Upstream First: Meta's Linux Userspace, meet Linux Distributions

Working across distributions for fun and profit

Michel Salim
Production Engineer

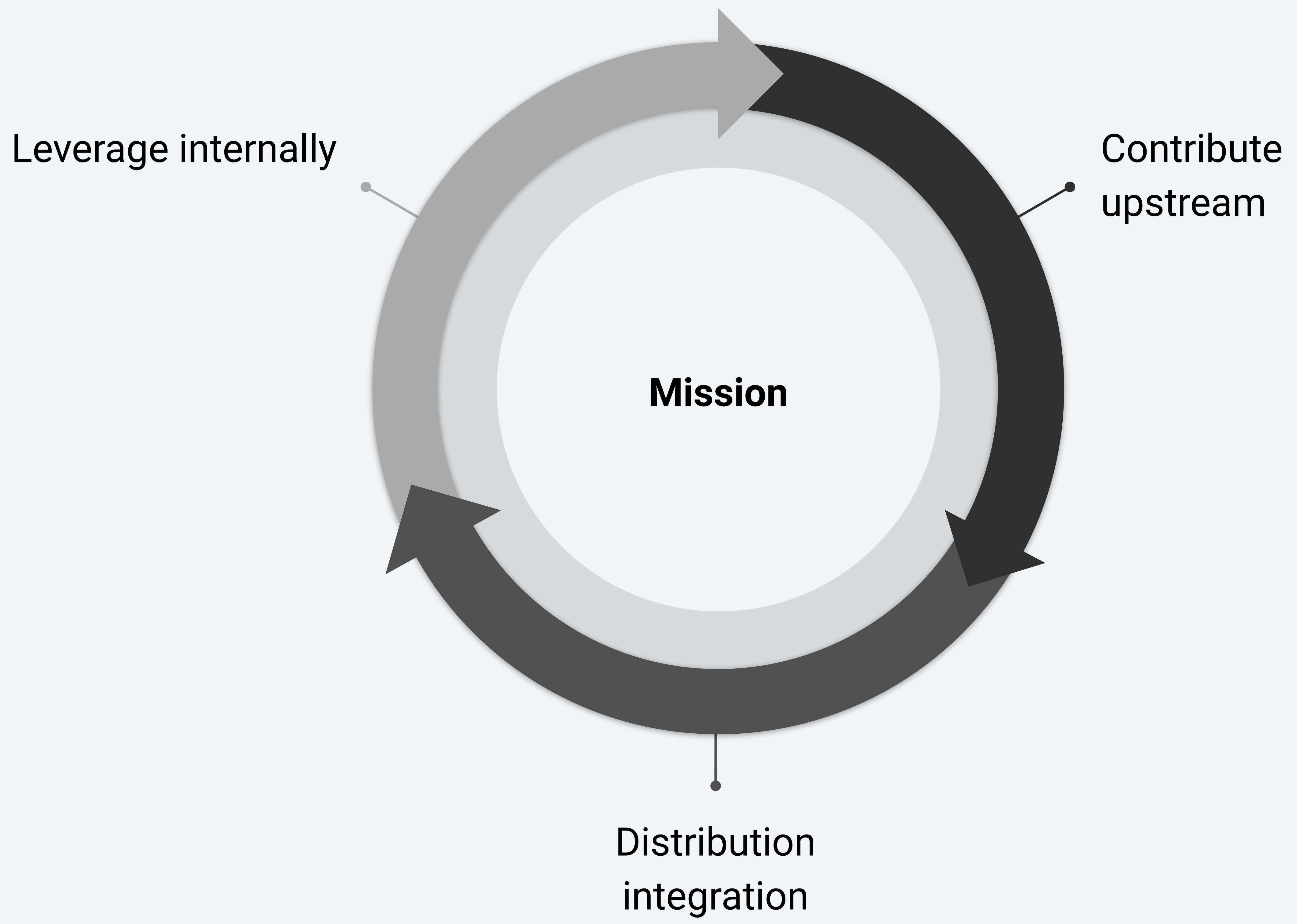
Meta

Agenda

About me

- Production Engineer at Meta
- Fedora contributor since 2003 (FAS: salimma)
- Debian Maintainer since 2022 (username: michelin)
- Mastodon: @michel_slm@floss.social
- Matrix: @michel:one.ems.host / @salimma:fedora.im
- Web: michel-slm.name

