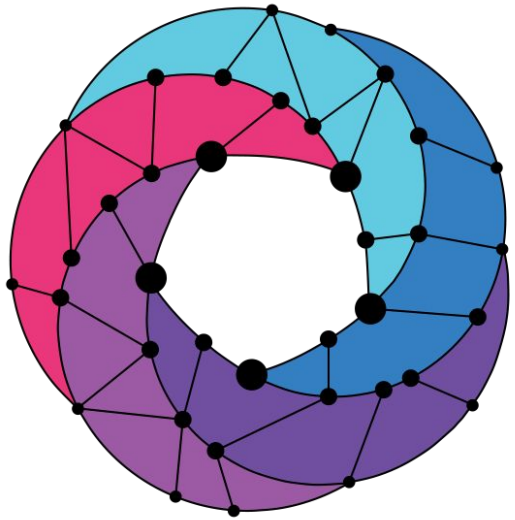


Open Source
MANO

LTS for OSM Community

Fabián Bravo, Mark Beierl (Canonical)

ETSI

What is LTS?

Long-term support (LTS) is a product lifecycle management policy in which a stable release of computer software is maintained for a longer period of time than the standard edition

Short term support (STS) is a term that distinguishes the support policy for the software's standard edition. STS software has a comparatively short life cycle, and may be afforded new features that are omitted from the LTS edition to avoid potentially compromising the stability or compatibility of the LTS release.

https://en.wikipedia.org/wiki/Long-term_support

Why Do We Need it?

- Production use
  - Upgrading with every release is not feasible
  - OSM might drop support before production is ready to upgrade
  - Consumers of OSM rely on having bug and security fixes
- Reduction of overhead
  - Community actively supports 3 releases: master, latest, latest-1
  - Bug fixes must go into all branches
  - With LTS, point releases for STS can be reduced

