* and tick the *enabled* checkbox at the top of the window. The mod will turn green as shown in Figure 2. Do the same with the *mobs_animal* mod.

Now, when you visit your world, there will be animals roaming around. You can hunt them, domesticate them, or just hang out with them – if they are friendly.

Different types of animals hang around different environments. I met warthogs on a savanna-like expanse, but none turned up on a green prairie where cows were roaming (Figure 3).

Many animals, like cows, pigs, warthogs and sheep, give meat when they are killed, or other products when they are right-clicked. For example, right-click a cow with an empty bucket, and you'll get milk. Right-click a sheep with a dye, and you will change its color. Then, right-click it with shears, and you'll get colored wool. In the `mobs_animal` directory, you'll find a README document describing different ways you can interact with your four-legged friends.

If you want the whole survival experience, get yourself some monsters from GitHub [4] and the Minetest Wiki [5] and battle it out against zombies, ghosts, and all sorts of creepy-crawlies.

Making Your Own

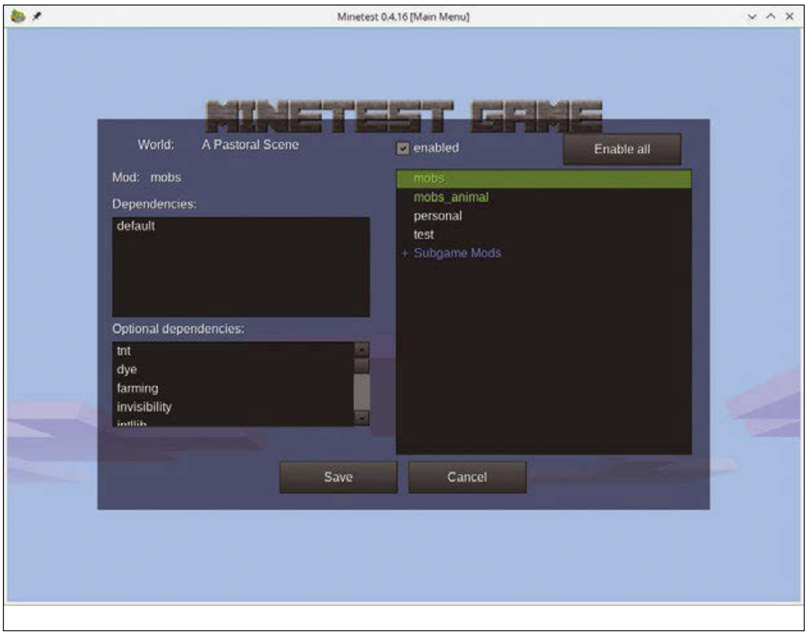
But, again, you are just playing someone else's game. Although Minetest's gameplay is great, it is not, I think, the most interesting feature. For us tinkerers, that feature would be the mod system, and making your own mods is relatively easy. Minetest has made the API for creating mods part of its framework from the beginning. To program mods, you use Lua [6], a high-level language vaguely reminiscent of Python.

Although we won't go into Lua in much depth here (there are plenty of good guides and tutorials online), it is worth mentioning that in Lua it is all about tables. Lua tables combine the features from arrays, associative arrays, lists, dictionaries, and so on from other languages into one powerful data structure. Even Lua's pseudo object-oriented programming paradigm is table-based. Unsurprisingly, the Minetest modding API relies heavily on tables.

Speaking of which, the basic building block, never better said, of Minetest modding is the *node*, a glorified table. All physical things in a Minetest world, including invisible ones, are nodes, and every different kind of node is defined by a mod. Let's see how that works by building a basic node.

Listing 1: init.lua – Reinforced Steel Node

```
minetest.register_node("personal:reinforced_
steel", {
})
```



World: A Pastoral Scene

Mod: mobs

Dependencies:
default

Optional dependencies:
tnt
dye
farming
invisibility
intllib

☑ enabled

mobs
mobs_animal
personal
test
+ Subgame Mods

Figure 2: Activate mods in the Configure screen.

Open your text editor, and copy what you see in Listing 1.

In your `mods/` directory create a subdirectory and call it `personal`. Save your file there as `init.lua`.

Congratulations: You just made your first mod.

The `register_node()` function is a Minetest module [7] that inserts a new kind of node into the Minetest world. It takes two parameters: the name and location of the material (`"personal:reinforced_steel"`), and a table that contains all the material's characteristics. The table is enclosed in curly brackets (`{}`) and, in this case, is empty, so you will see a cube with placeholder values when you use the node.

