

GNU Planet - Latest News

- [FSF Blogs: January GNU Spotlight with Amin Bandali featuring twelve new GNU releases: GRUB, Units, and more!](#) (2026/02/02 13:20)

- [gettext @ Savannah: GNU gettext 1.0 released](#) (2026/01/29 17:30)

Download from [https:// ... p.gnu.o ... ttext-0.26.tar.gz](https://...p.gnu.o...ttext-0.26.tar.gz) New in this release: Improvements for maintainers and distributors: In a po/ directory, the PO files are now exactly those that the translators submitted or committed in version control, or a translation project's daemon committed on behalf of the translators. They are no longer regularly updated with respect to the POT file in the same directory. The advantage for maintainers is that the maintainer may commit the PO files in version control, without getting lots of modified files shown by "git status", frequent merge conflicts when merging between branches, a voluminous version control history. The advantage for distributors is that the role of files in a release tarball are clearer: The PO files are source code, whereas the POT file and the *.gmo files are generated files. ATTENTION translators! Translators who work directly on a package's source code (without going through a translation project) now need to run "msginit" before starting work on a PO file. A new program 'po-fetch' is provided, that fetches the translated PO files from a translation project's site on the internet, and updates the LINGUAS file accordingly. In a po/ directory, a new script 'fetch-po' is now added by 'gettextize'. It provides the standard interface for fetching the translated PO files. It typically either invokes the 'po-fetch' program or does nothing. Improvements for translators: msginit: When the PO file already exists, 'msginit' now updates it w.r.t. the POT file, like 'msgmerge' would do. Previously, 'msginit' failed with an error message in this situation. Pretranslation: Two new programs, 'msgpre' and 'spit', are provided, that implement machine translation through a locally installed Large Language Model (LLM). 'msgpre' applies to an entire PO file, 'spit' to a single message. The documentation has a new chapter "Pretranslation". Improvements for maintainers: xgettext: The refactoring suggestion when a translatable string contains an URL or email address can now be inhibited through a command-line option '--no-check=url' or '--no-check=email', or through a comment in the source code of the form

```
/* xgettext: no-url-check */ or      /* xgettext: no-email-check */
```

Programming languages support: OCaml: xgettext now supports OCaml. 'msgfmt -c' now verifies the syntax of translations of OCaml format strings. A new example 'hello-ocaml' has been added. Rust: xgettext now recognizes 'gettextrs::gettext' invocations, like 'gettext' invocations. libgettextpo library: The function 'po_message_get_format' now supports distinguishing whether a negative format string mark, such as 'no-c-format', is set or not. The new functions po_message_has_workflow_flag, po_message_set_workflow_flag, po_message_workflow_flags_iterator, po_flag_next, po_flag_iterator_free can be used to manipulate or inspect the workflow flags of a message. The new functions po_message_has_sticky_flag, po_message_set_sticky_flag, po_message_sticky_flags_iterator, po_flag_next, po_flag_iterator_free can be used to manipulate or inspect the sticky flags of a message. Emacs PO mode: Restore syntax highlighting in Emacs version 30 or newer.

- [parallel @ Savannah: GNU Parallel 20260122 \('Maduro'\) released \[stable\]](#) (2026/01/27 23:44)

GNU Parallel 20260122 ('Maduro') has been released. It is available for download at: lbrary://@GnuParallel:4 Quote of the month: 64コアで、64並列でsimulationを回してtopコマンドで状況を見るのは心地よい。簡単に並列処理を実現できるGNU parallelコマンドは素晴らしい□ -- Daisuke lizuka @diizuka@twitter New in this release: No new features. Bug fixes. GNU Parallel - For people who live life in the parallel lane. If you like GNU Parallel record a video testimonial: Say who you are, what you use GNU Parallel for, how it helps you, and what you like most about it. Include a

command that uses GNU Parallel if you feel like it. About GNU Parallel GNU Parallel is a shell tool for executing jobs in parallel using one or more computers. A job can be a single command or a small script that has to be run for each of the lines in the input. The typical input is a list of files, a list of hosts, a list of users, a list of URLs, or a list of tables. A job can also be a command that reads from a pipe. GNU Parallel can then split the input and pipe it into commands in parallel. If you use xargs and tee today you will find GNU Parallel very easy to use as GNU Parallel is written to have the same options as xargs. If you write loops in shell, you will find GNU Parallel may be able to replace most of the loops and make them run faster by running several jobs in parallel. GNU Parallel can even replace nested loops. GNU Parallel makes sure output from the commands is the same output as you would get had you run the commands sequentially. This makes it possible to use output from GNU Parallel as input for other programs. For example you can run this to convert all jpeg files into png and gif files and have a progress bar: `parallel --bar convert {1} {1.}.{2} ::: *.jpg ::: png gif` Or you can generate big, medium, and small thumbnails of all jpeg files in sub dirs: `find . -name '*.jpg' | parallel convert -geometry {2} {1} {1//}/thumb{2}_{1/} ::: - ::: 50 100 200` You can find more about GNU Parallel at: <http://www.gnu.org/s/parallel/> You can install GNU Parallel in just 10 seconds with: `$ (wget -O - pi.dk/3 || lynx -source pi.dk/3 || curl pi.dk/3/ || \ fetch -o - http://pi.dk/3) > install.sh $ sha1sum install.sh | grep c555f616391c6f7c28bf938044f4ec50 12345678 c555f616 391c6f7c 28bf9380 44f4ec50 $ md5sum install.sh | grep 707275363428aa9e9a136b9a7296dfe4 70727536 3428aa9e 9a136b9a 7296dfe4 $ sha512sum install.sh | grep b24bfe249695e0236f6bc7de85828fe1f08f4259 83320d89 f56698ec 77454856 895edc3e aa16feab 2757966e 5092ef2d 661b8b45 b24bfe24 9695e023 6f6bc7de 85828fe1 f08f4259 6ce5480a 5e1571b2 8b722f21 $ bash install.sh` Watch the intro video on <http://www.youtube.com/watch?v=L284C9FF2488BC6D1> Walk through the tutorial (man parallel_tutorial). Your command line will love you for it. When using programs that use GNU Parallel to process data for publication please cite: O. Tange (2018): GNU Parallel 2018, March 2018, <https://doi.org/10.1101/284601> ... 81/zenodo.1146014. If you like GNU Parallel: Give a demo at your local user group/team/colleagues Post the intro videos on Reddit/Diaspora*/forums/blogs/Identi.ca/Google+/Twitter/Facebook/Linkedin/ mailing lists Get the merchandise <https://gnuparall.org/merch/> igns/gnu-parallel Request or write a review for your favourite blog or magazine Request or build a package for your favourite distribution (if it is not already there) Invite me for your next conference If you use programs that use GNU Parallel for research: Please cite GNU Parallel in you publications (use --citation) If GNU Parallel saves you money: (Have your company) donate to FSF <https://my.fsf.org/donate/> About GNU SQL GNU sql aims to give a simple, unified interface for accessing databases through all the different databases' command line clients. So far the focus has been on giving a common way to specify login information (protocol, username, password, hostname, and port number), size (database and table size), and running queries. The database is addressed using a DBURL. If commands are left out you will get that database's interactive shell. When using GNU SQL for a publication please cite: O. Tange (2011): GNU SQL - A Command Line Tool for Accessing Different Databases Using DBURLs, ;login: The USENIX Magazine, April 2011:29-32. About GNU Niceload GNU niceload slows down a program when the computer load average (or other system activity) is above a certain limit. When the limit is reached the program will be suspended for some time. If the limit is a soft limit the program will be allowed to run for short amounts of time before being suspended again. If the limit is a hard limit the program will only be allowed to run when the system is below the limit.

- [FSF Blogs: GPL-compliant reasonable legal notices and author attributions](#) (2026/01/27 19:20)
- [FSF Events: Free Software Directory meeting on IRC: Friday, February 6, starting at 12:00 EST \(17:00 UTC\)](#) (2026/01/26 19:25)
Join the FSF and friends on Friday, February 6 from 12:00 to 15:00 EST (17:00 to 20:00 UTC) to help improve the Free Software Directory.
- [FSF Events: Free Software Directory meeting on IRC: Friday, January 30, starting at 12:00 EST \(17:00 UTC\)](#) (2026/01/26 19:18)

Join the FSF and friends on Friday, January 30 from 12:00 to 15:00 EST (17:00 to 20:00 UTC) to help improve the Free Software Directory.

- **GNU Guix: GNU Guix 1.5.0 released** (2026/01/23 14:00)

We are pleased to announce the release of GNU Guix version 1.5.0! The release comes with ISO-9660 installation images, virtual machine images, and with tarballs to install the package manager on top of your GNU/Linux distro, either from source or from binaries—check out the download page. Guix users can update by running `guix pull`. It's been 3 years since the previous release. That's a lot of time, reflecting both the fact that, as a rolling release, users continuously get new features and update by running `guix pull`; but it also shows a lack of processes, something that we had to address before another release could be made. During that time, Guix received about 71,338 commits by 744 people, which include many new features; the project also got a new decision-making process, migrated to Codeberg and started a fundraising campaign. That's just the surface among so many great changes, so keep reading! Illustration by Luis Felipe, published under CC-BY-SA 4.0. This post provides highlights for all the hard work that went into this release. There's a lot to talk about so make yourself comfortable, relax, and enjoy. Guix ecosystem To start with, the Guix ecosystem has seen many exciting developments to the way we collaborate and make decisions! Firstly, the project adopted with unanimity a new consensus-based decision making process. This process fills a need to be able to gather consensus on significant changes to the project, something that was getting very complicated with the growing number of contributors to the project. Now, the process provides a clear framework for any contributor to propose and implement important changes. These can be submitted as Guix Consensus Documents (GCDs), each GCD goes through the multiple steps of consensus decision making before being accepted or withdrawn. Secondly, using this process, the project was able to collectively migrate to Codeberg. This means that all repositories, and bug trackers are now at the same place on Codeberg and that contributions are now made with pull requests instead of patch series. Thirdly, a new release process was adopted to bring an annual release cycle to Guix. This release is the first to follow this process, with hopefully many others to come! Lastly, a "Planet" website for Guix is now available at <https://planet.guix.gnu.org>. It aggregates blogs from various Guix hackers and contributors to bring you the latest and greatest in Guix news. Stronger distribution Three years is a long time for free and open source software! Enough time for 12,525 new packages and 29,932 package updates to the Guix repository. Here are the best highlights: To start, KDE Plasma 6.5 is now available with the new `plasma-desktop-service-type`! Continuing on desktops; GNOME has been updated from version 42 to 46 and now uses Wayland by default. The `gnome-desktop-service-type` was made more modular to better customize the default set of GNOME applications. Guix System is now using version 1.0 of the GNU Shepherd, which now supports timed services, `kexec` reboot and has new services for system logs and log rotation which are now used by Guix System instead of `Rottlog` and `syslogd`. There are around 40 new system services to choose from, including `Forgejo Runner`, `RabbitMQ`, `iwd`, and `dhcpcd` to name a few. `setuid`-programs has been replaced with `privileged`-programs in operating-system definitions to support giving specific Linux capabilities. Additionally, the `nss-certs` package is now included in `%base-packages`. More than 12,500 packages were added, keeping Guix in the top-ten biggest distributions according to Repology! Among the many noteworthy updates, we now have GCC 15.2.0, Emacs 30.2, Icecat and Librewolf 140, LLVM 21.1.8 and Linux-libre 6.17.12. Team activity In the last release, we introduced structured cooperation using teams. There are now 50 teams distributing the many aspects of the distribution. We have per-language teams like `python`, `rust` and `zig` ensuring updates for packages and build systems as well as thematic teams like `electronics`, `hpc` and `bioinformatics` working on specific application domains. Here are what some of these teams have been up to: The HPC team published their annual activity report 2024, showing the exciting developments of Guix in High-Performance Computing. The electronics team is maintaining free software based Electronic Design Automation (EDA) packages to cover the needs of professionals and hobbyists in the domain