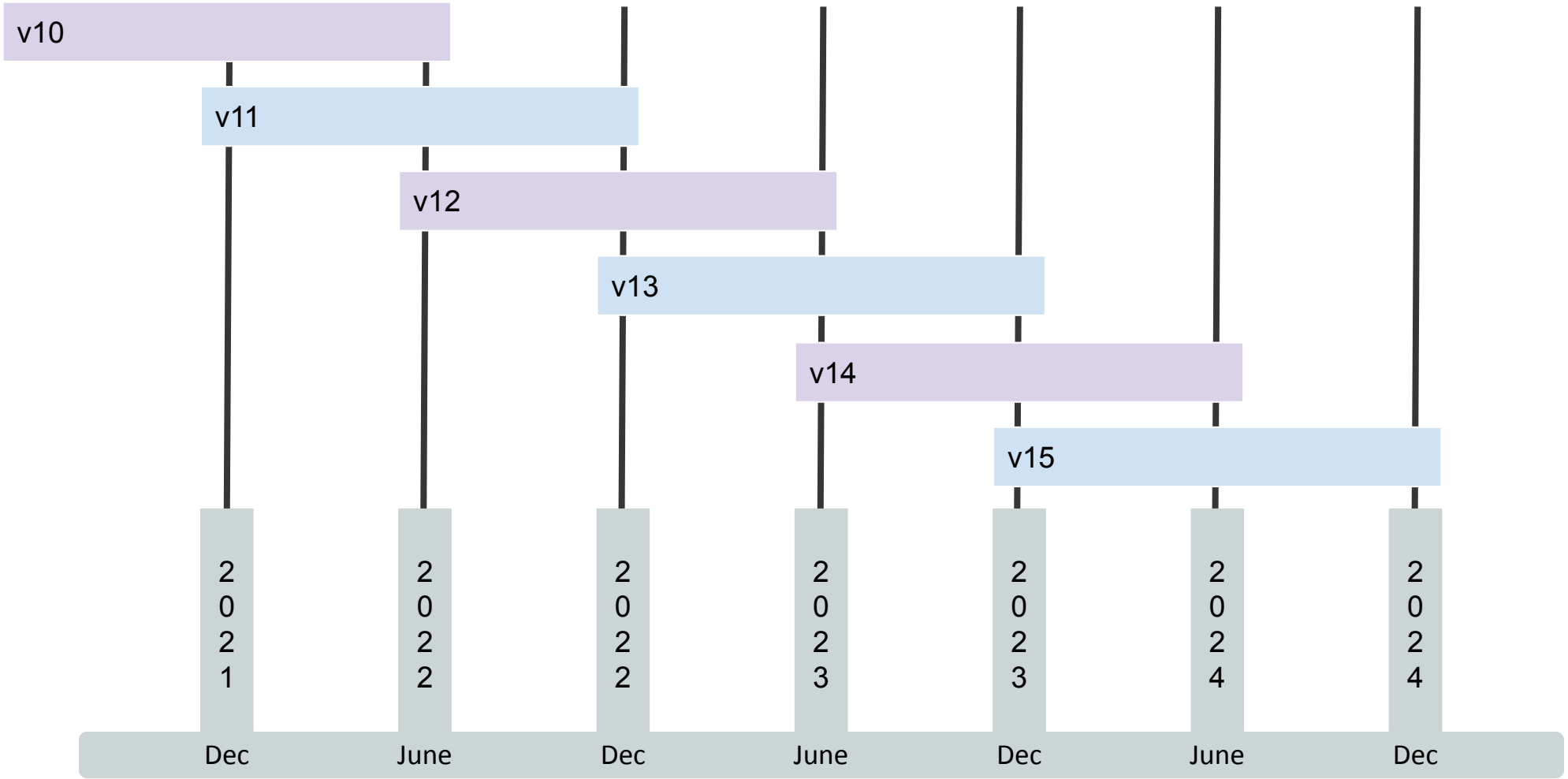
Patterns of LTS

- Linux kernel: even/odd
  - Even for production use, odd for feature development/unstable
- Ubuntu
  - Designate specific releases as LTS, interim releases are STS
- OpenStack
  - Maintained, Extended Maintenance, EOL
- K8s
  - No LTS
- ArchLinux
  - Rolling release, it has a stable branch and a testing branch

