



Full Circle

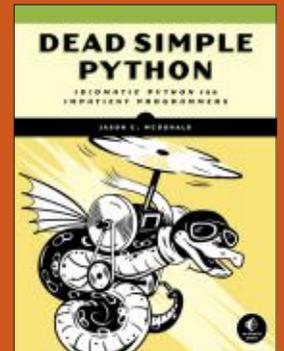
THE INDEPENDENT MAGAZINE FOR THE UBUNTU LINUX COMMUNITY

ISSUE #186 - October 2022



NixOS LAST OF THE NON-DEBIAN REVIEWS

BOOK REVIEW



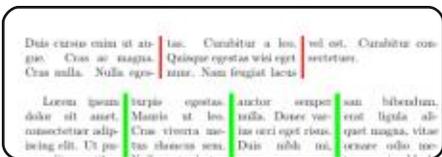
HowTo



Python p.17



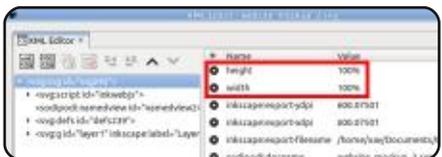
Blender p.21



Latex p.23



p.XX



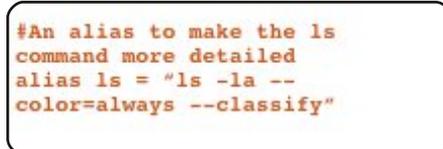
Inkscape p.25

Graphics



Full Circle

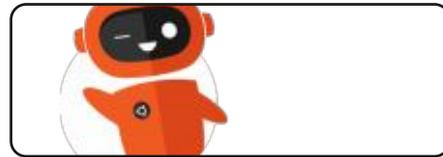
THE INDEPENDENT MAGAZINE FOR THE UBUNTU LINUX COMMUNITY



Command & Conquer p.15



... p.XX



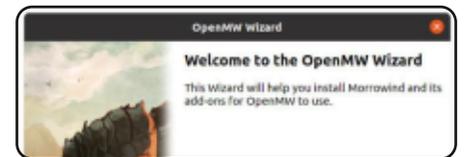
Ubuntu Devices p.XX



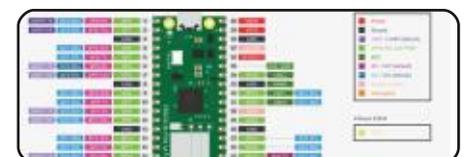
The Daily Waddle p.30



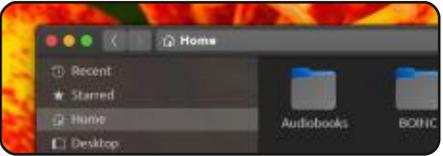
Linux News p.04



Everyday Ubuntu p.31



Micro This Micro That p.33



My Story p.40



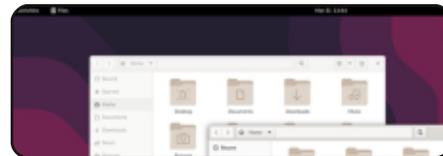
Letters p.XX



Review p.43



Q&A p.54



Review p.47



Ubuntu Games p.XX



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

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



WELCOME TO THE LATEST ISSUE OF FULL CIRCLE

We're firing on all cylinders this issue with Python, Blender, Latex, Micro This Micro That and Inkscape.

I'm sure that, by now, some of you will have upgraded to 22.10, but if you're still on the fence then next month will start Adam's reviews of 22.10. This issue, though, we have a look at Ubuntu Budgie 22.04 and NixOS. And as a bonus we have a book review from Greg.

Also, we have the start of a new (possibly bi-monthly) column from Josh Hertel who's going to introduce you to the delights of digital tabletop games. Way back in FCM#110 I did a quick look at Vassal which is great for digital wargames.

Don't forget: if you're looking for some help, advice, or just a chit chat: remember, we have a **Telegram** group: <https://t.me/joinchat/24ec1oMFO1ZjZDc0>. I hope to see you there. Come and say hello.

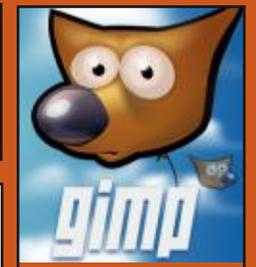
Anyway, all the best, and stay safe!

Ronnie

ronnie@fullcirclemagazine.org



This magazine was created using :



Find Full Circle on:



facebook.com/fullcirclemagazine



twitter.com/#!/fullcirclemag



<http://issuu.com/fullcirclemagazine>



<https://mastodon.social/@fullcirclemagazine>

Weekly News:



<http://fullcirclemagazine.org/feed/podcast>



<http://www.stitcher.com/s?fid=85347&refid=stpr>



<http://tunein.com/radio/Full-Circle-Weekly-News-p855064/>

FCM PATREON: <https://www.patreon.com/fullcirclemagazine>



LINUX KERNEL RELEASE 6.0:

03/10/2022

After two months of development, Linus Torvalds released the Linux 6.0 kernel. A major change in the number of the version is made for aesthetic reasons and is a formal step that relieves discomfort due to the accumulation of a large number of issues in the series (Linus joked that the reason for changing the number of the branch is rather that his fingers on his hands and legs count the numbers of versions). Among the most notable changes: support for asynchronous buffered recording in XFS, block driver ublk, task scheduler optimization, mechanism for verifying the correctness of the kernel, support for the ARIA block cipher.

The new version accepted 16585 fixes from 2129 developers, patch size was - 103 MB (changes affected 13939 files, added 1420093 lines of code, removed 318741 lines). About 40% of all changes presented in 6.0 are related to device drivers, about 19% of the changes are related to

updating code specific to hardware architectures, 12% related to the network stack, 4% to file systems and 2% with internal kernel subsystems.

<https://lkml.org/lkml/2022/10/2/255>

RELEASE OF THE STELLARIUM 1.0:

03/10/2022

After 20 years of development, the Stellarium 1.0 project was released, providing a free 'planetarium' for three-dimensional navigation on the star sky. The basic catalog of celestial objects has more than 600,000 stars and 80,000 objects of deep space (additional catalogs cover more than 177 million stars and more than a million objects of deep space), and also includes information about constellations and nebulae. The project code is written in C++ using the Qt framework and distributed under the GPLv2 license. The builds are supplied for Linux, Windows

and macOS.

The interface provides flexible scaling, 3D-visualization and simulation of various objects. It supports projection on the dome of the 'planetarium', the creation of mirror projections and integration with a telescope. Plugins can be used to expand the functionality and control of the telescope. It is possible to add your own space objects, simulate artificial satellites and the implementation of their looks.

The new version has implemented the transition to the Qt6 framework and provides an acceptable level of accuracy of reproduction of past states. There is increased detail during simulation of eclipses. They expanded the capabilities of the astronomical calculator, improved work on screens with high pixel density (HiDPI), added information about the perception of objects of the starry sky by the culture of the peoples of the Samoa archipelago.

<https://stellarium.org/release/2022/10/01/stellarium-1.0.html>

RELEASE OF ICEWM 3.0.0 WITH TAB SUPPORT:

03/10/2022

The release of the lightweight window manager IceWM 3.0.0 is available. IceWM provides full control through keyboard combinations, the ability to use virtual desktops, task bar and application menus. The window manager is configured through a fairly simple configuration file, themes are also supported. Built-in applets for monitoring CPU, memory, traffic. Separately, several third-party GUIs for customization, desktop implementations and menu editors are available. The code is written in C++ and is distributed under the GPLv2 license.

A major change in the version number is a natural continuation of the version numbered in the project, 2.99 to 3.0. However, a

major innovation is also presented in the 3.0 branch - the ability to switch between windows using tabs. The window in IceWM can now include several client windows, switching between them is carried out using tabs. To merge windows via a tab, it is enough to drag the header of one window to the header of another window, using the middle mouse button. To navigate the tabs using the keyboard, you can use the combinations of Alt+F6 and Alt+Shift+Esc. The tabs are also displayed in the window submenu.

<https://github.com/ice-wm/icewm/releases/tag/3.0.0>

COSMIC CUSTOM ENVIRONMENT WILL USE ICED INSTEAD OF GTK

04/10/2022

Michael Aaron Murphy, the leader of the developers of the Pop!_OS distribution and a participant in the development of the Redox operating system, spoke about the work on the new edition of the COSMIC user environment. COSMIC is transformed into a self-sufficient project that does not use

GNOME Shell and is developed in Rust. The environment is planned to be used in Pop!_OS, on laptops and PCs of System76.

The Iced library is fully written in Rust, uses safe types, modular architecture and reactive programming. There are several drawing engines that support Vulkan, Metal, DX12, OpenGL 2.1+ and OpenGL ES 2.0+, as well as a window shell and engine for Web integration. Applications based on Iced can be collected for Windows, macOS, Linux and launch in a web browser. The developers have offered a ready-made set of widgets, the ability to create asynchronous handlers and use an adaptive layout of the interface elements depending on the size of the window and the screen. The code is distributed under the MIT license.

<https://twitter.com/mmstick>
<https://iced.rs/>

RELEASE OF OPENSSH 9.1:

05/10/2022

After six months of development, the OpenSSH

9.1, open client implementation and server for SSH 2.0 and SFTP protocols were released. The release is characterized as mainly containing bug fixes, including the elimination of several potential vulnerabilities caused by problems when working with memory.

<https://lists.mindrot.org/pipermail/openssh-unix-dev/2022-October/040438.html>

ALP PROTOTYPE, REPLACES SUSE ENTERPRISE LINUX:

05/10/2022

SUSE has published the first prototype of the ALP (Adaptable Linux Platform), positioned as a continuation of the development of SUSE Enterprise Linux. The key difference between the new system is the division of the very basics of the distribution into two parts: a trimmed "host OS" to work on top of the metal and a layer for application support, focused on launching in containers and virtual machines. The builds are available for x86_64 at the moment.

As the basis for "host OS" is SEL

Micro, based on the MicroOS project. For centralized management, Salt (pre-installed) and Ansible configuration systems are offered. Podman and K3s (Kubernetes) tools are available for launching isolated containers. Among the system components put in containers are yast2, podman, k3s, cockpit, GDM (GNOME Display Manager) and KVM.

<https://www.suse.com/c/the-first-prototype-of-adaptable-linux-platform-is-live/>

RELEASE OF WIRESHARK 4.0:

06/10/2022

The release of a new stable branch of the network analyzer Wireshark 4.0 has been published. Recall that initially the project was called Ethereal, but in 2006 due to the conflict with the owner of the trademark Ethereal, the developers were forced to rename the project to Wireshark. The project code is distributed under the GPLv2 license.

<https://www.wireshark.org/news/20221004.html>

REDCORE LINUX 2201

RELEASE:

06/10/2022

A year after the last release, the Redcore Linux 2201 distribution has been released, which is trying to combine Gentoo's functionality with simple convenience for ordinary users. The distribution provides a simple installer that allows you to quickly deploy a working system without requiring building components from source code. Users are provided with a repository with ready-made binary packages, accompanied by a continuous update cycle (rolling-model). To manage the packages involved, they use their own package manager sisyphus. For installation,

an iso-image with a KDE desktop, of 4.2 GB (x86_64) is offered.

<https://redcorelinux.org/news/redcore-linux-hardened-2201-rastaban-stable>

RETROARCH 1.11:

06/10/2022

RetroArch 1.11 has been published, a superstructure for emulation of various game consoles, which allows you to run classic games with a simple unified graphical interface. The emulators include consoles such as Atari 2600/7800/Jaguar/Lynx, Game Boy, Mega Drive, NES, Nintendo 64/DS, PCEngine, PSP, Sega 32X/CD, SuperNES, etc. Gamepads from existing game consoles, including the Playstation 3, Dualshock 3, 8bitdo, Xbox 1 and Xbox360, as

well as general purpose gamepads such as the Logitech F710. The emulator supports advanced capabilities like multiplayer games, save state, improvement of the quality of the image of old games with the help of shaders, rewinding the game back, hot connection of game consoles and video streaming.

<https://www.libretro.com/index.php/retroarch-1-11-0-release>

CANONICAL LAUNCHES FREE EXTENDED UPDATES SERVICE FOR UBUNTU:

07/10/2022

Canonical has provided a free subscription to the Ubuntu Pro commercial service (formerly Ubuntu Advantage), which provides

access to extended updates for Ubuntu LTS. The service provides an opportunity to receive updates for vulnerabilities for 10 years (the full term of support for LTS-branch 5 years) and opens access to live patches that allow you to apply updates to the Linux kernel without rebooting on the fly.

A free subscription to Ubuntu Pro is provided for individuals and small businesses with up to 5 physical hosts in their infrastructure (the program also covers all virtual machines placed on these hosts). To get access tokens to the service, Ubuntu Pro free requires an account in Ubuntu One, which anyone can get. To subscribe to extended updates, use the "pro attach" command or the "Software & Updates" graphical application (Livepatch tab).

Additionally, they announced the development of upgrades for new categories of applications for workstations and data centers. For example, the release of extended updates will now cover packages such as Ansible, Apache Tomcat, Apache Zookeeper, Docker, Drupal, Najos, Node.js, phpMyAdmin, Puppet, PowerDNS, Python 2, Redis, Rust and WordPress.



DistroWatch.com

Put the fun back into computing. Use Linux, BSD.

<https://ubuntu.com//blog/ubuntu-pro-beta-release>

KAOS 2022.10:

08/10/2022

The release of KaOS 2022.10, a distribution with a continuous update model aimed at providing a desktop based on fresh releases of KDE and applications using Qt, is out. Out of all the features specific to the design, you can note the placement of a vertical panel on the right side of the screen, like MX Linux. The distribution is evolving with an eye on Arch Linux, but supports its own independent repository, with over 1,500 packages and offers a number of its own graphical utilities. XFS is used as a default file system. Builds are published for x86_64 (2.9 GB).

<https://kaosx.us/news/2022/kaos10/>

PARROT 5.1 RELEASED:

08/10/2022

Parrot 5.1, based on Debian 11 and includes a selection of

tools to verify system security, forensic analysis and reverse engineering is available for download. Several iso-images with an MATE environment designed for everyday use, security testing, installation on Raspberry Pi 4 boards and the creation of specialized installations, for example, for use in cloud environments, are proposed.

The Parrot distribution is positioned as a portable lab environment for security experts and criminologists, which focuses on means to verify cloud systems and devices of the Internet. The distro also includes cryptographic tools and programs to ensure secure access to any network, including TOR, I2P, anonsurf, gpg, tccf, zulucrypt, veracrypt, truecrypt and luks.

<https://parrotsec.org/blog/2022-09-24-parrot-5.1-release-notes/>

KDE 6 BRANCH SOON:

09/10/2022

At the KDE Academy 2022 conference held in Barcelona, the KDE 6 branch development plan

was mentioned. The release of Plasma 5.27 desktop will be the last in the KDE 5 series and after that, the developers will begin the KDE 6 branch. A key change in the new branch will be the transition to Qt 6 and the delivery of an updated basic set of libraries and runtime components KDE Framework 6, which forms the KDE software stack.

At the end of December, they plan to freeze the KDE Framework 5 branch for new features and begin to build KDE Frameworks 6. In addition to adapting to work on top of the Qt 6 in KDE Frameworks 6, processing of the API is also planned, in the new branch it will be possible to revise some concepts

As for the desktop KDE Plasma 6.0, the main focus in the preparation of this release is error correction. The release is expected in about a year - KDE Plasma 5.27 is expected in February, after which the summer issue (5.28) will be missed and the release of KDE Plasma 6.0 will be released in the fall of 2023 instead of the release of 5.29.

<https://blog.broulik.de/2022/10/physical-akademy-2022-in-barcelona/>

THE WAYLAND-PROTOCOLS

1.27:

10/10/2022

The wayland-protocols 1.27 package containing a set of protocols and extensions that complement the capabilities of the Wayland protocol and provide the capabilities necessary to build composite servers and user environments was published.

All protocols consistently pass three phases - development, testing and stabilization. After the development stage the protocol is placed in the staging branch and is officially included in the modelland-protocols set, and after the completion of this test, it is moved to stable. Protocols from the category "staging" can already be used in composite servers and clients, where the functionality associated with them is required. Unlike the "unstable" category, "staging" prohibits changes that break compatibility, but in the case of problems and shortcomings

during testing, it is not excluded to replace a new version of the protocol or other Wayland extension.

<https://lists.freedesktop.org/archives/wayland-devel/2022-October/042466.html>

RELEASE OF DRBD 9.2.0: 10/10/2022

The release of a distributed replicated block device DRBD 9.2.0, which allows you to implement something similar to a RAID-1 array, made up from network-connected disks of different machines (mirroring on the network). The system is designed as a module for the Linux kernel and is distributed under the GPLv2 license. The drbd 9.2.0 branch can be used to transparently replace drbd 9.x.x and is fully compatible at the protocol level, configuration files and utilities.

DRBD makes it possible to combine cluster node drives into a single failover storage. For applications and systems, this storage looks like a block device for all systems. When using DRBD, all

operations with a local disk are sent to other nodes and synchronized with the disks of other machines. In case of failure of one node, the storage will automatically continue to work at the expense of the remaining nodes. When the availability of the failed node is resumed, its condition will be automatically brought to the current type.

The cluster forming the storage can include several dozen nodes located both in a local network and geographically distributed to different data centers. Synchronization in such branched storages are performed using mesh-network technologies (data spread along the chain from node to node). The mapping of nodes can be performed both in synchronous mode and in asynchronous mode. For example, locally placed nodes can use synchronous replication, and for remote sites, asynchronous replication with additional compression and traffic encryption can be used.

<https://lists.linbit.com/pipermail/drbd-announce/2022-October/000594.html>

RELEASE OF VIRTUALBOX 7.0: 11/10/2022

After almost three years since the last major release, Oracle has released the VirtualBox 7.0 virtualization system. Ready-made installation packages are available for Linux (Ubuntu, Fedora, openSUSE, Debian, SLES, RHEL builds for AMD64 architecture), Solaris, macOS and Windows.

<https://www.mail-archive.com/vbox-announce@virtualbox.org/msg00218.html>

RELEASE OF KDE PLASMA 5.26: 11/10/2022

The user shell, KDE Plasma 5.26, built using the KDE Framework 5 platform and the Qt 5 library using OpenGL/OpenGL ES to speed up the graphics, was announced. To evaluate the new version, you can access it through the Live-build from the openSUSE project and builds from the KDE Neon User Edition project. Packages for various distributions can be found

on this page. Release 5.26 is likely to be the penultimate before the KDE Plasma 6.0 line, built on Qt 6.

<https://kde.org/announcements/plasma/5/5.26.0/>

THE DAILY BLENDER BUILDS INCLUDE WAYLAND SUPPORT: 11/11/2022

Developers of Blender reported the inclusion of support for the Wayland protocol in daily updated test builds. In stable releases, Wayland's native support is planned to be offered in Blender 3.4. The decision to support Wayland is due to the desire to get rid of restrictions when using XWayland and improve the quality of work in Linux distributions that use Wayland by default.

To work with the environment on Wayland, you need to install the libdecor library for decorating windows on the client's side. Among the features that are not yet available in the builds on Wayland, there is a lack of support for tablets, 3D mice (NDOF), screens with high pixel density, window frames and cursor

alignment.

<https://code.blender.org/2022/10/wayland-support-on-linux/>

RELEASE OF KATA CONTAINERS 3.0:

11.10.2022

After two years of development, the Kata Containers 3.0 project is out, developing a stack to organize containers using insulation based on full-fledged virtualization mechanisms. The project was created by Intel and Hyper by combining Clear Containers and runV technologies. The project code is written in Go and Rust and is distributed under the Apache 2.0 license. The development of the project is supervised by a working group created under the auspices of the independent organization OpenStack Foundation, which includes companies such as Canonical, China Mobile, Dell/EMC, EasyStack, Google, Huawei, NetApp, Red Hat, SUSE and ZTE.

The basis of Kata is runtime, which allows you to create compact virtual machines performed using a full-fledged hypervisor, instead of

using traditional containers that use the common Linux kernel and isolated by the namespaces and cgroups. The use of virtual machines allows you to achieve a higher level of security, protecting against attacks caused by vulnerabilities in the Linux kernel.

<https://github.com/kata-containers/kata-containers/releases/tag/3.0.0>

A ROBOT NAMED FIGHT GAME CODE:

13/10/2022

The source code of 'A Robot Named Fight', developed as a roguelike, has been published. The player controls a robot to investigate the procedurally generated non-repeating levels of the mazelike, collect artifacts and bonuses, perform tasks to access new content, destroy attackers and in the finale to fight the main monster.

The code is written in C using the Unity engine and published under its own proprietary license prohibiting the distribution of derivative works for commercial purposes. However, the author of

the game said that he was considering the possibility of translating the code to the GPL or a similar license.

<https://store.steampowered.com/news/app/603530/view/3293844171243621529>

THE MACHINE LEARNING SYSTEM FOR THE GENERATION OF REALISTIC HUMAN MOVEMENTS:

13/10/2022

A group of researchers from Tel Aviv University have opened source code associated with the MDM (Motion Diffusion Model) machine learning system, which allows you to generate realistic human movements. The code is written in Python using the PyTorch framework and is distributed under the MIT license. For experimentation, you can use both ready-made models and train models independently with the help of the proposed scripts, for example, using a collection of three-dimensional images of a person HumanML3D. GPUs with CUDA support are required to train

the system.

The proposed system attempts to use diffusion models to generate movements that are inherently better suited for simulating human movements, but not without disadvantages, such as high requirements for computing resources and management complexity. To minimize the diffusion deficiencies in MDM, a neural network with the "transformer" architecture and sample prediction (sample) is used instead of predicting noise at each stage, making it easier to prevent anomalies, such as loss of contact with the surface with the leg.

<https://guytevet.github.io/mdm-page/>

RELEASE OF DBMS POSTGRESQL 15:

13/10/2022

After a year of development, a new stable branch of PostgreSQL 15 DBMS was published. Updates for the new branch will be released for five years until November 2027.