with tools such as KiCad, LibrePCB, Xschem, Qucs-S and Ringdove's EDA, as well as Verilog, SystemVerilog and VHDL compilers and a toolchain for programmable designs on GateMate FPGAs. They are also collaborating with the Free Silicon Foundation (F-Si) to push free software in the EDA space! The science team has been able to add a myriad of Astronomy related packages, accompanied by the Python team bringing the move to the new pyproject.toml-based build system as well as the NumPy's 2 update. Finally, the rust team created a new packaging model to efficiently package rust crates, and was able to migrate the Rust collection, 150+ packages with 3,600+ libraries, in just under two weeks; making the Rust packaging process much easier for everyone. Full source bootstraps Full-source bootstraps of the Zig and Mono compilers are now available, and the existing bootstrap of Guix has been reduced once again! Full-source bootstraps are Guix's solution to the trusting trust problem: compilers are usually compiled by themselves, so how can you build a compiler without trusting an existing binary? Read these posts to learn more about this fascinating problem: The Full-Source Bootstrap: Building from source all the way down Zig reproduced without binaries Restoring Zig bootstrap chain in Guix (in Traditional Chinese) Adding a fully bootstrapped Mono Improved CLIThe guix graph command has new backends for GraphML and CycloneDX's JSON, meaning Guix can now be used to generate complete Software Bill of Material (SBOM) down to the first bootstrap binary! guix shell containers have been improved with a --nesting option to use Guix within the container and a --emulate-fhs option that can be used to run software expecting a Filesystem Hierarchy Standard (FHS) compliant filesystem. The guix pack command also received new backends to create RPM packages and AppImages that can be used to publish your Guix packages to non-Guix users. Lastly, a new guix locate command is now available to find which packages provide a given file. Security improvements It is now possible to run the Guix daemon without root privileges, reducing the impact of privilege escalation vulnerabilities. This "rootless" mode is now the default when installing Guix 1.5.0 on distros other than Guix System; on Guix System, it currently has to be explicitly enabled by setting (privileged? #f) in guix-configuration. Existing installation on distros other than Guix System can also be migrated to "rootless". This is possible thanks to the user namespaces. It might be possible that on your system, the user namespaces are not allowed for guix due to the lack of an AppArmor profile. Because of that, we've also included AppArmor profiles that are installed by default on foreign systems. Finally, the Guix daemon received security fixes for CVE-2024-27297, CVE-2024-52867, CVE-2025-46415, CVE-2025-46416 and CVE-2025-59378. Widened architecture support Release tarballs are now available for the RISC-V 64-bit architecture (riscv64-linux). The x86_64 architecture saw some development as well, with the experimental support of the GNU's Hurd kernel (x86_64-gnu), aiming to be another significant step in the adoption and development of the Hurd. Overall support for the Hurd was greatly improved, it is now an option in the installer, childhurd can be automatically created with a system service and it can even run on a Thinkpad X60! Fundraising campaign Surprisingly, making a completely free software distribution does not come for free! The Guix project needs your help to pay the infrastructure costs of build farms, web servers and QA tools that are essential to making this release happen. If you appreciate all of the work that is done to bring you this one-of-a-kind distro: please donate to the Guix Foundation! Acknowledgments For the release, thanks to all the release team members: Rutherford, Rodion's Goritskov, Efraim's Flashner, and NoÃ©'s Lopez. Thanks as well to the release helpers: Andreas's Enge, Mothacehe, Dariqq and Ludovic's CourtÃ's. For creating the release process, thanks to Steve's George. For their Guix contributions, thanks to the 744 wonderful people who contributed and whose names we don't list here (it would be a bit long). They can be listed with git log --oneline v1.4.0..v1.5.0 --format="%an" | sort -u. Every commit counts and is always appreciated! About GNU Guix GNU Guix is a transactional package manager and an advanced distribution of the GNU system that respects user freedom. Guix can be used on top of any system running the Hurd or the Linux kernel, or it can be used as a standalone operating system distribution for i686, x86_64, ARMv7, AArch64, RISC-V and POWER9 machines. In addition to standard package

management features, Guix supports transactional upgrades and roll-backs, unprivileged package management, per-user profiles, and garbage collection. When used as a standalone GNU/Linux distribution, Guix offers a declarative, stateless approach to operating system configuration management. Guix is highly customizable and hackable through Guile programming interfaces and extensions to the Scheme language.

- [GNU Guix: Meet Guix at FOSDEM](#) (2026/01/22 13:00)

It's that time of the year again: next week is FOSDEM time! As in previous years, many Guix people will be in Brussels. Right after FOSDEM, about sixty of us will gather on February 2–3 for the Guix Days! First things first: Guix presence at FOSDEM. On Saturday, January 31st: In Name resolution in package management systems — A reproducibility perspective, Gábor Boskovits will look at how several package managers refer to packages and how this affects reproducibility. Simon Tournier will give a lightning talk in the Bioinformatics track, Guixifying workflow management system: past, present, maybe future?, discussing the Guix Workflow Language (GWL), the ccwl, and ravanan. On Sunday, February 1st, the Declarative & Minimalistic Computing track will once again be a Guile & Guix lair. The whole track is amazing, with top-notch talks and speakers; particularly relevant to Guix and Guile hackers are the following: Sergio Pastor Pérez will give a talk entitled BLUE — A generic build system crafted entirely in Guile. There are clear connections with Guix but BLUE could well become the new standard build system for Guile developers! In Modern Development Tools and Practices for GNU Guile Andrew Tropin will talk about live programming at the REPL, in particular with the Ares/Arei interactive development environment. On the same theme, Jessica Talon of Spritely will talk about Guile development outside of Emacs—addressing a real need of Guile and Guix outreach efforts. In Lisp is clay: the power of composable DSLs, the inimitable Christine Lemmer-Webber will talk about this Lisp foundation that makes Spritely and Guix so powerful. David Thompson (also of Spritely!) will talk about Functional reactive programming with propagators, which sounds like an exciting topic for any functional programming person and any programmer who's worked on user interfaces and other kinds of “reactive” programs. Simon Josefsson will talk about Guix Container Images — and what you can do with them, showing how to add Guix container images to registries and how to use them in continuous integration and continuous delivery (CI/CD) pipelines. You can have more Guix bliss on Sunday afternoon: In a talk entitled Package management in the hands of users: dream and reality, I (Ludovic Courtès) will reflect on successes and failures bringing package management to HPC supercomputer users. Samuel Thibault will share Updates on GNU/Hurd progress, which includes Guix goodness and good news for a practical empowering operating system. Guix Days will take place on Monday and Tuesday right after FOSDEM, at our usual venue. Sixty people already registered, which is our maximum capacity—don't just show up and hope for the best. As always, this will be unconference style: we'll make the program as we go, discussing hot topics such as the crowdfunding campaign, an update on Guix Foundation, processes and governance, as well as the more technical topics we're fond of. This year marks the tenth anniversary of the Guile/Declarative & Minimalistic Computing track, and the eighth Guix Days. Shout out to our friends Pjotr Prins and Manolis Ragkousis, who have spearheaded the two events during all these years, and to all the volunteers who helped them on the way! This yearly Brussels gathering has been instrumental in building, shaping, and strengthening our community; to those who can be present, it's the energizing and refreshing moment of the year. To Pjotr, to Manolis: thank you! Guix Days graphics are copyright © 2024 Luis Felipe López Acevedo, under CC-BY-SA 4.0, available from Luis' Guix graphics repository. Picture of “Au Bon Vieux Temps” sign © 2025 Ludovic Courtès, under CC-BY-SA 4.0.

- [FSF Blogs: You came through for free software!](#) (2026/01/20 21:50)
- [FSF Events: Free Software Directory meeting on IRC: Friday, January 23, starting at 12:00 EST \(17:00 UTC\)](#) (2026/01/20 15:33)

Join the FSF and friends on Friday, January 23 from 12:00 to 15:00 EST (17:00 to 20:00 UTC) to help improve the Free Software Directory.

- [FSF Events: Free/libre software and our freedom: Our shield against many digital injustices.](#) (2026/01/15 21:00)

Join FSF founder Richard M. Stallman for his talk at the Georgia Institute of Technology.

- [FSF Events: Meet up with FSF staff and Librephone developer at FOSDEM 2026](#) (2026/01/15 17:31)

FOSDEM is a free event for software developers to meet, share ideas and collaborate. Every year, thousands of developers of free and open source software from all over the world gather at the event in Brussels. You don't need to register. Just turn up and join in!

- [GNU Taler news: Critical analysis of digital euro published](#) (2026/01/13 23:00)

The Springer journal "Digital Finance" has recently published "The proposed design of the digital euro: A critical analysis" by Mikolai Gütschow and Bernd Lucke. They describe serious flaws in the digital euro design as proposed by the European Commission and propose GNU Taler as an alternative technology for a potential CBDC with tangible benefits for Europeans.

- [FSF Blogs: Our members help secure the future of a free society](#) (2026/01/13 21:47)

- [Simon Josefsson: Debian Libre Live 13.3.0 is released!](#) (2026/01/13 13:53)

Following up on my initial announcement about Debian Libre Live I am happy to report on continued progress and the release of Debian Libre Live version 13.3.0. Since both this and the previous 13.2.0 release are based on the stable Debian trixie release, there really isn't a lot of major changes but instead incremental minor progress for the installation process. Repeated installations has a tendency to reveal bugs, and we have resolved the apt sources list confusion for Calamares-based installations and a couple of other nits. This release is more polished and we are not aware of any known remaining issues with them (unlike for earlier versions which were released with known problems), although we conservatively regard the project as still in beta. A Debian Libre Live logo is needed before marking this as stable, any graphically talented takers? (Please base it on the Debian SVG upstream logo image.) We provide GNOME, KDE, and XFCE desktop images, as well as text-only "standard" image, which match the regular Debian Live images with non-free software on them, but also provide a "slim" variant which is merely 750MB compared to the 1.9GB "standard" image. The slim image can still start a debian installer, and can still boot into a minimal live text-based system. The GNOME, KDE and XFCE desktop images feature the Calamares installer, and we have performed testing on a variety of machines. The standard and slim images does not have a installer from the running live system, but all images support a boot menu entry to start the installer. With this release we also extend our arm64 support to two tested platforms. The current list of successfully installed and supported systems now include the following hardware: Desktop ADLINK Ampere Altra Developer Platform arm64 Neoverse N1 Desktop MSI Z790-P WIFI PRO i9-14900K Dasharo Laptop Framework 13 AMD AI 9 HX 370 Laptop Lenovo X201 i7-620M Laptop NovaCustom NV56 Intel Ultra 7 155H i915 Dasharo Server Dell PowerEdge R630 2xE2680v4 Server/Router Protectli VP2440 Server Supermicro MegaDC ARS-110M-NR Ampere Altra Max 128 core 2x25GBe This is a very limited set of machines, but the diversity in CPUs and architecture should hopefully reflect well on a wide variety of commonly available machines. Several of these machines are crippled (usually GPU or WIFI) without adding non-free software, complain at your hardware vendor and adapt your use-cases and future purchases. The images are as follows, with SHA256SUM checksums and GnuPG signature on the 13.3.0 release page. Amd64 GNOME debian-live-13.3.0-amd64-libre-gnome.iso Amd64 KDE debian-live-13.3.0-amd64-libre-kde.iso Amd64 XFCE debian-live-13.3.0-amd64-libre-xfce.iso Amd64 Standard debian-live-13.3.0-amd64-libre-standard.iso Amd64 Slim debian-live-13.3.0-amd64-libre-slim.iso Arm64 GNOME debian-live-13.3.0-arm64-libre-gnome.iso Arm64 KDE debian-live-13.3.0-arm64-libre-kde.iso Arm64 XFCE debian-live-13.3.0-arm64-libre-xfce.iso Arm64 Standard debian-live-13.3.0-arm64-libre-standard.iso Arm64 Slim debian-live-13.3.0-arm64-libre-slim.iso Curious how the images were made? Fear not, for the Debian Libre Live project README has documentation, the run.sh

script is short and the .gitlab-ci.yml CI/CD Pipeline definition file brief. Happy Libre OS hacking!