LINUX USERSPACE



Audience

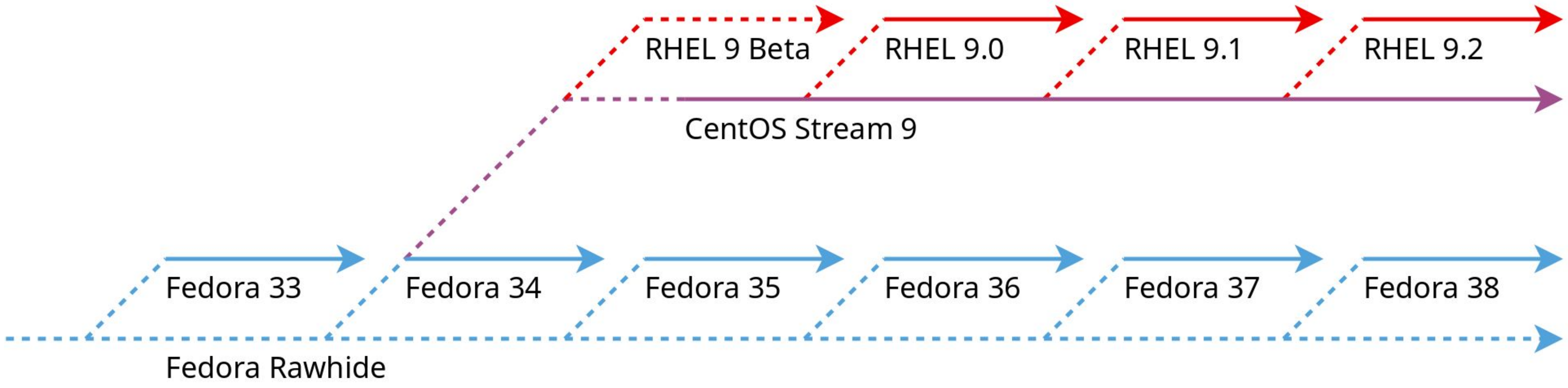
- Open source community members
- Companies / company employees interested in engaging with open source communities

What we use (and why)

- Custom data centers
- Millions of servers
- Containerized workloads using Twine
- CentOS Stream (currently migrating from 8 -> 9)
-

CentOS Stream

- Downstream of Fedora, upstream of Red Hat Enterprise Linux / Alma / Rocky
- Collaborative development
- ABI compatibility
- We upgrade key components (kernel, systemd, etc.)


RHEL 9 Beta
RHEL 9.0
RHEL 9.1
RHEL 9.2
CentOS Stream 9
Fedora 33
Fedora 34
Fedora 35
Fedora 36
Fedora 37
Fedora 38
Fedora Rawhide

Fedora

- Supported on desktop/laptop systems
- Chosen for hardware support* and compatibility with prod
- Ideal for upstream development work

Ubuntu

- CI for open source projects
- Used to be community supported for desktop use

Rationale for contributing

Why contribute to distributions?