<https://www.postgresql.org/about/news/postgresql-15-released-2526/>

PROJECT GENODE PUBLISHED OS SCULPT 22.10:

14/10/2022

Sculpt 22.10, from Genode OS Framework, a general-purpose operating system that can be used by ordinary users to perform everyday tasks, is out. The original code of the project is distributed under the AGPLv3 license. LiveUSB image, 28 MB, is available for download. Work on systems with x86 processors and Intel graphics subsystem with included VT-d and VT-x extensions is supported

<https://genode.org/news/sculpt-os-release-22.10>

GOOGLE HAS OPENED THE CODE OF KATAOS (AND SPARROW):

15/10/2022

Google has announced the opening of the code related to

the KataOS project, aimed at creating a secure operating system for embedded equipment. The KataOS system components are written in Rust and executed on top of the seL4 microkernel, for which RISC-V systems provide mathematical proof of reliability, indicating full compliance with the code specifications specified in a formal language. The project code is open under the Apache 2.0 license.

The current GitHub release includes most of the KataOS core pieces, including the frameworks we use for Rust (such as the seL4-sys crate, which provides seL4 syscall APIs), an alternate rootserver written in Rust (needed for dynamic system-wide memory management), and the kernel modifications to seL4 that can reclaim the memory used by the rootserver. And we've collaborated with Antmicro to enable GDB debugging and simulation for our target hardware with Renode.

<https://opensource.googleblog.com/2022/10/announcing-kataos-and-sparrow.html>

RELEASE OF ARDOUR 7.0:

16/10/2022

After more than a year of development, the release of the free sound editor Ardour 7.0, designed for multi-channel recording, processing and mixing of sound, was published. Ardour provides a multi-track timeline, an unlimited level of redo of changes throughout the file (even after the program is closed), support for a variety of hardware interfaces. The program is positioned as a free analogue of professional tools like, ProTools, Nuendo, Pyramix and Sequoia. The code is distributed under the GPLv2 license. Ready-made builds for Linux are available in Flatpak format.

<https://ardour.org/whatsnew.html>

UPDATING VOID LINUX INSTALLATION BUILDS:

17/10/2022

New bootable builds of the Void Linux distribution have been released, which is an independent project that does not use the

developments of other distributions and is developed using a continuous program version update cycle. Past builds were published a year ago. In addition to the appearance of current boot images based on a more recent snapshot of the system, updating the builds of functional changes makes sense only for new installations (in already installed package update systems are delivered as ready).

The builds are available in variants based on Glibc and Musl system libraries. For x86_64, i686, armv6l, armv7l and aarch64, there are live-images with Xfce and basic console builds. The builds for ARM support; BeagleBone/BeagleBone Black, Cubieboard 2, Odroid U2/U3, RaspberryPi (ARMV6) and Raspberry Pi boards. In contrast to previous releases, the new builds for the Raspberry Pi are now combined into universal images for Raspberry Pi boards based on armv6l (1 A, 1 B, 1 A+, 1 B+, Zero, Zero, Zero), WH arm7l (2 B) and aarch64 (3 B, 3 A+, 3 B+, 3 B+, Zero 2W, 4 B,400).

The distribution is used for initialization and management of services by the runit system

manager. To manage packages, they are making xbps and a xbps-src packet assembly system. Xbps allows you to install, delete and update applications, identify incompatibility of shared libraries and manage dependencies. As a standard library, instead of Glibc, you can use Musl. The systems is distributed under the BSD license.

<https://voidlinux.org/news/2022/10/new-images.html>

RHINO LINUX, A ROLLING DISTRIBUTION BASED ON UBUNTU:

17/10/2022

The developers of the Rolling Rhino Remix announced the transformation of the project into a separate distribution - Rhino Linux. The reason for the creation of a new product was the revision of the goals and model of the project, which has already outgrown the state of amateur development and began to go beyond the simple respin of Ubuntu. The new distribution will still continue to be based on Ubuntu, but will include additional utilities and a team of several developers (two more

members have joined the work).

A slightly redesigned version of Xfce will be offered as a desktop. The main line-up will include a Pacstall packet manager, positioned as an analogue of the AUR (Arch User Repository) repository for Ubuntu, allowing third-party developers to distribute their packages without inclusion in the main repository of the distribution. Through the repository implemented with Pacstall, Xfce desktop components, the Linux kernel, bootable screensaver and Firefox browser will be distributed.

<https://rhinolinux.org/more.html>

RELEASE OF TAILS 5.5:

17/10/2022

The release of the specialized distribution Tails 5.5 (The Amnesic Incognito Live System), based on Debian and designed for anonymous access to a network, is out. Anonymous access to Tails is provided by the Tor system. All connections, except traffic over the Tor network, are blocked by the default packet filter. Encryption is used to store user data between

launches.

The new version uses the Linux kernel 5.10.140, improved support for new graphics cards and wireless devices. Tor Browser has been updated until release 11.5.4, which included vulnerabilities patches transferred from Firefox ESR 102.3. For the wget utility, the use of different Tor chains at each start is provided. The Thunderbird email client has been upgraded to 102.102.

https://tails.boum.org/news/version_5.5/index.en.html

RELEASE OF THE OPEN 3D ENGINE 22.10:

18/10/2022

The Open 3D Foundation (O3DF) has unveiled Open 3D Engine 22.10 (O3DE), suitable for the development of modern AAA games and high-precision simulators capable of working in real time and providing quality cinematics. The code is written in C++ and published under the Apache 2.0 license. It supports Linux, Windows, macOS, iOS and Android platforms.

The original code of the O3DE engine was opened in July 2021 by Amazon and are based on the code of the previously developed proprietary Amazon Lumberyard engine, built on CryEngine engine technology licensed from Crytek in 2015. After that, the development of the engine is supervised by the non-profit organization Open 3D Foundation, created under the auspices of the Linux Foundation. In addition to Amazon, companies such as Epic Games, Adobe, Huawei, Microsoft, Intel and Niantic joined the joint work on the project.

The engine includes an integrated game development environment, a multithreaded Atom Renderer photoreal rendering system with Vulkan, Metal and DirectX 12 support, an extensible 3D model editor, character animation system (Emotion FX), a semifab development system, a real-time physical process simulation engine and mathematical libraries that use SIMD instructions. Visual programming environment (Script Canvas) and Lua and Python languages can be used to define game logic.

<https://www.o3de.org/blog/posts/o3de-22-10-release/>

RELEASE OF ERGOFramework 2.2:

18/10/2022

The next release of ErgoFramework 2.2 is out, implementing the full Erlang network stack and its OTP library in Go. The framework provides a developer with flexible tools from the world of Erlang for the creation of distributed solutions in the Go language using ready-made general-purpose design templates.

In addition, the framework provides proxies functionality with the possibility of end-to-end encryption, not available in Erlang/OTP and Elixir. Since there is no direct analogue of the Erlang process in the Go language, the goroutine process is used in the framework as the basis for gen.Server with a wrapper "recover" for the possibility of handling exceptional situations. The project code is distributed under the MIT license.

The network stack in

ErgoFramework fully implements the DIST specification of the Erlang protocol. This means that applications written on the basis of ErgoFramework work natively with any applications written in Erlang or Elixir (an example of interaction with Erlang node). It is also worth noting that the design template gen.Stage is implemented according to the Elixir GenStage specification and fully compatible with it (example implementation).

<https://github.com/ergo-services/ergo>

RELEASE OF STRATIS 3.3:

19/10/2022

The Stratis 3.3 project, developed by Red Hat and the Fedora community, has been published to unify and simplify the pool configuration and management tools from one or more local drives. Stratis provides things dynamic allocation of space in storage, homeshots, integrity and layer creation for caching. Stratis support is integrated into the Fedora and RHEL distributions since the releases of Fedora 28 and RHEL 8.2. The project code is

distributed under the MPL 2.0 license.

The system largely repeats in its capabilities, advanced tools for managing ZFS and Btrfs partitions, but implemented as a layer (demonstrated), running on top of the device-mapper subsystem of the Linux kernel (dm-thin, dm-cache, dm-current, dm-inpool, dm-ray-ray and dm-integrity modules are used. Unlike ZFS and Btrfs, Stratis components only work in user space and do not require the core specific modules.

<https://github.com/stratis-storage/stratisd/releases/tag/v3.3.0>

RELEASE OF ANTI X 22:

19/10/2022

The release of the lightweight Live distribution AntiX 22, built on Debian and focused on outdated equipment, was released. Though the release is based on Debian 11, it comes without systemd manager and with eudev instead of udev. Runit or sysvinit can be used for initialization. The user environment is built by default with the help of the window manager IceWM, but

fluxbox, jwm and herbstluftwm are additionally included in the delivery. The size of iso-images: 1.5 GB (full, includes LibreOffice), 820 MB (basic), 470 MB (without graphics) and 191 MB (networking). Builds are prepared for x86_64 and i386 architectures.

<https://antixlinux.com/antix-22-released/>

COREBOOT 4.18:

19/10/2022

The release of the CoreBoot 4.18 project has been published, a free alternative to proprietary firmware and BIOS is being developed. The project code is distributed under the GPLv2 license. More than 200 developers took part in the creation of the new version, who prepared more than 1,800 changes.

<https://blogs.coreboot.org/blog/2022/10/18/announcing-coreboot-4-18/>

RELEASE OF ASTERISK 20:

20/10/2022

After a year of development, a new stable branch of the open communication platform Asterisk 20, used to deploy software PBXs, voice communication systems, VoIP-cluses, IVR-systems (voice menu), voice mail, telephone conferences and call centers, were released. The source code of the project is available under the GPLv2 license.

Asterisk 20 is classified as extended support (LTS), which will be released for five years instead of typical two years. Support for the last Asterisk 18 LTS branch will last until October 2025, and the Asterisk branches will last until October 2023. In the preparation of LTS-releases, the focus is on ensuring stability and optimizing performance, the priority of conventional releases are to increase functionality.

<https://github.com/asterisk/asterisk/>

RELEASE OF UBUNTU 22.10:

20/10/2022

On the day of the project's eighteenth anniversary, the Ubuntu 22.10 "Kinetic Kudu" distribution is available, which is attributed to the interim releases, for which updates are available only for 9 months (support will be carried out until July 2023). Installation images are designed for Ubuntu, Ubuntu Server, Lubuntu, Kubuntu, Ubuntu Mate, Ubuntu Budgie, Ubuntu Studio, Xubuntu, UbuntuKylin (edition for China) and Ubuntu Unity.

<https://ubuntu.com/download/desktop>

RELEASE OF OPENBSD 7.2:

21/10/2022

OpenBSD 7.2 is out. The OpenBSD project was founded by Theo de Raadt in 1995 after a conflict with NetBSD developers, which resulted in the closure of Teo's NetBSD repository. After that, Tao de Raadt and a group of like-minded people created a new open operating system based on the NetBSD source tree, the main goals

of which were portability (supported by 13 hardware platforms), standardization, correct operation, proactive security and integrated cryptographic tools. The size of the full installation ISO-image of the base system OpenBSD 7.2 is 556 MB.

In addition to the operating system, the OpenBSD project is known for its components, which have been distributed in other systems and have proven themselves as one of the most secure and high-quality solutions. Among them: LibreSSL (OpenSSL fork), OpenSSH, packet filter PF, pharmacological routing demons OpenBGPD and OpenOSPF, NTP server OpenNTPD, mail server OpenSMTPD, multiplexer of the text terminal (similar to GNU screen) tmux, OpenRSYNC file synchronization utility, etc.

<https://www.mail-archive.com/announce@openbsd.org/msg00449.html>

FULL CIRCLE WEEKLY NEWS



Join our host Moss Bliss as he presents you with a short podcast (<10min) with just the news. No chit-chat. No time wasting. Just the latest FOSS/Linux/Ubuntu news.

RSS: <http://fullcirclemagazine.org/feed/podcast>



THE

VIRTUALBOX NETWORKING PRIMER

Connecting and Configuring
Virtual Machines



Robin Catling

The VirtualBox Networking Primer

Connecting and Configuring Virtual Machines

The VirtualBox Networking Primer is a no-nonsense guide for the VirtualBox user taking their next steps into virtual networks.

While Oracle VM VirtualBox is a great free tool, the real power of virtualisation comes when you start connecting virtual machines to each other and to the wider world. Software development, sales, education and training are just some of the areas in which network access to virtual machines offers endless opportunities. But the world of computer networks is filled with complex technical jargon.

Complete with principles, practice, examples and glossary, The VirtualBox Networking Primer takes the frustration and confusion out of connecting real-world projects.

Author: **Robin Catling**

Publisher: Proactivity Press

ISBN13 : 9781916119482

Amazon US link:

[https://www.amazon.com/dp/1916119484?
ref=pe_3052080_397514860](https://www.amazon.com/dp/1916119484?ref=pe_3052080_397514860)

Amazon UK link:

[https://www.amazon.co.uk/VirtualBox-Networking-Primer-Connecting-Configuring/dp/1916119484/ref=sr_1_1?
dchild=1&keywords=virtualbox+networking+primer&qid=1600253699&s=books&sr=1-1](https://www.amazon.co.uk/VirtualBox-Networking-Primer-Connecting-Configuring/dp/1916119484/ref=sr_1_1?dchild=1&keywords=virtualbox+networking+primer&qid=1600253699&s=books&sr=1-1)

Kobo:

<https://www.kobo.com/us/en/ebook/the-virtualbox-networking-primer>



COMMAND & CONQUER

Written by Erik

For the newbies out there, I want to say, reach out and touch the keyboard. In modern Linux Distributions like Ubuntu, the focus is a lot on the DE. That being said, it is nice to have the mouse-driven menus at your fingertips, but no matter how comfortable you are with your mouse, there will always be something that is not covered. We will start slow and keep the pace easy, so anyone, grandma included, can follow along and not be afraid of the computer. Windows has taught us that you can break the system. While it is possible in Linux, it is a lot more resilient. Now this month's 'command n conquer' is not going to assume anything about the desktop you are using. I will try to write in broad terms, using words like "file manager". This does not – in any way – mean "files" aka Nautilus only; this also goes for Thunar, Rox, whatever.

To start, let us do just that, open your file manager. With the file manager having focus (what I mean by this is it is on top of other windows and the title bar is in the foreground color, if you have a title

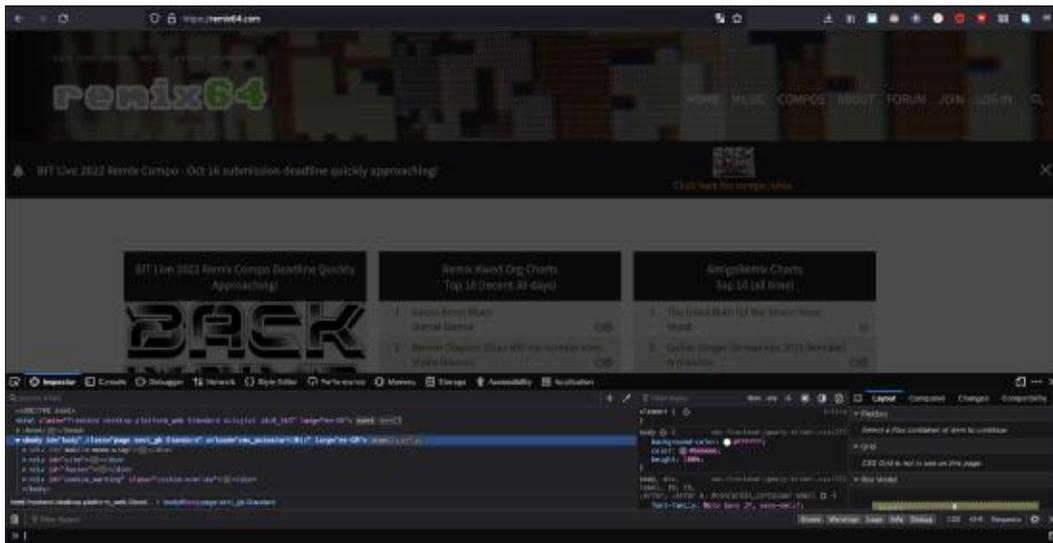
bar), I want you to press F1 and see what happens. Now that you know, I want you to press F2 and see what happens, all the while keeping the file manager in focus. The reason I say this is that, with some desktops, when you close a window that popped up, like the help that popped up when you pressed F1, will not automatically go back to the previous window that had focus, but some will focus on the start button, for instance. I want you to repeat the exercise with each F-key until you reach F12, and make a mental note of what just happened. For instance, some file managers may split in two by pressing F3, some may not. Feel free to do it multiple times to see what changes. I had a discussion with someone on Telegram who wanted eye-candy only. That is fine too, and I understand that "fake it till you make it" has become a very big part of life, but it can be fun too, to dazzle your peers with your fancy finger work. A lot of the time, the keyboard is going to be faster than the mouse, but, other times, the mouse will be faster at a task. I don't want to push you in a

direction, but I want you to know your system and have fun. As I mentioned before, some file managers will split your view in two, so you can get away without installing an extra two-pane file manager like 'midnight commander' or something similar. Though midnight commander is a godsend when you are working in the terminal.

So you are happy you know your file manager a bit better now... or you learned something new (kudos to you!). Now, at the same pace as before, I want you to open your favourite browser and repeat the exercise above. If you have multiple browsers, I want you to take one from another family and do it again. What I mean by this is some browsers, like Chromium, Brave, Vivaldi, are loosely based on Chrome, while Waterfox, Pale Moon and the TOR browser are based on Firefox. Very broadly speaking, they will not all conform to Chrome standard keys, for instance F3 on one will open a search whilst in others it will have no effect. We have not touched on any of the

other keys (modifiers) but you can already see how simple F-keys (which some people use so rarely that they have keyboards without F-keys!), can be handy as all heck. Now that we are in your browser, you may notice that most of them support F12 as "inspect". This is where YOU can make changes to the website you are using. This is more of an advanced topic we can cover another time, but I want you to see the power in this. I want you to feel like the genie in Disney's Aladdin. "unlimited power at your fingertips!!" :)

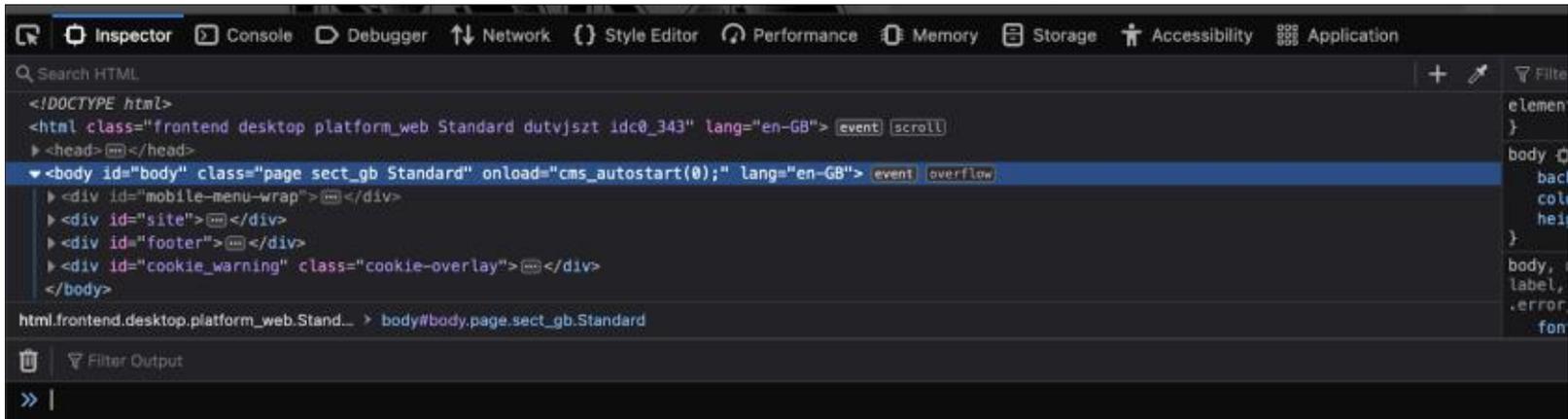
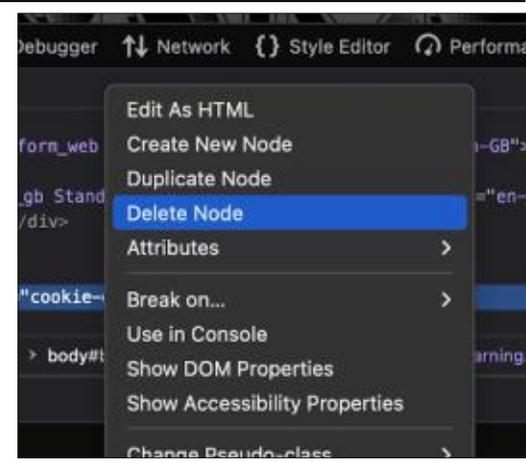
Let's all navigate to remix64 - <https://remix64.com/>, and you should immediately see a dark grey "film" being pulled over your browser window and a message that you HAVE to accept cookies. Annoying as all heck, right? Now, before we go further, some advanced users may point out there are add-ons for the browser that will do this for you, and there are, but the object here is to learn. Pop that F12 (shown next page, top left). In the following images, I will be using Firefox, but the method is



the same for most other browsers.

Let's zoom in on the "inspector" (shown below).

Can you spot the line that says, 'cookie overlay'? I want you to click on it once, to highlight it, then, right-click on it. You should see the following menu:



Now click on 'Delete Node'.

Don't worry, you won't break anything if you slip up and delete the wrong line, you can simply reload the page with the F-key you found when you tried it earlier.

Once the line is deleted, simply press F12 again and the page will be rid of the dirty film that was overlaid on it. Do you feel powerful yet? I'm glad if this was an eye opener for you. Your computer is your own, and should be more than a spotify station.

The takeaway here is that YOU can do things. You don't have to rely on others to do things for you. I mean, the cookie add-on I was using was causing rekt.network not to play (who needs Spotify?), so instead I disabled it. This in turn

caused cookie popups on sites I was browsing, so I took out my spanner and 'fixed' the problem in one simple stroke.

I could make a clickbait title, like "cookie companies hate him, see how he fixes overlays with one simple trick" and slap it on the web, but the more you know....

Any corrections? Contact us at: misc@fullcirclemagazine.org



Erik has been in IT for 30+ years. He has seen technology come and go. From repairing washing machine sized hard drives with multimeters and oscilloscopes, laying cable, to scaling 3G towers, he's done it.



A few months ago, one of the Python newsfeed aggregator services had a short blurb about a new project named Flet. For some reason, it caught my eye and I decided to bite the bullet and take a look. Their main website (<https://flet.dev/>) has the following headline...

"The fastest way to build Flutter apps in Python. Flet enables developers to easily build realtime web, mobile and desktop apps in Python. No frontend experience required."