- [FSF Blogs: Turning freedom values into freedom practice with the FSF tech team](#) (2026/01/09 16:29)

Ian Kelling, FSF senior systems administrator, and also our president, outlines the complex steps the FSF tech team goes through to ensure the software we use is free. The tech team — currently just two people — is vital to our collective work for software freedom, which itself helps guarantee many of our other basic freedoms. We depend on people just like you to support our work: we have an associate membership drive to welcome 100 new members by January 16. Please join the FSF and help keep this work going.

- [www-zh-cn @ Savannah: Summary 2025](#) (2026/01/07 07:46)

Dear GNU CCT: Here is summary of GNU website from GNU: 2025 had a splash of activity; a few teams who were dormant in 2024 made a notable progress, in terms of new translations or updating the existing ones. General Statistics About 2/3 new translations were made by the Chinese (zh-cn) team this year; then the Greek and Albanian teams followed. The Polish and Dutch teams considerably reduced the amount of their outdated translations. Currently, the total amount of translations is over 3400; the overall percentage of outdated translations was about 5% lower than in 2024. The table below shows the number and size of newly translated articles in important directories and typical number of outdated GNUified translations throughout the year.

team	new	outdated	el	eo *	es	fr	ml	nl	pl	ru	sq	tr	zh-cn	total
	5 (81.9Ki)	19 (41%)	4 (34.2Ki)	1 (33.6Ki)	2.0 (0.9%)	1 (7.9Ki)	1 (11.2Ki)	0 (0.0Ki)	1 (9.9Ki)	2 (26.9Ki)	4 (73.9Ki)	4 (52.5Ki)	39 (797.9Ki)	62 (1130Ki)
	2.0 (0.9%)	2.0 (0.9%)	2.0 (0.9%)	1.4 (0.4%)	2.0 (0.9%)	1.4 (0.4%)	2.0 (0.9%)	0.0 (0.0%)	1.4 (0.4%)	2.2 (0.7%)	9.0 (10%)	1.9 (1.4%)	1.3 (0.5%)	2.0 (0.9%)

The Esperanto translation was installed by GNU Translation Managers without establishing a new team. For the reference: 2 new articles were added, amounting to 27Ki (vs. 4 articles and 44Ki in 2024); the number of commits (about 500 changes in approximately 90 English files) was almost twice as many as in 2024. Orphaned Teams, New and Reformed Teams No teams were orphaned, and no new teams were established. Greek and Dutch teams changed their status to active without a reorganization. A volunteer requested creating the Georgian team, with no further progress. Thank you for your contribution. I wish you all a happy and successful 2026. Happy hacking. wxie

- [GNU Taler news: P15 CoNetWorking Space accepts GNU Taler payments in eCHF](#) (2026/01/05 23:00)

The P15 CoNetWorking Space in Biel/Bienne right next to the train station (and the BFH) is the first shop to accept GNU Taler payments in Swiss francs (eCHF) issued by Taler Operations AG and thus the first merchant accepting Taler payments in fiat currency. P15 is a great space to network, so go check it out!

- [GNU Taler news: GNU Taler presented at 39C3](#) (2026/01/04 23:00)

Mikolai Gütschow and signum gave a talk at the 39th Chaos Communication Congress (39C3) in Hamburg, Germany, where they reported on their good experiences with offering GNU Taler as a local payment system at LugCamp 2024 and Datenspuren 2024 and 2025.

- [Amin Bandali: The People of Emacs](#) (2025/12/31 14:09)

GNU Emacs has been my primary computing environment of choice for over a decade. Emacs has enabled me to perform a wide array of tasks involving human and computer languages, such as reading and writing notes, emails, chats, programs, and more, all in a cohesive and consistent environment that I can tailor exactly to my needs and liking. Coming from a Vim background, I started my Emacs journey trying some

configuration frameworks that provided vi-like key bindings, and after a few Emacs bankruptcies, ended up with my current homegrown configuration that I wrote from scratch gradually over the last 7 years, with inspiration from the configurations of some folks who shared theirs publicly. Though my configuration has been mostly stable for a few years now and I consciously keep the number of external packages I use very small, I occasionally add small bits and pieces to my configuration when I'm inspired after learning about a neat feature or package on the blogs aggregated on Planet Emacslife, the messages sent to the Emacs mailing lists, or the videos from the annual EmacsConf conference. I like getting a glimpse of other people's worlds through the lens of their creative works such as writings, be it prose or Emacs Lisp. That's only possible when people share freely, free as in freedom. I'm thankful to Richard Stallman for his foresight to imbue GNU Emacs with that freedom from the very beginning and for his lifelong fight for computer user freedom, and to the many other folks who have joined the free software movement since then and have fought the good fight. I've been inspired and encouraged by many awesome Emacs people through the years. People like Corwin Brust with his joyful creative energy around Emacs and the road to software freedom, Sacha Chua and her philosophy of leading a life of learning, sharing, and scaling, Gopar and his enthusiasm for Emacs and its intersection with the Python world, folks like Protesilaos Stavroutou and Greg Falarow who discovered Emacs initially as non-programmers yet were enamoured by its embodiment of software freedom in practice and went on to integrate it into their everyday lives, and shoshin of the Cicadas cooperative at the intersection of humanity and technology sharing his passion for the human element and community by developing and contributing input methods for his ancestral language of Lakota to GNU Emacs. I'm deeply inspired by each of these wonderful people, and grateful for having known them and for each of their unique perspectives and life stories with which they have enriched my experience in Emacs and the free software world. As wonderful and impactful as Emacs has been in the lives of the many who have come to know it throughout the decades that it's been around, it would not have become what it has been, what it is today, and what it may become in the future without its community of passionate users and contributors. The People of Emacs are all of us. Here's to many more of us, enjoying many more years of Emacs and software freedom together even if spread far apart. Take care, and so long for now. Inspired by the Emacs Carnival theme for this month, The People of Emacs. Thanks to George Jones for hosting.

- [FSF News: Eko K. A. Owen joins the FSF board as the union staff pick](#) (2025/12/29 22:45)
BOSTON, Massachusetts, USA (December 29, 2025) – The Free Software Foundation (FSF) announced today that Eko K. A. Owen will follow in Ian Kelling's footsteps by becoming the second union staff-elected board member on the organization's board of directors.
- [Jose E. Marchesi: Gemini capsule capsule.jemarch.net](#) (2025/12/29 19:00)
Recently I have been using Gemini, a sort of a modernized Gopher, more and more, and have finally decided to create and maintain my own Gemini capsule, that you can find at gemini://capsule.jemarch.net. The plan, moving forward, is to publish basically the same contents in both www and gemini versions of this homepage. Salud!
- [Jose E. Marchesi: Gemini capsule jemarch.srht.site](#) (2025/12/29 19:00)
Recently I have been using Gemini, a sort of a modernized Gopher, more and more, and have finally decided to create and maintain my own Gemini capsule, that you can find at gemini://jemarch.srht.site. The plan, moving forward, is to publish basically the same contents in both www and gemini versions of this homepage. Salud!
- [FSF News: Free Software Foundation receives historic private donations](#) (2025/12/24 22:45)
Boston, Massachusetts, USA (Wednesday, December 24, 2025) -- The Free Software Foundation (FSF) today announced it received two major

contributions totaling around \$900,000 USD.

- [parallel @ Savannah: GNU Parallel 20251222 \('Bondi'\) released \[stable\]](#) (2025/12/24 01:34)

GNU Parallel 20251222 ('Bondi') has been released. It is available for download at: [lbry://@GnuParallel](#):4 Quote of the month: Used? gnu parallel is my new favorite toy -- Eytan Adar [@eytan.adar.prof](#) New in this release: No new features. Bug fixes. GNU Parallel - For people who live life in the parallel lane. If you like GNU Parallel record a video testimonial: Say who you are, what you use GNU Parallel for, how it helps you, and what you like most about it. Include a command that uses GNU Parallel if you feel like it. About GNU Parallel GNU Parallel is a shell tool for executing jobs in parallel using one or more computers. A job can be a single command or a small script that has to be run for each of the lines in the input. The typical input is a list of files, a list of hosts, a list of users, a list of URLs, or a list of tables. A job can also be a command that reads from a pipe. GNU Parallel can then split the input and pipe it into commands in parallel. If you use xargs and tee today you will find GNU Parallel very easy to use as GNU Parallel is written to have the same options as xargs. If you write loops in shell, you will find GNU Parallel may be able to replace most of the loops and make them run faster by running several jobs in parallel. GNU Parallel can even replace nested loops. GNU Parallel makes sure output from the commands is the same output as you would get had you run the commands sequentially. This makes it possible to use output from GNU Parallel as input for other programs. For example you can run this to convert all jpeg files into png and gif files and have a progress bar: `parallel --bar convert {1} {1.}.{2} ::: *.jpg ::: png gif` Or you can generate big, medium, and small thumbnails of all jpeg files in sub dirs: `find . -name '*.jpg' | parallel convert -geometry {2} {1} {1//}/thumb{2}_{1/} ::: - ::: 50 100 200` You can find more about GNU Parallel at: <http://www.gnu.org/s/parallel/> You can install GNU Parallel in just 10 seconds with: `$ (wget -O - pi.dk/3 || lynx -source pi.dk/3 || curl pi.dk/3 || \ fetch -o - http://pi.dk/3) > install.sh $ sha1sum install.sh | grep c555f616391c6f7c28bf938044f4ec50 12345678 c555f616391c6f7c 28bf9380 44f4ec50 $ md5sum install.sh | grep 707275363428aa9e9a136b9a7296dfe4 70727536 3428aa9e 9a136b9a 7296dfe4 $ sha512sum install.sh | grep b24bfe249695e0236f6bc7de85828fe1f08f4259 83320d89 f56698ec 77454856 895edc3e aa16feab 2757966e5092ef2d 661b8b45 b24bfe24 9695e023 6f6bc7de 85828fe1 f08f4259 6ce5480a 5e1571b2 8b722f21 $ bash install.sh` Watch the intro video on <http://www.youtube.com/watch?v=L284C9FF2488BC6D1> Walk through the tutorial (`man parallel_tutorial`). Your command line will love you for it. When using programs that use GNU Parallel to process data for publication please cite: O. Tange (2018): GNU Parallel 2018, March 2018, <https://doi.org/10.1111/zenodo.1146014>. If you like GNU Parallel: Give a demo at your local user group/team/colleagues Post the intro videos on Reddit/Diaspora*/forums/blogs/ Identi.ca/Google+/Twitter/Facebook/LinkedIn/ mailing lists Get the merchandise <https://gnuparall.org/igns/gnu-parallel> Request or write a review for your favourite blog or magazine Request or build a package for your favourite distribution (if it is not already there) Invite me for your next conference If you use programs that use GNU Parallel for research: Please cite GNU Parallel in you publications (use --citation) If GNU Parallel saves you money: (Have your company) donate to FSF <https://my.fsf.org/donate/> About GNU SQL GNU sql aims to give a simple, unified interface for accessing databases through all the different databases' command line clients. So far the focus has been on giving a common way to specify login information (protocol, username, password, hostname, and port number), size (database and table size), and running queries. The database is addressed using a DBURL. If commands are left out you will get that database's interactive shell. When using GNU SQL for a publication please cite: O. Tange (2011): GNU SQL - A Command Line Tool for Accessing Different Databases Using DBURLs, ;login: The USENIX Magazine, April 2011:29-32. About GNU Niceload GNU niceload slows down a program when the computer load average (or other system activity) is above a certain limit. When the limit is reached the program will be suspended for some time. If the limit is a soft limit the program will be allowed to run for short amounts of time before being suspended again. If the limit is a hard limit the program will

only be allowed to run when the system is below the limit.