Although you could load your new node the same way you loaded the pre-built mods you saw above, there is not much to see (Figure 4). Be-

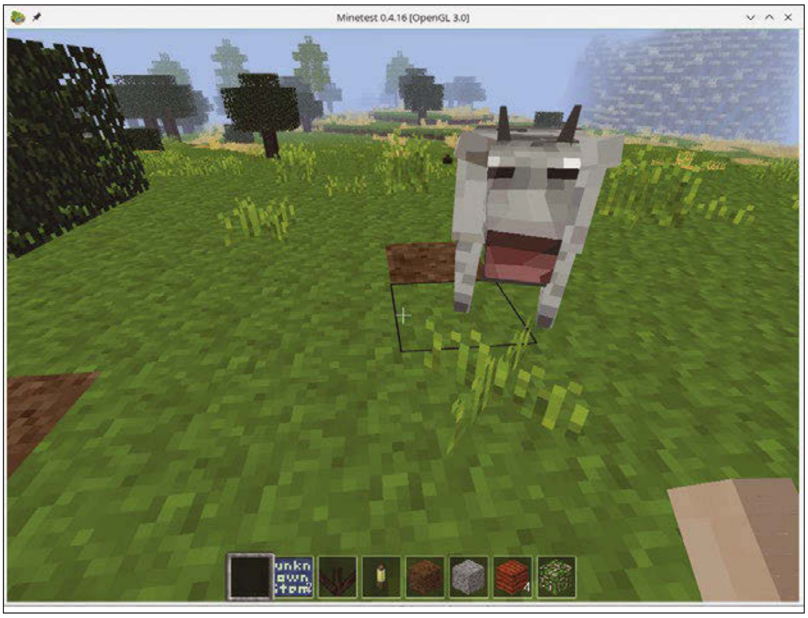


Figure 3: It is exciting to meet your first cow.

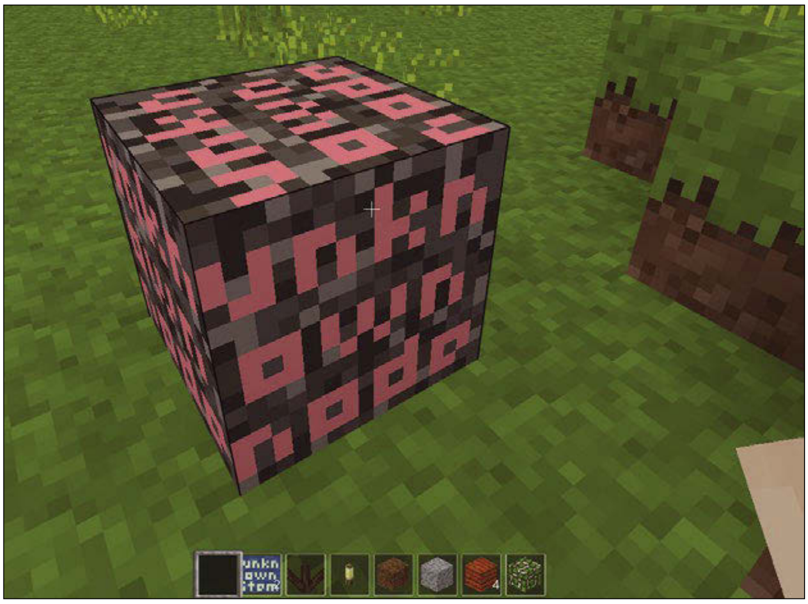


Figure 4: Your first node.

Listing 2: init.lua – Reinforced Steel Node

```
minetest.register_node("personal:reinforced_steel", {
  tiles = {"reinforced.png"},
})
```

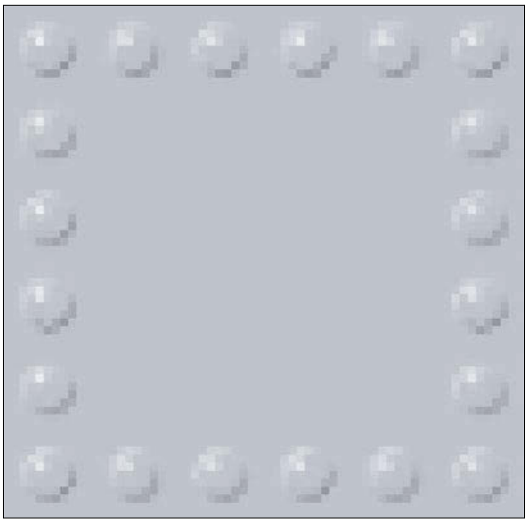


Figure 5: A reinforced steel texture created with Inkscape.