- Make it easier for the community to consume our projects
- Discover correctness issues in less commonly used configurations (architectures / compiler / compiler settings)
- Make it easier for us to keep up with distribution releases
  - Minimize changes we carry internally
- DRY
  - Distributions are actually quite good at … building Linux distributions

Fedora

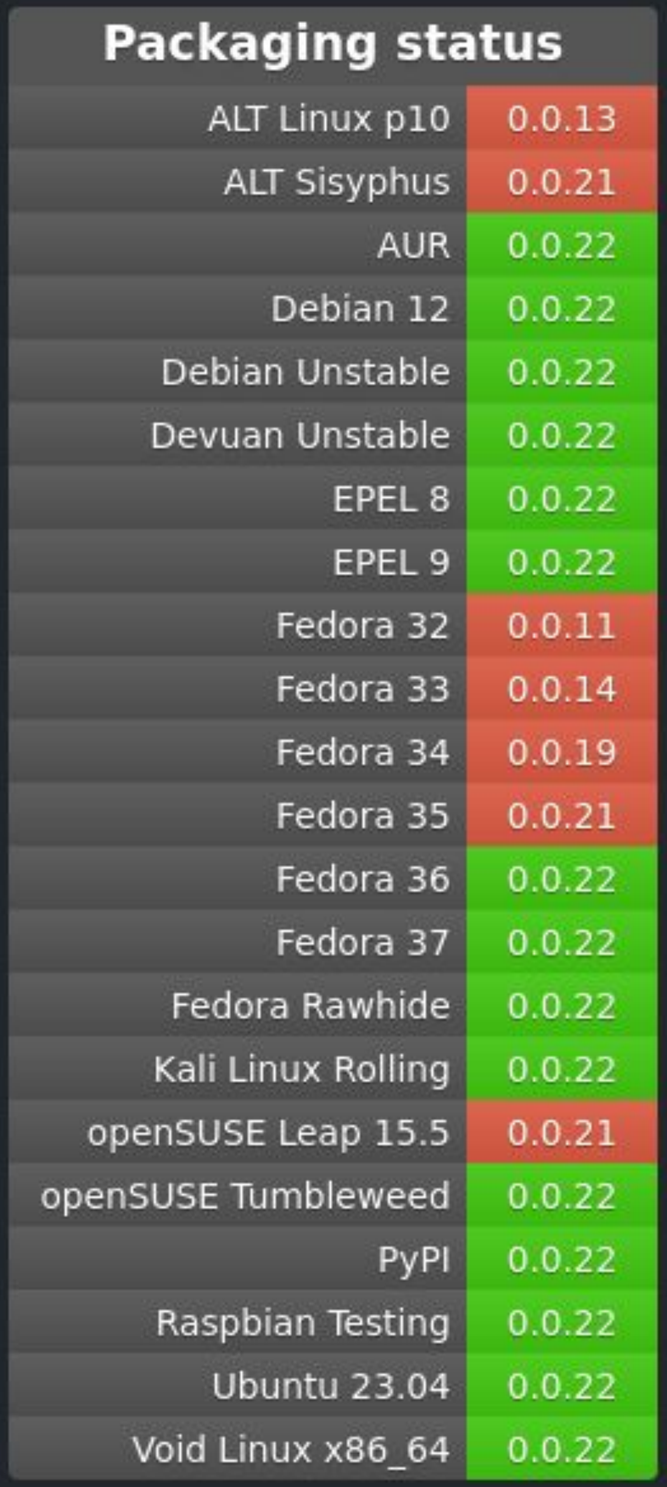
- Upstream for CentOS Stream
- Indirect upstream of RHEL and rebuilds (AlmaLinux, Rocky Linux)
- Can’t run CentOS Stream etc. without Extra Packages for Enterprise Linux (EPEL), which is part of Fedora

CentOS Stream

- Need to customize stock offering
- Sharing the burden
- Better Engineering: avoid “fire and forget” internal packages
- Minimize the delta between what we use and stock CentOS Stream

Debian

Upstream for Ubuntu and many other derivatives



Packaging status	
ALT Linux p10	0.0.13
ALT Sisyphus	0.0.21
AUR	0.0.22
Debian 12	0.0.22
Debian Unstable	0.0.22
Devuan Unstable	0.0.22
EPEL 8	0.0.22
EPEL 9	0.0.22
Fedora 32	0.0.11
Fedora 33	0.0.14
Fedora 34	0.0.19
Fedora 35	0.0.21
Fedora 36	0.0.22
Fedora 37	0.0.22
Fedora Rawhide	0.0.22
Kali Linux Rolling	0.0.22
openSUSE Leap 15.5	0.0.21
openSUSE Tumbleweed	0.0.22
PyPI	0.0.22
Raspbian Testing	0.0.22
Ubuntu 23.04	0.0.22
Void Linux x86_64	0.0.22