- [GNUnet News: GNUnet 0.26.2](#) (2025/12/22 23:00)

GNUnet 0.26.2 This is a bugfix release for gnunet 0.26.1. It fixes some regressions and minor bugs. Links Tarball download (tar.gz) (signature) You can find a detailed list of changes in the git log and noteworthy changes in the NEWS file. The GPG key used to sign is: D842 3BCB 326C 7907 0339 29C7 939E 6BE1 E29F C3CC Note that due to mirror synchronization, not all links may be functional early after the release. For direct access try <https://ftp.gnu.org/gnu/gnunet/>

- [GNU Taler news: GNU Taler 1.3 released](#) (2025/12/22 23:00)

We are happy to announce the release of GNU Taler v1.3.

- [health @ Savannah: GNU Health HIS 5.0.5 - Ready for Python 3.14](#) (2025/12/16 12:46)

Dear community I am happy to announce that since patchset 5.0.5, GNU Health Information system is ready for Python 3.14. All GNU Health HIS packages have been updated so they allow Python 3.14. The GNU Health GTK client and GNU Health control installation and instance manager have also been upgraded. Operating systems like Void Linux already upgraded to Python 3.14, and now you can enjoy this Python release in GNUHealth. As usual, backup your database, local modules/packages and GNU Health filesystem before upgrading, and report any issue you may find. For detailed information and changes, please visit the GNU Health HIS repositories at Codeberg (<https://codeberg. ... org/gnuhealth/his>) You can download the source code directly from GNU.org and the packages from PyPI. Happy hacking ♥

- [health @ Savannah: GNU Health HIS server 5.0.4 patchset bundle released](#) (2025/12/13 20:53)

Dear community The patchset 5.0.4 of GNU Health Information System is out! This is a small patch related to the calendar package. If you use the vanilla / standard installation, you can update the server and the dependencies from the gnuhealth control center (<https://docs.gnuh ... ontrolcenter.html>) Backup As usual, before you upgrade your instance, make sure you have made a backup of your DB instance and "attach" resource ! Happy hacking ♥ Changelog for 5.0.4 5bbd80c38: health_calendar: Fix issue #164 - AttributeError when creating work schedule Tue Dec 2 11:13:26 2025 +0000 Luis Falcon For more information, please check our Codeberg page: <https://codeberg. ... org/gnuhealth/his>

- [mailutils @ Savannah: GNU mailutils version 3.21](#) (2025/12/11 14:31)

GNU mailutils version 3.21 is available for download. Short list of changes in this version follows. Refer to its NEWS file for a detailed discussion. Optionally create intermediate directories when creating mailboxes. New configuration section homedir controls creation of home directories. Imap4d configuration statements create-home-dir and home-dir-mode declared obsolete. Use homedir section, instead. Changes in sieve language Fileinto action: new option :interdir New commands in string expansions: localuser and detail. Immediate values allowed in Imap.field_map configuration statement. Bugfixes Sieve: fix coredump on parsing fileinto :permission action. Sieve: fix optimizer. Library: fix parsing ls-compatible permission strings. Library: fix mu_sieve_machine_clone function. Configuration: use backslash to escape delimiters in some statements. Library: fix localized help output.

- [radius @ Savannah: GNU radius version 1.7](#) (2025/12/11 12:07)

Version 1.7 of GNU radius is available for download. This is a maintenance release. Noteworthy changes: Support for Guile 3.x. Code cleanup. Improved testsuite. Improved documentation. Various bugfixes.

- [FSF News: Free Software Awards winners announced: Andy Wingo, Alx Sa, Govdirectory](#) (2025/12/09 16:40)

BOSTON, Massachusetts, USA (Tuesday, December 9, 2025) – The Free Software Foundation (FSF) announced today the recipients of the 2024

Free Software Awards, which are given annually by the FSF to groups and individuals in the free software community who have made significant contributions to the cause for software freedom.

- [GNU Guix: A Planet for Guix](#) (2025/12/05 12:00)

I am pleased to announce the availability of Planet Guix, an Atom and RSS aggregator covering all things Guix. You can browse posts on the website or use your favourite feed reader to subscribe to the aggregate feed. Planet Guix already has subscriptions to 19 blogs from around the community; if you write about Guix (no matter how infrequently) and would like your blog to be included, or if you would like to suggest another blog I missed, please create a pull request against the repository in Codeberg — you'll see that the subscriptions are simply configured as association lists in `planet/config.scm`. Background Back in September, Sébastien Gendre asked on the help-guix mailing list if there were any plans to create a Planet website for Guix. The discussion drifted into how this might be implemented in Guile, and I thought it sounded like an interesting project for the dark autumn evenings. The original Planet aggregator was written in Python and many Planet websites are still using its successor, Venus. The Venus code base has not seen much activity in the last decade and still uses Python 2, which was sunset in 2020. This was all the incentive I needed to implement a new Planet aggregator and static site generator in Guile. Implementation We already know from the likes of Haunt that Guile has all the tools needed to generate a static web site. Both Atom and RSS are XML formats, and Guile also has great support for working with XML. The Guile Planet implementation uses the following built-in modules: `(web client)` to fetch the feeds. `(sxml simple)` for reading the Atom/RSS feeds and writing the aggregate Atom feed. `(sxml xpath)` for searching the feeds to extract the data of interest. `(sxml transform)` for sanitizing HTML in the entry summaries. Many feeds include HTML content in the entry summary, which we need to parse. This is where `htmlprag` from `guile-lib` comes in. I used this both to parse HTML embedded in feeds and to generate the static content from an SXML data structure. With these libraries to hand the code for the planet aggregator almost wrote itself! I was trying to keep dependencies to a minimum, but `guile-filesystem` is too useful to do without and, later in the development process, I pulled in `guile-srfi-235` which provides some useful combinators. At the moment I'm only using `apply-chain` to build a function for post-processing one of the feeds, but why re-invent the wheel? Deployment I initially deployed the Planet to a test site running on one of my servers, but the idea was received enthusiastically by the Guix maintainers and I was happy that they wanted to host it on their infrastructure. Of course they are using Guix to manage their virtual machines in Hetzner cloud! While they could have picked up the Planet code and run with it, instead they pointed me at the server configuration and invited me to make a pull request against `hydra/guix-hetzner-2.scm`. They suggested I base the configuration on their existing static-web-site-configuration so I started reading the code which proved very educational (I admit that I had to sleep on it for a week before coming up with a plan!) The static-web-site-configuration did almost everything needed to build the Planet aggregator, only the build step runs like a Guix package build in an isolated environment with no network - so we cannot fetch the feeds in this build step. Luckily, I had already implemented functionality in the Planet code base to build the static site from feeds cached on disk. So it was simply a case of adding support for a pre-build script to the static-web-site-configuration and using this step to download the feeds. The pull request was merged after some short discussion, and a few days later the site was live in its new home. Community This was my second time contributing to the Guix project and I'm pleased to report that it was a smooth experience both times. When it came to the deployment, I was glad that I was encouraged to add the service configuration myself instead of being spoon-fed: working with computers, you learn best by doing. I'd like to give a shout-out to @civodul, @cbaines, and @apteryx for their help with the deployment, and to the several people who sent merge requests to add their blogs before I even got around to writing this announcement. I think the Planet site is already a great place to discover people writing about Guix, and I hope it grows and becomes an asset to

the community. Happy reading!

- [a2ps @ Savannah: a2ps 4.15.8 released \[stable\]](#) (2025/12/04 17:34)

I am delighted to announce a new release of GNU a2ps, the “anything to PostScript” system. This is to announce a2ps-4.15.8, a stable release. This release fixes a buffer overflow, and a failure to build on some older systems. There have been 13 commits by 1 people in the 21 weeks since 4.15.7. See the NEWS below for a brief summary. Thanks to everyone who has contributed! The following people contributed changes to this release: Reuben Thomas (13) Reuben [on behalf of the a2ps maintainers]

===== Here is the GNU a2ps home page:

<https://gnu.org/s/a2ps/> Here are the compressed sources and a GPG detached signature: <https://ftpmirror.gnu.org/a2ps/a2ps-4.15.8.tar.gz>
<https://ftpmirror.gnu.org/a2ps/a2ps-4.15.8.tar.gz.sig> Use a mirror for higher download bandwidth: <https://www.gnu.org/order/ftp.html> Here are the SHA256 and SHA3-256 checksums: File: a2ps-4.15.8.tar.gz SHA256 sum:
8d13915a36ebbf8e7b236b350cc81adc714acb217a18e8d8c60747c0ad353f9 SHA3-256 sum:
0dce19c25df3be0ce1fc2b92710c33a724595b3617686cbb904ab60dcdd15b34 Verify the SHA256 checksum with either sha256sum, sha256, or shasum -a 256. Verify the SHA3-256 checksum with cksum -a sha3 --check from coreutils-9.8. Use a .sig file to verify that the corresponding file (without the .sig suffix) is intact. First, be sure to download both the .sig file and the corresponding tarball. Then, run a command like this: gpg --verify a2ps-4.15.8.tar.gz.sig The signature should match the fingerprint of the following key: pub rsa2048 2013-12-11 [SC] 2409 3F01 6FFE 8602 EF44 9BB8 4C8E F3DA 3FD3 7230 uid Reuben Thomas <rrt@sc3d.org> uid keybase.io/rrt <rrt@keybase.io> If that command fails because you don't have the required public key, or that public key has expired, try the following commands to retrieve or refresh it, and then rerun the 'gpg --verify' command. gpg --locate-external-key rrt@sc3d.org gpg --recv-keys 4C8EF3DA3FD37230 wget -q -O- 'https://savannah.gnu.org/project/release-gpgkeys.php?group=a2ps&download=1' | gpg --import - As a last resort to find the key, you can try the official GNU keyring: wget -q https://ftp.gnu.org/gnu/gnu-keyring.gpg gpg --keyring gnu-keyring.gpg --verify a2ps-4.15.8.tar.gz.sig This release is based on the a2ps git repository, available as git clone https://https.git.savannah.gnu.org/git/a2ps.git with commit c41304d1745017322adb12ba40e7f06984e7f14e tagged as v4.15.8. For a summary of changes and contributors, see: <https://gitweb.git.savannah.gnu.org/gitweb/?p=a2ps.git;a=shortlog;h=v4.15.8> or run this command from a git-cloned a2ps directory: git shortlog v4.15.7..v4.15.8 This release was bootstrapped with the following tools: Autoconf 2.72 Automake 1.18.1 Gnulib 2025-12-04 481064c5c22c8137188eeeb6662ebeb03fc6d0b8 NEWS * Noteworthy changes in release 4.15.8 (2025-12-04) [stable] * Bug fixes: - Fix a buffer overflow when a long value supplied to -E. - Include some header files with system paths, not user paths. * Build system: - Fix building on systems that need gnulib's malloc wrapper. - Remove a generated file from git. - Update the version of gettext used. * Documentation: - Update copyright notices to point to GPL online.