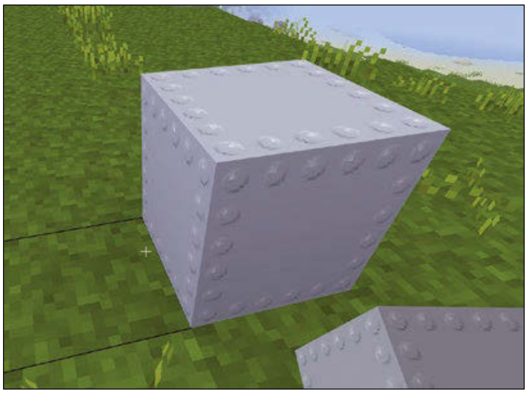


Figure 6: A texture node is a good-looking node.

sides, you will not find your node "in nature," as it were; you will have to give it to yourself.

To do this, open Minetest's terminal (inside the game, press F10) and type `/giveme personal:reinforced_steel 50` (if the engine complains, give yourself `give` privileges with `/grantme give`). This will add 50 blocks of your new material to your inventory.

To lay down a block, right-click anywhere on the floor, and you should see something like what you see in Figure 4. It doesn't look very good, does it? That is because it is using a default placeholder texture, with the words "unknown node" scrawled all over it.

Let's prettify it.

Inside your `personal/` folder, create new subfolder called `textures/`. This is where the Minetest engine will go looking for pictures to plaster over your node.

Create a square image using your preferred image editor. The creators of Minetest recommend using an editor that will allow you to edit individual pixels, so as to keep in line with the legacy look of the game. That is good advice, but I have found that using Inkscape, exporting to a PNG, then downsizing the image to 80x80 pixels using a Cubic interpolation in Gimp, and then blowing it up with no interpolation to 320x320 works fine. If you choose your colors carefully, it gives your designs a nice 16-bit vibe like what you see in Figure 5.

Save your graphic in your `textures/` folder as `reinforced.png`, and modify your mod to look like what is shown in Listing 2.

You don't have to indicate the path to the image, because Minetest looks into all of the mods' `textures/` directories. The downside of this is that you have to make the name of your material unique, or you risk getting the wrong material plastered over your node.

In Listing 2, you can see that we have added the first element of the table that gets passed on to `register_node()`. It shows how this Lua table is basically the same as an associative array: `tiles` is the key, and the value is another table containing the designs for the nodes' faces.

If you now visit your world again, your `reinforced_steel` node will look like Figure 6. I am sure you will agree that it looks quite good.

You can have different designs on each face. In that case, the `tiles` table would look like this:

```
tiles = {"top.png", "bottom.png", "right.png", ↩
  "left.png", "back.png", "front.png"}
```

You must, of course, change `top.png`, `bottom.png`, and son for the names of the images you want to use for the respective faces of your node.

Listing 3: init.lua – Safe node

```
...
minetest.register_node("personal:safe", {
 tiles = {
  "reinforced.png",
  "reinforced.png",
  "reinforced.png",
  "reinforced.png",
  "reinforced.png"
  "reinforced.png^safe_front.png"
 },
})
```

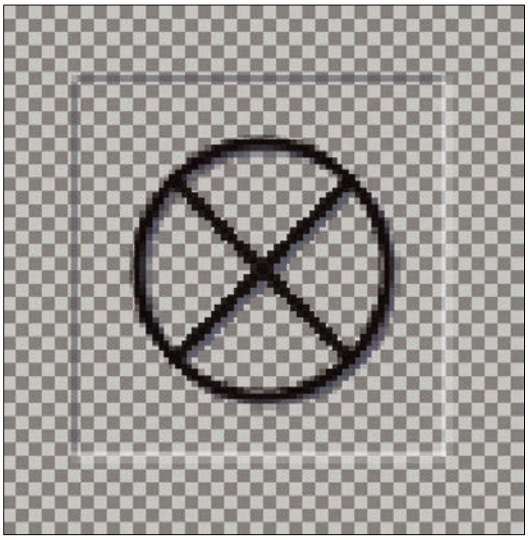


Figure 7: A transparent texture can be used as an overlay on top of another texture.