Ubuntu

- We don’t contribute directly yet, because
- Roughly similar to RHEL, packages get imported from upstream (Debian) until freeze
- Post-freeze changes need to be justified
  - New package process
  - Stable release updates

How and what do we contribute

How

- Follow established processes
- Contribute as individuals, not as a corporate entity
- No special treatment

Fedora

- Change Proposals
- Package maintenance
- Extra Packages for Enterprise Linux
  - Governance
  - Packaging

Fedora Changes

Sometimes we succeed…

- Fedora 33: Btrfs By Default
- Fedora 34: Enable systemd-oomd by default for all variants
- Fedora Linux 35: Fedora Cloud Btrfs By Default
- Fedora Linux 38: -fno-omit-frame-pointer

Fedora Changes

… sometimes we don't

- Fedora Linux 37: Enable fs-verity in RPM

Shameless plug: some cool projects we maintain

- On Fedora, they are a `dnf install` away
- below (an interactive tool to view and record historical system data)
  - “not atop”
- drgn (a programmable debugger written in Python)
- pystemd (Python library to talk to systemd over dbus; see also the Friday workshop)
- systemd-mkosi (build bespoke OS images)

What is EPEL?

- Extra Packages for Enterprise Linux
- See Carl George's talk from Saturday: The Road to EPEL 9

Fedora, RHEL/CentOS Stream, EPEL

- A subset of Fedora is branched off for CentOS Stream
- RHEL minor releases are cut from CentOS Stream
- Packages in RHEL get official Red Hat support
- Anything else is eligible for EPEL (Extra Packages for Enterprise Linux)
- For the old timers, remember the Fedora Core vs Extras split?

Stale requests

- Many Fedora maintainers are not interested in EPEL
- Most are volunteers so they might not check BZ that often
- For general maintenance, provenpackagers can help
- But branch requests require someone in the ACL
- The EPEL Steering Committee created Stalled EPEL Requests for this

ebranch

- Calculates transitive closure of missing build time dependencies
  - Upcoming: adding support for install time dependencies
- Calculates chain build ordering
- File Bugzilla issues requesting missing builds
- Talks
  - Bootstrapping new EPEL releases with ebranch
    - CentOS Dojo, FOSDEM 2022
  - One year on: Experiences using ebranch to bring over Fedora packages to EPEL
    - CentOS Connect, FOSDEM 2023

Fedora: ebranch walkthrough

```
$ ebranch
Usage: ebranch [OPTIONS] COMMAND [ARGS]...

  Tool for branching Fedora packages for EPEL

Options:
  --help  Show this message and exit.

Commands:
  dependencies  Commands for working with dependencies
  issues        Commands for issue tracker integration
  version       Display ebranch version information
```

What BRs are missing?

```
$ ebranch dependencies missing-build-reqs -f epel9.json python-b4 epel9
{
  "python-b4": {
    "build": {
      "python-dkimpy": [
        "(python3dist(dkimpy) >= 1 with python3dist(dkimpy) < 2)",
        "(python3dist(dkimpy) >= 1.0.5 with python3dist(dkimpy) < 1.1)"
      ],
      "python-patatt": [...] } } }
```

Filing branch requests

```
$ ebranch issues file-request --fas salimma --sig --blocked
<python_b4_bzid> python-dkimpy
```