- [GNU Guile: GNU Guile 3.0.11 released](#) (2025/12/01 14:00)

We are pleased to announce the release of GNU Guile 3.0.11! This release is mainly a bug-fix release, though it does include a number of new features, including support for SRFI 197: Pipeline Operators, support for SRFI 207: String-notated bytevectors (bytestrings), and JIT (just-in-time) compilation for the RISC-V architecture. It also overhauls SRFI-64 (testing) and includes many other improvements. For full details, see the release notes and check out the download page. Happy Guile hacking!

- [Amin Bandali: Free software activities in November 2025](#) (2025/11/30 23:26)

Hello and welcome to my November free software activities report. I've been working on a number of things throughout this month but they're not quite ready for reporting yet, so this month's report will be quite short. GNU & FSF EmacsConf: I recorded the video for my Gnus talk for this year's conference. The video will be available along with the other EmacsConf talks from the conference website, but if you're feeling particularly impatient you can sneak a peek at it. :) <https://archive.org/details/emacsconf-2025-gnus> GNU Spotlight: I prepared and sent the November GNU Spotlight to the FSF campaigns team for publication on the FSF's community blog and the monthly Free Software Supporter newsletter. Take care, and so long for now.

- [Jose E. Marchesi: Algol 68 Front-End merged in GCC](#) (2025/11/30 00:00)

I am very happy to announce that, after almost a year in the works, today the Algol 68 front-end has been merged in GCC proper in its development trunk branch. This means that we are no longer off-tree, and that GCC 16 will be featuring a full-fledged and modern Algol 68 compiler once it gets released. An Algol 68 module This doesn't mean the work is done. The modules system, which is already functional, needs to be completed and polished, parallel clauses are still to be implemented, and the quality of the generated code should be generally improved. We also want to replace the Boehm GC with a tightly integrated customized, exact one, and to continue modernizing and expanding the language, always carefully and respectfully, via GNU extensions: exception handling, FFI, etc. At this point I would like to thank Marcel van der Veer, Pietro Monteiro, Mohammad-Reza Nabipour, Thomas Schwinge, Sam James, Matthias Klose, Iain Buclaw, Andrew Pinski, Segher Boessenkool, Iain Sandoe, the GCC global reviewers and the overall GCC community. Without their help, and Marcel's nifty Algol 68 parser, this front-end would simply not exist. And now, the real fun starts... ;) Happy algoling!

- [Simon Josefsson: Container Images for Debian with Guix](#) (2025/11/28 16:32)

The debian-with-guix-container project build and publish container images of Debian GNU/Linux stable with GNU Guix installed. The images are like normal Debian stable containers but have the guix tool and a reasonable fresh guix pull. Supported architectures include amd64 and arm64. The multi-arch container is called: `registry.gitlab.com/debdistutils/guix/debian-with-guix-container:stable` It may also be accessed via `debian-with-guix` at Docker Hub as: `docker.io/jas4711/debian-with-guix:stable` The container images may be used like this: `$ podman run --privileged -it --hostname guix --rm registry.gitlab.com/debdistutils/guix/debian-with-guix-container:stable root@guix:/# hello bash: hello: command not found root@guix:/# guix describe guix c9eb69d repository URL: https://gitlab.com/debdistutils/guix/mirror.git branch: master commit: c9eb69ddb05e77300b59f49f4bb5aa50cae0892 root@guix:/# LC_ALL=C.UTF-8 /root/.config/guix/current/bin/guix-daemon --build-users-group=guixbuild & [1] 21 root@guix:/# GUIX_PROFILE=/root/.config/guix/current; . "$GUIX_PROFILE/etc/profile" root@guix:/# guix describe Generation 2 Nov 28 2025 10:14:11 (current) guix c9eb69d repository URL: https://gitlab.com/debdistutils/guix/mirror.git branch: master commit: c9eb69ddb05e77300b59f49f4bb5aa50cae0892 root@guix:/# guix install --verbosity=0 hello accepted connection from pid 55, user root The following package will be installed: hello 2.12.2 hint: Consider setting the necessary environment variables by running: GUIX_PROFILE="/root/.guix-profile" . "$GUIX_PROFILE/etc/profile" Alternately, see `guix package --search-paths -p "/root/.guix-profile". root@guix:/# GUIX_PROFILE="/root/.guix-profile" root@guix:/# . "$GUIX_PROFILE/etc/profile" root@guix:/# hello Hello, world! root@guix:/# Below is an example GitLab pipeline job that demonstrate how to run guix install to install additional dependencies, and then download and build a package that pick up the installed package from the system. test-wget-configure-make-libksba-amd64: image: registry.gitlab.com/debdistutils/guix/debian-with-guix-container:stable before_script: - env LC_ALL=C.UTF-8 /root/.config/guix/current/bin/guix-daemon --build-users-group=guixbuild $GUIX_DAEMON_ARG & - GUIX_PROFILE=/root/.config/guix/current; . "$GUIX_PROFILE/etc/profile" - guix`

describe - guix install libgpg-error - GUIX_PROFILE="/root/.guix-profile"; . "\$GUIX_PROFILE/etc/profile" - apt-get install --update -y --no-install-recommends build-essential wget ca-certificates bzip2 script: - wget <https://www.gnupg.org/ftp/gcrypt/libksba/libksba-1.6.7.tar.bz2> - tar xfa libksba-1.6.7.tar.bz2 - cd libksba-1.6.7 - ./configure - make V=1 - make check VERBOSE=t V=1 The images were initially created for use in GitLab CI/CD Pipelines but should work for any use. The images are built in a GitLab CI/CD pipeline, see .gitlab-ci.yml. The containers are derived from official Debian stable images with Guix installed and a successful run of guix pull, built using buildah invoked from build.sh using image/Containerfile that runs image/setup.sh. The pipeline also push images to the GitLab container registry, and then also to Docker Hub. Guix binaries are downloaded from the Guix binary tarballs project because of upstream download site availability and bandwidth concerns. Enjoy these images! Hopefully they can help you overcome the loss of Guix in Debian which made it a mere apt-get install guix away before. There are several things that may be improved further. An alternative to using podman --privileged is to use --security-opt seccomp=unconfined --cap-add=CAP_SYS_ADMIN,CAP_NET_ADMIN which may be slightly more fine-grained. For ppc64el support I ran into an error message that I wasn't able to resolve: guix pull: error: while setting up the build environment: cannot set host name: Operation not permitted For riscv64, I can't even find a Guix riscv64 binary tarball for download, is there one anywhere? For arm64 containers, it seems that you need to start guix-daemon with --disable-chroot to get something to work, at least on GitLab.com's shared runners, otherwise you will get this error message: guix install: error: clone: Invalid argument Building the images themselves also require disabling some security functionality, and I was not able to build images with buildah without providing --cap-add=CAP_SYS_ADMIN,CAP_NET_ADMIN otherwise there were errors like this: guix pull: error: cloning builder process: Operation not permitted guix pull: error: clone: Operation not permitted guix pull: error: while setting up the build environment: cannot set loopback interface flags: Operation not permitted Finally on amd64 it seems --security-opt seccomp=unconfined is necessary, otherwise there is an error message like this, even if you use --disable-chroot: guix pull: error: while setting up the child process: in phase setPersonality: cannot set personality: Function not implemented This particular error is discussed upstream, but I think generally that these error suggest that guix-daemon could use more optional use of features: if some particular feature is not available, gracefully fall back to another mode of operation, instead of exiting with an error. Of course, it should never fall back to an insecure mode of operation, unless the user requests that. Happy Hacking!

- [gnuastro @ Savannah: Gnuastro 0.24 released](#) (2025/11/27 01:35)

The 24th release of GNU Astronomy Utilities (Gnuastro) is now available. See the full announcement for all the new features in this release and the many bugs that have been found and fixed: [https://lists.gnu.org ... -11/msg00001.html](https://lists.gnu.org/archive/html/astropack/2025-11/msg00001.html)

- [www @ Savannah: Malware in Proprietary Software - Latest Additions](#) (2025/11/25 15:00)

The initial injustice of proprietary software often leads to further injustices: malicious functionalities. The introduction of unjust techniques in nonfree software, such as back doors, DRM, tethering, and others, has become ever more frequent. Nowadays, it is standard practice. We at the GNU Project show examples of malware that has been introduced in a wide variety of products and dis-services people use everyday, and of companies that make use of these techniques. Here are our latest additions October 2025 Apple's Operating Systems Are Malware Apple repeatedly sabotaged Beeper Mini, a client to replace its iMessage instant messaging service, interfering with people's ability to use their installed software just to keep a dominant position in that market by avoiding competition. Malware In Cars Jeep forced a software change into certain cars. In addition to being unjust, this one was dangerous too. Google's Software is Malware Motorola ships Android phones with a locked bootloader, offering a method to unlock the devices. The method involves creating an account, which requires running nonfree JavaScript and

disclosing personal data as well as identifying at least your phone's model. This puts users in danger of privacy breaches in exchange for permission to modify the software that runs in a device they own. Users should be free to modify this and all software as they wish, without interference from the manufacturer or developer. Back in 2013 (when the company was owned by Google) someone found a way to crack the bootloader restrictions. Android developers also provide a lock/unlock method. September 2025 Malware in Appliances Echelon forcefully downgraded the firmware of its home gym equipment so that the devices will provide their full capabilities only if connected to Echelon's servers and only with a paid subscription, all the while breaking compatibility with third party apps that offer additional functionalities. Efforts to restore offline functionality were successful, but the fix can't be released due to section 1201 of the DMCA. Note that those articles mention "open source"; the GNU Project recommends the expression free/libre software instead. Google's Software is Malware Google has announced the inclusion of a "security" measure in Android "smartphones," which will require any software installed in certified Android devices to come from a developer who has gone through Google's new developer verification program. The problem here is not that there's a system that provides trust on the origin of the software. A system like that might be useful, but the end user should still be able to select which organization provides that service, or maybe set up such an organization or renounce the service altogether. Making this verification exclusive to Google makes us question which is the threat here. Is it a user installing malware inadvertently? Or is it the user installing software that makes Google lose money? This will also kill projects such as F-Droid that promote privacy and freedom by distributing free (as in freedom) apps.

- [Gary Benson: Slow boot?](#) (2025/11/24 08:41)

Does your Linux box take forever to boot? The command you're looking for is systemd-analyze blame

- [parallel @ Savannah: GNU Parallel 20251122 \('Mamdani'\) released \[stable\]](#) (2025/11/24 05:27)