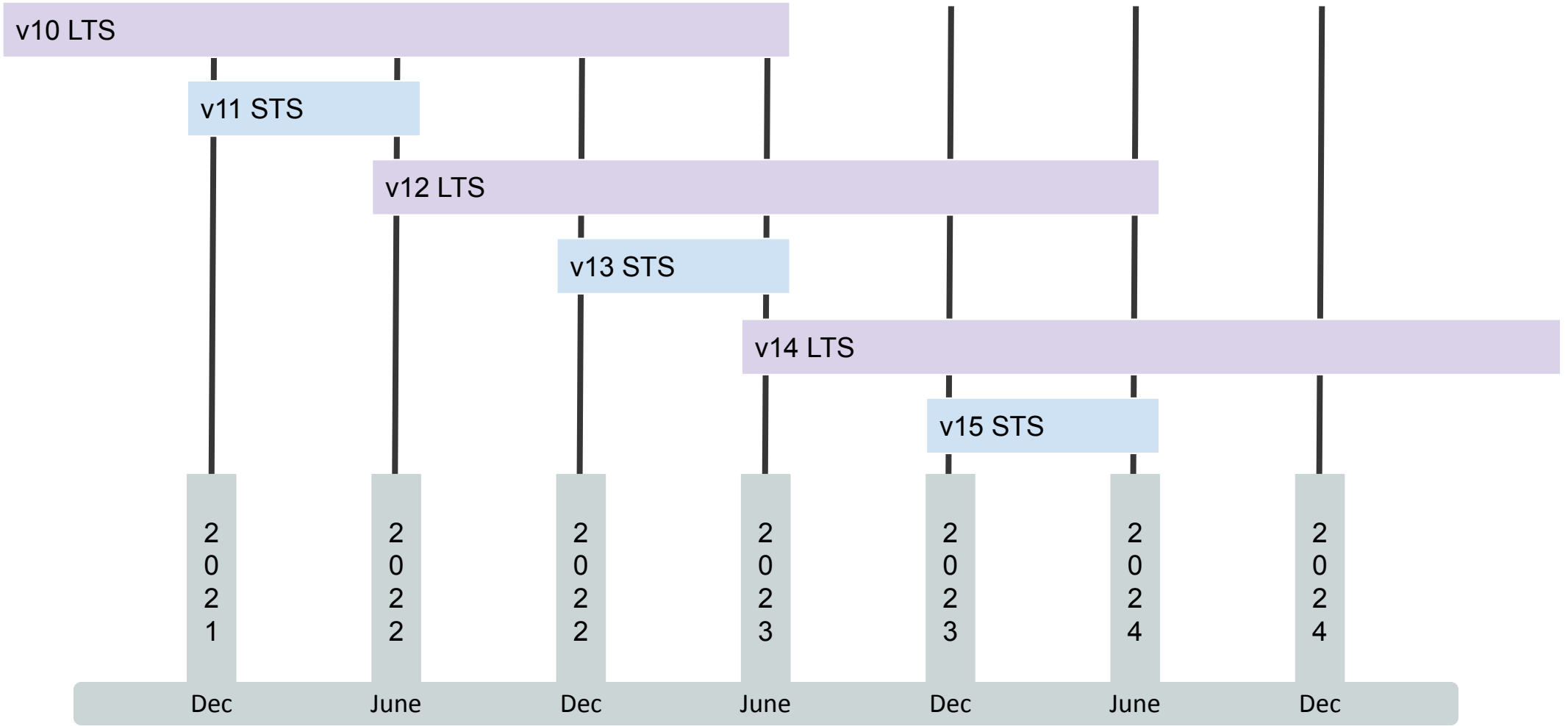
Examples of LTS

Software	LTS Length	STS Length
Debian GNU/Linux	5 Years	3 Years
Django	3 Years	16 Months
Firefox	1 Year	6 Weeks
Linux Mint	5 Years	6 Months
Node.js	18 Months	12 Months
Ubuntu	5 Years / 10 Years ESM	9 Months
Windows 10	10 Years	18 Months

What We Do Now



LTS for 24 Months, STS for 6 Months



What does that mean for bug fixes?

- LTS
  - Production bugs
  - Security patches
  - Minor bugs or bugs not experienced in production might not be back ported
  - Protect integrity of branch
  - **No new features in LTS point releases**
  - Point releases on demand for security/critical issues
- STS
  - Any bug reported against that version
  - Possibility of new feature in point release
  - Point releases on fixed schedule

New Features

- LTS
  - Firm cut off date: feature must be merged 1 month before LTS release
  - Release notes must include list of all bugs fixed
  - Rigorous testing needed
  - Documented and tested upgrade procedures from LTS to LTS
- STS
  - Lax cut off date for features
  - Features can be put into point release
  - Release notes may be limited to features only
  - Upgrades not guaranteed

Procedure for Approving Changes in LTS

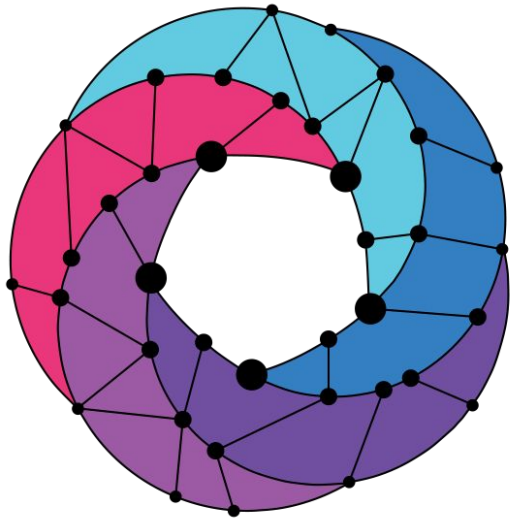
- LTS must be tightly controlled and kept stable
- Only approved bug fixes should go into LTS
- Need to define procedure
  - How to recognize production bugs
  - What to do with trivial bugs from STS/master development
  - What level of testing is needed
  - Robot test or at least unit tests added for every bug fix!

Challenges

- 2 year span and support for helper libraries
- VIM and K8s cluster connections/API can be deprecated along the way, the upgrade path is not clear sometimes
- Helper "modules" or applications also have their lifecycle

What version to keep as LTS

- V10.0 [X]
- v11.0
- v12.0



Open Source
MANO

Thank you