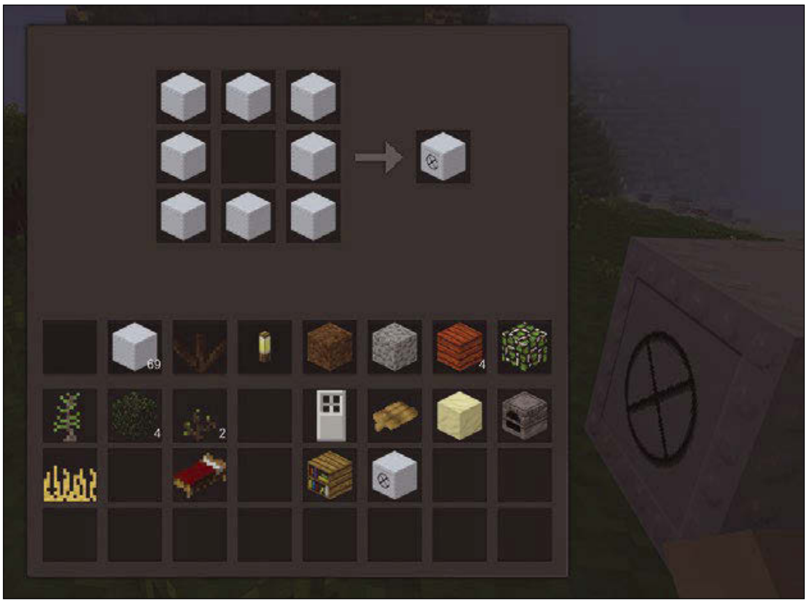


Figure 8: Crafting a safe using reinforced steel nodes.

Texturing can be much more sophisticated, too, since you can combine several textures together. Take a look at Listing 3. Notice how in this new node the front face gets a combined texture of `reinforced.png` with `safe_front.png` [8]. Figure 7 shows `safe_front.png`, a PNG with transparency. You can overlay it on top of `reinforced.png` using the ^ operator.

Add the `safe` node to your existing `init.lua`. But this time, when you go to test it, instead of "giving" it to yourself when you play (which is kind of cheating), let's give it a `recipe`, so you can build it from other materials. You can do this by adding the code shown in Listing 4 to `init.lua`.

The recipe tells the Minetest engine that players can build a safe in the crafting area by using steel nodes surrounding an empty space, as shown in Figure 8.

Listing 4: init.lua – Safe Recipe

```
...
minetest.register_craft ({
 output = "personal:safe",
 recipe = {
   {"personal:reinforced_steel", "personal:reinforced_steel",
"personal:reinforced_steel"},
   {"personal:reinforced_steel", "", "personal:reinforced_steel"},
   {"personal:reinforced_steel", "personal:reinforced_steel",
"personal:reinforced_steel"}
 },
})
```

Listing 5: init.lua – Reinforced Steel Node

```
minetest.register_node("personal:reinforced_steel", {
 tiles = {"reinforced.png"},
 drawtype = "nodebox",
 node_box = {
  type = "fixed",
   fixed = {
    {-0.5, -0.5, -0.5, 0.5, -0.3, 0.5},
   }
 }
})
```

Non-Cubic Nodes

There are several more things you may want to do with your nodes. One of them is to change their shapes. As with other mining and crafting games, you can have nodes in the shapes of steps, flat slabs, etc. For your reinforced steel nodes, you may want them to be flat, and you can achieve that with node boxes [9] (see Listing 5).

There are a few new things in here. First `drawtype` tells the Minetest engine that the node may not be a 1x1x1 cube, that it has to take the coordinates for the cuboids that make up the node from a table within the `nodebox` table. The `nodebox` item itself contains the kind of node box this is going to be, as well as the coordinates of the cuboid(s) that will make up the node.

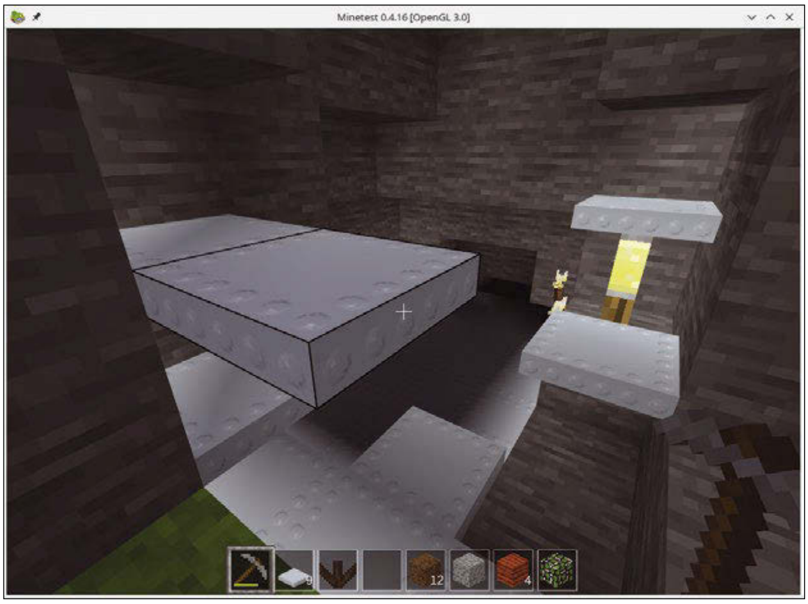
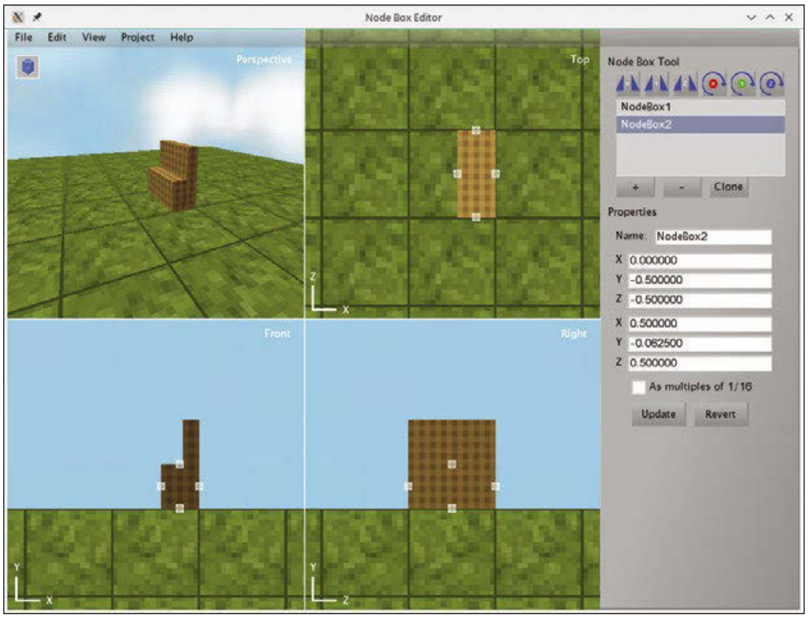


