

Overview of the Debian project



whoami

Nicolas Dandrimont

Backend engineer at [Software Heritage](#)

Contributes to Debian since 2009; Member of the project since 2013

Debian Account Manager; Maintainer of the piuparts.debian.org QA system



What is Debian?

One of the oldest community distros (since 1993!)

A community of thousands of volunteers building *The Universal Operating System*

- a distribution of Linux
- almost 65 thousand software packages
- stable release every 2 years, each supported for 5+ years
- across 9 supported architectures



Project organization

```
apt install anarchism
```



Foundational documents



Debian Social Contract

- Debian will remain 100% free
- We will give back to the free software community
- We will not hide problems
- Our priorities are our users and free software



Debian Free Software Guidelines

10 key points to delineate what is Free and what is not.
Used as base for the Open Source Definition.



Diversity statement and Code of conduct

Documents that our community has agreed to uphold and abide by.



Debian Constitution

The interaction between parts of the project are codified in the Debian Constitution



Contributors

Individual contributors, and teams, are largely self-organized.

Being a group of volunteers, the project cannot impose work on anyone, but conversely, noone should be impeding anyone else's work



Project members

Members of the project (“[Debian Developers](#)”) are coopted through the [New Member process](#) after having shown their social and technical abilities. Some members of the project have unrestricted upload permissions to the archive (“[Uploading Debian Developers](#)”).



Member Rights

All members who have gone through the NM process have voting rights to:

- pass general resolutions
- elect the Debian Project Leader

General Resolutions are our last-resort means of taking decisions as a project.

- 5 project members need to second (co-sign) a GR proposal
- Condorcet (ranked) voting system



Sponsored Maintainers

Anyone can propose a package to be uploaded. An established Debian Developer needs to “sponsor” the upload by reviewing it and signing off on it (and taking responsibility).

Debian Maintainers

Debian Maintainer status allow DDs to delegate upload permissions (and only those) of some packages to the people that they have vetted.



Technical Committee

When the jurisdiction of different developers overlap, on technical matters related to the distribution, the constitution empowers the [Technical Committee](#) to make a decision.

The tech-ctte is a body of 8 developers who are self-selected for a 4-year term.

TC decisions can always be overridden by project members (with a 2:1 majority GR).



The Debian Project Leader

The Debian Project Leader has authority on anything that doesn't fall under the jurisdiction of any individual developer.

The DPL is elected for a 1-year term, renewable.

The DPL position is explicitly not technical, the DPL sets a vision for the project in their platform and serves as representative of the project.

The Project Leader can delegate other members of the project to have the authority to perform certain tasks.



Delegates

Account managers, archive maintainers (“FTP team”), release team, system administrators, community team, treasurer team, ...

Decisions by delegates and the Project Leader can be overridden by developers (via a General Resolution).

Overview of the Debian release process



Development cycle

unstable and testing

New versions of packages are uploaded to the unstable distribution. After a delay, allowing time for some automated and manual QA processes, packages migrate to the testing suite.



Freeze, release

Regularly (once every two years), the Release team enacts a “freeze” which gradually prevents the automated migration of packages, in favor of stabilizing the testing suite to make it releasable.

Once enough bugs have been fixed in the testing suite, the Release Team performs a stable release by copying the testing suite into the (now new) stable suite.



Release QA testing

The aim of the freeze process is to reduce the number of “release-critical” bugs in the testing suite to zero.

There’s an important focus on supporting upgrades (from one stable version to the next). In the age of cattle, we still care (a lot!) about a pet or two.



Debian release schedule

“Debian releases when it’s ready”

Version	Codename	Release date	Interval
3.1	Sarge	6 June 2005	
4.0	Etch	8 April 2007	22 mos
5.0	Lenny	14 February 2009	22 mos
6.0	Squeeze	6 February 2011	24 mos
7	Wheezy	4 May 2013	27 mos



Version	Codename	Release date	Interval
8	Jessie	25–26 April 2015	23 mos
9	Stretch	17 June 2017	26 mos
10	Buster	6 July 2019	25 mos
11	Bullseye	14 August 2021	25 mos
12	Bookworm	10 June 2023	22 mos

2-year ($\pm 10\%$) release cycle, consistent over the past 20 years!