I've always been interested in cross-platform programming like Desktop to Android or IOS. You might remember many years ago I did an article on Kivy (FCM #63, 64 and 65 July, August and September, 2012) which allows Python programmers to create applications for IOS, Android, Mac, Linux, Windows and Kiosks. So it seemed like a logical thing to look at.

I wasn't really familiar with Flutter, so I took a look at that before I went much further. A simple web search provided their

main page (<https://flutter.dev/>) with the main headline of "Build apps for any screen" – which intrigued me – and I kept scrolling. I finally found what I was looking for, a paragraph that said...

"Flutter is an open source framework by Google for building beautiful, natively compiled, multi-platform applications from a single codebase."

So Flutter is a frontend framework for the Dart language and Dart was developed by Google. I don't need to learn another programming language at this point in my life, but still, I was getting excited by the possibility that Flet might be a good extension for my Python programming.

I returned to the Flet website and started to explore.

INSTALLATION

Flet is a Python library, so we'll use pip to install it.

```
pip3 install flet --upgrade
```

Once Flet is installed, we can

```
import flet
from flet import Page

def main(page: Page):
    #add or update controls on the Page
    pass

flet.app(target=main)
```

give it a test.

USING FLET

According to the website, the basic structure of a flet app is as shown above.

When you break this basic program structure down, the function main() is the entry point for the Flet app and the Page is the "canvas" that holds the controls (or widgets to Tkinter programmers). In the sample above, the application opens in a native OS window, which is the default, but if you want to open the app in a

browser window, you can modify the last line as follows:

```
flet.app(target=main,
view=flet.WEB_BROWSER)
```

FIRST FLET APPLICATION

Let's create a REALLY simple flet app just to see how it all ties together. Use your normal IDE or text editor to create a new file. Call it FletTest1.py (shown below).

To run the app, simply type:

```
$ python FletTest1.py
```

You should see something like

```
import flet
from flet import Page, Text

def main(page: Page):
    page.add(Text(value="Hello, Full Circle Magazine!"))

flet.app(target=main)
```

this:



Please notice that I've resized the app to take the window image, since the default shows up very large on my screen.

Now the really important line in this small demo is:

```
page.add(Text(value="Hello, Full Circle Magazine!"))
```

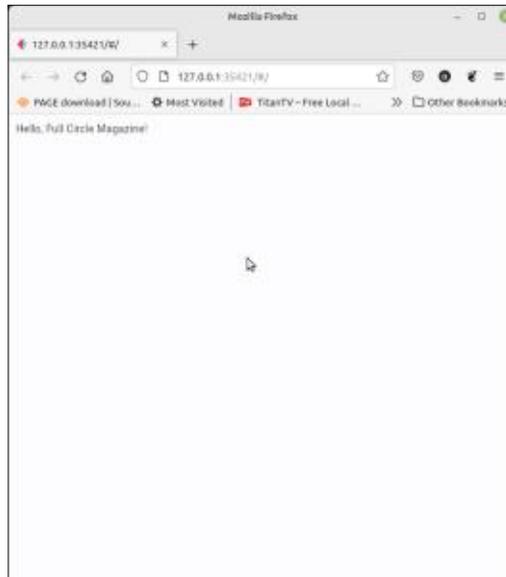
Which is the only line within the entry point function named main. You can see that this line adds a Text control and we set the parameter value to "Hello, Full Circle Magazine!". The Text control seems to be one of the simplest controls. It has a number of attributes to customize that instance of the control, like size, color, background color, font, and much more. Also, you might notice that this simple app doesn't have a title. You can add one within the main function by simply adding the line:

```
page.title = "Flet Test #1"
```

Now, let's try modifying the app to run in our default Web browser. Simply change the last line in the program to:

```
flet.app(target=main, view=flet.WEB_BROWSER)
```

And re-running the application.



Again, I've resized the app before I took the screenshot.

Now, I'm going to show a more complex application, a To Do app, from their tutorial. You can find the complete code at <https://github.com/flet-dev/examples/tree/main/python/tutorials/todo>, but I

suggest that you follow the complete tutorial, which is at <https://flet.dev/docs/tutorials/python-todo>. They break it down into logical steps as they show how the program is created. I'm going to whet your appetite by showing you only a few snippets of the full code.

We'll start with the import section of the code.

```
import flet
from flet import (
    Checkbox,
    Column,
    FloatingActionButton,
    IconButton,
    OutlinedButton,
    Page,
    Row,
    Tab,
    Tabs,
    Text,
    TextField,
    UserControl,
    colors,
    icons,
)
```

You can see that the app

```
def main(page: Page):
    page.title = "ToDo App"
    page.horizontal_alignment = "center"
    page.scroll = "adaptive"
    page.update()

    # create application instance
    app = TodoApp()

    # add application's root control to the page
    page.add(app)
```

contains a number of controls that get combined to create various custom controls, which is one of the big draws of Flutter. By using Flet, you don't have to learn Dart to utilize the power of Flutter and its control set.

Next, I'll show you the main function. Remember, this is the entry point of the app (shown below).

So you can see that the main function creates an instance of a class called TodoApp which is the workhorse of the application. Of course, there is much more to the class than I present here (next page, top right).

When you look at the class definition line, you will see that it uses UserControl, which is a reusable UI component. They don't go into a lot of explanation on this,

HOWTO - PYTHON

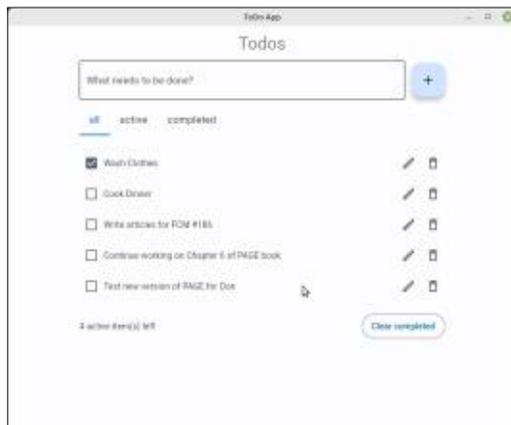
unfortunately, but it is an important concept that you will get the general idea from building the project (see bottom right).

As you look through the rest of this portion of the class, you can see the beginnings of the layering of the various controls to create a column container which shows its children in a vertical array (middle right).

You can also tell that the `TodoApp` class uses a `UserControl` which, in this case, is part of the `Task` class (next page, top right).

There is a lot more for the `Task` class, but I won't spoil the tutorial by showing you all of that class's code either.

So when the code is all created, here is what the app looks like.



```
class TodoApp(UserControl):
    def build(self):
        self.new_task = TextField(hint_text="What needs to be done?", expand=True)
        self.tasks = Column()

        self.filter = Tabs(
            selected_index=0,
            on_change=self.tabs_changed,
            tabs=[Tab(text="all"), Tab(text="active"), Tab(text="completed")],
        )
```

```
        Column(
            spacing=25,
            controls=[
                self.filter,
                self.tasks,
                Row(
                    alignment="spaceBetween",
                    vertical_alignment="center",
                    controls=[
                        self.items_left,
                        OutlinedButton(
                            text="Clear completed", on_click=self.clear_clicked
                        ),
                    ],
                ),
            ],
        ),
    ],
),
```

```
        self.items_left = Text("0 items left")
# application's root control (i.e. "view") containing all other controls
return Column(
    width=600,
    controls=[
        Row([Text(value="Todos", style="headlineMedium")], alignment="center"),
        Row(
            controls=[
                self.new_task,
                FloatingActionButton(icon=icons.ADD, on_click=self.add_clicked),
            ],
        ),
    ],
)
```

HOWTO - PYTHON

The bottom line, if you want to start to create portable applications between Web apps, Mac, Windows and Linux, a Portable Web app or (according to their website) via the Flet app for IOS and Android, you really should consider giving Flet a try. It would be a good idea to do some research into Flutter, since that's the backbone of the whole thing. While their documentation is somewhat sparse on exactly how to build very complex applications, you can get up to speed fairly quickly – enough that you can start giving Flet a chance to create apps for just about any platform. Looking at their Roadmap page, there are some tasks that are a bit behind, but the entire schedule is very aggressive so that's somewhat understandable. Learning Flet right now, wouldn't be a bad idea, getting ready for the future.

Until next time, as always; stay safe, healthy, positive and creative!



Greg Walters is a retired programmer living in Central Texas, USA. He has been a programmer since 1972 and in his spare time, he is an author, amateur photographer, luthier, fair musician and a pretty darn good cook. He still is the owner of RainyDaySolutions a consulting company and he spends most of his time writing articles for FCM and tutorials. His website is www.thedesignatedgeek.xyz.



HOW-TO

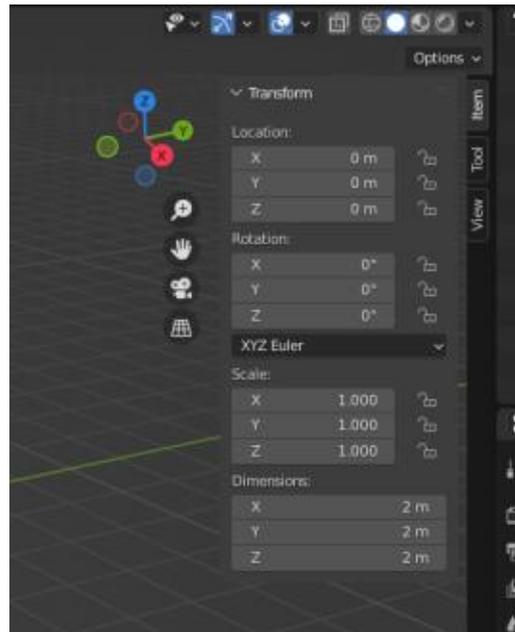
Written by Erik

Blender Pt.9

Fire up blender and let's begin. (with my PC out of action, I will be using version 3,0 on my Solus OS laptop. As long as your version is not older than 2.8, you should be able to follow along 100%)

I will TRY to answer all your questions in this issue, as I think it is important not to just blunder ahead but understand what and why you are doing it. I must stress that our egg shaped penguin or character project is a basic one. I work with metadata all day and only touch blender when I have a chance. It gives me joy when I do and I want to pass that on. For those who reached out and said I was going too fast, I'm sorry you feel that way, but I just want you to have something to say is your own. We can delve deeper into sculpting at another stage, those that asked I'll dedicate a few issues to it, though I am by no means a master. The few of you that asked for more explanation, we will get there, as I said, I just want you to get something out of the door as hands-on experience is king.

Right, thanks to Tom, I will point out the "n" key. When you click within the viewport and press "N" a menu will slide out from the left. This menu corresponds to the place your object is, in the "world" relative to the centre.



You will see an X, Y and Z in Location. If you move your mouse over any of the bars, you can press the primary mouse button and slide it left to right and as the values change, so will the position of your object, and if you don't have an object yet, it will move the default

cube. You may also note the locks beside the sliders (that don't look like sliders) that will lock the object in place on that axis. Then there are the tabs down the right side and this is where things may look a little different, depending on installed add-ons or vanilla blender. I will keep everything vanilla as I explain.

Jaydeen & Co, we can cover the settings menu at the end, I do not want to confuse people now. You CAN use blender at stock defaults. I understand there are ways of setting up blender to speed up your work flow, but I really just want to get newbies to make something, as crooked and broken as it may be. Once they find their feet, they can get comfortable. I will address the "industry compatible" toggle you guys are on about. As a newbie to Ubuntu and possibly Blender, you probably have never worked with other drawing/sculpting packages, and even if you have, is it really a necessity for grandpa Sebs who retired long ago? Keeping the keys standard, means everyone who has a go at it, will get the same result. See key above.

To answer Miroslav's question, on how to change the value of the vertices?(smoothness) once the sphere is moved - there is no way to change the smoothness of our ball once created and moved, you need to delete the object with the 'X' key and recreate it and play with the sliders again as far as I know, I did try to find something in the documentation, but I could not, so let's assume my statement is fact. This is the reason I said to play with it when you were creating, it is literally the first step you do, so deleting and recreating is not too much time lost.

As to the questions about plug-in's, I really cannot comment as I am not a professional user making money from Blender, so I would not spend lots of money on them. Places like Gumroad list their pricing in USD and I need to pay 20x more, so I just don't. I enjoy using blender as is and I do realise they can make the quality of life a lot easier, I am a simple home user who has other priorities with my hard-earned cash. It was only when

HOWTO - BLENDER

looking up the plug-in's mentioned, and the messages on Telegram that I found out you can put in \$0 for some of them. Though I will look into it in the future, we won't need any to continue making, this series is for Joe Bloggs at home wanting to play some.

For those of you who are now wondering about plug-in's, you can get a look at the plug-in's that Blender ships with, (Yes, it comes with quite a few) by simply clicking on the menu, Edit -> Preferences -> Add-ons. To install any of them, simply click the tick boxes of those you wish to install and click on the install button in the top right. What you need to be aware of is the icons on the right, they indicate Blender foundation plug-in's or community plug-in's. The other icon you need to keep an eye open for, is the warning triangle. To see what the warning is, you need to click the expand point on the left of the name and you will see. Usually those found in Blender by default are not destructive, but beware of ones you find on shady websites, the warnings you will most likely encounter will be software that is still in development, but do think twice.

This tip from Roland, I would like to pass on; When you deselect an object, like an eye that is embedded in another object, like a body, when you have multiple objects selected, the SHIFT+double click, will start at the object that is the furthest away. In other words, it will deselect the body behind the eye, instead of the eye. To get away from this behaviour, you need to rotate your object so you cannot see anything behind it and then SHIFT+double click.

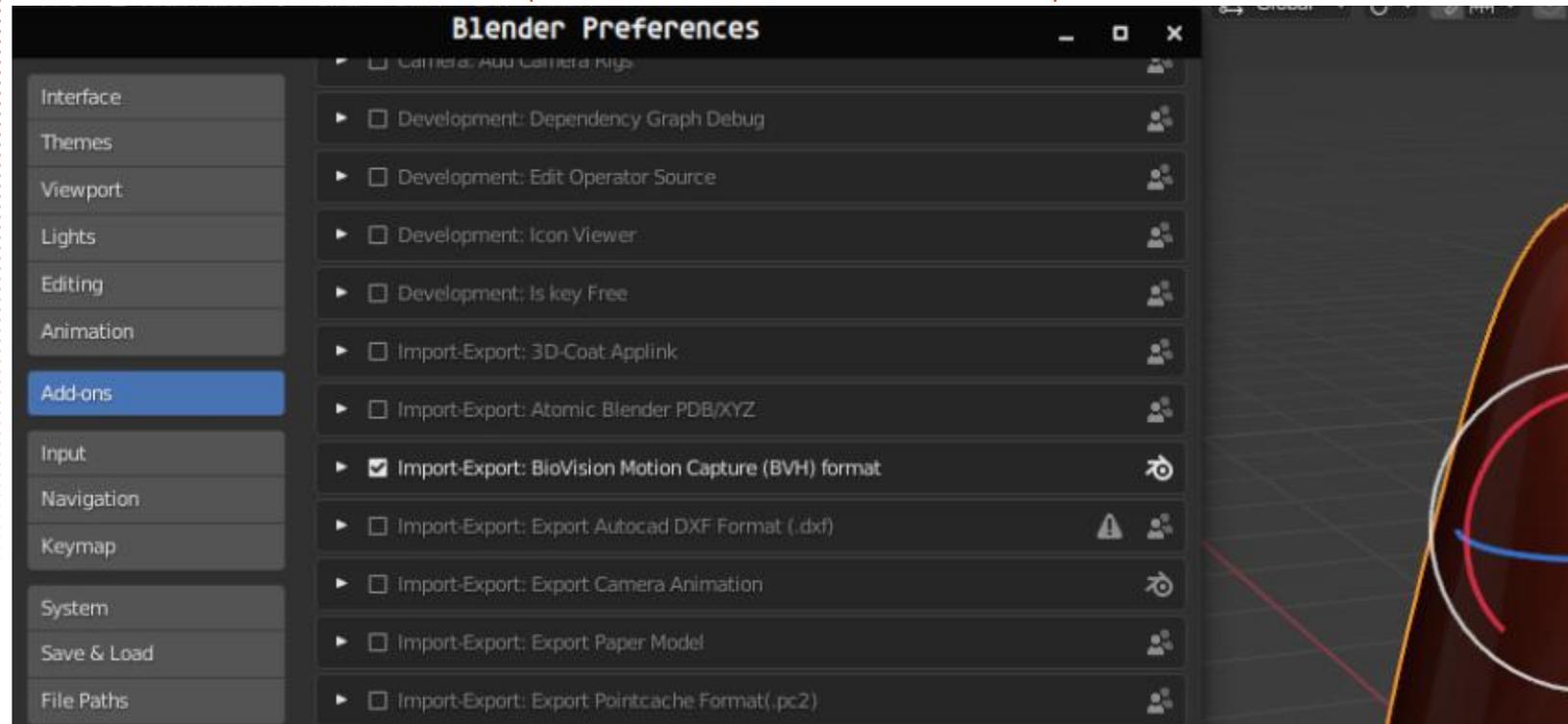
As I am writing this I am nowhere near home, so I cannot

confirm, but I really thought I explained it. The difference between the X,Y,Z tool at the top right we talked about and the "Rotate" tool is that the X,Y,Z tool in the top right, rotates the "camera" or your viewport, if that makes it easier for you. The Rotate tool, see when an object is selected, and you press 'R' or click on the rotate button on the left, rotates the object within your viewport. Thank you for that Terry. Please, if this is confusing for you guys, play with it. Mangle a cube and rotate it with the different options.

If more of you feel I left something out, or I did not explain something clearly, feel free to reach out to me on Telegram, or drop us a line at: misc@fullcirclemagazine.org.



Erik has been in IT for 30+ years. He has seen technology come and go. From repairing washing machine sized hard drives with multimeters and oscilloscopes, laying cable, to scaling 3G towers, he's done it.





HOW-TO

Written by Robert Boardman

The use of text in columns is fairly common. You are not likely to use columns when writing an essay for school, or in a resume, or writing a story. Newspapers, magazines (like FCM), journals, business reports, all set type in columns in which the text flows down the page, then up to the next column, then down again, and on and on. As is usual in Latex, there are different ways to accomplish the task, the current task being to set text in columns. For small amounts of text, it is possible to use a table. (Formatting tables was reviewed in Full Circle Magazine #182.) For longer sections of text, or to avoid using tables, there are several tools available to format columns. If you search on CTAN for “multicol”, you will find more than one dozen possible choices.

As is true for many features in Tex / Latex, there is little or no information to tell a new user how to include a new feature in their installation and how to use it. I checked information for the following environments: balance, balanced, cuted, doublecol,

multicol, threecol, twocolumns, vwc. Only multicol and vwc had README files (which are identical, and generic, and do not provide any instructions for installation or use). Cuted, multicol and vwc had informative documentation files.

Here is an explanation of cuted: Standard LATEX will switch between `\onecolumn` and `\twocolumn` only at the top of a page; the commands themselves clear the previous page. This package does away with the restriction, and allows you to mix one- and two-column modes on the same page. The PDF documentation is less than two pages, and explains what the options do and how to turn them on.

The documentation for vwc is 24 pages. It sets out the purpose of the environment clearly at the beginning. Vwc is designed for paragraphs of text to be in multiple columns of various widths. The columns cannot span multiple pages. The environment is not designed for anything but text.

```
\documentclass{article}
\usepackage{vwc}
\usepackage{lipsum}
\vwcsetup{widths={0.25,0.25,0.5},rule=1pt}
\begin{document}
\begin{vwc}
  \lipsum[1]
  \lipsum[2]
  \lipsum[3]
  \lipsum[4]
\end{vwc}
\lipsum[5]
\end{document}
```

The documentation for multicol is 39 pages, a lot to read if you are only trying to set up the layout of a document that is overdue. The environment multicol allows for single-column and multi-column layouts on the same page.



I am going to experiment only with multicol and vwc in this article. I will use the lipsum package

to generate fake text, as usual. I will experiment to find out if tables and images can be included in columns. I will start with vwc since it was developed first (see code above).

I hope the code is reasonably self-explanatory. The option “widths” sets up the number and size of the columns. In this example I used decimals, per-cents are also acceptable, as are physical units like inch and cm. If Latex calculates the total of the column widths plus the spacing between columns exceeds the margin widths, the file will not compile. The “rule=1pt” tells Latex to put a line 1 point wide between the columns.

Vwc is restricted to putting

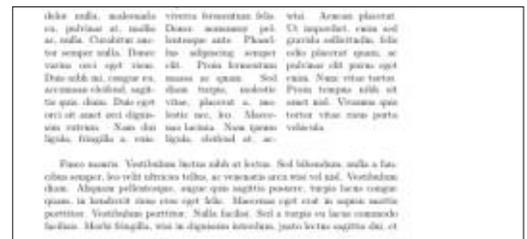
```

\documentclass{article}
\usepackage{multicol}
\usepackage{lipsum}
\begin{document}
\begin{multicols}{3}
  \lipsum[1]
  \lipsum[2]
  \lipsum[3]
  \lipsum[4]
\end{multicols}
\lipsum[5]
\end{document}

```

columns on one page. When I added lipsum[5] inside the vwcol environment, all of the columns shifted to the next page of the document and text filled to the physical bottom of the page, there was text in the bottom margin area. This is clearly an error.

If you are writing or laying out a newsletter or magazine you want to be able to have articles that flow from one page to the next. Vwcol is not adequate for this work.



I will next experiment with the multicol environment using similar code to what I used with vwcol (top left).

Notice the name of the package is multicol but, to begin and end the environment, an "s" is added to the end of the package name. I am sure you can see that the three columns in the multicol example are equal width and that the column separator rule is missing. The column separator rule can be inserted using `\setlength{\columnseprule}{2pt}`. If the document uses the package color then the colour of the column rule can also be changed:

```

\def\columnseprulecolor{\color{red}}

```



If I add a few more paragraphs I

```

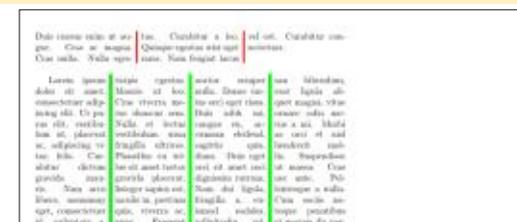
\documentclass{article}
\usepackage{multicol}
\usepackage{lipsum}
\usepackage{color}
\begin{document}
\begin{multicols}{3}
  \setlength{\columnseprule}{2pt}
  \def\columnseprulecolor{\color{red}}
  \lipsum[1]
  \lipsum[2]
  \lipsum[3]
  \lipsum[4]
  \lipsum[5]
\end{multicols}
\lipsum[6]
\end{document}

```

can make the columns go onto the second page. I have also added a paragraph outside the multicols environment in a single column (see code above).



