ARE DISTRIBUTIONS STILL RELEVANT?

VINCENT BERNAT — EXOSCALE

VIENNA LINUX MEETUP — 2017-11-07

ABOUT ME

- Vincent Bernat
- Working at **Exoscale**
- Linux user since 1999
- Debian user since 2002
- Debian Developer since 2008

WHAT IS A DISTRIBUTION?

- A software collection (including a kernel)
- A package manager
- A release cycle
- A philosophy
- People

EXAMPLE OF DISTRIBUTIONS

- **Debian** (*The universal operating system*)
  - 50k+ packages, Linux kernel
  - `dpkg` and `apt`
  - a release every 2 years (when it's ready), supported 5 years
  - a **Constitution**, a **Social Contract**, **Free Software Guidelines**
  - about 2500 contributors, including 1000 **Debian Developers**
  - community-based, **meritocracy**, elected leader

EXAMPLE OF DISTRIBUTIONS

- Ubuntu, a Debian-derivative
  - 50k+ packages, Linux kernel
  - `dpkg`, `apt` and now **Snappy**
  - a release every 6 months and a longer-supported version (5 years) every 2 years (fixed schedule)
  - commercial distribution, backed by Canonical with participation from community
  - 500+ employees

EXAMPLE OF DISTRIBUTIONS

- Red Hat Enterprise Linux
  - 10k packages, Linux kernel
  - `rpm`, `yum`
  - release about every 3 years, support for 10 years
  - **commercial distribution** with mandatory contract support
  - no community, only Red Hat employees
- Fedora, base for RHEL
  - 20k packages, Linux kernel
  - `rpm`, `dnf`
  - release every 6 months, 1 year support
  - **community-based**, sponsored by Red Hat

EXAMPLE OF DISTRIBUTIONS

- CentOS, free version of RHEL
  - same as RHEL without Red Hat trademark
  - community-based

EXAMPLE OF DISTRIBUTIONS

- Gentoo (community-based, source-based)
- ArchLinux (community-based, rolling release)
- OpenBSD (no Linux kernel, release every 6 months, 1 year support)
- Container Linux (Gentoo-based, minimal, run containers)
- Homebrew (no kernel, for macOS)
- Alpine Linux (musl and busybox, small footprint)
- NixOS (source-based, reproducibility, atomic upgrades)
- …

STABILITY

- For distributions, **stability** means **nothing changes** during the lifetime of the distribution
- No new functionalities, no new bugs, no configuration changes
- Security and critical bugs are usually fixed (by backporting them to minimize changes)

USERS WANT STABILITY

- Most distributions provide stable releases for a given timeframe:
  - 1 year for OpenBSD
  - 5 years for Debian and Ubuntu
  - 10 years for RHEL
- Updating software everyday is OK for one personal computer (or a phone)
- Imagine managing thousand of servers where, every day, one piece of software:
  - changes its configuration format
  - gets a new bug
  - removes a deprecated functionality

USERS WANT UP-TO-DATE SOFTWARE

- When in the middle of a release cycle, all software are **old**
- Usually, users care only for a selection of packages to be up-to-date
- Several solutions available:
  - Debian provides **backports** for some software, no stability
  - Ubuntu has PPA for users to get software from **third-parties**, no guarantee
  - Do your own packages (do-able when not many)
  - As a trade-off, use the older version (it worked fine 3 years ago)

DISTRIBUTIONS CANNOT SHIP TOO RECENT SOFTWARE

- Frequently, people complain Debian was just released with an out-of-date software
- We cannot ship the latest version of every software because everything needs to work together
- We cannot fix bugs once released (except very critical ones)
- We ship buggy software but with **bugs we know** and document

EXAMPLE: OPENSSL

- In Debian Stretch (the latest one), we ship OpenSSL 1.1.0
- OpenSSL 1.1.0 was released in August 2016
- In November 2016, Debian scheduled to use it for its next release
- Most software were unable to compile against this version
- Many complaints
- Basically, the same person can tell us:
  - "*We want you to ship the latest version of my software*"
  - "*We want you to ship a stable version of others' software*"

FREEDOM

- For most distributions, freedom is important:
  - Debian has the **DFSG**
  - Fedora/Red Hat are even stricter in what they accept (no patents)
- Freedom is usually the ability to **distribute**, to **modify** and to distribute modified versions

TODAY

So, today, are classic distributions (like Debian) still relevant?

PACKAGING

- Packaging can be replaced by a `Dockerfile` (just a few shell commands) or another similar mechanism
- Packaging can be done by upstream and be cross-distribution
- Packaging can be done with less restrictions
- Too many software for distributions to package them all
- However:
  - Every Docker image is different (no uniformity)
  - Upstream may only maintain the latest version (no stability)
  - Most Docker images are still using a distribution
  - Security status of various components are usually unknown

FREEDOM

- Many opensource software published today
- Many people using opensource software
- However:
  - People care less about freedom
  - Impossibility to get an opensource phone
  - Closed firmware running on your CPU like Intel AMT
  - Software running in the cloud are usually closed

CONCLUSION?

- Stability is still high in most users need (more important than having the latest version): distributions still have a public
- Many continue to use a classic distribution as a base image
- However:
  - Classic distributions are **challenged by newcomers** (Container Linux for running containers, Alpine for small footprints, NixOS for managing configuration and software together)
  - Classic distributions are **unable to attract more developers** (packaging appears outdated and boring, even if it's easier now than 10 years ago)
  - Classic distributions can **put roadblocks** to new technology (systemd adoption, new ecosystems)

QUESTIONS?