Lifecycle of stable releases

Security support

The [Security Team](#) provides security support for stable releases on all* packages, for up to 1 year after the release of the next stable (~3 years)

Security updates are provided in real-time, and published on the [debian-security-announce](#) mailing-list.

[unattended-upgrades](#) can be configured to automatically install security (and other) updates.



Stable updates

The [Stable Release Managers](#) handle updates for all packages in stable, in point-releases published bi-monthly.

Maintainers upload updated packages into proposed-updates until they're merged into stable.

Useful QA practice: enable proposed-updates on some canary systems to test updates.

The final point release for non-security bugs is done around the same time as the end of support by the security team (3 years after release).



Long Term Support

Long Term Support is provided by another team for 2 more years (5 years of total support).

The scope of LTS is reduced (to a limited subset of architectures, and some packages, such as web browsers, are excluded)



LTS infrastructure and sponsorship

Long Term Support is based on community infrastructure (it uses the same archive as the security updates so that users get a seamless transition). [Regular reports](#) are published.

LTS is supported by [a sponsorship program](#) run by Freexian, a consultancy started by a Debian Developer to fund his (and other Developers') activities around and within Debian.



Beyond LTS

Freexian offers a bespoke **Extended LTS service** where the support for requested packages is extended up to 10 years.

Other consultancies offer similar services, as well as support for upgrade planning and execution.



Release lifecycle recap

- New stable releases every 2 years, with upgrade support
- Support by Security team & SRM for 3 years
- Scope-reduced, sponsored LTS support for 2 additional years
- Targeted paid ELTS support for 5 additional years



Debian users



Who is using Debian?

No formal survey or “analytics”

Lots of informed guessing from interactions at community events (e.g. DebConf, meetups, etc.)

Millions of direct users (plausibly), and millions of users of derivatives (undoubtedly).

Deployments small and large (fleets of thousands of machines are not unheard of). Public administrations, private institutions, hobbyists, professionals.



Challenges of developing Debian

- Focus on the DFSG / license orthodoxy
- Strong aversion to vendored dependencies
- Tooling updates for static linked ecosystems (rust, go) are lagging



Challenges of using Debian

- (some) lack of upstream recognition for software support
- lack of recognition by hardware vendors

Not any different from other community operating systems, in my experience.



Why Debian?

Single-vendor open source solutions aren't all that different from proprietary systems. In the end, you're stuck with their business decisions.

Debian provides a community-driven, multi-vendor base that you can build your systems on, with extensive generic and bespoke support options available.

Extensive upgrade support, built by the distro, is a strong differentiator (kudos to AlmaLinux for building that for RHEL variants)



Getting help, giving back



Packaging teams

Debian provides a lot (more than 50 thousand!) packages in its distribution. Of course there is disparity in the level of support and care, even though security support and stable updates theoretically apply to the whole distribution.

The best way to effect the direction of the distribution is to contribute to packaging teams. Packaging teams are also the first point of contact if you need help with some packages.



Teams of interest to this audience:

- Science Team
- Debian HPC mailing-list
- lots of language-specific packaging teams (Rust, Python, Java, etc.)



Other support resources

For ad-hoc support, the project maintains a bunch of [public mailing lists](#) with searchable archives.

Bugs for the distribution are tracked in the [Debian Bug tracking system](#), which is largely driven by email.

There's real-time discussions on a bunch of [IRC channels](#), a lot of which are bridged to Matrix



Getting support

Aside from community resources, there are many companies, large and small, that offer contracts to support users of Debian with their specific needs.

See the [very extensive list of consultants](#)