I can introduce another multicols environment with either the same or a different number of columns.



The ability to switch from multiple columns to a single column allows for images and tables and other items to go from left margin to right margin without interfering with the columns of text.

The multicol environment has a great deal of flexibility. It should satisfy the needs of most writers and editors who need multi-column layouts. I strongly recommend reading the accompanying PDF file and experimenting with various options before attempting to use the multicol environment in production.



HOW-TO

Written by Mark Crutch

Inkscape - Part 126

Last month, I looked at the Web > Interactive Mockup extension using a simple website mock-up. I demonstrated how you can use this to create a demo that works well on a specific screen size. But even though SVG files are scalable by their very nature, this extension doesn't do a great job of producing demos that will work across a variety of devices. This is a limitation we'll try to address this month, following on with the same file I created last time, so make sure you've read last month's column before pressing on with this one.

First, I need to manage your expectations. Although we'll be trying to make our presentation work better on different screen sizes, there is a limit to what can be done with this simple extension or the small amount of code we'll be writing. All we're aiming to do is to make the output scalable – we're not going to be creating a fully responsive mock-up where parts of the design move around or change appearance in order to suit mobile, tablet, and PC displays from a single file. Think of it instead as

creating a file that will work on both a laptop and desktop PC, even if the screens are slightly different sizes.

Let's begin by exposing the problem we have. Last month we created a mock-up that was specifically sized at 1920px x 1080px – the dimensions of a high-definition screen. To display at that size we loaded the file into a web browser and pressed the F11 key in order to remove the browser UI and render the content full-screen. But let's see what the same file looks like before the F11 button is

pressed, when the available resolution for the SVG content is smaller.

Vertically we can no longer see the bottom of the page. The available space to draw the image has been reduced by the browser's UI, by the window's title bar, and by the top and bottom panels of my Mate desktop environment.

Horizontally all the content fits, because the browser isn't showing any scrollbars and the window theme has thin borders, so almost the whole width of the monitor is

available when drawing the content. If I were to resize the window, however, or open the developer tools to one side, the right-hand side of the image would be cut off.

To make it easier to experiment with different screen sizes, we're going to use the browser's developer tools to simulate the page's behaviour across a range of screen sizes. These instructions are for Firefox, but all modern browsers have similar tools available, and even the keyboard shortcuts tend to be common between them.

- Open the Developer Tools via the menu, or by pressing F12.
- Switch to Responsive Design Mode using the button at the top of the developer tools (it looks like a couple of rectangles, representing a phone and a tablet), or by pressing Ctrl-Shift-M.
- The screen content should switch to some sort of mobile view.

You can use the controls at the top to select specific devices to emulate, and to switch between





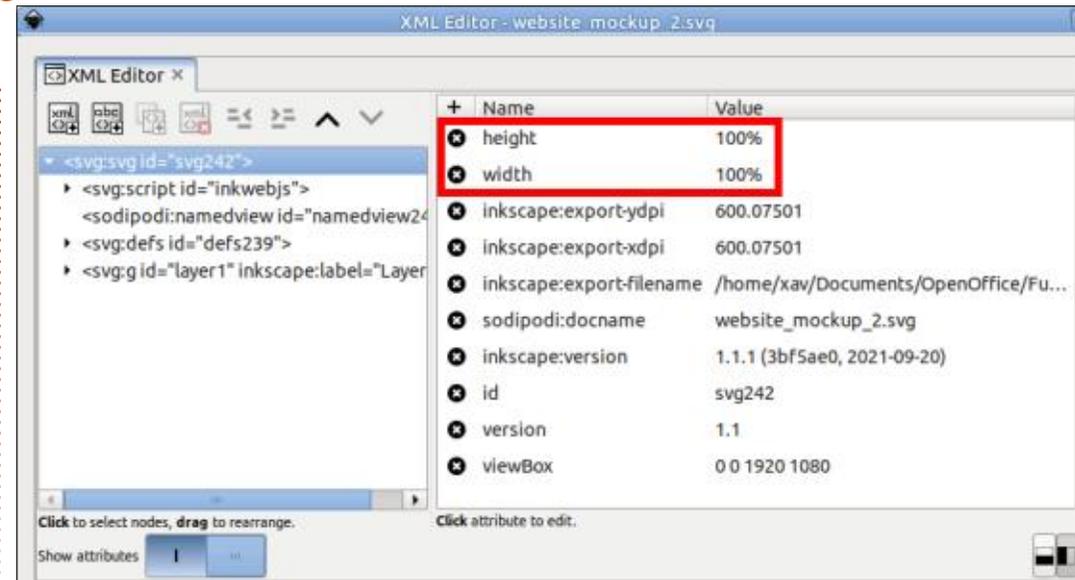
landscape and portrait mode. For our purposes, however, just drag the handle at the bottom-right corner of the screen around to quickly try the page in various sizes and aspect ratios. On Chrome and Chromium, the handle may not be visible: in that case you first need to select “Responsive” from the pop-up menu at the top of the page area.

Now it should be pretty clear that the content of the file is not scaling to suit the window, and is being cut off along the right or the bottom (or both) depending on the size of the content area.

The reason for this is pretty straightforward. Inkscape requires us to specify a page size when we create a document, and in this case we set it to our target size of

1920px × 1080px. When loading the image into a web browser, however, it’s trying to honour those dimensions still, meaning that if the available document size isn’t large enough, the excess will just get cut off. What we need is to replace the width and height in the file with something more dynamic – something like “100%” or “auto”. Or even to remove those fixed values altogether and just let the browser do the right thing. Unfortunately none of these approaches work within Inkscape’s Document Properties dialog.

Although “%” appears as a unit in many of Inkscape’s unit pickers, the size pop-up in the Document Properties dialog isn’t one of them. Don’t get fooled by the “pc” unit – that’s “picas” (1/8th of an inch). Trying to clear the Width or Height



fields, or typing “auto” into them, results in them being populated with a value of 0.00001 instead. Typing “100%”, with the units appended to the number, just gets converted to a value of 100, using whatever units are selected in the pop-up.

Although we can’t set the values we want via the Document Properties dialog, there are other options open to us. If you’re familiar with editing XML or HTML then you could load the SVG file into a text editor and remove or change the “width” and “height” attributes on the <svg> element directly. Alternatively you can achieve the same via Edit XML Editor, by selecting the top entry in

the left-hand pane (the <svg> element), and then editing the attributes in the other pane.

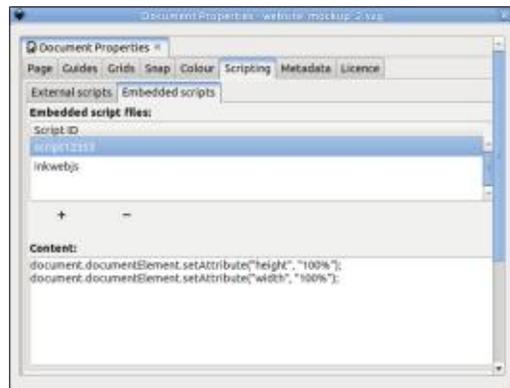
Both these approaches work, but they also have a drawback. If you have reason to go into the Document Properties dialog and edit other fields in there, you may find that these attributes return with their original values, leaving you playing a constant game of cat-and-mouse, having to remember to check or edit them just before saving each time you edit the file.

Instead I propose a more elegant solution. We know Inkscape wants some ‘real’ units to define its page size. But we also know that we want to change those units to

HOWTO - INKSCAPE

something more abstract when the page is viewed in a web browser. My solution, therefore, is to dynamically make that change when the image is loaded into the browser, by including a small snippet of JavaScript in the Inkscape file. Don't worry, it's not as tricky as it sounds...

- Open File → Document Properties.
- Switch to the "Scripting" tab.
- Within that, switch to the "Embedded scripts" tab.
- There will probably already be an entry for "inkwebjs" which you can ignore.
- Click on the "+" button to create a new embedded script. It will appear in the list with a random ID.
- Select the new entry in the list.
- Put the cursor into the "Content" field, then enter the lines of code below.



The code itself consists of two nearly identical lines. We need to

access the <svg> element first: since this is the main container for the whole file, it can be accessed as "document.documentElement" in JS (make sure you type in the right case). We then need to call the setAttribute() method, telling it the name of the attribute we wish to set or update ("width" or "height"), together with the new value it should have ("100%"). The complete code, therefore, looks like this:

```
document.documentElement.setAttribute("height", "100%");
```

```
document.documentElement.setAttribute("width", "100%");
```

Because that code isn't inside a function, it will automatically run when the page loads in the browser. Job done, and we don't need to keep remembering to adjust any values each time we save an edit to the file.

That solves one problem: our content now scales to suit the width and height of the browser, while still maintaining its original aspect ratio. But it also exposes another issue with the way this extension works. You may have already seen the problem when playing with the resize handle in the

responsive view: if the window size is tall enough, then we can also see some corners of the other pages in our mock-up (outlined in red in this image).

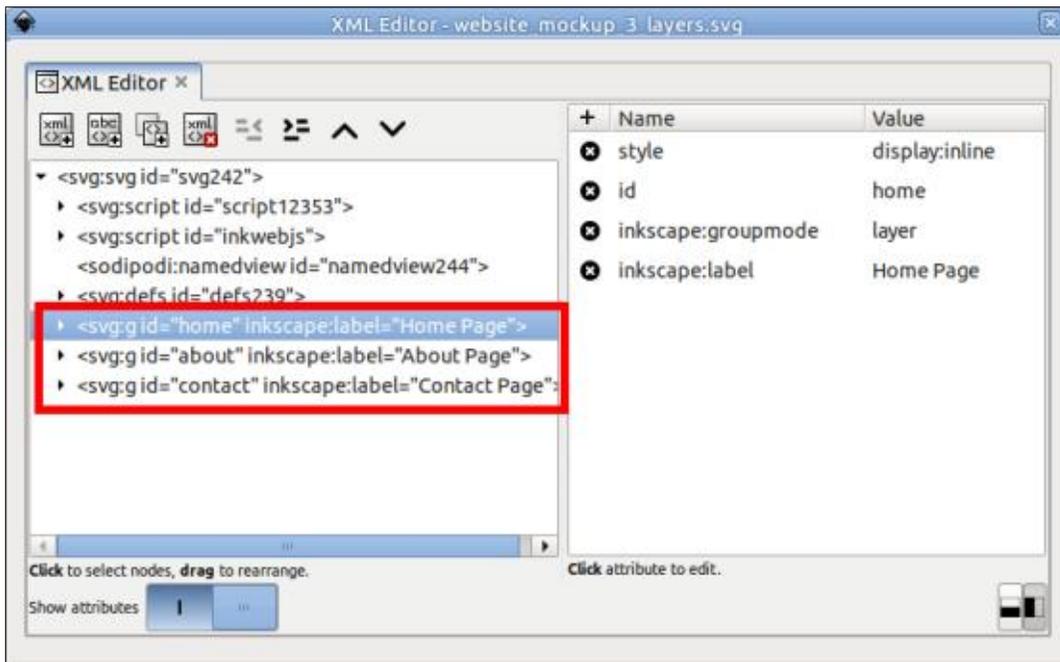


The reason for this is that our new width and height values tell the browser how to size the main content (the bit inside the viewBox), but the browser will happily render anything outside that area, if there's space on the screen to do so. We're simply seeing the parts of our document that fall outside the current viewBox region. There's an obvious and easy fix for this: just move the pages further apart in the original document. If they're way outside

the bounds of the viewBox then they're far less likely to appear on screen when they shouldn't.

This is, of course, something of a band-aid. Although it works in most practical cases, there will always be some extreme aspect ratio which is sufficient for the other pages to just creep in at the edge. For most situations it's probably good enough, but it would be better if we could have a solution that caters for all cases.

What we require is some more JavaScript that hides all the pages in our mock-up except the one we're currently viewing. This requires us to have some simple way to define what counts as a "page" – and it strikes me that simply putting each page onto a separate layer is the easiest way to do that. Yes, I know Inkscape 1.2 has multi-page support, but using layers will work for older releases as well. In the case of our demo file, we need three top-level layers, one for each page. These will all be direct children of the <svg> element. To make our code more readable, we'll change the IDs for the layers to "home", "about" and "contact" using the XML editor.



To simplify the code we have to write, we'll also move all the layers back onto the canvas, stacking them on top of each other. That means you'll have to turn them on and off in order to edit the content, but it also means our JS code only has to deal with the visibility of the layers, not with also moving the viewBox around.

The code itself is a little more

complex than the two lines we created earlier, though not by much. We'll want a function that accepts a layer ID, and which turns off all of the layers before turning on the one we've provided. We'll also need to call that function when the page loads to ensure the first layer is visible. Finally we'll call that function from each of the "interactive" elements in our mock-up, by attaching the function call to

```
function showLayer(id) {
  const layers = document.querySelectorAll("svg > g[*|groupmode=layer]");
  layers.forEach(layer => layer.style.display = "none");

  const layerToShow = document.querySelector("#" + id);
  layerToShow.style.display = "inline";
}
```

an onclick event or one of the other interactivity events we've seen before.

Let's begin by adding the function we need. Once again open the Document Properties dialog, select the "Scripting" tab, the "Embedded scripts" tab, and then the script we created earlier. Append a blank line or two, then add the following code (shown below - again, remember it's case-sensitive)

The first line in the body of this function uses the `querySelectorAll()` method to search the document for any elements that match the supplied CSS selector. The gobbledegook in quotes actually means "find any <g> elements that have an attribute called 'groupmode' (in any namespace), with a value of 'layer', but only if they're an immediate child of an <svg> element". This is good enough to pull out only our

top-level layers, without accidentally catching any sub-layers or other <g> elements. The second line then iterates over the collection we've just created, hiding each of them by setting their CSS "display" property to "none".

The last two lines are similar, but only deal with a single element. This time we use `querySelector()` (without the "All") to just find an element with an id that matches the one we've provided. In CSS terms an ID starts with a "#" character, so we'll add that to construct the selector, meaning we can just supply a page name, such as "home", rather than having to pass "#home". The last line sets the CSS display property for this one element back to "inline" in order to make it visible.

Now we need to add another line that will call this function to make one layer visible by default when we load the file into the browser. This needs to happen after a short delay (we're using 100ms) to give the page a chance to load and settle down before we start poking with it. Append a line like this at the end, after the closing curly brace, replacing "home" with the name of your own initial layer.

```
setTimeout ( () =>  
showLayer ("home"), 100);
```

To avoid any visible glitching, it's also worth ensuring that your default page is the topmost layer within Inkscape, and is not hidden.

Now all that remains is to add a call to this function to each of our interactive elements. We'll have to do this one-by-one, without the convenient shortcuts we had when using the extension for adding multiple items at once. That makes it a little more fiddly, but I think it's worth it for the better end result.

- Select an interactive element.
- Open the Object Properties dialog.
- Expand the "Interactivity" section, if necessary.
- Choose the field for the type of interactivity you want (usually "onclick").
- If there's already code in there from the Interactive Mockup extension, remove it (this will be a call to the `InkWeb.moveViewbox()` function). We don't actually need anything from this extension any more!
- Type in a call to the `showLayer()` function, using the ID of the layer that should be displayed when the

element is clicked – e.g.
`showLayer("home")`

- Repeat for each interactive element on each page. Hint: you don't need to close and reopen the dialog, it'll update as you select each element.

Save the file and load it into your browser. If you've done everything correctly you should now have an interactive mock-up that scales correctly for any screen size, and doesn't suffer from other pages peeking into view. All it took was a few lines of JavaScript, and a single function call added to each interactive element. And to clarify, this is a replacement for the Interactive Mockup extension, not an enhancement of it: you can add this JS to a new Inkscape file to create interactive mock-ups without ever going near the extension.

Using the extension is definitely simpler, especially if you're not comfortable with JavaScript. If you know you only have to target one specific screen size, then that's probably the approach for you. But if you need the flexibility of scaling to suit any screen size, or prefer to keep all your pages stacked in layers rather than distributed

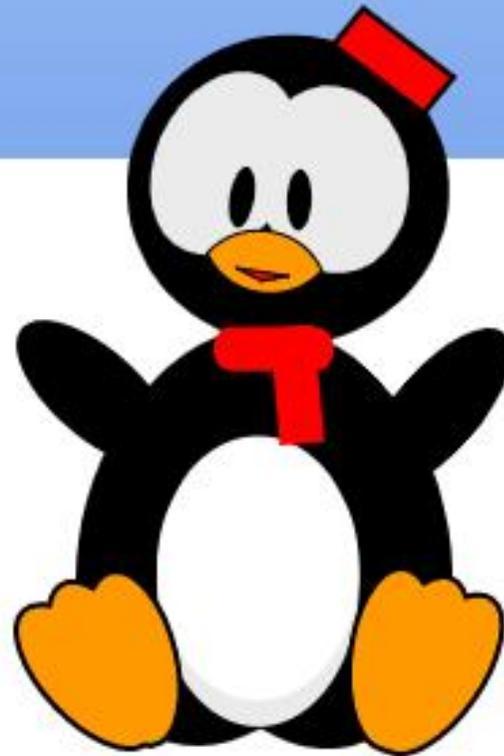
across the canvas, these few lines of JS may be just what you need to create an interactive demo that suits your needs.



Mark uses Inkscape to create comics for the web (www.peppertop.com/) as well as for print. You can follow him on Twitter for more comic and Inkscape content:
[@PeppertopComics](https://twitter.com/PeppertopComics)

The daily waddle

HEY WE WERE JOKING!!
YOUR PHONE IS IN MY
CAR...





In 1996, Bethesda Softworks published the second in their nascent Elder Scrolls series of fantasy role-playing computer games, Daggerfall (see FCM137's Everyday Ubuntu column for details on the game and how to run it in Linux). It came in a now highly-collectible gold holofoil box (I actually have TWO of those boxes), and was advertised as "Your Newest Obsession", which was a pretty apt claim. The world of Daggerfall, Tamriel, was stunning at the time, if for nothing else, for its sheer scope. As far as I know, it's still the largest computer game world ever created, with over 60,000 square miles and more than 15,000 towns to explore.

First person perspective gaming was still pretty new at the time, and probably fairly low resolution and cartoony looking to modern sensibilities, but it was (and still is) a very entertaining and engaging game. It was also very popular, so Bethesda soon announced a sequel, Elder Scrolls III: Morrowind. Due originally in 2001, Morrowind was repeatedly delayed and

rescheduled until it finally came out in 2003.

AN AMAZING UPGRADE

When Morrowind did finally come out, the wide consensus was that it was WELL worth the wait. The graphics were a phenomenal upgrade over the original, the gameplay was much improved, Jeremy Soule's soundtrack was fantastic, and the game soon became a much-lauded instant classic. The backgrounds and landscapes are particularly great, in my opinion, and hold up pretty well even today.

You start out your adventures in a seaside village called Seyda Neen, which has somewhat swampy forests shrouded in fog, pretty realistic looking ground, water, and flora, and human or humanoid figures that are a bit lacking in polygon count, but still look pretty okay in my humble opinion, especially for the time. The voice acting is also on point. It's actually surprisingly entertaining to simply walk around the town and its

environs and just soak up the atmosphere, checking out the fantastic plants and animals, looking at the memorable silt striders, and just virtually living in a compelling and believable fantasy world.

Morrowind had such lasting popularity that there are still active online communities modifying it to this day, particularly upgrading the graphics even more, in order to conform to modern standards. You can play a thoroughly modified Morrowind today and easily believe it's a fairly current game, rather than a 20-year old one. The game was popular enough to give birth to the also critically-acclaimed Oblivion and Skyrim chapters in the Elder Scrolls series.

GETTING STARTED WITH OPENMW

Fortunately, it's a pretty simple task to play Morrowind on a modern Linux computer, thanks to the existence of OpenMW. OpenMW is an open-source

Morrowind implementation that encourages continued modifications and upgrades to the existing Morrowind base. In order to play Morrowind, though, you will need to have the data files, as they are not public domain and cannot be included with OpenMW.

Fortunately, Morrowind is a very inexpensive purchase on GOG.COM, \$14.99 at the time of this column's publishing (it's actually on sale for only \$4.49 as I write, but that will be over long before you see this, dear readers). I have the fully digital GOG version of Morrowind installed on a Windows 10 computer, so I can get the data files needed for OpenMW from that location. Simply drag your Morrowind folder from your Windows PC hard drive to a portable drive (I used a thumb drive, but an external hard drive or an SD/MicroSD card would also work). Once you have the game folder transferred to a folder on your Linux computer's drive, you'll be ready to install OpenMW and import the Morrowind data.

INSTALLING OPENMW

Hit the key combination CTRL-ALT-T on your keyboard. That key combination will start a Terminal command line session. In Terminal, type:

```
sudo apt-get install openmw
```

and hit Enter. Answer any prompts appropriately and OpenMW will be installed.

LAUNCHING OPENMW

Now that OpenMW is installed, click the App Drawer (9 white boxes in a grid, bottom left of your screen).