GNU Parallel 20251122 ('Mamdani') has been released. It is available for download at: [lbry://@GnuParallel:4](#) Quote of the month: ainda não inventaram palavras capazes de expressar minha gratidão aos desenvolvedores do GNU Parallel -- @nueidris.kawaii.social New in this release: No new features. Bug fixes. GNU Parallel - For people who live life in the parallel lane. If you like GNU Parallel record a video testimonial: Say who you are, what you use GNU Parallel for, how it helps you, and what you like most about it. Include a command that uses GNU Parallel if you feel like it. About GNU Parallel GNU Parallel is a shell tool for executing jobs in parallel using one or more computers. A job can be a single command or a small script that has to be run for each of the lines in the input. The typical input is a list of files, a list of hosts, a list of users, a list of URLs, or a list of tables. A job can also be a command that reads from a pipe. GNU Parallel can then split the input and pipe it into commands in parallel. If you use xargs and tee today you will find GNU Parallel very easy to use as GNU Parallel is written to have the same options as xargs. If you write loops in shell, you will find GNU Parallel may be able to replace most of the loops and make them run faster by running several jobs in parallel. GNU Parallel can even replace nested loops. GNU Parallel makes sure output from the commands is the same output as you would get had you run the commands sequentially. This makes it possible to use output from GNU Parallel as input for other programs. For example you can run this to convert all jpeg files into png and gif files and have a progress bar: `parallel --bar convert {1} {1.}.{2} ::: *.jpg ::: png gif` Or you can generate big, medium, and small thumbnails of all jpeg files in sub dirs: `find . -name '*.jpg' | parallel convert -geometry {2} {1} {1//}/thumb{2}_{1//} ::: - ::: 50 100 200` You can find more about GNU Parallel at: <http://www.gnu.org/s/parallel/> You can install GNU Parallel in just 10 seconds with: `$ (wget -O - pi.dk/3 || lynx -source pi.dk/3 || curl pi.dk/3/ || \ fetch -o - http://pi.dk/3) > install.sh $ sha1sum install.sh | grep c555f616391c6f7c28bf938044f4ec50 12345678 c555f616 391c6f7c 28bf9380 44f4ec50 $ md5sum install.sh | grep 707275363428aa9e9a136b9a7296dfe4 70727536 3428aa9e 9a136b9a 7296dfe4 $ sha512sum install.sh | grep`

b24bfe249695e0236f6bc7de85828fe1f08f4259 83320d89 f56698ec 77454856 895edc3e aa16feab 2757966e 5092ef2d 661b8b45
 b24bfe24 9695e023 6f6bc7de 85828fe1 f08f4259 6ce5480a 5e1571b2 8b722f21 \$ bash install.sh Watch the intro video on <http://www.youtube.com/watch?v=L284C9FF2488BC6D1> Walk through the tutorial (man parallel_tutorial). Your command line will love you for it. When using programs that use GNU Parallel to process data for publication please cite: O. Tange (2018): GNU Parallel 2018, March 2018, <https://doi.org/10.1101/zenodo.1146014>. If you like GNU Parallel: Give a demo at your local user group/team/colleagues Post the intro videos on Reddit/Diaspora*/forums/blogs/Identi.ca/Google+/Twitter/Facebook/Linkedin/ mailing lists Get the merchandise <https://gnuparall.com/merch> ... igns/gnu-parallel Request or write a review for your favourite blog or magazine Request or build a package for your favourite distribution (if it is not already there) Invite me for your next conference If you use programs that use GNU Parallel for research: Please cite GNU Parallel in you publications (use --citation) If GNU Parallel saves you money: (Have your company) donate to FSF <https://my.fsf.org/donate/> About GNU SQL GNU sql aims to give a simple, unified interface for accessing databases through all the different databases' command line clients. So far the focus has been on giving a common way to specify login information (protocol, username, password, hostname, and port number), size (database and table size), and running queries. The database is addressed using a DBURL. If commands are left out you will get that database's interactive shell. When using GNU SQL for a publication please cite: O. Tange (2011): GNU SQL - A Command Line Tool for Accessing Different Databases Using DBURLs, ;login: The USENIX Magazine, April 2011:29-32. About GNU Niceload GNU niceload slows down a program when the computer load average (or other system activity) is above a certain limit. When the limit is reached the program will be suspended for some time. If the limit is a soft limit the program will be allowed to run for short amounts of time before being suspended again. If the limit is a hard limit the program will only be allowed to run when the system is below the limit.

- [Jose E. Marchesi: Version 6 of the Algol 68 GCC Front-End posted](#) (2025/11/22 03:00)

Today I submitted the version 6 of the patch series for the Algol 68 GCC Front-End:

<https://gcc.gnu.org/pipermail/gcc-patches/2025-November/701589.html> Since last submission we have added a modules system based on the Modules and Separate Compilation Facility designed by Charles Lindsey and Hendrik Boom and released by the IFIP Working Group 2.1 Standing Subcommittee on ALGOL 68 Support. To our knowledge, this is the first time the modules facility ever gets implemented. This is the deal: Jose E. Marchesi (50): a68: top-level misc files a68: build system a68: build system (regenerated files) a68: documentation a68: command-line options a68: DWARF language codes a68: darwin specific support a68: powerpc specific support a68: gcc/algol68 misc files a68: ga68 compiler driver a68: a681 compiler proper a68: unicode support routines a68: front-end diagnostics a68: modules exports a68: modules imports a68: parser: entry point a68: parser: AST nodes attributes/types a68: parser: scanner a68: parser: keyword tables management a68: parser: top-down parser a68: parser: parenthesis checker a68: parser: bottom-up parser a68: parser: syntax check for declarers a68: parser: standard prelude definitions a68: parser: parsing of modes a68: parser: symbol table management a68: parser: static scope checker a68: parser: debug facilities a68: parser: extraction of tags from phrases a68: parser: dynamic stack usage in serial clauses a68: parser: pragmas infrastructure a68: low: lowering entry point and misc handlers a68: low: plain values a68: low: stowed values a68: low: standard prelude a68: low: clauses and declarations a68: low: runtime a68: low: builtins a68: low: ranges a68: low: units and coercions a68: low: modes a68: libga68: sources, spec and misc files a68: libga68: build system a68: libga68: build system (generated files) a68: testsuite: infrastructure a68: testsuite: execution tests 1/2 a68: testsuite: execution tests 2/2 a68: testsuite: compilation tests a68: testsuite: revised MC Algol 68 test set a68: testsuite: mcgt tests

- [remotecontrol @ Savannah: Google is collecting troves of data from downgraded Nest thermostats](#) (2025/11/17 15:06)

[https://www.theve ... d-data-collection](https://www.theve...d-data-collection)

- [GNUnet News: GNUnet 0.26.1](#) (2025/11/14 23:00)

GNUnet 0.26.1 This is a bugfix release for gnunet 0.26.0. It fixes some regressions and minor bugs. Links Source:

<https://ftpmirror.gnu.org/gnunet/gnunet-0.26.1.tar.gz> (<https://ftpmirror.gnu.org/gnunet/gnunet-0.26.1.tar.gz.sig>) Detailed list of changes: <https://git.gnunet.org/gnunet.git/log?h=v0.26.1> NEWS: <https://git.gnunet.org/gnunet.git/tree/NEWS?h=v0.26.1> The GPG key used to sign is: 3D11063C10F98D14BD24D1470B0998EF86F59B6A Note that due to mirror synchronization, not all links may be functional early after the release. For direct access try <https://ftp.gnu.org/gnu/gnunet/>

- [Simon Josefsson: Introducing the Debian Libre Live Images](#) (2025/11/12 23:16)

The Debian Libre Live Images allows you to run and install Debian GNU/Linux without non-free software. The general goal is to provide a way to use Debian without reliance on non-free software, to the extent possible within the Debian project. One challenge are the official Debian live and installer images. Since the 2022 decision on non-free firmware, the official images for bookworm and trixie contains non-free software. The Debian Libre Live Images project provides Live ISO images for Intel/AMD-compatible 64-bit x86 CPUs (amd64) built without any non-free software, suitable for running and installing Debian. The images are similar to the Debian Live Images distributed as Debian live images. One advantage of Debian Libre Live Images is that you do not need to agree to the distribution terms and usage license agreements of the non-free blobs included in the official Debian images. The rights to your own hardware won't be crippled by the legal restrictions that follows from relying on those non-free blobs. The usage of your own machine is no longer limited to what the non-free firmware license agreements allows you to do. This improve your software supply-chain situation, since you no longer need to consider their implication on your computing environment for your liberty, privacy or security. Inclusion of non-free firmware is a vehicle for xz-style attacks. For more information about the advantages of free software, see the FSF's page on What is Free Software?. Enough talking, show me the code! Err, binaries! Download images: `wget https://gitlab.com/api/v4/projects/74667529/packages/generic/debian-libre-live/main/live-image-amd64.hybrid.iso` `wget https://gitlab.com/api/v4/projects/74667529/packages/generic/debian-libre-live/main/live-image-amd64.hybrid.iso.SHA256SUMS` `sha256sum -c live-image-amd64.hybrid.iso.SHA256SUMS` Run in a virtual machine: `kvm -cdrom live-image-amd64.hybrid.iso -m 8G` Burn to an USB drive for installation on real hardware: `sudo dd if=live-images-amd64.hybrid.iso of=/dev/sdX # use sdX for USB drive` Images are built using live-build from the Debian Live Team. Inspiration has been taken from Reproducible Live Images and Kali Live. The images are built by GitLab CI/CD shared runners. The pipeline .gitlab-ci.yml container job creates a container with live-build installed, defined in container/Containerfile. The build job then invokes run.sh that includes a run to lb build, and then upload the image to the package registry. This is a first initial public release, calibrate your expectations! The primary audience are people already familiar with Debian. There are known issues. I have performed successful installations on a couple of different machines including laptops like Lenovo X201, Framework AMD Laptop 13" etc. Are you able to install Debian without any non-free software on some hardware using these images? Happy Hacking!

- [GNUnet News: GNUnet 0.26.0](#) (2025/11/12 23:00)

GNUnet 0.26.0 released We are pleased to announce the release of GNUnet 0.26.0. GNUnet is an alternative network stack for building secure, decentralized and privacy-preserving distributed applications. Our goal is to replace the old insecure Internet protocol stack. Starting from an application for secure publication of files, it has grown to include all kinds of basic protocol components and applications towards the creation of a GNU internet. This is a new major release. Major versions may break protocol compatibility with the 0.25.X versions. Please be aware that Git

master is thus henceforth (and has been for a while) INCOMPATIBLE with the 0.25.X GNUnet network, and interactions between old and new peers will result in issues. In terms of usability, users should be aware that there are still a number of known open issues in particular with respect to ease of use, but also some critical privacy issues especially for mobile users. Also, the nascent network is tiny and thus unlikely to provide good anonymity or extensive amounts of interesting information. As a result, the 0.26.0 release is still only suitable for early adopters with some reasonable pain tolerance. If it were not for compatibility-breaking changes in the crypto API of libgnunetutil this would only be a maintenance release. The changes hopefully protect users of the library from misuse of GNUnet's cryptographic key objects in ways that may be detrimental to security. Since this change breaks backwards compatibility, this is a new major release. Download links [gnunet-0.26.0.tar.gz](#) (signature) [gnunet-fuse-0.26.0.tar.gz](#) (signature) The GPG key used to sign is: 3D11063C10F98D14BD24D1470B0998EF86F59B6A Note that due to mirror synchronization, not all links might be functional early after the release. For direct access try <http://ftp.gnu.org/gnu/gnunet/> Changes A detailed list of changes can be found in the git log, the NEWS. Known Issues There are known major issues with the TRANSPORT subsystem. There may be some regressions in the new CORE subsystem. There are known moderate implementation limitations in CADET that negatively impact performance. There are known moderate design issues in FS that also impact usability and performance. There are minor implementation limitations in SET that create unnecessary attack surface for availability. The RPS subsystem remains experimental. In addition to this list, you may also want to consult our bug tracker at bugs.gnunet.org which lists about 190 more specific issues. Thanks This release was the work of many people. The following people contributed code and were thus easily identified: Christian Grothoff, Florian Dold, TheJackiMonster, ch3, and Martin Schanzenbach.