Figure 9: You can change the shape of a node using node boxes.

In this case, the node box is very simple: It is `fixed`, in the sense that it doesn't change whether you mount it on the floor, wall, or ceiling, and consists of one rather flat cuboid. You define the shape of each cuboid that makes up the node box using a table that contains the coordinates of two of its opposing corners. The cuboid in Listing 5 is 1 unit square and 0.2 units high (Figure 9).

If you find defining the shape of your node-box nodes using coordinates confusing, you can download the Node Box Editor utility [10]. The Node Box Editor (Figure 10) lets you edit the size and shape of multiple cuboids visually and even apply textures to them. Then you can export your node boxes to Lua code and incorporate them into your mods.



Name: NodeBox2
X 0.000000
Y -0.500000
Z -0.500000
X 0.500000
Y -0.062500
Z 0.500000
□ As multiples of 1/16

Figure 10: Use the Node Box Editor to create complex shapes.

Interacting

One final problem with your nodes is that, as they stand, they are unbreakable. This means that, once you lay them down, you can't pick them up again or bash them until they smash. They are indestructible.

You can make nodes destructible by adding the `groups` directive to the nodes table. A line that would make our reinforced steel crack and break after three or four consecutive blows with a pickax would be:

```
groups = {cracky = 3},
```

The `groups` directive tells the Minetest engine how the node is destroyed [11]. Hard objects, like stone and bricks, crack, so they belong to the `cracky` group. Softer objects, like dirt or sand, belong to the `crumbly` group. There are a wide variety of groups that cover flammability, wetness, liquid porosity, and so on.

Listing 6: init.lua Complete

```
minetest.register_node ("personal:reinforced_
                         steel", {
 tiles = {"reinforced.png"},
 groups = {cracky = 3},
 drawtype = "nodebox",
 node_box = {
  type = "fixed",
  fixed = {{-0.5, -0.5, -0.5, 0.5, -0.3, 0.5}},
 }
})

minetest.register_node ("personal:safe", {
 tiles = {
  "reinforced.png",
  "reinforced.png",
  "reinforced.png",
  "reinforced.png",
  "reinforced.png",
  "reinforced.png^safe_front.png"
 }
})

minetest.register_craft ({
 output = "personal:safe",
 recipe = {
   {"personal:reinforced_steel",
    "personal:reinforced_steel",
    "personal:reinforced_steel"},
   {"personal:reinforced_steel", "",
"personal:reinforced_steel"},
   {"personal:reinforced_steel",
    "personal:reinforced_steel",
    "personal:reinforced_steel"}
 }
})
```

The value assigned to the group tells Minetest how resistant the node is. The higher the value, the harder it is to break. As it stands, the so-called "reinforced" steel is pretty weak and cracks and breaks easily. You may want to increase that 3 to something higher.

Your final mod will look like Listing 6.

Conclusion

This article has only just scratched the surface of mod making for Minetest. You can interact with your nodes in many other ways. You have a whole slew of events you can trigger when you punch, dig, use, or right-click an object in your world. You could, for example, make your safe very, very resistant to digging, but, when right-clicked, it will ask players for a combination that will allow them to open the safe.

Minetest's possibilities are pretty much endless, and the modding community is thriving. Players and developers from all over the world are creating mods that help turn Minetest worlds into role playing games, educational environments, places where visitors can express their creativity, and so much more.