Type OpenMW at the top of the screen in the search box. There will be two icons found. Click OpenMW Launcher and the application will launch:



IMPORTING THE GAME DATA

Click the Run Installation Wizard button, and the data import process will begin:



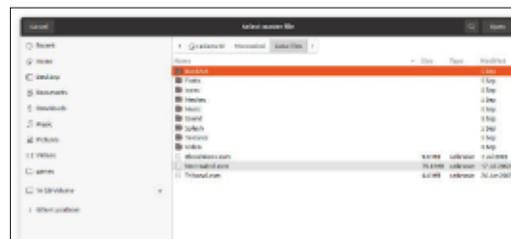
Click Next to continue:



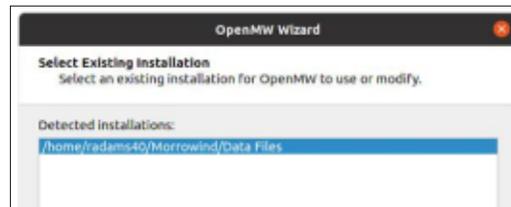
Click browse to find the folder where you placed the Morrowind files. Once you've found the folder, navigate into the Data Files subfolder:



Within the Data Files folder, you should see an .ESM file. Select the one you want to load (we'll start with the main Morrowind.esm file here):



Once you double-click the .ESM file, the wizard will continue:

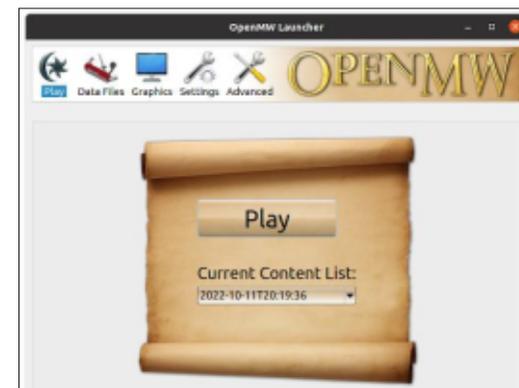


Click Next. We'll go with the default of English, with all due respect to our friends in non-English speaking countries (my French is nowhere near good enough for playing this game: Vraiment, je parle francais comme une vache espagnole!)

Click Next again. We'll continue with the default setting and click Next once more:



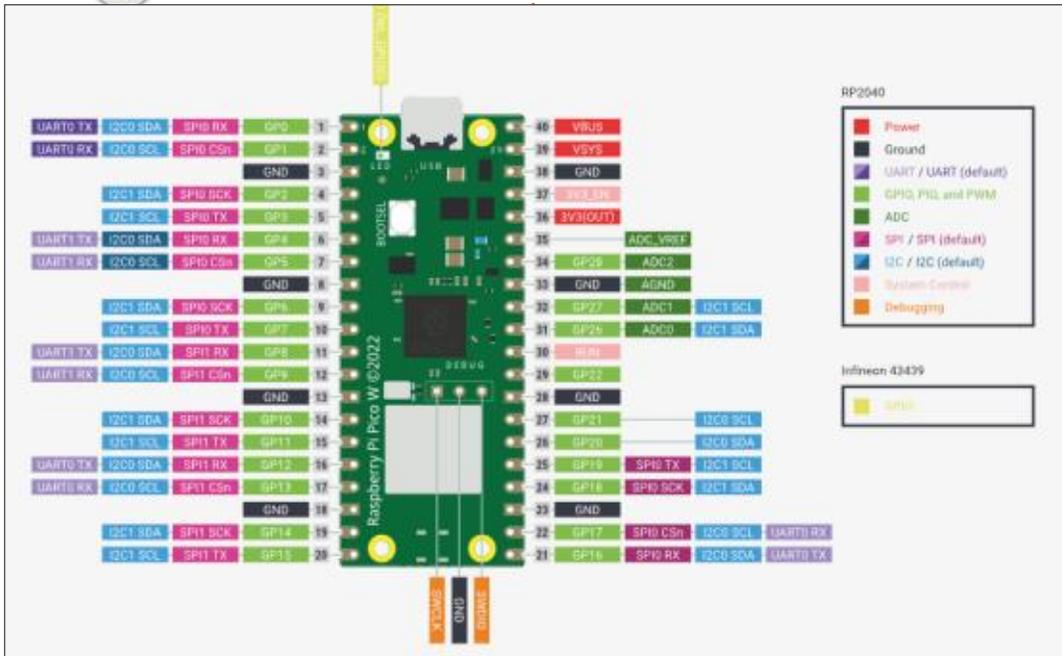
And we're done! Click Finish and we'll be transported to the very doorstep of our upcoming adventure in Tamriel!



Next month: Finally, running and playing Morrowind!



Richard 'Flash' Adams lives in rural north Alabama and has been a computer support technician, a business analyst, a software salesman, a sales analyst, a QC team lead, and is now disabled/retired. He enjoys reading, NFL football, computer and video games, cooking, and playing with Baby, his cockatiel. Feedback and suggestions are welcome at acer11kubuntu@gmail.com.



It's been a while since I've been able to write a 'Micro This Micro That' article. I apologize that it's been so long. Too many things have been in my way, but at least for this month, I can provide you with something.

I also know that I had promised a long time ago that we would complete the compass app and I still intend to do that, but so much has changed in the Microcontroller world, I felt that I needed to discuss some of the new things. With any

luck, the next 'Micro This Micro That' article will finish that.

For this month, we'll be discussing the new Raspberry Pi Pico board, the Pico-W. You can find out a good bit of information from <https://www.raspberrypi.com/documentation/microcontrollers/raspberry-pi-pico.html>. This webpage covers the entire RPi Pico family, the Original board, the Pico-H, and the Pico-W. We'll ignore the Pico-H for now, other than to say that the Pico-H is pretty much the

```
# i2cscan.py
# Searches the i2c bus for i2c devices

import machine
sda = machine.Pin(8)
scl = machine.Pin(9)
i2c=machine.I2C(0, sda=sda, scl=scl, freq=400000)
devices=i2c.scan()
if devices:
    for dev in devices:
        print(hex(dev), dev)
```

same board as the original Pico, but it has a three-pin keyed socket for the Serial wire debug connections. Here is the pinout for the Pico-W

You can see that the pinout is pretty much the same as the original Pico, so, for the most part, you can use the software you've already developed, but you need to get the latest MicroPython version that is designed to support the Pico-W. You can get that from the website above or from the MicroPython.org site. The process of flashing the board is the same as the original board, so I won't bore you with that.

One of the new things that the Pico-W brings to the table is that the onboard LED pin is not

connected to a GPIO pin on the board, but is connected to a GPIO pin on the wireless module (and the MicroPython version has been modified to handle it). So it makes it much easier to access it in your code. Here is a quick demo program that shows how to do it.

```
import machine
import time

led=machine.Pin("LED", machine.Pin.OUT)
led.on()
time.sleep(3)
led.off()
```

As you can see, it is very simple. We define the led object, then we simply call the led.on() and led.off

MICRO THIS MICRO THAT

methods to turn on and off the led. You can use this to provide a visual indication that you have a wireless connection.

Since the pinout of the Pico-W is pretty much the same as the original, we can use our original i2cscan program to probe the i2c bus and provide the address of all the i2c devices attached to our Pico-W (shown previous page, top right).

As I usually do, I connected the sda (data) line to physical pin 8 and the scl (clock) line to physical pin 9. When I ran the program I got the following output.

```
0x19 25
0x1e 30
0x40 64
```

Address 0x19 is the lsm303 accelerator and 0x1e is the lsm303 magnetometer. Address 0x40 is the SI7021 temperature/humidity sensor. (See, I AM going to finish the compass project!).

Since there are now three RPi Pico boards, how can you tell which device you are working with (other than looking at it)?

The micropython versions for the Pico boards now support a sys method named implementation (shown top right). Of course you have to import the sys module first. I threw together a simple program that shows how to access the information.

When you run the program, here's what it returns.

```
(name='micropython',
version=(1, 19, 1),
_machine='Raspberry Pi Pico W
with RP2040', _mpy=4102)
Raspberry Pi Pico W with
RP2040
Pico with Wireless
```

So, which1 returns a tuple that holds the fact that we are using Micropython, the version of Micropython installed, the type of machine, and a value for _mpy. The important part of this is the value located at position 2 of the tuple (remember this is zero based). Which1[2] returns a string which we can use the in operator of Python. So by checking for "Pico W" within the string, if we are running on a Pico-W, we will get back a True.

This whole thing can be distilled down to a simple function (shown bottom right).

```
# whichPico.py
# =====
import sys
which1 = sys.implementation
print(which1)
print(which1[2])
if "Pico W" in which1[2]:
    print('Pico with Wireless')
else:
    print('Either Pico or Pico-H')
```

```
import network
import time
import secret

# Setup Network
wlan=network.WLAN(network.STA_IF)
wlan.active(True)
# Provide SSID and PASSWORD from secret.py file
ap=secret.SSID
pwd=secret.PASSWORD
# Try to connect to the wireless network
wlan.connect(ap,pwd)
# Loop until we get connected
while not wlan.isconnected() and wlan.status() >= 0:
    print("Waiting to connect")
    time.sleep(1)
# Print the status value and the ifconfig values
print(wlan.status())
print(wlan.ifconfig())
```

```
import sys

def GetWhichPico():
    which1 = sys.implementation
    if "Pico W" in (sys.implementation[2]):
        return True
    else:
        return False

print(GetWhichPico())
```

MICRO THIS MICRO THAT

And we get back **True**. I've saved this as `WhichPico2.py`.

Now, let's look at how to work with the wireless portion of the Pico-W. It is very similar to what we've done with the ESP-32 and the ESP-8266 microcontrollers.

As a minimum, you need to import the `network`, `time` and a special file named `secret.py`.

`Secret.py` contains

```
# secret.py
# =====
SSID = ""
PASSWORD = ""
```

This way, you can keep the wireless access point and your password secret and have to provide only this sample file to anyone you share your code with. Be sure to modify the two values on your Pico-W before trying to run the program.

Once you have the imports section complete, you need to add only the following code (previous page, middle right).

We create the `wlan` object and set the active method to `True` in order to connect. Then we read the

SSID and `PASSWORD` values from the `secrets.py` file and pass these to the `connect` method. We then need to wait until we get a `True` value from the `isconnected` method and a status greater than 0. This can sometimes take a few seconds, so be patient. Once we are connected, we print the status and the `ifconfig` information, and we can carry on with what we need to do.

Here is the output from the program

```
3
('192.168.1.195',
'255.255.255.0',
'192.168.1.1', '192.168.1.1')
```

The status value of 3 means that we are properly connected to the wireless network. The `ifconfig` information is (from left to right) IP address, netmask, gateway, DNS. Here are all the possible values for the status method.

```
// Return value of
cyw43_wifi_link_status
#define CYW43_LINK_DOWN (0)
#define CYW43_LINK_JOIN (1)
#define CYW43_LINK_NOIP (2)
#define CYW43_LINK_UP (3)
#define CYW43_LINK_FAIL (-1)
#define CYW43_LINK_NONET (-2)
#define CYW43_LINK_BADAUTH (-3)
```

```
def GetWhichPico():
    which1 = sys.implementation
    if "Pico W" in (sys.implementation[2]):
        return True
    else:
        return False
```

If the board actually is a Pico-W, we can try to connect to the wireless network:

```
if GetWhichPico():
    # Setup onboard LED
    led=machine.Pin("LED",machine.Pin.OUT)
    led.off()
```

```
# Setup Network
wlan=network.WLAN(network.STA_IF)
wlan.active(True)
# Provide SSID and PASSWORD from secret.py file
ap=secret.SSID
pwd=secret.PASSWORD
# Try to connect to the wireless network
wlan.connect(ap,pwd)
```

Now loop until we get connected:

```
# Loop until we get connected
while not wlan.isconnected() and wlan.status() >= 0:
    print("Waiting to connect")
    time.sleep(1)
```

Print the status value and the `ifconfig` values, and turn the onboard LED on:

```
print(wlan.status())
print(wlan.ifconfig())
# Turn on the onboard LED
led.on()
# The rest of your code goes here.
```

MICRO THIS MICRO THAT

I've created a simple program that rolls all of the test programs we've used into one. I've named it net1a.py. I've changed most of the comments into explanation lines here, to make it easier for the editors of FCM to include the full code.

```
import machine
import network
import time
import secret
import sys
```

Check that the board is a Pico-W with wireless support (previous page, top right).

First, set up the led object and turn off the led (just in case). Next, set up all the things we need to do for the wireless support, including getting the SSID and PASSWORD from the secret.py file, and call the wlan.connect with these values (previous page, bottom right).

In the event the board is NOT a Pico-W, print that we can't deal with wireless on this board and end the program (top right).

So that's using the RPi Pico-W networking in a nutshell. The site I mentioned earlier has a very nice PDF on using the wireless portion

```
else:
    print('This microcontroller board does not support Wireless!')
```

When the program runs (assuming it is a Pico-W board), here is the output from my machine.

```
Waiting to connect
Waiting to connect
3
('192.168.1.195', '255.255.255.0', '192.168.1.1', '192.168.1.1')
```

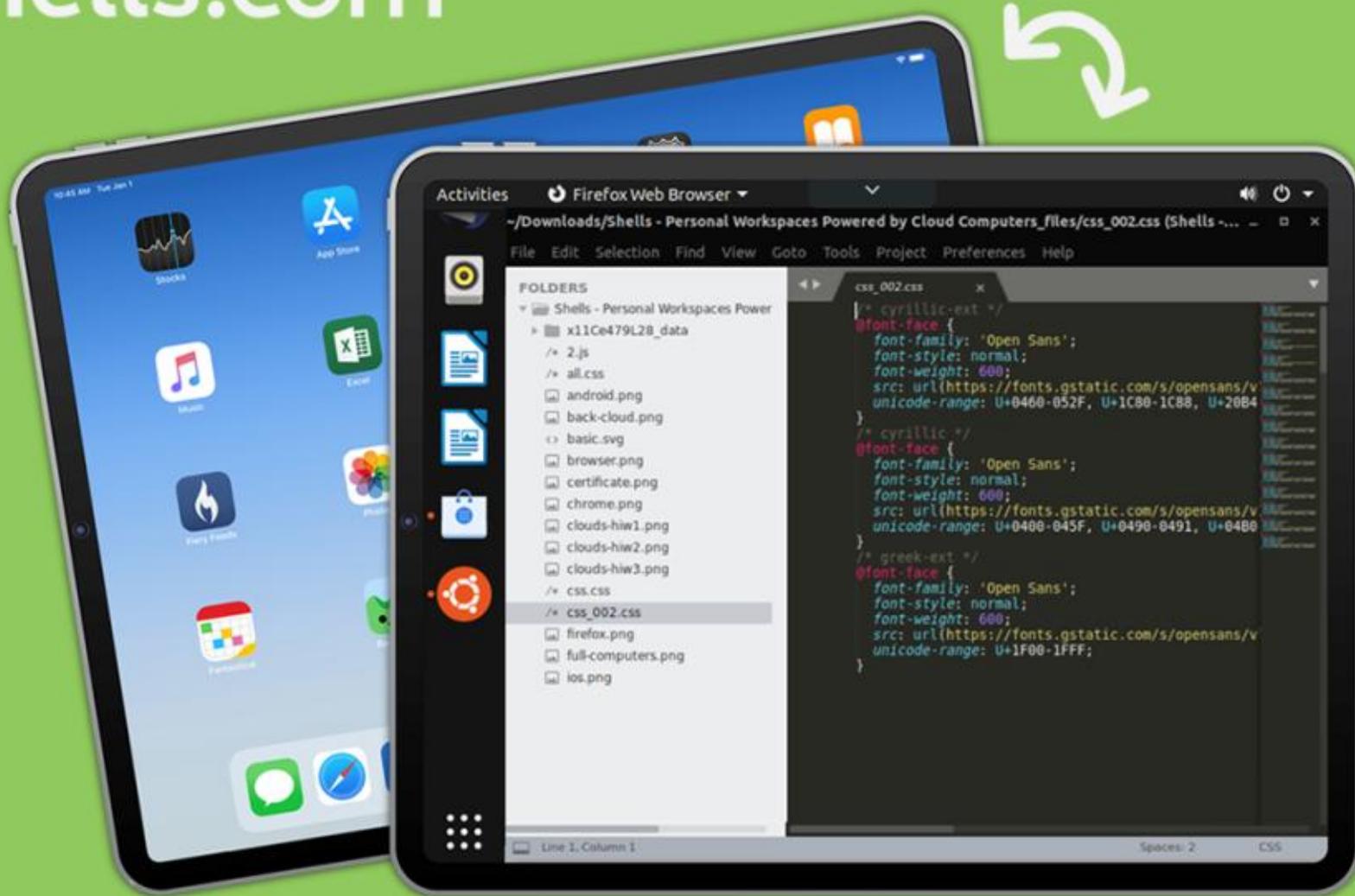
of the Pico-W at <https://datasheets.raspberrypi.com/picow/connecting-to-the-internet-with-pico-w.pdf>, as well as other important information about the board.

I've set up a github repository for the code that we've used this month. You can find it at https://github.com/gregwa1953/FCM-185_MicroThisMicroThat.

Until next time, as always; stay safe, healthy, positive and creative!



Greg Walters is a retired programmer living in Central Texas, USA. He has been a programmer since 1972 and in his spare time, he is an author, amateur photographer, luthier, fair musician and a pretty darn good cook. He still is the owner of RainyDaySolutions a consulting company and he spends most of his time writing articles for FCM and tutorials. His website is www.thedesignedgeek.xyz.



Linux on Your iPad

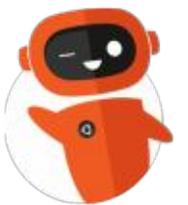
For as low as \$4.95, you can have your own personal Linux cloud computer in minutes on any device.





UBPORTS DEVICES

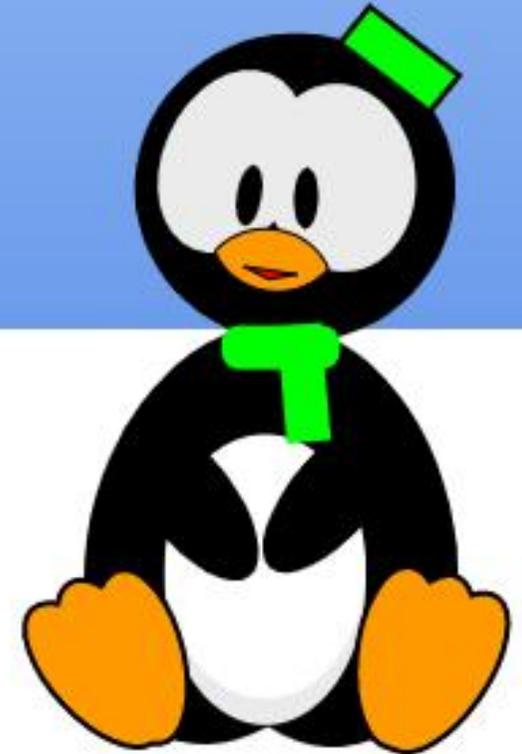
Written by UBports Team



The daily waddle

THEY SHOULD ADD SKIN LESIONS
TO THE KERNEL, IT'S OPEN SORES
AFTER ALL....

ERM...





MY STORY

Written by Brian Bogdan

One Ubuntu Journey

Let me start this story by mentioning that I have been using Ubuntu since version 10.04. I really loved the OS interface that came with 10.04. I updated regularly from Ubuntu LTS to LTS, with a full clean install for 16.04 and 20.04 and finally upgraded to 22.04.

I hate to say this (as I've been behind Ubuntu for so long), but login issues were a problem... when trying to resume from the screen blanker – the gnome-screensaver. Only a reboot, or using tty to kill gnome-shell, would help. It was happening about every 4 days of up-time, where Gnome Shell started using more and more RAM, and the system became unstable after those few days where the RAM footprint of Gnome Shell was over 1.5GB.

Granted, I had a lot of extensions running, but this also happened when the Gnome Shell extensions were not started/loaded. Please understand, I am NOT slamming Ubuntu. It is still my favourite OS, with the Gnome

Desktop environment. But for me, the issues had become far more annoying than I could handle.

ENTER UBUNTU MATE.

I decided that I would just add the desktop environment by opening my Gnome Terminal and

typing:

```
sudo apt update && sudo apt upgrade -y
sudo apt install ubuntu-mate-desktop
```

After a few minutes, the install was complete. I logged out of

Ubuntu, and logged in, and chose Ubuntu MATE. The OS loaded extremely fast! I was greeted with the MATE desktop, which looked almost exactly like Ubuntu 10.04 (prior to the Ubuntu dock).

I immediately went into the Control Center and clicked on





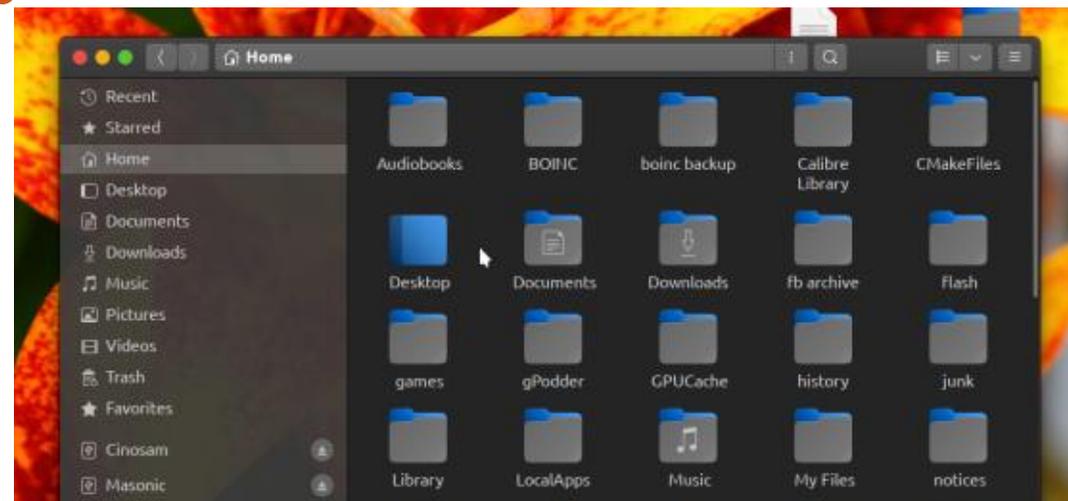
Appearance. I was pleased that, in addition to several coloured Yaru themes, My MC-OS-Mojave theme was still there!

There are so many applets to customize MATE. I chose the Pantheon theme, with the Plank dock. Note that the Clock/weather widget is supplied by My Weather Indicator, and the CPU/RAM widget is supplied by Conky and Conky Manager.

The MATE install added Caja as File Manager, MATE Terminal, Pluma as the text editor, among others, but I have chosen (thus far)

to continue to use Nautilus, Gnome terminal and Gedit. I guess that is one of the cool things of adding the Desktop environment rather than a fresh install of Ubuntu MATE.

The theme choices in Ubuntu MATE are many. Many many! You can change just about every aspect in the windows, icons, window borders, and the pointer. You can also choose from several built-in panels. You can have the top panel set with the MATE application menu and the indicators for running programs, as well as a panel on the bottom of the desktop. There are several types with predefined



layouts but each of these is also customizable. You can choose from one that works like a MAC with the top bar displaying the options available for the particular program. There is also one that resembles the Windows 9 and 10 taskbar and Menu of applications. The small Ubuntu MATE logo at the top-left is the applications menu. (I removed the word menu to keep it cleaner, in my opinion).