- [health @ Savannah: GNU Health HIS server 5.0.3 patchset bundle released](#) (2025/11/12 17:52)

Dear community I am happy to announce that the GNU Health Hospital Information System 5.0.3 patchset has been released! In addition to the patches, you will notice a tarball significantly smaller than previous bundles, mainly from not excluding large translation files that need work. You can of course always download the most current translation of any GNU Health package from our weblate instance at Codeberg. (<https://translate.alexio.tf/projects/gnuhealth/>) If you use the vanilla / standard installation, you can update the server and the dependencies from the gnuhealth control center (<https://docs.gnuhealth.org/en/latest/development/development.html>) Backup As usual, before you upgrade your instance, make sure you have made a backup of your DB instance and "attach" resource ! Happy hacking ♥ Changelog for 5.0.3 2025-11-11 * [SKIP CI] Backport po files from devel branch. (c7f8f3a3c) (Feng Shu) 2025-11-11 * Add tryton/health_icd9procs/locale/el.po (5acbb83f2) (Feng Shu) 2025-11-10 * [SKIP CI] Backport po files from devel branch. (57aeec18f) (Feng Shu) 2025-11-07 * [SKIP CI] Backport po files from devel branch. (489f1b225) (Feng Shu) 2025-11-05 * Fix bug core:Dx command should filter by arguments #153 (dc410b014) (Luis Falcon) 2025-11-05 * [SKIP CI] Backport po files from devel branch. (40a327c3b) (Feng Shu) 2025-11-05 * Update tryton/health/locale/health.pot (a41f324ed) (Feng Shu) 2025-11-05 * [SKIP CI] Backport po files from devel branch. (583837f1b) (Feng Shu) 2025-11-05 * fix bug core: healthprof attribute duplicated in gnuhealth.patient.disease model (2c6976f6b) (Luis Falcon) 2025-11-04 * Readd sv.po file, for Umeåman will become a team leader for Swedish. [do not merge] (31ef5422c) (Feng Shu) 2025-11-04 * [SKIP CI] Backport po files from devel branch. (bc914fc3d) (Feng Shu) 2025-11-02 * Delete health module po files which translate progress < 30%, [do not merge] (48e76223a) (Feng Shu) 2025-10-29 * [SKIP CI] Backport po files from devel branch. (c2497b4de) (Feng Shu) 2025-10-28 * translation: Missing translation terms. #156. Fix spanish string for time in health_surgery (c76d20d36) (Luis Falcon) 2025-10-28 * Issue #155. Clean po files, backport from devel branch [do not merge] (59b8f3eb5) (Feng Shu) 2025-10-28 * Issue #155. genetics_uniprot: Remove empty translation po files (01806d006) (Luis Falcon) 2025-10-28 * [SKIP CI] Backport po files from devel branch.

(e719e0f9c) (Feng Shu) 2025-10-28 * Update tryton/health/locale/health.pot [do not merge] (3b9ec6200) (Feng Shu) 2025-10-28 * Merge remote-tracking branch 'origin/po-backport' into patchset/5.0.3 (b6847391a) (Feng Shu) 2025-10-27 * [SKIP CI] Backport po files from devel branch. (32c1903b2) (po-backport) 2025-10-25 * Update package version to 5.0.3 (bbcc6c9ed) (Luis Falcon) 2025-10-25 * Fix bug #150. core: Include children in the family history (46f6293ad) (Luis Falcon) For a more detailed list, please go to our project page at Codeberg: <https://codeberg.org/gnuhealth/his>

- [GNU Taler news: GNU Taler 1.2 released](#) (2025/11/11 23:00)

We are happy to announce the release of GNU Taler v1.2.

- [health @ Savannah: GNUHealthCon 2025 - December 20th](#) (2025/11/11 08:51)

The GNU Health Conference 2025 will take place December 20th, and it will be online! Join us and get the latest on Free Software in healthcare and social medicine! #GHCon2025 More info and registration: <https://www.gnuhealth.org/ghcon/2025/>

- [unrtf @ Savannah: unrtf 0.21.11](#) (2025/11/11 06:59)

unrtf 0.21.11 is released, fixing recently submitted security issues and a number of older bugs. Until the tar file can be uploaded to the proper location on <ftp.gnu.org>, you can find it on the project home page

- [coreutils @ Savannah: coreutils-9.9 released \[stable\]](#) (2025/11/10 14:16)

This is to announce coreutils-9.9, a stable release. This is primarily a stabilization release, details of which are summarized in the NEWS below. There have been 106 commits by 10 people in the 7 weeks since 9.8. Thanks to everyone who has contributed! The following people contributed changes to this release: Bernhard Voelker (4) Mathieu Bordere (1) Bruno Haible (4) Nicolas Boichat (1) Collin Funk (28) Paul Eggert (9) Grisha Levit (1) Pádraig Brady (57) Hannes Braun (1) Sylvestre Ledru (1) Pádraig [on behalf of the coreutils maintainers]

===== Here is the GNU coreutils home page: <https://gnu.org/s/coreutils/> Here are the compressed sources: <https://ftp.gnu.org/gnu/coreutils/coreutils-9.9.tar.gz> (15MB)

<https://ftp.gnu.org/gnu/coreutils/coreutils-9.9.tar.xz> (6.1MB) Here are the GPG detached signatures:

<https://ftp.gnu.org/gnu/coreutils/coreutils-9.9.tar.gz.sig> <https://ftp.gnu.org/gnu/coreutils/coreutils-9.9.tar.xz.sig> Use a mirror for higher download bandwidth: <https://www.gnu.org/order/ftp.html> Here are the SHA1 and SHA256 checksums: File: coreutils-9.9.tar.gz SHA1 sum:

c66ec935ab7e0ef32c40153fcf67dcf67579171a SHA256 sum: 91a719fcf923de686016f2c8d084a8be1f793f34173861273c4668f7c65af94a File: coreutils-9.9.tar.xz SHA1 sum: 456b5c69f3ce8fbdb926a11652673ecf12bfc44 SHA256 sum:

19bcb6ca867183c57d77155eae946c5eced88183143b45ca51ad7d26c628ca75 Use a .sig file to verify that the corresponding file (without the .sig suffix) is intact. First, be sure to download both the .sig file and the corresponding tarball. Then, run a command like this: `gpg --verify coreutils-9.9.tar.gz.sig` The signature should match the fingerprint of the following key: pub rsa4096/0xDF6FD971306037D9 2011-09-23 [SC]

Key fingerprint = 6C37 DC12 121A 5006 BC1D B804 DF6F D971 3060 37D9 uid [ultimate] Pádraig Brady <P@draigBrady.com> uid [ultimate] Pádraig Brady <pixelbeat@gnu.org> If that command fails because you don't have the required public key, or that public key has expired, try the following commands to retrieve or refresh it, and then rerun the 'gpg --verify' command. `gpg --locate-external-key P@draigBrady.com gpg --recv-keys DF6FD971306037D9 wget -q -O-`

'<https://savannah.gnu.org/project/release-gpgkeys.php?group=coreutils&download=1>' | `gpg --import -` As a last resort to find the key, you can try the official GNU keyring: `wget -q https://ftp.gnu.org/gnu/gnu-keyring.gpg gpg --keyring gnu-keyring.gpg --verify coreutils-9.9.tar.gz.sig` This

release is based on the coreutils git repository, available as `git clone https://https.git.savannah.gnu.org/git/coreutils.git` with commit `0ae5bdc7a8311efd3efe43363050710d6ea1c367` tagged as v9.9. For a summary of changes and contributors, see: <https://gitweb.git.savannah.gnu.org/gitweb/?p=coreutils.git;a=shortlog;h=v9.9> or run this command from a git-cloned coreutils directory: `git shortlog v9.8..v9.9` This release was bootstrapped with the following tools: Autoconf 2.72.97-cf8b9 Automake 1.18.1 Gnulib 2025-11-06 862a81c0e15448adde6a6e7473ec47e3a4bd91a6 Bison 3.8.2 NEWS * Noteworthy changes in release 9.9 (2025-11-10) [stable] ** Bug fixes

``basenc --base58`` would not operate correctly with input > 15561475 bytes. [bug introduced with `--base58` in coreutils-9.8] `'cksum --check'` now supports base64 encoded input in untagged format: - for all length adjustable algorithms (blake2b, sha2, sha3), - if that base64 input starts with a tag like "SHA1" etc. Previously an error was given, about invalid input format. [bug introduced in coreutils-9.2] `'cksum --check -a sha2'` has better support for tagged format. Previously an unneeded but explicit `'-a sha2'` did not match standard tags like SHA256. Also non standard SHA2 tags with a bad length resulted in undefined behavior. [bug introduced in coreutils-9.8] `'cp'` restores performance with transparently compressed files, which regressed due to the avoidance of copy offload, seen with OpenZFS at least. [bug introduced in coreutils-9.8] ``env`` on macOS, for now only when built with `--disable-nls`, will no longer always set a `__CF_USER_TEXT_ENCODING` environment variable. [bug introduced in coreutils-9.8] `'nice'` now limits the adjusted niceness value to its supported range on GNU/Hurd. [This bug was present in "the beginning".] `'numfmt'` no longer reads out-of-bounds memory with trailing blanks in input. [bug introduced with numfmt in coreutils-8.21] `'numfmt'` no longer outputs invalid characters with multi-byte blanks in input. [bug introduced in coreutils-9.5] `'rm -d DIR'` no longer fails on Ceph snapshot directories. Although these directories are nonempty, `'rmdir DIR'` succeeds on them. [bug introduced in coreutils-8.16] `'sort --compress-program'` now diagnoses if it can't write more data to an exited compressor. Previously sort could have exited silently in this case. [bug introduced in coreutils-6.8] `'tail'` outputs the correct number of lines again for non-small `-n` values. Previously it may have output too few lines. [bug introduced in coreutils-9.8] `'unexpand'` no longer triggers a heap buffer overflow with `--tabs` arguments that use the GNU extension `/NUM` or `+NUM` formats. [bug introduced in coreutils-8.28] ** Changes in behavior `'cp'` with default options may again, like with versions before v9.8, miss opportunities to create holes with file systems that support `SEEK_HOLE` only trivially. This change is a consequence of the abovementioned copy offload fix. `'sort --compress-program'` will continue without compressing temporary files if the specified program cannot be executed. Also malformed shell scripts without a "shebang line" will no longer be executed. ** New Features `'numfmt'` now accepts the `--unit-separator=SEP` option, to output or accept a separator between the number and unit. For e.g. "1234 M". ** Improvements `'fmt'`, `'date'`, `'nl'`, and `'pr'` will now exit promptly upon receiving a write error, which is significant when reading large / unbounded inputs. `install`, `sort`, and `split` now use `posix_spawn()` to invoke child programs more efficiently and more independently from their own memory usage. `'numfmt'`: - parses numbers with a non-breaking space character before a unit - parses numbers containing grouping characters from the current locale - supports a multi-byte `--delimiter` character - no longer processes input indefinitely in the presence of write errors `wc -l` now operates 10% faster on hosts that support AVX512 instructions. ** Build-related `chcon` and `runcon` are not built by default if selinux headers are not present, or if the `--without-selinux` configure option is specified. This can be overridden with the `--with-selinux` configure option. `nproc` no longer fails to build with Android API level <= 20. [build issue introduced in coreutils-9.8]

- [Gary Benson: Terminal colours](#) (2025/11/06 22:48)
`\x1B[38;5;105m` and `\x1B[38;5;141m` are a great combo.
- [FSF News: Announcing the winners of the FSF40 Photo Contest](#) (2025/11/06 21:20)

BOSTON, Massachusetts, USA (Thursday, November 6, 2025) – The Free Software Foundation (FSF) today announced the winning photo submissions in the FSF40 Photo Contest held in August.