Minetest is a cracking good game in its own right, and its mod system offers a whole new level of control and flexibility that you won't find in the alternatives. ∎∎∎

Info

[1] Minetest: *https://www.minetest.net/*

[2] "Minetest" by Mike Saunders, *Linux Magazine*, issue 204, November 2017, pg. 70: *http://www.linux-magazine.com/Issues/2017/204/Minetest*

[3] Customize Minetest with mods: *https://www.minetest.net/customize/*

[4] Creatures, including zombies and ghosts: *https://github.com/BlockMen/cme*

[5] More creepy creatures: *https://wiki.minetest.net/Mods/Not_So_Simple_Mobs*

[6] Lua programming language: *https://www.lua.org/*

[7] Register a node: *http://dev.minetest.net/minetest.register_node*

[8] Node textures: *http://dev.minetest.net/texture*

[9] Node boxes: *http://dev.minetest.net/Node_boxes*

[10] Node Box Editor: *https://rubenwardy.com/NodeBoxEditor/*

[11] Node groups: *http://dev.minetest.net/Groups/Custom_groups*

FEATURED EVENTS

Users, developers, and vendors meet at Linux events around the world. We at *Linux Magazine* are proud to sponsor the Featured Events shown here.

For other events near you, check our extensive events calendar online at *http://linux-magazine.com/events.*

If you know of another Linux event you would like us to add to our calendar, please send a message with all the details to *events@linux-magazine.com*.

ISC High Performance 2018

Date: June 24–28, 2018

Location: Frankfurt, Germany

Website: *https://www.isc-hpc.com/*

ISC High Performance is dedicated to tackling HPC technological development and its application in scientific fields, as well as its adoption in commercial environments. This year the program team has selected 13 topics, which will be addressed in a number of thought-provoking sessions.

Akademy 2018

Date: August 11-17, 2018

Location: Vienna, Austria

Website: *https://akademy.kde.org/2018*

Akademy 2018 is expected to draw hundreds of attendees from the global KDE Community to discuss and plan the future of the Community and its technology. Many participants from the broad free and open source software community, local organizations, and software companies also will attend.

Linux Security Summit

Date: August 27-28, 2018

Location: Vancouver, Canada

Website: *https://events.linuxfoundation.org/events/linux-security-summit-north-america-2018/*

The Linux Security Summit (LSS) is a technical forum for collaboration between Linux developers, researchers, and end users to foster community efforts in analyzing and solving Linux security challenges.

Events

Texas Linux Fest	June 8–9, 2018	Austin, Texas	https://2018.texaslinuxfest.org/
Tübix	June 9, 2018	Tübingen, Germany	https://www.tuebix.org/anfahrt/
CeBIT 2018	June 11–15, 2018	Hannover, Germany	http://www.cebit.de/en/
Open Source Data Center Conf.	June 12–13, 2018	Berlin, Germany	https://osdc.de/
ISC High Performance 2018	June 24–28, 2018	Frankfurt, Germany	http://www.isc-hpc.com/
GUADEC	July 6–11, 2018	Almería, Spain	https://2018.guadec.org/
DebConf18	July 29–August 5, 2018	Hsinchu, Taiwan	https://wiki.debconf.org/wiki/DebConf18
Akademy 2018	August 11–17, 2018	Vienna, Austria	https://akademy.kde.org/2018
Atlassian Summit Europe	September 3–5, 2018	Barcelona, Spain	https://www.atlassian.com/company/events/summit-europe
Drupal Europe	September 10-14, 2018	Darmstadt, Germany	https://www.drupaleurope.org/
DevOpsDays Berlin	September 12–13, 2018	Berlin, Germany	https://www.devopsdays.org/events/2018-berlin/welcome/
Open Source Firmware Conf.	September 12–15, 2018	Erlangen, Germany	https://osfc.io/
Storage Developer Conf.	September 24–27, 2018	Santa Clara, California	https://www.snia.org/events/storage-developer
Open Networking Summit Europe	September 25–27, 2018	Amsterdam, Netherlands	https://events.linuxfoundation.org/events/open-networking-summit-europe-2018/
Open Source Backup Conference 2018	September 26–27, 2018	Cologne, Germany	https://upcoming.org/event/open-source-backup-conference-2018-0xlw0zogn5
All Systems Go	September 28-30, 2018	Berlin, Germany	https://all-systems-go.io/
All Things Open	October 21-23, 2018	Raleigh, North Carolina	https://allthingsopen.org/
Open Source Summit Europe	October 22–24, 2018	Edinburgh, UK	http://events.linuxfoundation.org/events/open-source-summit-europe
Linux Presentation Day 2018.2	November 10, 2018	Europe-wide in many cities	http://www.linux-presentation-day.de/
Linux Kernel Summit	November 12–15, 2018	Vancouver, Canada	https://events.linuxfoundation.org/events/linux-kernel-summit-2018/
IT-Days 2018	December 10-13, 2018	Frankfurt, Germany	https://www.ittage.informatik-aktuell.de/