I also use a custom theme Mc-OS-MJV dark. While my theme isn't 100% MAC, it is certainly MAC OS inspired.

I really like the retro looking desktop and enjoy the thoroughly modernized version of Ubuntu that is MATE.

I immediately enrolled in the Ubuntu MATE tech forums: <https://ubuntu-mate.community>

I absorbed myself into reading as much as I could about my new OS as well as looking for any issues I might face, and stated bugs, etc, by reading the forum.



Erik has been in IT for 30+ years. He has seen technology come and go. From repairing washing machine sized hard drives with multimeters and oscilloscopes, laying cable, to scaling 3G towers, he's done it.



HOW-TO

Written by Ronnie Tucker

Write For Full Circle Magazine

GUIDELINES

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

RULES

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

• For advice, please refer to the **Official Full Circle Style Guide:** <http://bit.ly/fcmwriting>

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

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

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

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

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

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

TRANSLATIONS

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

REVIEWS

GAMES/APPLICATIONS

When reviewing games/applications please state clearly:

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

HARDWARE

When reviewing hardware please state clearly:

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

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



We now come to Ubuntu Budgie 22.04 LTS, the sixth and final of my reviews of the April, 2022, Ubuntu family of long term support (LTS) releases. This version is the 12th release for Ubuntu Budgie and its fourth LTS version. In fact, the very first release of Ubuntu Budgie was an LTS, in April, 2016.

As an LTS, this new release is supported for three years until April 2025.

INSTALLATION

I downloaded the .iso file from the official website via BitTorrent. At 2.7 GB, this release is actually 300 MB smaller than the last release, 21.10, although it is not clear why.

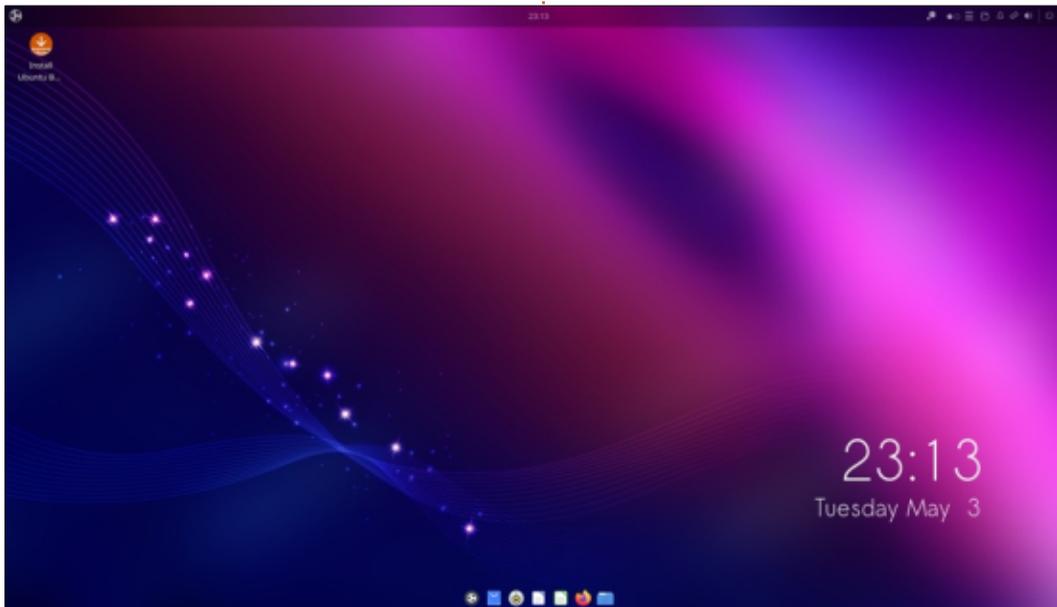
I completed a command-line SHA256 sum check on the file to confirm it was not corrupted and then dropped it onto a USB stick equipped with Ventoy 1.0.74. Ventoy allows multiple ISO files on one USB stick, limited only by the capacity of the device, and then

allows choosing which one to try on boot-up. It is a great boon to users testing out a series of Linux distributions.

SYSTEM REQUIREMENTS

The minimum system requirements for Ubuntu Budgie 22.04 LTS are a 64-bit Intel or AMD processor and 4 GB of RAM. UEFI PCs booting in CSM mode and modern Intel-based Apple Macs are also specifically supported.

NEW



The Budgie interface is still undergoing active development and this release uses a new version, 10.6, which brings a host of small improvements.

The project held a wallpaper competition specifically for this LTS release back in February, 2022, and grandly titled it The Ubuntu Budgie 22.04 Wallpaper Contest. It produced some nicely done wallpapers, 12 of which are included in the selection of 16 wallpapers provided. Oddly, the contest winner, while included, did

not become the new default wallpaper, and instead the same one used since 19.04 remains the default. I guess the developers are fairly attached to it.

Among the many small changes, this release does include Mesa 22, the 3D graphics layer that translates graphics requests to the graphics driver, something that gamers will like.

SETTINGS

Ubuntu Budgie 22.04 LTS does give users a wide range of choices in themes, icons and other user settings. The only drawback is that the controls for these are scattered around a number of places, including Budgie Themes and Layouts, Budgie Control Center, Budgie Desktop Settings, and Budgie Extras. It does take a while to learn where all of these settings are squirreled away.

It is in “Extras” where the desktop applets are hidden. These applets are small programs that can

REVIEW

put weather, clocks, CPU monitors and such on your desktop. There are 37 of them provided, one more than in the last release. Many of these applets have been recently updated, too.

This release has eight available themes, also one more than last time. I guess they accumulate over time. On Ubuntu Budgie, the themes provide more than just window colors, they include a unique wallpaper and icon set to go with them. A few themes are actually installed and the rest downloaded from Personal Package Archives (PPAs).

The default theme remains Pocillo but I tried out a few of the

others. Most of them are quite dark and a lot look very similar. Ubuntu Budgie is still hard to “lighten up” if you are not a fan of dark themes.

The main Budgie menu is fairly flexible, with three ways of finding applications: pages of alphabetical application tiles, categorized lists, or just typing in a search for what you are looking for. The menu can be opened by mouse-click on the top-left icon, or by hitting the “super” (i.e. Windows) key. The menu is easy and intuitive to use, and, in many ways, it is comparable to the Xubuntu Whisker Menu.

Plank is the small Budgie dock usually found bottom-center. It is actually very configurable and

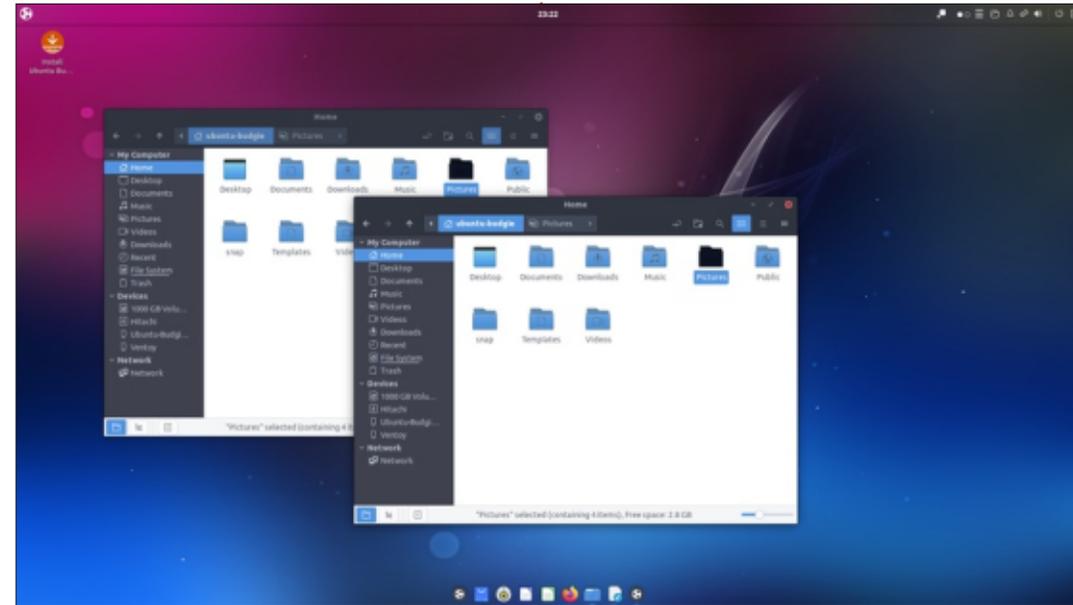
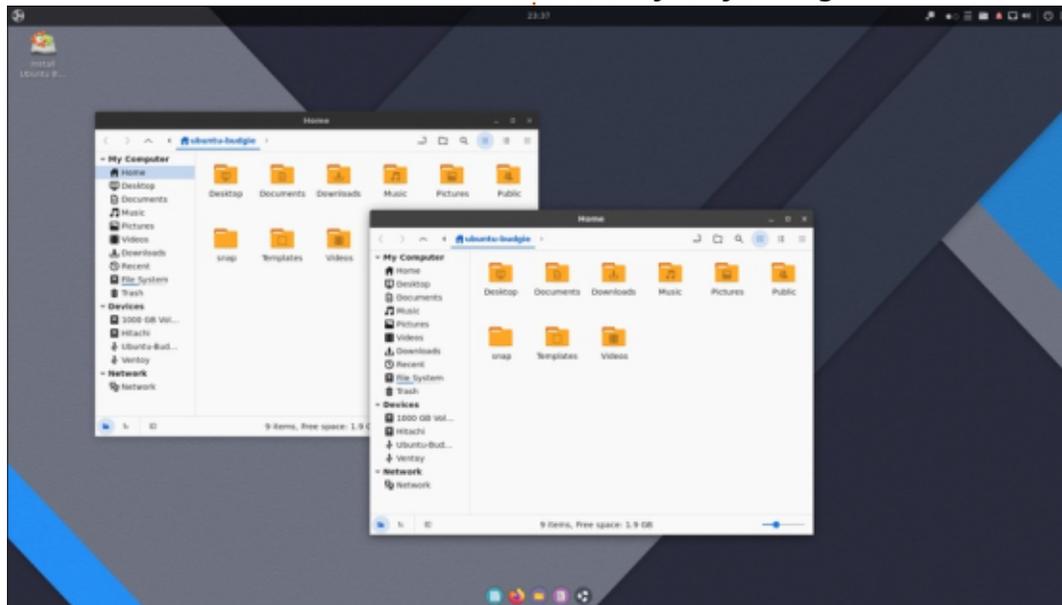
works well, launching applications and keeping track of which ones you have open, with a white dot for each instance.

APPLICATIONS

Some of the applications included with Ubuntu Budgie 22.04 LTS are:

- Archive Manager (File Roller) 3.42.0 archiver
- Celluloid 0.20 movie player*
- Cheese 41.1 webcam application
- CUPS 2.4.1 printing system
- Document Viewer (Evince) 42.1 PDF viewer
- Document Scanner (Simple Scan) 42.0 optical scanner
- Drawing 0.8.5 image editor

- Firefox 99.0.1 web browser**
- Gnome Calendar 41.2 desktop calendar
- Gnome Disks 42.0 disk manager
- Gnome Screenshot 41.0 screenshot tool
- Gnome Software 41.5 package management system
- Gnome System Monitor 42.0 system resource monitor
- Gparted 1.3.0 partition editor
- gThumb 3.12.0 image viewer
- LibreOffice 7.3.1 office suite
- Nemo 5.2.4 file manager
- PulseAudio 15.99.1 audio controller
- Rhythmbox 3.4.4 music player*
- Text Editor (gedit) 41.0 text editor
- Thunderbird 91.8.0 email client
- Tilix 1.9.4 terminal emulator
- Transmission 3.0.0 bittorrent client*



REVIEW

* indicates same application version as used in Ubuntu Budgie 21.10
** supplied as a snap, so version depends on the upstream package manager

As can be seen, most of the application suite is from the Gnome desktop. The notable exception is the Nemo file manager which is actually a fork of the standard Gnome file manager, Nautilus. Nemo restores many functions that Nautilus once had but were removed in a simplification drive a few years ago, such as file bookmarks and the “up one file level” arrow. Nemo lacks only bulk file-renaming.

The last release, Ubuntu Budgie

21.10, included the Catfish file search utility from the Xfce desktop. I noted at the time that it was an odd inclusion, as Nemo has native file searching, so I was not surprised to see that Catfish was removed from Ubuntu Budgie 22.04 LTS.

Like most other Ubuntu flavors, Ubuntu Budgie 22.04 LTS includes LibreOffice, complete less only LibreOffice Base, the infrequently-used database application. It can be installed if needed, however.

The potential controversy in the whole 22.04 LTS series of releases has been around switching the Firefox web browser from a .deb file to a snap. Mainstream Ubuntu

and Ubuntu Unity made the switch in 21.10 but none of the other Ubuntu flavors did for that release. With 22.04 LTS, all have now moved to the snap version of Firefox, with some, like Xubuntu, providing a detailed list of reasons why that made sense. Ubuntu Budgie’s release notes just dryly state, “for 22.04, the deb version of Firefox has been removed since only the snap is now available.”

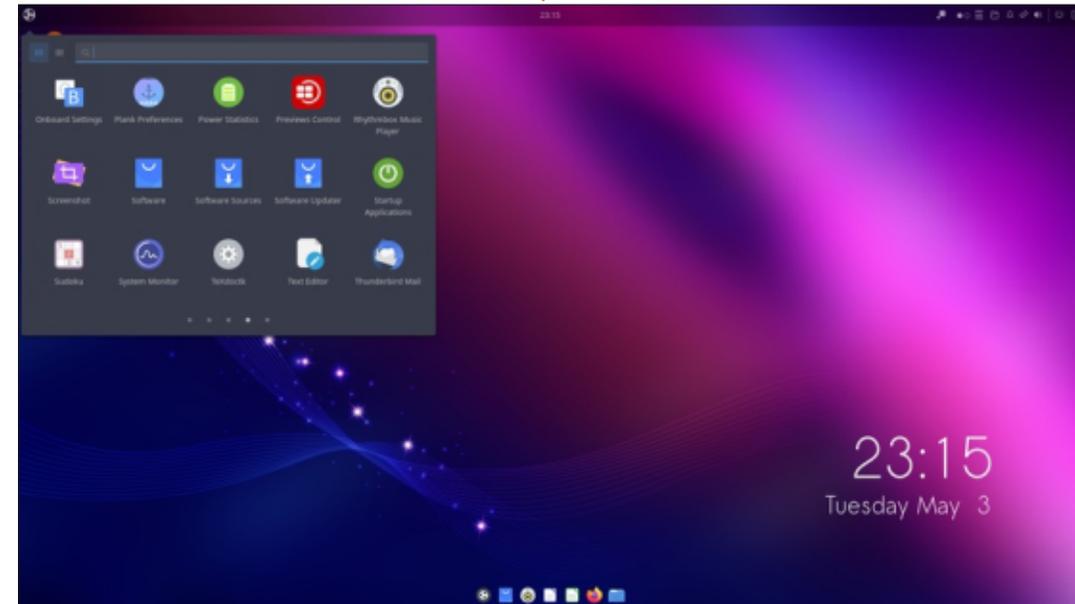
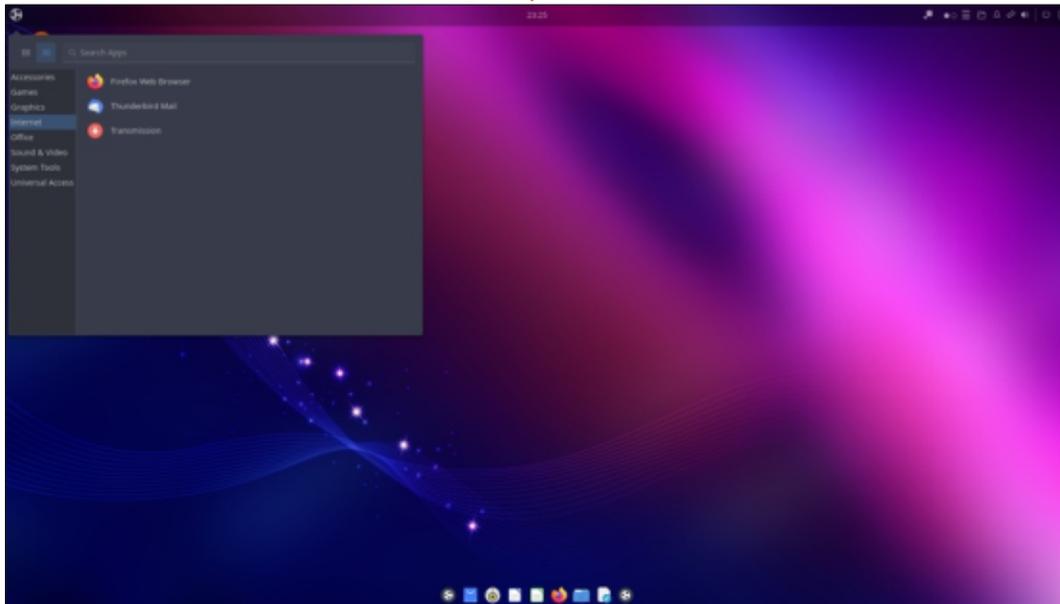
The default text editor is the venerable gedit from Gnome, now at version 41.0, which displays on menus simply as “text editor”. It is a simple text editor but comes with spell-checking pre-installed, a tabbed interface, and has a choice of seven different syntax

highlighting color schemes – although all but two are dark schemes. It works well and is suitable for writing websites, text documents, or simple coding.

CONCLUSIONS

Ubuntu Budgie 22.04 LTS is a highly-polished and mature distribution that looks good and works well. With three years of support, this LTS will end up as many users’ desktop for the next while.

Ubuntu Budgie’s only shortcomings are its scattered controls for user settings, which will take a new user some time to



REVIEW

master. Once located, there are a wide range of settings that can be used to personalize the Budgie desktop, although they do tend towards darker themes.

Ubuntu Budgie is aimed at new users from the Mac and Windows worlds, although it has its own Linux fanbase as well. For anyone who likes Gnome applications, but not the sort of interface found on Ubuntu, Ubuntu Budgie may be worth a serious look.

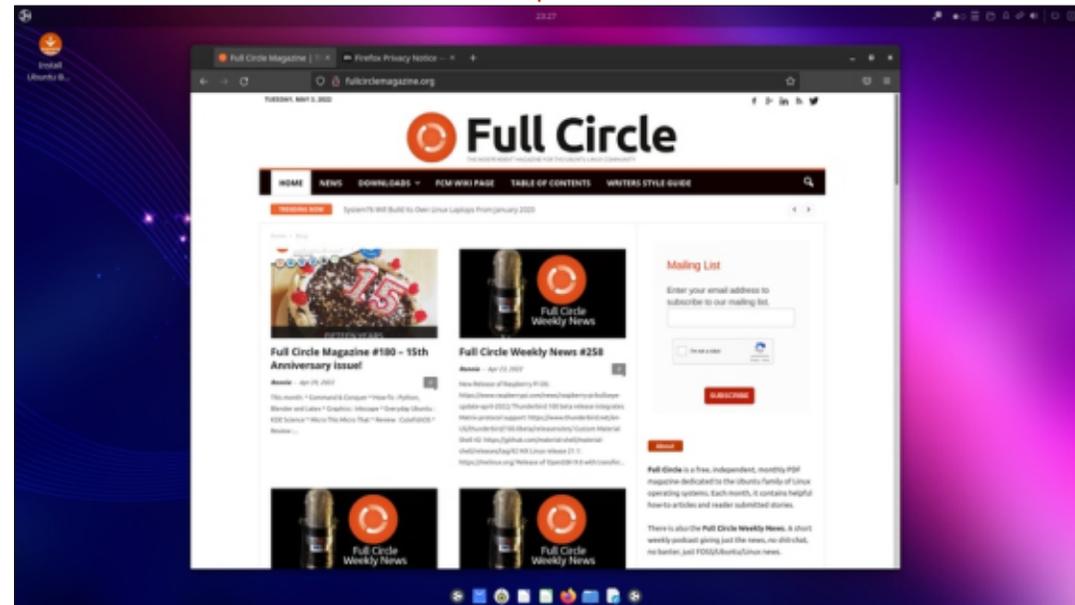
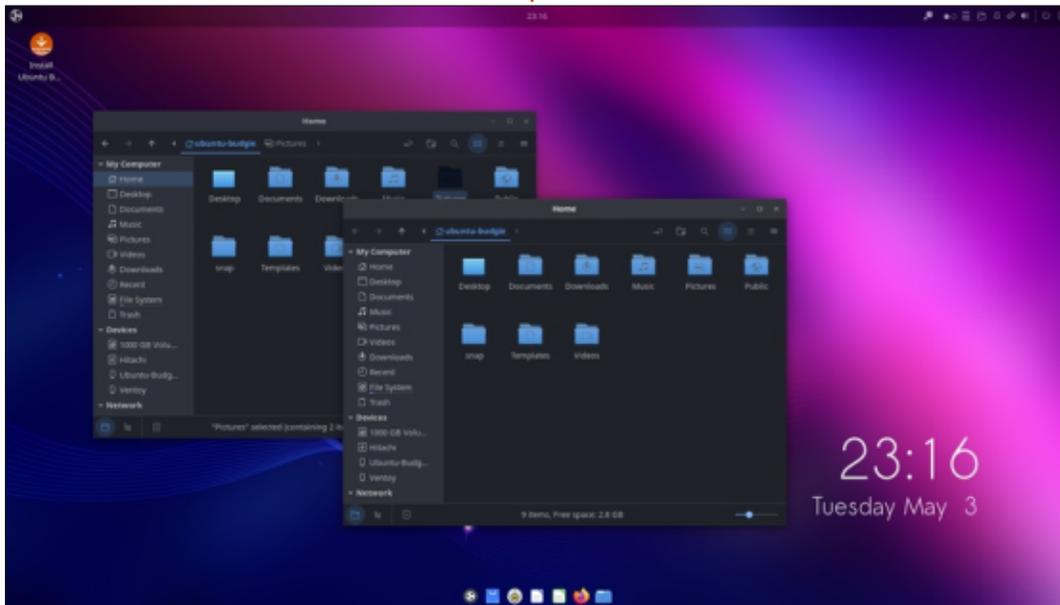
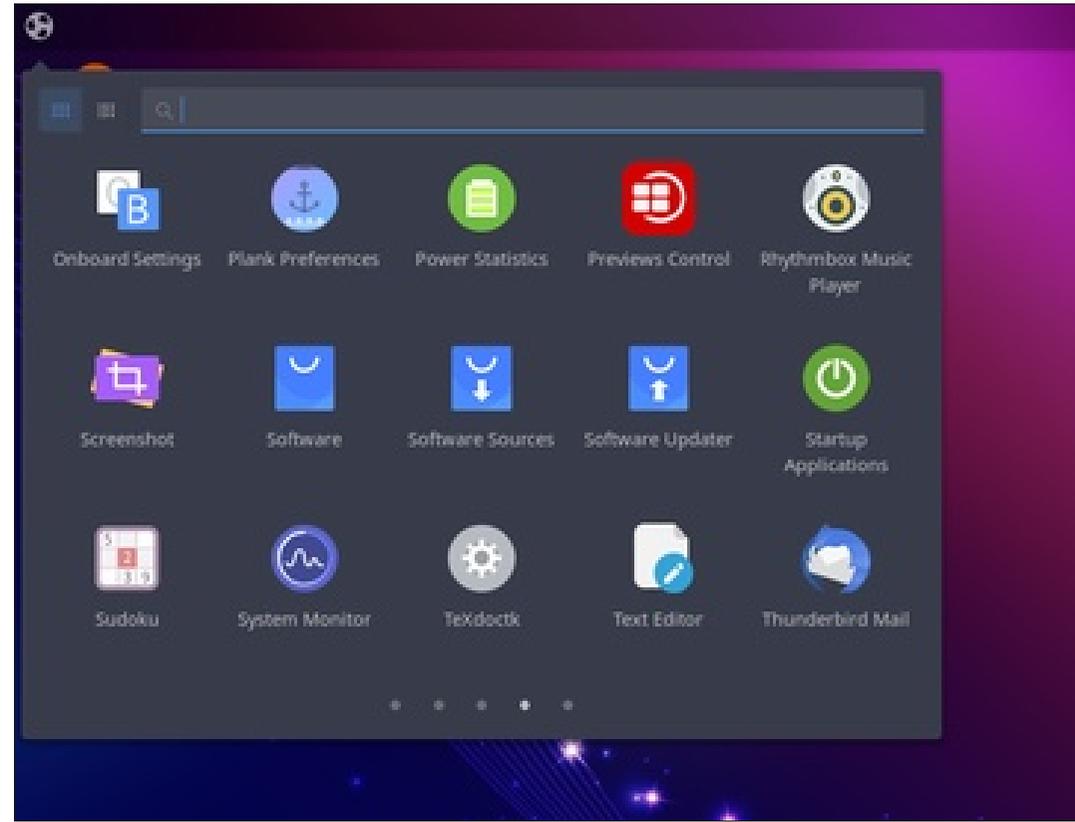
EXTERNAL LINKS

Official website:

<https://ubuntubudgie.org/>



Adam Hunt started using Ubuntu in 2007 and has used Lubuntu since 2010. He lives in Ottawa, Ontario, Canada, in a house with no Windows.