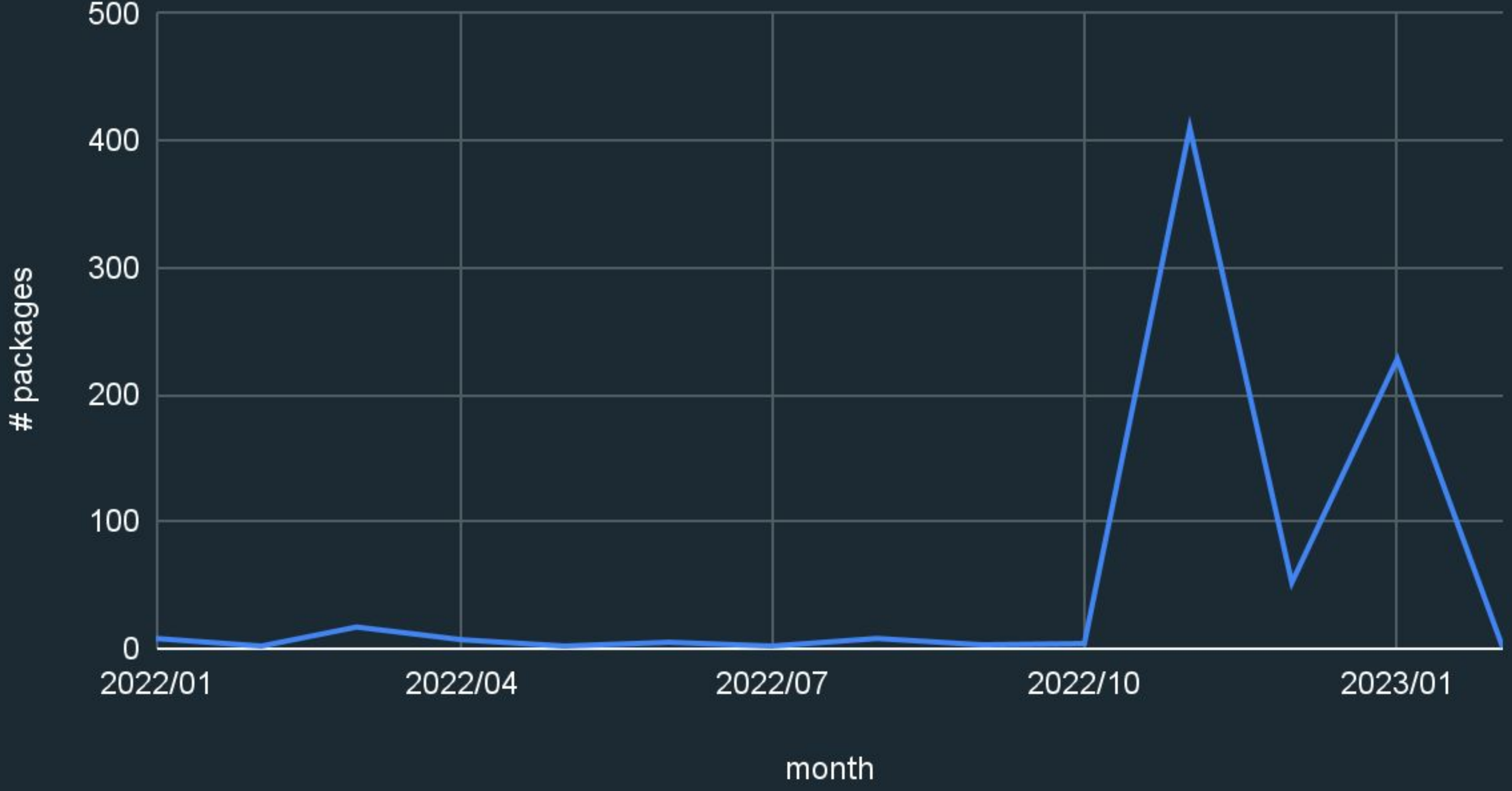
Chain building

```
$ ebranch dependencies calculate-chain-build epel9.json
python-dkimpy python-patatt : python-b4

# invoke fedpkg chain-build from any of the projects in the last
group, after removing it from the list
# will be nice to make fedpkg chain-build support out-of-directory
builds (by passing the branch explicitly)
```

```
$ bodhi-cli query-updates --releases EPEL-9 --users salimma \
  --type newpackage --submitted-since 2022-01-01 --status stable \
  | grep -E '\.el9$' | wc -l
749
$ bodhi-cli query-updates --releases EPEL-9 --users salimma \
  --type newpackage --submitted-since 2022-01-01 --status stable \
  | grep -E '^rust-.*\.el9$' | wc -l
686
```