- [GNU Guix: Update on the Guix Fundraising](#) (2025/11/03 11:00)

We're on our way! It's been a month since we started the fundraising campaign to Sustain and Strengthen Guix. So far we've raised €6562 which is around 40% of our €15000 annual goal. If you'd like to support the project's fundraiser there's still time, pop over to the donate page now! **DONATE NOW** There have been a range of donations, both one-off and recurring. A few people have made large one-off donations, one of over €2150! There have been a couple between €500-€250 and a few more in the €100 range. These are big contributions to our goal, so I want to thank those individuals for helping out so generously. Just over 100 people (115 right now) have stepped forward to become recurring supporters, pledging a monthly amount to help the project. This is key because it means the project knows there's a regular stream of donations that can pay for the shared resources that we all use. There's been great support with a few people donating €30-€50 a month which is fantastic, the rest at the €10-€15 a month - and one person managed to use the recurring button multiple times to get precisely the amount they wanted to donate monthly! The result is that Open Collective estimates €657.50 a month of recurring donations, and Stripe estimates €720 a month of recurring donations. This is significant because if each person is able to continue giving monthly then annually we'd estimate around €16500 of donations. The maths is simple, the impact significant - a recurring donation of €10 a month is worth €120 a year, that's why recurring donations make such a difference! Of course, people's situations change and they may stop supporting Guix - we've had a couple of cancellations already. So in terms of the actual money we've received we're at ~40% of the €15000 target which I think is pretty good! Thanks to everyone who's supported Guix by donating, you're making a difference and we really appreciate it! If you haven't done it yet, and would like to jump in to support the project then now's a great time! A recurring donation is ideal, but we appreciate any support you can give and every donation gets us a bit closer! **DONATE NOW** Spreading the word Guix is a global community of people, we've had donations from so many places. Where ever you are, it's amazing to think of so many people enjoying, supporting and contributing to Guix. As we're distributed all over the globe we don't have that many ways to keep people informed about the project. I'm sure there are many Guix users who don't know the project needs support. You can help spread the word that Guix is running a fundraiser by talking about it and using this badge. Put it on your social media, your web site or your Git forge account! Thanks to Luis Felipe for creating it. What's next The next few weeks will tell us how many people are able to donate to Guix and the annual budget the project has so that it's sustainable. Then we'll be able plan where we can sustain Guix and where we can do new things to strengthen the project. My goal is for the next blog post is to provide an update on both our fundraising campaign and how we're using the donations that we've received.

- [unifont @ Savannah: Unifont 17.0.03 Released](#) (2025/11/01 19:13)

1 November 2025 Unifont 17.0.03 is now available. This is a minor release aligned with Unicode 17.0.0. This release updates and adds over 100 Chinese ideographs. Download this release from GNU server mirrors at: [https://ftpmirror ... /unifont-17.0.03/](https://ftpmirror.gnu.org/unifont-17.0.03/) or if that fails, [https://ftp.gnu.o ... /unifont-17.0.03/](https://ftp.gnu.org/pub/fonts/unifont-17.0.03/) or, as a last resort, [ftp://ftp.gnu.org ... /unifont-17.0.03/](ftp://ftp.gnu.org/pub/fonts/unifont-17.0.03/) These files are also available on the unifondry.com website: [https://unifondr ... /unifont-17.0.03/](https://unifondry.com/unifont-17.0.03/) Font files are in the subdirectory [https://unifondr ... 0.03/font-builds/](https://unifondry.com/unifont-17.0.03/font-builds/) A more detailed description of font changes is available at [https://unifondr ... nifont/index.html](https://unifondry.com/unifont/index.html) and of utility program changes at [https://unifondr ... nt-utilities.html](https://unifondry.com/unifont-utilities.html) Enjoy! Paul Hardy GNU Unifont Maintainer

- [Parabola GNU/Linux-libre: dovecot >= 2.4 requires manual intervention](#) (2025/11/01 19:07)

From Arch: The dovecot 2.4 release branch has made breaking changes which result in it being incompatible with any ≤ 2.3 configuration file. Thus, the dovecot service will no longer be able to start until the configuration file was migrated, requiring manual intervention. For guidance on the 2.3-to-2.4 migration, please refer to the following upstream documentation: [Upgrading Dovecot CE from 2.3 to 2.4](#) Furthermore, the dovecot 2.4 branch no longer supports their replication feature, it was removed. For users relying on the replication feature or who are unable to perform the 2.4 migration right now, we provide alternative packages available in [extra]: dovecot23 pigeonhole23 dovecot23-fts-elastic dovecot23-fts-xapian The dovecot 2.3 release branch is going to receive critical security fixes from upstream until stated otherwise.

- [Amin Bandali: Free software activities in October 2025](#) (2025/10/31 23:44)

Hello and welcome to my October free software activities report. GNU & FSF GNU Spotlight: I prepared and sent the October GNU Spotlight to the FSF campaigns team, who will review and publish it on the FSF's community blog and as part of the next issue of the monthly Free Software Supporter newsletter. GNU Emacs: bug#79629: I noticed that I was unable to customize the holiday-other-holidays variable using the setopt macro: my change did not seem to take effect. As Eli Zaretskii helpfully pointed out, this was because customizing holiday-other-holidays did not recompute the value of calendar-holidays, which is computed once, when the package is loaded. So I prepared and sent a patch 500a2d0cc55 to recompute calendar-holidays when its components are set. bbabc1db258: While reading about custom-reevaluate-setting in the Startup Summary node of the GNU Emacs Lisp reference manual I noticed a small typo, so I committed a patch to fix it. Misc The Free Software Foundation celebrated its fortieth birthday on 4 October 2025 online and in person in Boston! I was not able to attend the event in person, so I recorded a video for the FSF40 volunteer panel held at the venue. This month at work one of our Elasticsearch clusters experienced partial failure, and we needed to extract document IDs from a backup of one of the cluster's shards. Elasticsearch uses Lucene under the hood and each shard is a standalone Lucene index, so I used Lucene's Java API to write a little GetIDS class to query the index for all of its documents, and for each document print its _id field, decoding the binary-valued BytesRef as needed. The gotcha was that all of the BytesRefs seemed to have a -1 byte in the beginning, throwing off the recommended BytesRef.utf8ToString() method, so I had to reimplement that method's logic in my program and have it use an adjusted offset + 1 and length - 1 instead. That's about it for this month's report. Take care, and so long for now.

- [Amin Bandali: Free software activities in October 2025](#) (2025/10/31 23:44)

Hello and welcome to my October free software activities report. GNU & FSF GNU Spotlight: I prepared and sent the October GNU Spotlight to the FSF campaigns team, who will review and publish it on the FSF's community blog and as part of the next issue of the monthly Free Software Supporter newsletter. GNU Emacs: bug#79629: I noticed that I was unable to customize the holiday-other-holidays variable using the setopt macro: my change did not seem to take effect. As Eli Zaretskii helpfully pointed out, this was because customizing holiday-other-holidays did not recompute the value of calendar-holidays, which is computed once, when the package is loaded. So I prepared and sent a patch 500a2d0cc55 to recompute calendar-holidays when its components are set. bbabc1db258: While reading about custom-reevaluate-setting in the Startup Summary node of the GNU Emacs Lisp reference manual I noticed a small typo, so I committed a patch to fix it. Misc The Free Software Foundation celebrated its fortieth birthday on 4 October 2025 online and in person in Boston! I was not able to attend the event in person, so I recorded a video for the FSF40 volunteer panel held at the venue. This month at work one of our Elasticsearch clusters experienced partial failure, and we needed to extract document IDs from a backup of one of the cluster's shards. Elasticsearch uses Lucene under the hood and each shard is a standalone Lucene index, so I used Lucene's Java API to write a little GetIDS class to query the index for all of its documents, and for each document print its _id field, decoding the binary-valued BytesRef as needed. The gotcha was that all of the BytesRefs seemed to have a -1

byte in the beginning, throwing off the recommended `BytesRef.utf8ToString()` method, so I had to reimplement that method's logic in my program and have it use an adjusted offset + 1 and length - 1 instead. That's about it for this month's report. Take care, and so long for now.

- [parallel @ Savannah: GNU Parallel 20251022 \('Goodall'\) released \[stable\]](#) (2025/10/30 22:58)

GNU Parallel 20251022 ('Goodall') has been released. It is available for download at: `lbry://@GnuParallel:4` Quote of the month: idk who built GNU parallel but I owe them a beer -- ram @h4x0r1ng New in this release: No new features. Bug fixes. GNU Parallel - For people who live life in the parallel lane. If you like GNU Parallel record a video testimonial: Say who you are, what you use GNU Parallel for, how it helps you, and what you like most about it. Include a command that uses GNU Parallel if you feel like it. About GNU Parallel GNU Parallel is a shell tool for executing jobs in parallel using one or more computers. A job can be a single command or a small script that has to be run for each of the lines in the input. The typical input is a list of files, a list of hosts, a list of users, a list of URLs, or a list of tables. A job can also be a command that reads from a pipe. GNU Parallel can then split the input and pipe it into commands in parallel. If you use `xargs` and `tee` today you will find GNU Parallel very easy to use as GNU Parallel is written to have the same options as `xargs`. If you write loops in shell, you will find GNU Parallel may be able to replace most of the loops and make them run faster by running several jobs in parallel. GNU Parallel can even replace nested loops. GNU Parallel makes sure output from the commands is the same output as you would get had you run the commands sequentially. This makes it possible to use output from GNU Parallel as input for other programs. For example you can run this to convert all jpeg files into png and gif files and have a progress bar: `parallel --bar convert {1} {1.}.{2} ::: *.jpg ::: png gif` Or you can generate big, medium, and small thumbnails of all jpeg files in sub dirs: `find . -name '*.jpg' | parallel convert -geometry {2} {1} {1//}/thumb{2}_{1/} :::: - ::: 50 100 200` You can find more about GNU Parallel at: <http://www.gnu.org/s/parallel/> You can install GNU Parallel in just 10 seconds with: `$ (wget -O - pi.dk/3 || lynx -source pi.dk/3 || curl pi.dk/3 || \ fetch -o - http://pi.dk/3) > install.sh $ sha1sum install.sh | grep c555f616391c6f7c28bf938044f4ec50 12345678 c555f616391c6f7c 28bf9380 44f4ec50 $ md5sum install.sh | grep 707275363428aa9e9a136b9a7296dfe4 70727536 3428aa9e 9a136b9a 7296dfe4 $ sha512sum install.sh | grep b24bfe249695e0236f6bc7de85828fe1f08f4259 83320d89 f56698ec 77454856 895edc3e aa16feab 2757966e5092ef2d 661b8b45 b24bfe24 9695e023 6f6bc7de 85828fe1 f08f4259 6ce5480a 5e1571b2 8b722f21 $ bash install.sh` Watch the intro video on <http://www.youtub...L284C9FF2488BC6D1> Walk through the tutorial (`man parallel_tutorial`). Your command line will love you for it. When using programs that use GNU Parallel to process data for publication please cite: O. Tange (2018): GNU Parallel 2018, March 2018, <https://doi.org/10.1111/zenodo.1146014>. If you like GNU Parallel: Give a demo at your local user group/team/colleagues Post the intro videos on [Reddit/Diaspora*/forums/blogs/](#) [Identi.ca/Google+/Twitter/Facebook/Linkedin/](#) mailing lists Get the merchandise <https://gnuparall...igns/gnu-parallel> Request or write a review for your favourite blog or magazine Request or build a package for your favourite distribution (if it is not already there) Invite me for your next conference If you use programs that use GNU Parallel for research: Please cite GNU Parallel in your publications (use `--citation`) If GNU Parallel saves you money: (Have your company) donate to FSF <https://my.f...org/donate/> About GNU SQL GNU sql aims to give a simple, unified interface for accessing databases through all the different databases' command line clients. So far the focus has been on giving a common way to specify login information (protocol, username, password, hostname, and port number), size (database and table size), and running queries. The database is addressed using a DBURL. If commands are left out you will get that database's interactive shell. When using GNU SQL for a publication please cite: O. Tange (2011): GNU SQL - A Command Line Tool for Accessing Different Databases Using DBURLs, ;login: The USENIX Magazine, April 2011:29-32. About GNU Niceload GNU niceload slows down a program when the computer load average (or other system activity) is above a certain limit. When the limit is reached the program will be suspended for some time. If the limit is a

soft limit the program will be allowed to run for short amounts of time before being suspended again. If the limit is a hard limit the program will only be allowed to run when the system is below the limit.

From:

<https://wiki.tromjaro.alexio.tf/> - **TROMjaro wiki**

Permanent link:

<https://wiki.tromjaro.alexio.tf/doku.php?id=news:planet:gnu>

Last update: **2021/10/30 11:41**