For the sixth and final look at some independent Linux distributions, this month we review NixOS, a unique and unusual operating system.

NixOS is entirely built around its own custom package manager, Nix. This not only installs and removes application packages, but also powers some unusual features, including system reversion rollbacks, reliable upgrades, and the ability to run different versions of the same applications side-by-side. It also installs applications for single users in a computer, meaning each user has their own applications.

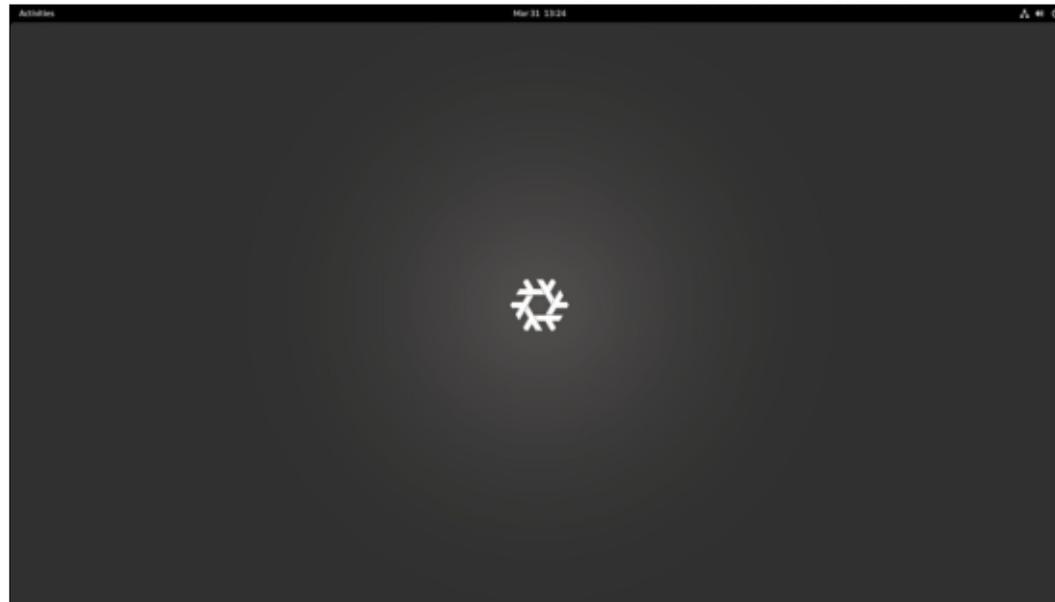
The Nix package manager is used to build the entire operating system including the applications, system packages, configuration files, and even the Linux kernel itself.

To make this all work, NixOS has a completely new file system organization with all the components isolated to give enhanced reliability and security.

The resulting operating system is advertised as being able to create "reproducible, declarative and reliable systems".

NixOS is basically designed to be a dream operating system for developers, system administrators and people who work in IT security, and seems to be mostly commercially deployed in companies working in those fields. While optimized for these roles, it is a general desktop distribution as well.

Because it is an independent



distribution, NixOS is not based on any other Linux distribution. In fact, it is built from scratch using Nix.

So, we know developers love it, but my task was to see if this unusual distribution is suitable for the average desktop Linux user.

BACKGROUND

NixOS traces its origins back almost 20 years, to 2003, when Eelco Dolstra started the Nix package manager as a PhD research project at Utrecht University in the Netherlands. From those

beginnings, it drew in more developers, and on 31 October, 2013, the first public release of the Nix-based operating system was made. Called NixOS 13.10, it adopted the same "year.month" format that Ubuntu uses. That initial release was codenamed "Aardvark", and each release since then has been in alphabetical order and carried the name of an animal. Unlike Ubuntu, there are no adjectives, though.

The current release is 21.11, out on 30 November, 2021, and codenamed "Porcupine", is the 16th release.

NixOS has grown so that it now has a foundation backing it, the NixOS Foundation. It has the NixCon series of developer conventions, regional meetups, community-organized teams for security, infrastructure, convention, marketing, platform moderation and much more. The 21.11 release had 1,541 developers contribute to it, so the project is now fairly big and quite well organized.

REVIEW

There are also a dozen companies that provide commercial support for NixOS enterprise deployments.

Despite its radically different file system focused on isolating applications and processes, it does superficially have the basic and familiar Linux file system organization, but the directories are all redirects to the underlying new system.

The only package manager is Nix, so there is no APT or similar available. The Nix repositories have over 80,000 application files, all in its unique format, so just about every Linux application is there.

NixOS is available in three basic forms. There is a "minimal ISO image" which has no graphical interface, and two desktop versions with Gnome and KDE. Other desktops, like Xfce, LXDE and LXQt, are available, but as package downloads for installation.

Because the developer emphasis seems to be on the package and file systems, the desktops are unmodified, plain, vanilla versions.

GETTING NIXOS

I downloaded the Gnome version of NixOS 21.11 as an ISO file directly from the official website downloads page via https, as there

are no bittorrents offered. The Gnome desktop version is 2.1 GB in size, which compares with Ubuntu 21.10 at 2.9 GB.

There are updated builds since the 30 November, 2021 release so that, even though the current version is five months old, it arrives fully updated.

SHA 256 sums are provided, and it is always a good idea to run the test on the download to make sure your ISO file is not corrupted or otherwise compromised.

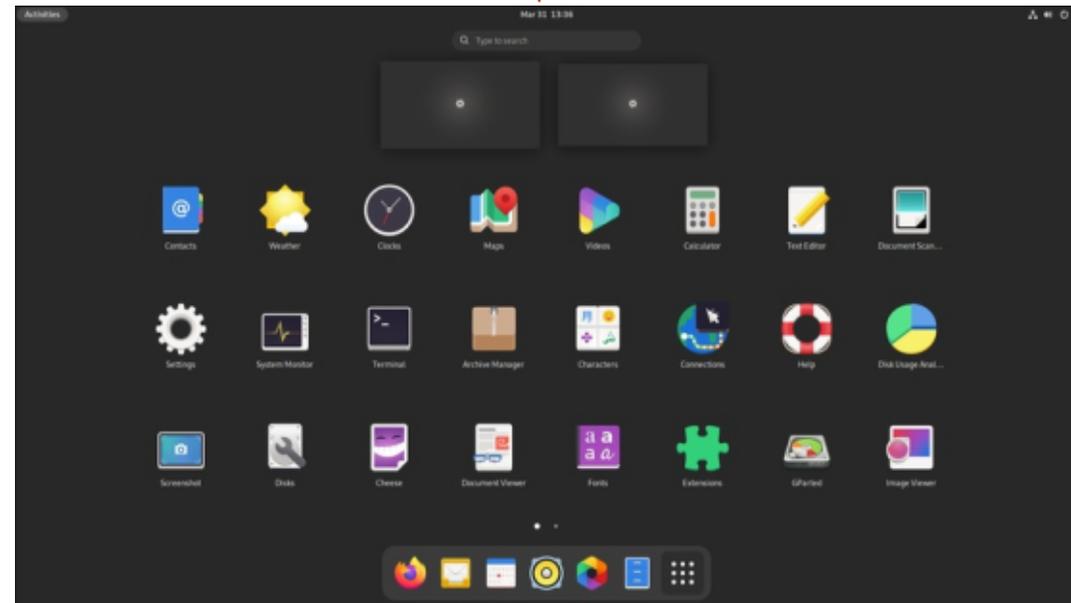
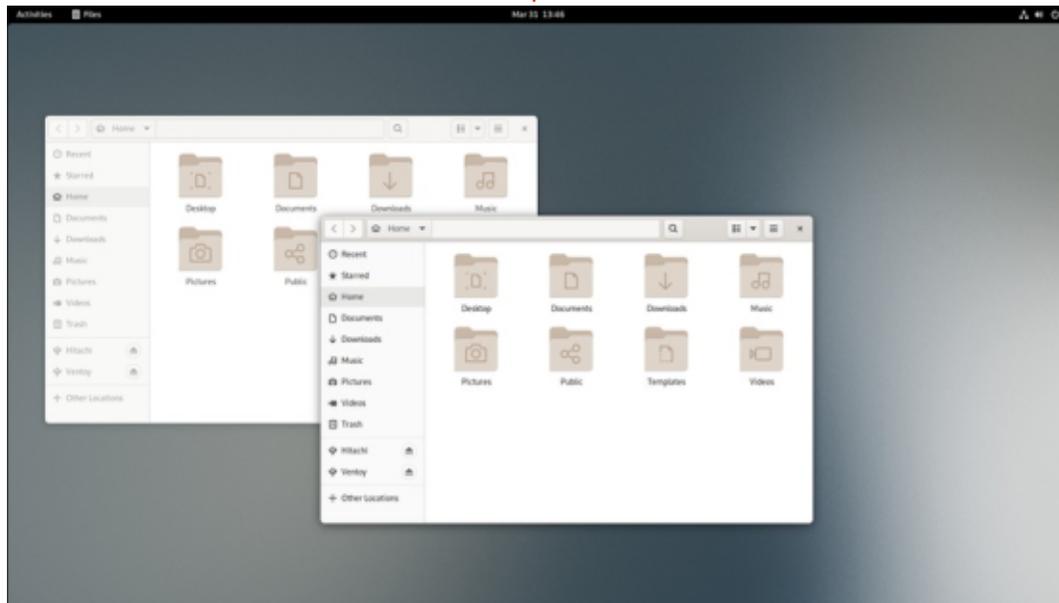
INSTALLING

I used Ventoy 1.0.72 to put the

ISO file on a USB stick. With Ventoy already installed on the stick, this is as easy as just copying and pasting NixOS with the file manager. There is no need to unpack or write the ISO file, as Ventoy takes care of all that on boot-up. As always, Ventoy worked great and I was able to boot up NixOS without issues.

SYSTEM REQUIREMENTS

NixOS has a lot of documentation on its official website but no minimum system requirements are specified. In one place it does suggest that 4 GB of RAM are recommended to run some specific processes, though.



REVIEW

a dead end in the opaque file system.

The KDE desktop version likely comes with more user options than Gnome does.

APPLICATIONS

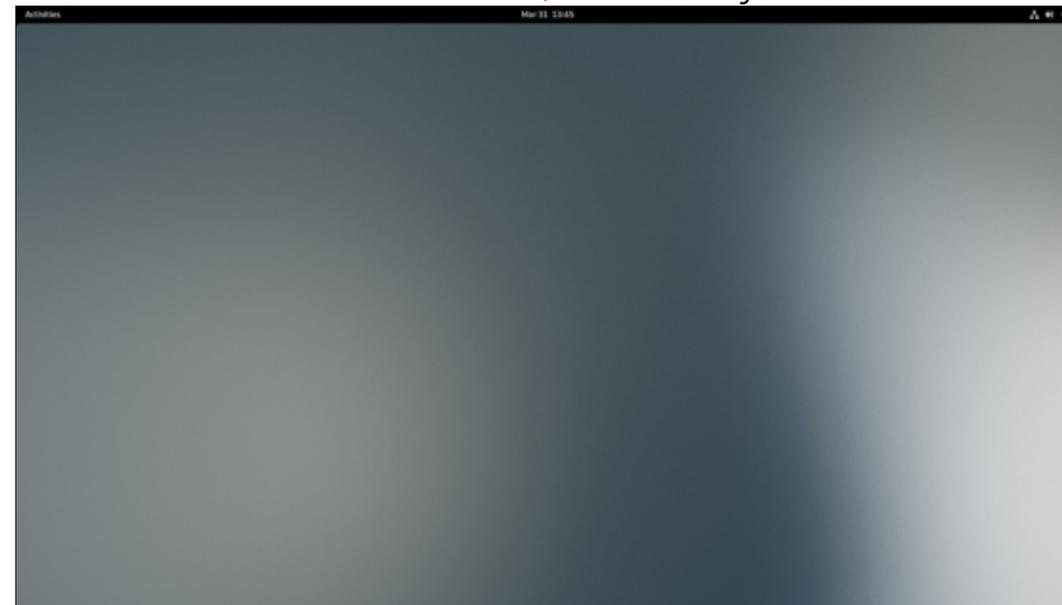
NixOS comes with a moderate collection of application software, almost all from Gnome. This includes:

- Archive Manager (File Roller) 3.40.0 file archiver
- Cheese 41.1 webcam application
- Document Viewer (Evince) 41.3 PDF viewer
- Document Scanner (Simple Scan) 40.6 optical scanner
- Firefox 98.0.2 web browser
- Gnome Calendar 41.0 desktop calendar
- Gnome Disks 40.0 disk manager
- Gnome Files (Nautilus) 41.1 file manager
- Gnome Music 41.0 music player
- Gnome Photos 40.1 photo manager
- Gnome Terminal 3.42.1 terminal emulator
- Gnome System Monitor 40.0 system monitor
- Gnome Web (Epiphany) 41.2 web browser
- Gparted 1.3.1 partition editor

- Image Viewer (Eye of Gnome) 41.0 image viewer
- Text Editor (gedit) 40.1 text editor
- Videos (Totem) 3.38.2 movie player

Notably missing from this list is an office suite such as LibreOffice or even a word processor. This is probably just more evidence that NixOS is more intended for developers than regular desktop users. There is also no default bittorrent client. As noted, though, with 80,000 packages to choose from, including LibreOffice and Transmission, it isn't hard to add what you need for any possible role.

CONCLUSIONS



Overall NixOS Gnome 21.11 impresses as serious, neat and elegant. If you are a fan of the unmodified Gnome desktop, then you will find a lot to like here.

The downside of this distribution is the steep learning curve for package management, including updates and the like. No matter which distribution you come from, you will have much to learn to be able to make Nix work well for you on the command-line. The lack of a beginner-level guide is a hindrance or at least creates a high bar for entrance here. You may need to rely on third-party guides on the internet to find the information you need.

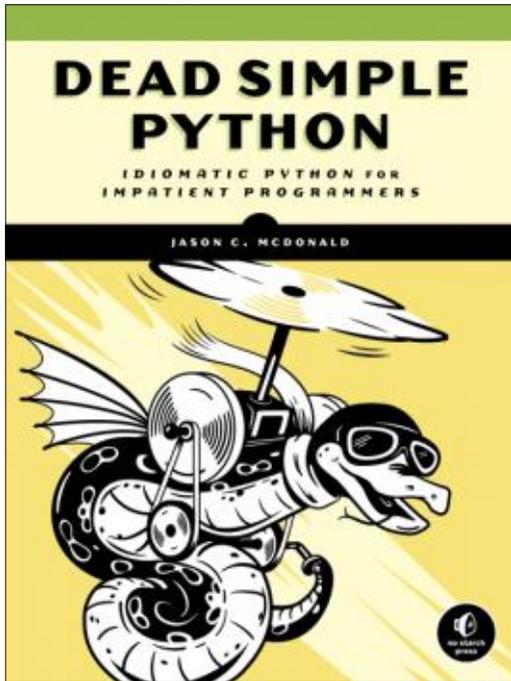
If you have a work-related or personal reason to use NixOS, such as enhanced security, packaging isolation, or even if you are just looking for a new Linux challenge, it could be worth the time invested. But, for most average Linux users just seeking a nice, plain, unmodified Gnome desktop distribution, something like Fedora or Debian would be an easier transition and an easier installation too.

EXTERNAL LINKS

Official website:
<https://nixos.org/>



Adam Hunt started using Ubuntu in 2007 and has used Lubuntu since 2010. He lives in Ottawa, Ontario, Canada, in a house with no Windows.



Dead Simple Python - Idiomatic Python for the Impatient Programmer

Jason C. McDonald

Publisher: No Starch Press
Release Date: November 22, 2022 (Amazon USA)
Pages: 752
ISBN-13: 9781718500921
Price: \$59.99 USD Paperback (Preorder on Amazon.com USA)
Website: <https://nostarch.com/dead-simple-python>

When I got notification that this book was available for review, I saw the title "Dead Simple Python" and thought "Oh brother! This is another of THOSE books." thinking that it would be for the quasi-beginner and so dumbed down that it would almost be unusable. However, I was wrong!

WHO IS THIS BOOK FOR?

The book is mainly designed for programmers coming to Python from another programming language and intermediate level Python programmers.

WHAT DOES SIMPLE MEAN?

To quote from his own explanation:
"The topics discussed herein may, at first blush, appear anything but simple. You may be wondering how simple a book this thick could actually be! When I titled this book Dead Simple Python, I was describing a retro-spective view of the topics,

rather than a forward-looking one. One should recognize that any topic worth learning, when first approached, will feel insurmountable. Similarly, any explanation worthy of an existing software developer should possess enough depth to utterly disqualify it from the forward-looking label of "simple."

The book is divided into five sections, "The Python Environment", "Essential Structures", "Data and Flow", "Advanced Concepts" and "Beyond the Code". The first section examines the basics of Python, its tools, basic syntax and project layout. The second (Beyond the Essential Structures) looks at variables, functions, classes and exceptions. Part 3 (Data and Flow) goes into ways to control execution flow and manipulating data like data structures, loops, iteration, generators and more. Part 4 (Advanced Concepts) looks into inheritance, introspection and concurrency. Finally part 5 digs into testing, debugging and deploying your projects (although the deploying portion gets its start in

part 1 chapter 4, Project Structure and Imports).

Going through chapter 5 (Variables and Types), with a very jaundiced eye expecting the most simple examples and topics possible here, I was surprised by the author discussing the functions `id()` and `isinstance()` (both of which are rarely discussed in "general" Python texts. In addition, the author goes through a very interesting examination of Python's garbage collection and even the `{interpreter shutdown}` module and then jumps into a very clear explanation of global scope.

That's just a quick glance at some of the topics that Mr. McDonald clearly demystifies and explains in this tremendous book. If I were to try to continue telling you about the other wonderful things that he explains, I would probably end up taking up 1/2 of the magazine for this month.

BOTTOM LINE

The author does a wonderful job

BOOK REVIEW

of making the complex and many times under explained concepts closer to truly simple. This is definitely a book that any Python programmer NEEDS to have in his bookshelf!



Table of Contents

- Foreword
- Acknowledgments
- Introduction
- Part I: The Python Environment
 - 1: The Python Philosophy
 - 2: Your Workbench
 - 3: Syntax Crash Course
 - 4: Project Structure and Imports
- Part II: Essential Structures
 - 5: Variables and Types
 - 6: Functions and Lambdas
 - 7: Objects and Classes
 - 8: Errors and Exceptions
- Part III: Data and Flow
 - 9: Collections and Iteration
 - 10: Generators and Comprehensions
 - 11: Text IO and Context Managers
 - 12: Binary and Serialization
- Part IV: Advanced Concepts

- 13: Inheritance and Mixins
- 14: Metaclasses and ABCs
- 15: Introspection and Generics
- 16: Asynchrony and Concurrency
- 17: Threading and Parallelism
- Part V: Beyond the Code
 - 18: Packaging and Distribution
 - 19: Debugging and Logging
 - 20: Testing and Profiling
 - 21: The Parting of the Ways
- Appendix A: Special Attributes and Methods
- Appendix B: Python Debugger (PDB) Commands
- Glossary

Dead Simple Python (Sample Chapter) © 4/13/22 by Jason C. McDonald

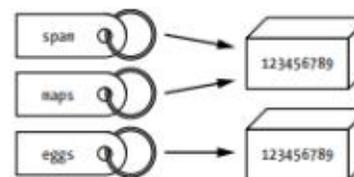


Figure 5-2: `spam` and `maps` share an identity; `eggs` is bound to an equivalent value, but it does not share identity.