new packages per month



CentOS Stream

- Hyperscale SIG (talks)
  - CentOS Dojo FOSDEM 2021: Hyperscale SIG Introduction (Davide Cavalca)
  - SCALE 19x: Building the Future with CentOS Stream (Davide Cavalca)
  - CentOS Dojo, DevConf.US 2022: Adventures with systemd in Hyperscale (Anita Zhang, Daan De Meyer)
- CentOS Board of Directors

CentOS Hyperscale SIG: Mission

The Hyperscale SIG focuses on enabling CentOS Stream deployment on large-scale infrastructures and facilitating collaboration on packages and tooling.

What is in Hyperscale?

- Packages that upgrade the ones in CentOS Stream
- Packages that enable additional functionalities
  - Btrfs
  - CPU optimizations (e.g. zlib in hs+intel)
- Packages where we need to build variants for internal use
  - E.g. Meta’s fish is compiled with additional logging that can’t be upstreamed

CentOS Hyperscale

- Packages released for CentOS Stream
  - Main: 8, 9
  - Experimental: 8, 9
  - Facebook: 8, 9
  - Intel: 8, 9
- Tooling repos: pagure.io/projects/centos-sig-hyperscale/*

Debian

Package maintenance

e.g.

- drgn (a programmable debugger in Python)
    - Debian tracker
- archlinux-keyring (for testing systemd-mkosi’s ability to generate Arch images)
    - Debian tracker

Ubuntu

- PPA!
  - ppa:michel-slm/kernel-utils

Lessons learned

Share the burden

- Have several active maintainers
  - Fedora
    - 2 packager sponsors
    - 1 proven packager
  - Debian
    - 2 Debian Developers (on affiliated teams)
    - 1 Debian Maintainer (me)
  - CentOS Hyperscale SIG
    - ~ a dozen Meta contributors

Share the burden

- These are all community projects (to a greater or lesser extent)
- Having coworkers review changes speed up the process
  - Caution: NOT an invitation to lower the quality bar!
  - Also review others' work to unblock

Where Fedora > Debian

- Less friction for building for supported releases
  - In Debian, new binary packages for each repos (unstable, backports, proposed-updates) need to go through the DD binary upload + ftp-master route
- Wider access to the official build system
  - Any packager can do a Koji scratch build
  - In Debian, porter boxes are accessible to DDs only by default

Where Debian > Fedora

- dh magic: default debian/rules works out of the box for many projects, much less customization needed
- e.g. for drgn:
  - debian/rules
  - Relevant parts of python-drgn.spec
- Discovered endianness issues in drgn’s libkdumpfile dependency when packaging in Debian
  - Full story

Where Debian > Fedora

- Parallel installability of shared components
  - This means there's extra friction if SONAME changes often
  - Arguably a good gating mechanism
  - See e.g. the_Foundation#13

Patience

Patience

- It takes at least months to get Debian Maintainer status
  - Still not enough to operate fully independently
- It takes several years to get Debian Developer status
- Even in Fedora, more radical changes require more consensus building
  - Fedora 33 - Btrfs by Default
  - Fedora 38 - -fno-omit-frame-pointer

Contribute (for fun and profit)

Conclusion

- To achieve the full potential of using Linux, you should contribute
- At least report bugs
  - Without a support contract, YMMV
- Work in the open as much as possible
  - Avoid accruing internal tech debt
  - Help each other instead of reinventing the wheel
- Go with the flow and build relationships
- Changes can happen surprisingly fast once you have momentum

Meta