Images © Alex White, 123RF.com

CALL FOR PAPERS

We are always looking for good articles on Linux and the tools of the Linux environment. Although we will consider any topic, the following themes are of special interest:

- System administration
- Useful tips and tools
- Security, both news and techniques
- Product reviews, especially from real-world experience
- Community news and projects

If you have an idea, send a proposal with an outline, an estimate of the length, a description of your background, and contact information to *edit@linux-magazine.com*.

The technical level of the article should be consistent with what you normally read in *Linux Magazine*. Remember that *Linux Magazine* is read in many countries, and your article may be translated into one of our sister publications. Therefore, it is best to avoid using slang and idioms that might not be understood by all readers.

Be careful when referring to dates or events in the future. Many weeks could pass between your manuscript submission and the final copy reaching the reader's hands. When submitting proposals or manuscripts, please use a subject line in your email message that helps us identify your message as an article proposal. Screenshots and other supporting materials are always welcome.

Additional information is available at:
http://www.linux-magazine.com/contact/write_for_us.

NOW PRINTED ON recycled paper from 100% post-consumer waste; no chlorine bleach is used in the production process.

Authors

Erik Bärwaldt	36, 68
Maximilian Batz	54
Gunnar Beutner	30
Swapnil Bhartiya	8, 16
Paul Brown	90
Zack Brown	12
Bruce Byfield	44, 62
Joe Casad	3
Mark Crutch	65
Marco Fioretti	84
Jon "maddog" Hall	67
Stefan Heinle	18
Frank Hofmann	74
Charly Kühnast	52
Vincent Mealing	65
Anzela Minosi	58
Graham Morrison	78
Mandy Neumeyer	74
Mike Schilli	48
Tim Schürmann	24

Contact Info

Editor in Chief
Joe Casad, jcasad@linux-magazine.com

Managing Editor
Rita L Sooby, rsooby@linux-magazine.com

Localization & Translation
Ian Travis

News Editor
Swapnil Bhartiya

Copy Editor
Amy Pettle

Layout
Dena Friesen, Lori White

Cover Design
Dena Friesen

Cover Image
© arrow , 123RF.com

Advertising
Brian Osborn, bosborn@linuxnewmedia.com
phone +49 89 3090 5128

Marketing Communications
Gwen Clark, gclark@linuxnewmedia.com
Linux New Media USA, LLC
2721 W 6th St, Ste D
Lawrence, KS 66049 USA

Publisher
Brian Osborn

Customer Service / Subscription
For USA and Canada:
Email: cs@linuxpromagazine.com
Phone: 1-866-247-2802
(Toll Free from the US and Canada)