It just goes to show, `spam` by any other name is still `spam`. Python isn't guaranteed to behave exactly like this, and it may well decide to reuse an existing value. For example:

```
answer = 42
insight = 42
print(answer is insight) # prints True
```

Listing 5-6: `assign_reuse.py`

When I assign the value 42 to `insight`, Python decides to bind that name to the existing value. Now, `answer` and `insight` happen to be bound to the same value in memory, and thus, they share an identity.

This is why the identity operator (`is`) can be sneaky. There are many situations in which `is` appears to work like the comparison operator (`==`).

GOTCHA ALERT

The `is` operator checks identity. Unless you really know what you're doing, only use this to check if something is `None`.

As a final note, the built-in function `id()` returns an integer representing the identity of whatever is passed to it. These integers are the values that the `is` operator compares. If you're curious about how Python handles names and values, try playing with `id()`.

PEDANTIC NOTE

In CPython, the value returned from the `id()` function is derived from the memory address for the value.

Data Types

As you've likely noticed, Python does not require you, the programmer, to declare a type for your variables. Back when I first picked up Python, I joined the `#python` channel on IRC and jumped right in.

"How do you declare the data type of a variable in Python?" I asked, in all the naivete of a first-year coder.

Within moments, I received a response that I consider to be my first true induction into the bizarre world of programming: "You're a data type."



LETTERS

If you would like to submit a letter for publication, compliment or complaint, please email it to: letters@fullcirclemagazine.org. PLEASE NOTE: some letters may be edited for space.

Join us on:



[facebook.com/
fullcirclemagazine](https://facebook.com/fullcirclemagazine)



twitter.com/#!/fullcirclemag



[linkedin.com/company/full-
circle-magazine](https://linkedin.com/company/full-circle-magazine)



[ubuntuforums.org/
forumdisplay.php?f=270](https://ubuntuforums.org/forumdisplay.php?f=270)

FULL CIRCLE NEEDS YOU!



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

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

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



Q&A

Compiled by EriktheUnready

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

Welcome back to another edition of Questions and Answers! In this section we will endeavour to answer your Ubuntu questions. Be sure to add details of the version of your operating system and your hardware. I will try to remove any personally identifiable strings from questions, but it is best not to include things like serial numbers, UUIDs, or IP addresses. If your question does not appear immediately, it is just because there is such a lot, and I do them, first-come-first-served.

I worked for an automation company back in the day. My direct boss, the IT Director, was unhappy with his laptop. To fill in the background, let me state that he would test ALL our software on this laptop. Needless to say, Windows needed a registry clean and temp file clean like it is nobody's business. (He also used it everywhere crappy, so the exhaust ports were probably all blocked inside.) I would venture to say that, daily, he would install and uninstall at least two programs downloaded

from the internet, sometimes sketchy as all heck. However, he was very impatient and had the opinion that everything should just work and it was Microsoft's baby to see that it did, as he paid good money for the OS. He could never set aside an hour or two just so I could get into the guts of the thing and see if I could rectify the problem. If the problem wasn't solved in the first five minutes, he would chase you away and say he would do it himself. This culminated in him taking the laptop after a few weeks and shoving it behind a forklift and simply driving over it, claiming a new one from the insurance company. He loudly blamed the laptop manufacturer, Microsoft, etc. - never himself. Ubuntu is an operating system only and things like the above probably drove them to the Snap store. For newer computers, this is not a problem, but with the current chip shortages, a lot of people are holding on to their old 2011 computers as replacing them is too costly. Should Ubuntu maybe not ship a version aimed at older computers? Sure, Linux Lite

supposedly fills that gap, but I think there is a lot of scope for improvement even there. Do you think you are up to the challenge? *nudge-nudge*, *wink-wink*.

Q: I did something and after an update - if I open a folder with another program, it opens with the terminal instead of files. I tried googling, but I am not sure which search terms will get me there. Nothing I have tried (from places like stack-overflow) has worked for me. I have even reinstalled Nautilus, hoping it would fix the defaults. I have it in the back of my mind that it is something to do with dconf editor, but I am not finding anything related I can put my finger on. I'm using Ubuntu Budgie with an i5 and 8 GB of RAM.

A: I'd suggest looking for mimeapps.list and adding: inode/ directory=org.gnome.Nautilus.desktop in the Added Associations section at the top, and inode/ directory=Nautilus.desktop in the Default Applications section.

Reboot and if it still does not work, let us know.

Q: Why would a folder show up in the terminal, but not in the file manager?

A: Two reasons; one, it is hidden (it has a . before the name), and two, the permissions are set to not include the user that is browsing.

Q: I have Ubuntu 22.04 installed on my system, but with i3 desktop. I use a Nvidia3060 with two 24" Dell displays via HDMI. I recently installed VirtualBox with Windows and wanted to know if I can play games on both screens with Windows in VirtualBox? I have 32 GB of RAM and a 10th gen i5 with 8 cores.

A: My guess would be probably no, but I would start asking that in the VirtualBox forums rather.

Q: I'm a bit clueless, so please bear with me. I installed my P2P apps in Ubuntu, set my network port to 119, and set my method to encrypt only and also my IP v6 to off. Which is what I used to do on Windows 7. Nothing else. I did not even change the interface colors. Yet I cannot get it to work. I tried 3 different P2P programs and they all will not communicate with the outside world. My friend suggested that it is UFW I need to install to make it work, but I have no idea how to set it up for P2P networking.

A: Ubuntu is a bit less forgiving than Windows in this regard, ports below 1024 will not work for anything other than listed. See: <https://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml>

Q: I have a Dell dock -K20A- connected to my laptop via USB-C with a widescreen monitor attached to it via HDMI. My fan goes up when I am working on the wide screen, but almost never comes on when I am using my laptop in bed. I have installed TLP, but it does not help. My laptop has only a Core i7 and no GPU. I still use

Ubuntu 20.04.

A: I am unsure what you want, but I suspect you are wondering why your fan is going bananas. Firstly, the laptop is charging when the USB-C plug is in, causing heat, as energy is converted from one range to another. Secondly, the GPU, yes, your laptop has one or you would not get a display, is working twice as hard for a second monitor and then extra because it is a wide screen. The laptop display needs to fill in only 1920x1080 pixels every second, whereas the external one needs to fill, say, 3840x2160 every second. Make sure the laptop is well ventilated off the table too, the height of the feet are usually "minimum" height.

Q: Hello!!! I am a Mac convert and I can't even go back to Mac any more, it feels like it is outdated when I do use it. I set up gnome boxes using this as my guide:: <https://www.funkyspacemonkey.com/how-to-use-gnome-boxes-to-set-up-virtual-machines>. I am sorta stuck now trying to get the display settings right. I want to set up a windows XP machine just to load

my old games like Crusader, no Regret. I don't need Xbox controller support or anything funny, just the basics.

A: Looks like everyone has display issues this week. What you are looking for is VirtualBox rather. I'd agree that Gnome Boxes are probably a lot "lighter", but you will need the "client addons" that VirtualBox provides, that is like a driver for the underlying OS to set display size.

Q: My Ubuntu 22.04 LTS version drains battery a lot during suspend mode on the cooling pad, maybe 6 hours. It lasts a lot longer without the cooling pad, maybe 2 days, why would that be?

A: That probably has more to do with the BIOS setup than Ubuntu. Make sure that your BIOS USB settings are set NOT to provide power in suspend mode or power-off mode, something like that. I have a sneaky suspicion you have a cooling pad with fans that continue spinning when you close your laptop lid. Another option would be to simply unplug the cooling pad from the laptop when it sleeps.

Q: I have a question about Ubuntu Studio and the new 'fixes' for speculative attacks. I use the realtime kernel, but my system is always updated, so I assume I have the latest patches. From what I understand, the 'fixes' basically kill any advantage the realtime kernel offers me. Is there still a point in using the realtime kernel?

A: Sorry, this is a bit out of my depth, as I never really used the RT or low-latency kernel. You can read more about it here: <https://unix.stackexchange.com/questions/553980/why-would-anyone-choose-not-to-use-the-lowlatency-kernel> - You do, however, know that you can disable all those patches, if you are sure you don't load untrusted software. See here: <https://magenaut.com/disable-spectre-and-meltdown-mitigations/>

Q: Our wifi access point sorta melted and I'm back to using cables in the lounge. I just can't afford to buy another at the moment. I have noticed that sometimes I get 1Gb/s to the switch and sometimes only 100Mb/s. It feels like I haven't used these cables since the early 2000s, so, on

Q&A

the one hand, it is good to get use from something I was going to put out with the trash. On the other hand, I used Windows 95 then, and I have no idea how to set all the nuances of Ubuntu wired networking. I am on Ubuntu 18.04 until I get tired of it.

A: My guess is that the issue is not Ubuntu, but old cabling that came across with Moses on the boat. Old cables are usually rated for only 10-100Mbit traffic. Try picking up some CAT5 cabling. (Make sure it is, as I have tested more than 5 brands in 2020 that said CAT5 but were not compliant).

Q: I used to use Ubuntu standard, but gnome is kinda boring, so I switched it up and went to Budgie. So far, I am liking it; just one small issue. When I save a file to my desktop, there is nothing there. Then, I open the file manager and click on the desktop folder and my files are there. I want my files on my desktop for easy access, not the desktop folder. How can I change this?

A: Sorry my friend, but you are not making any sense. The

“desktop” you see IS your desktop folder. The only thing I can think of is that you have the desktop icons turned off in the budgie settings. This also means it will hide any other files on the desktop.

Q: I can't get my scanner to actually scan in Ubuntu 22.04. It's detected, etc, but when push comes to shove, nothing happens.

A: Try another scanning program or “app”. The scanning software for Linux is not pretty by a long shot, but try another. For instance, if Simple Scan does not do it for you, try Gscan2pdf or something.

Q: After my Windows update earlier this week, I have not been able to get into my Ubuntu. I get just a blank screen. Help?

A: Hi, Unfortunately you have provided no information, so none can come out. Do you dual-boot, or do you use WSL or some other method of running Ubuntu? A good example would be to say I have Ubuntu version x with Windows 10 on VirtualBox on top of Proxmox or whatever, with 2GB of

memory and standard VGA drivers. When I run VirtualBox in full-screen, there is no display, or something to that effect, and you will get help a lot easier.

Q: Which terminal is better to use, like Kitty or Tabby or what modern terminal should I use?

A: Try them all and keep the one you like. If you want an opinion poll, we can ask our readers.

Q: Why would one use xdg-open in the place of Curl? It seems Curl can do everything and more?

A: The reason is that xdg-open will open the graphical application associated with whatever it is you are grabbing. For instance, if you grab a magnet file, xdg-open will open your torrent application using the file as an argument. Curl, on the other hand, is more of a terminal command-line tool. Though it is powerful, each has its niche.

Q: I am learning Ubuntu and I want to install the package manually, but I get an error. I am

attaching screenshots. I have few virtual machines to play with so breaking is not a problem. Get from here-> <https://pkgs.org/> <removed> <removed>

A: Package architecture is usually arranged by system, denoted in the package name. You cannot install a package meant for an ARM CPU on your x86 CPU. One is CISC and one is RISC, on a very basic level. It is the same reason you do not put Diesel in your petrol car. Both are fuel, but meant for very different engines. Dpkg is fine for installing packages, but try to stick to apt or apt-get to make sure all dependencies are satisfied.



Erik has been in IT for 30+ years. He has seen technology come and go. From repairing washing machine sized hard drives with multimeters and oscilloscopes, laying cable, to scaling 3G towers, he's done it.



What is Tabletop Ubuntu? This column aims to discuss the thriving genre of digital tabletop games. Prior to the pandemic, I did not play these games much myself, preferring to spend my time and money on boxes and boxes of wood, plastic, and paper. I stacked them nicely on shelves. They were beautiful. When I ran out of space, I bought more shelves. And they were beautiful. When COVID happened, I watched my games slowly gather (more) dust on the shelf. It seemed natural to try out digital versions that I could play with others. Pros of digital tabletop games: you save shelf space, nothing gets dusty, and the components are never lost. Cons: you are probably still going to buy games that you will not have time to play. Unless that's just me? I mean, it was such a good sale, and expansions were included! Anyway, my point is, if your physical board game library is taking up too much room, you can now fill up your hard drive instead (likely for a fraction of the cost). Okay, that is enough chatter, on to the game!

Do you have a minute? How about eight? In the time it takes to make a cup of bean juice*, you can play a game of Eight-Minute Empire. Developed by Acram Digital, Eight-Minute Empire is a fast-paced strategy game that uses card-driven area control. It is a port of the tabletop game of the same name, which was designed by Ryan Laukat and published by Red Raven Games. At the time of this writing, the game is available on Steam for \$5.99 (It also has Android and iOS versions, which are probably fun, but that's not my jam. I'm reviewing Tabletop Ubuntu games here).

In Eight-Minute Empire, players take turns selecting one card from six that are displayed. During setup,

players are given an amount of coins based on the total number of players. They then bid to determine turn order. On each player's turn, they select one of six cards from the card row. The first card is free and subsequent cards cost coins. Each card provides a resource and also has an immediate action that the player can use. Resources are worth victory points at the end of the game depending on type. Actions allow the player to recruit more troops, move troops on land, move troops over water, destroy other troops, or build a city where subsequent troops can be recruited. For example, the first card in Figure 1 is free, provides one iron, and lets the player recruit three troops. The fourth card costs

two coins, provides one tree, and lets the player destroy one soldier from another player or build a city.

If a player has the most troops in a region, they control the region and gain 1 victory point. Likewise, if a player controls the most regions on a continent, they control the continent and gain one victory point. For example, the map in Figure 2 shows individual regions controlled by blue, red, yellow, and green. This control is indicated by the border highlight color of each region. Because the red player controls the majority of regions, they control the continent.

The central tension in playing Eight-Minute Empire is balancing



TABLETOP UBUNTU

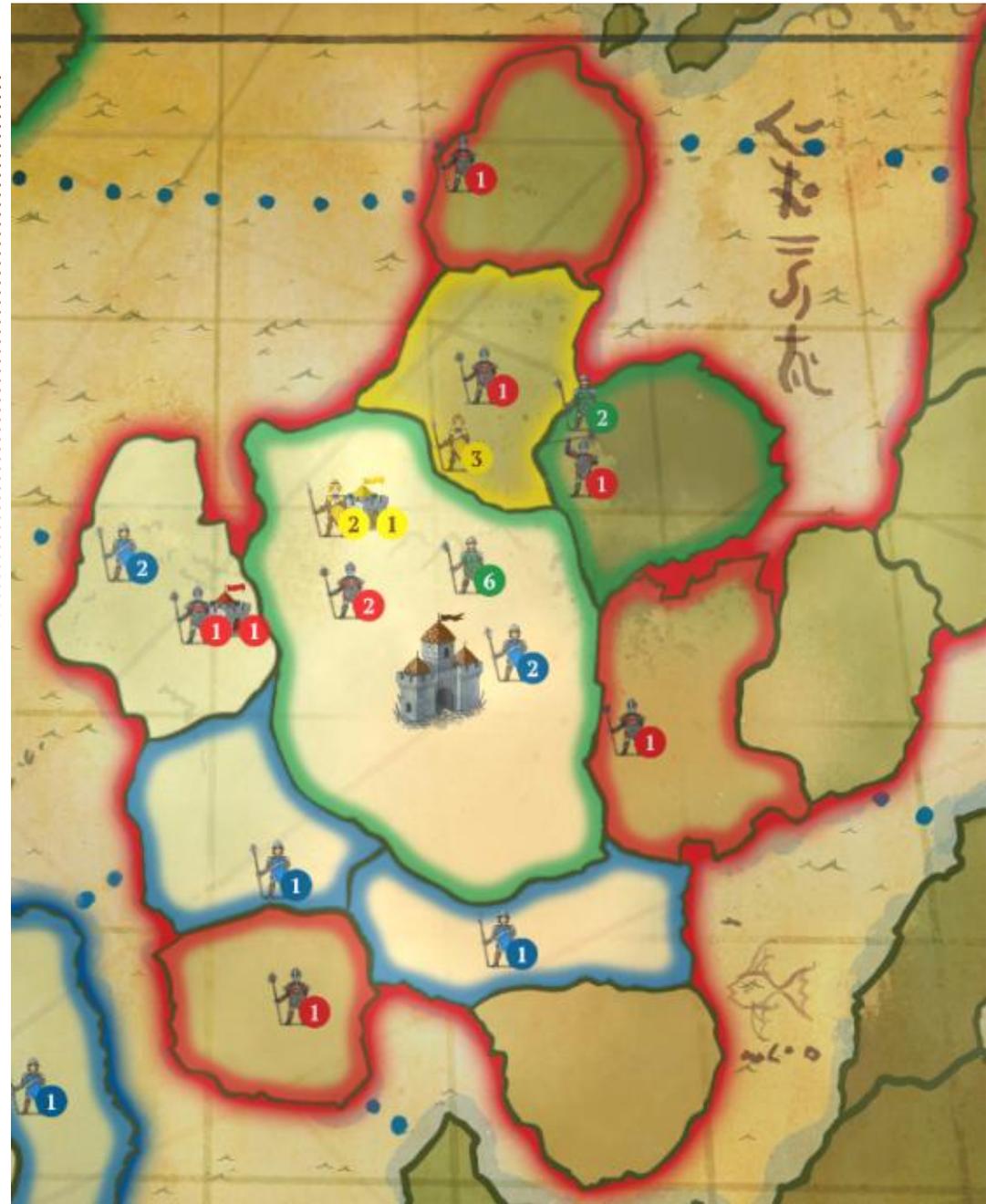
set collection of resources with recruiting and moving troops in order to control regions and continents. The simple rules, fast play, randomization of cards, and different AI levels give it a lot of replay value. Additional maps are also available for purchase.

For me, Eight-Minute Empire finds the sweet spot in terms of speed and content. It is easy for beginners and quite challenging if you turn up the AI. Despite its name, I can often play the game in five minutes. In fact, I played multiple games while writing this review (those screenshots weren't going to take themselves after all). If you are looking for a fast-paced, turn-based strategy game, I highly recommend it!



Josh Hertel is a husband, father, mathematics educator, tabletop gamer, techie, and geek.
<https://twitter.com/hertelj1>

* I'm a regular drip coffee-maker, folks. Not one of those pod monstrosities.





PATRONS

MONTHLY PATRONS

Alex Crabtree
 Alex Popescu
 Andy Garay
 Bill Berninghausen
 Bob C
 Brian Bogdan
 CBinMV
 Darren
 Dennis Mack
 Devin McPherson
 Doug Bruce
 Elizabeth K. Joseph
 Eric Meddleton
 Gary Campbell
 George Smith
 Henry D Mills
 Hugo Sutherland
 Jack
 Jack Hamm
 Jason D. Moss
 Joao Cantinho Lopes
 John Andrews
 John Malon
 John Prigge
 Jonathan Pienaar
 Joseph Gulizia
 JT
 Kevin O'Brien
 Lee Allen
 Leo Paesen

Linda P
 Mark Shuttleworth
 Moss Bliss
 Norman Phillips
 Oscar Rivera
 Paul Anderson
 Paul Readovin
 Rino Ragucci
 Rob Fitzgerald
 Robin Woodburn
 Roy Milner
 Scott Mack
 Sony Varghese
 Taylor Conroy
 Tom Bell
 Tony
 Vincent Jobard
 Volker Bradley
 William von Hagen

SINGLE DONATIONS

2021:
 Floyd Smith
 Dale Reisfield
 Jan Ågren
 Linda Prinsen
 melvyn smith
 Frits van Leeuwen
 Raymond Mccarthy
 Robert Kaspar
 Frank Dinger

Ken Maunder
 Brian Kelly
 János Horváth
 Ronald Eike
 John Porubek
 Hans van Eekelen
 Kees Moerman
 Jon Loveless
 Jim Hibbard

2022:
 Frits van Leeuwen
 Marcus Carlson
 Louis W Adams Jr
 Linda Prinsen
 James Flanagan
 Peter Horlings
 Patrick Martindale
 Frank Dinger
 János Horváth
 Dan Prizner
 TODD WINTERS
 Douglas Brown

The current site was created thanks to **Lucas Westermann** (ex-Command & Conquer) who took on the task of completely rebuilding the site, and scripts, from scratch, in his own time.

The Patreon page is to help pay the domain and hosting fees. The yearly target was quickly reached thanks to those listed on this page. The money also helps with the new mailing list that I set up.

Several people have asked for a PayPal (single donation) option, so I've added a button to the right side of the website

A big thank you to all those who've used Patreon and the PayPal button. It's a HUGE help.



<https://www.patreon.com/fullcirclemagazine>



<https://paypal.me/ronnietucker>



<https://donorbox.org/recurring-monthly-donation>



HOW TO CONTRIBUTE

FULL CIRCLE NEEDS YOU!

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

We are always looking for new articles to include in Full Circle. For help and advice please see the **Official Full Circle Style Guide**: <http://bit.ly/fcmwriting>

Send your **comments** or Linux experiences to: letters@fullcirclemagazine.org
Hardware/software **reviews** should be sent to: reviews@fullcirclemagazine.org
Questions for Q&A should go to: questions@fullcirclemagazine.org
Desktop screens should be emailed to: misc@fullcirclemagazine.org
... or you can visit our **site** via: fullcirclemagazine.org



FCM#187

Deadline:
Sunday 13th Nov. 2022.
Release:
Friday 25th Nov. 2022.



Full Circle Team



Editor - Ronnie Tucker
ronnie@fullcirclemagazine.org

Webmaster -
admin@fullcirclemagazine.org

Editing & Proofreading

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

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

Getting Full Circle Magazine:

For the Full Circle Weekly News:



You can keep up to date with the Weekly News using the RSS feed: <http://fullcirclemagazine.org/feed/podcast>



Or, if you're out and about, you can get the Weekly News via Stitcher Radio (Android/iOS/web):
<http://www.stitcher.com/s?fid=85347&refid=stpr>



and via TuneIn at: <http://tunein.com/radio/Full-Circle-Weekly-News-p855064/>



EPUB Format - Most editions have a link to the epub file on that issue's download page. If you have any problems with the epub file, email: mobile@fullcirclemagazine.org



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

FCM PATREON: <https://www.patreon.com/fullcirclemagazine>