For all other countries:
Email: subs@linux-magazine.com

www.linuxpromagazine.com – North America
www.linux-magazine.com – Worldwide

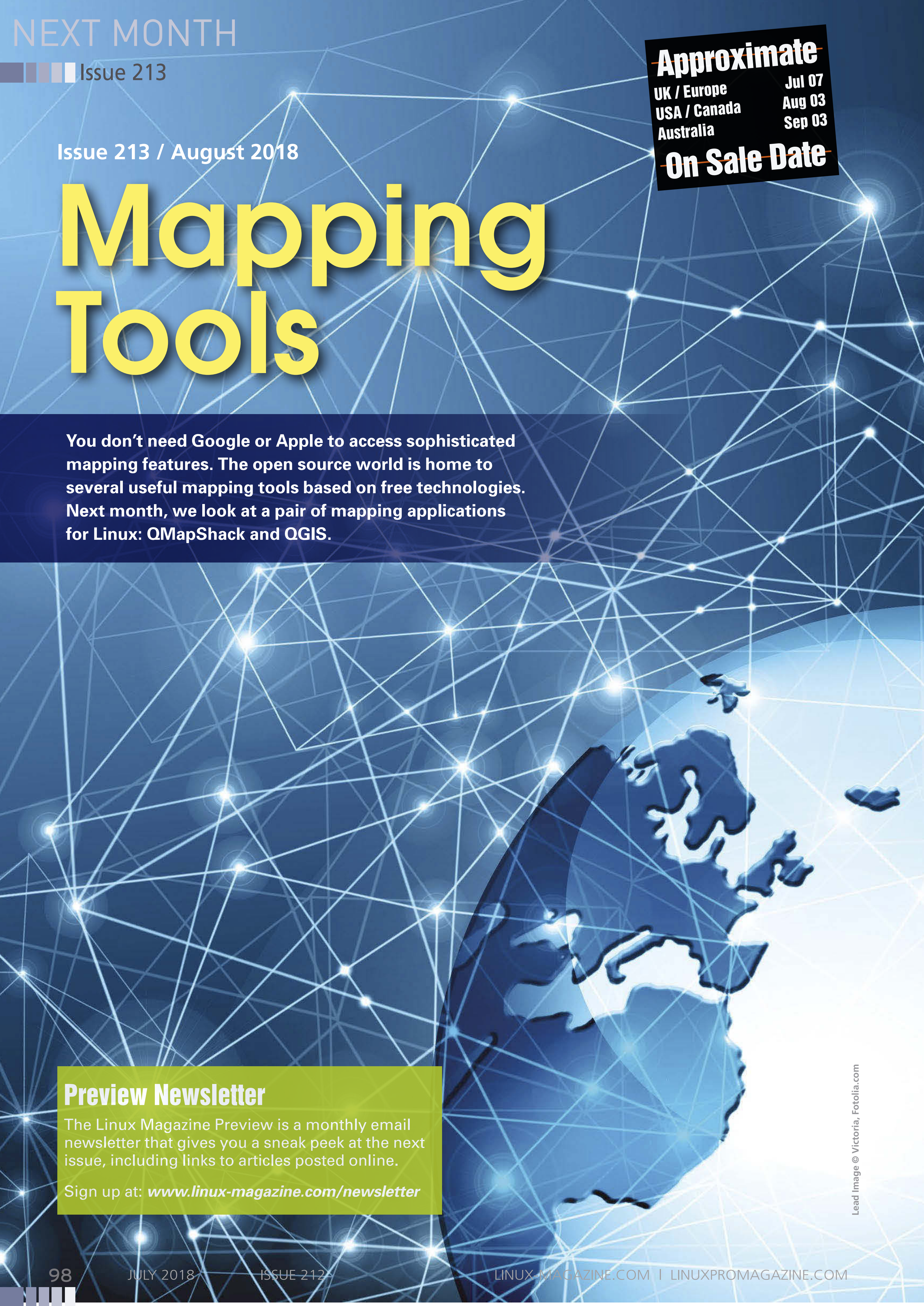
While every care has been taken in the content of the magazine, the publishers cannot be held responsible for the accuracy of the information contained within it or any consequences arising from the use of it. The use of the disc provided with the magazine or any material provided on it is at your own risk.

Printed in Nuremberg, Germany by hofmann infocom GmbH on recycled paper from 100% post-consumer waste; no chlorine bleach is used in the production process.

Distributed by Seymour Distribution Ltd, United Kingdom

LINUX PRO MAGAZINE (ISSN 1752-9050) is published monthly by Linux New Media USA, LLC, 2721 W 6th St, Ste D, Lawrence, KS, 66049, USA. Periodicals Postage paid at Lawrence, KS and additional mailing offices. Ride-Along Enclosed. POSTMASTER: Please send address changes to Linux Pro Magazine, 2721 W 6th St, Ste D, Lawrence, KS 66049, USA.

Published monthly in Europe as Linux Magazine (ISSN 1471-5678) by: Sparkhaus Media GmbH, Zieblandstr. 1, 80799 Munich, Germany.



NEXT MONTH

Issue 213

Issue 213 / August 2018

Approximate	
UK / Europe	Jul 07
USA / Canada	Aug 03
Australia	Sep 03
On Sale Date	

Mapping Tools

You don't need Google or Apple to access sophisticated mapping features. The open source world is home to several useful mapping tools based on free technologies. Next month, we look at a pair of mapping applications for Linux: QMapShack and QGIS.

Preview Newsletter

The Linux Magazine Preview is a monthly email newsletter that gives you a sneak peek at the next issue, including links to articles posted online.

Sign up at: ***www.linux-magazine.com/newsletter***

Lead Image © Victoria, Fotolia.com

Application-Optimized Server Solutions

Scientific Computing Built to Meet Your Specific Needs

The Fastest Image Processor in the World

With 41 of our GPU optimized SYS-7048GR-TR servers,
the New GPU based Supercomputer named
ARTS (Apertif Radio Transient System) is now the Fastest
in the Netherlands with a calculation capacity of two PetaFLOPS.
This supercomputer is Application-Optimized
to scan 20,000 images per second then uses machine learning
algorithms to detect if there are any fast explosions in space
using one of the largest Radio Telescopes in the World.
For Exactly the Best Enterprise Computing Solution
for your Application, Choose Supermicro.

intel XEON PLATINUM inside™

Intel Inside®. Powerful Productivity Outside.

SUPERMICRO®

Application-Optimized Server Solutions

Learn more at supermicro.com/application-optimized