



Integrating new major components on fast and slow moving distributions

How latest GNOME desktop was integrated into latest SUSE / openSUSE releases

Frédéric Crozat <fcrozat@suse.com>
SUSE Linux Enterprise Release Manager

What we don't do



What we do



Distribution delivery styles

Three distributions styles

- Rolling:
 - Bleeding edge
 - Release as soon as possible
 - Example: openSUSE Tumbleweed, ArchLinux, Gentoo
- Regular:
 - Release one to twice a year
 - Update their entire stack for each release
 - Example: Ubuntu, Fedora, Debian
- LTS / Enterprise:
 - Slow cadence (yearly or even less than that)
 - Very few things move between sub-releases
 - Example: openSUSE Leap, Ubuntu LTS, SLES/SLED, RHEL

openSUSE/SUSE terminology

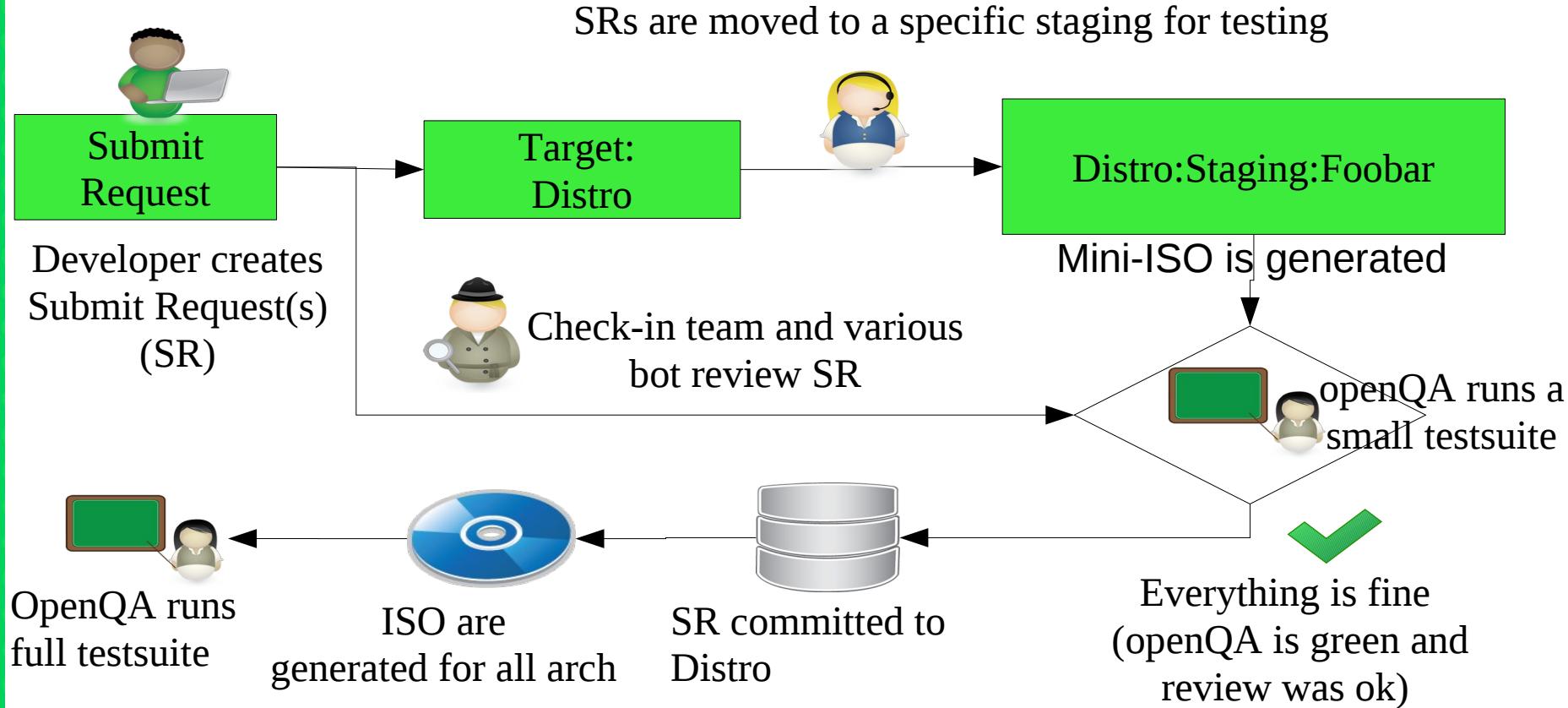
- OBS = OpenBuildService
- SLE = SUSE Linux Enterprise (Server / Desktop)
 - Enterprise distribution, developed by SUSE
- openSUSE Tumbleweed:
 - openSUSE Rolling release, by openSUSE, using only Factory packages, tested by openQA
- openSUSE Factory:
 - Development repository for Tumbleweed
- openSUSE Leap:
 - openSUSE Stable release, based on SLE common code + Packages from Factory (or specific repository)

Integration process

OBS and Devel project

- On OBS, every source package is handled in a project which can build several packages together
- openSUSE Tumbleweed uses devel project per “topic” (KDE, GNOME, X11, ...)
- Changes (patch, version update) are done in Devel projects and then, pushed to “main” distribution for integration, either by human or a bot.

Submitting to a distribution



Factory first policy (for SLE)

- New guidelines in effect for development since 2017
- Whenever possible, development should be done on OBS (openSUSE:Factory) and pushed back to SLE15-SP2
- When submission is reviewed and accepted on Factory, it is automatically submitted to SLE15 (unless packages was branched in SLE15). This was in place for most of the development period.
- When a submission is sent to SLE15, a automated check will ensure similar submission was done to openSUSE:Factory
- Based on this knowledge, review team decides what to do with those submit requests
- You can see SLE15(SP2) development “live” since we are in Beta phase

Updating to GNOME 3.34

Devel projects

- Two layered projects:
 - **GNOME:Factory**: contains the latest stable version of packages
 - **GNOME:Next** : packages for the upcoming release (usually packages are updated with version .90 aka upstream RC)
- Allow to get bugfixes into Tumbleweed while preparing next major version
- **GNOME:Next** is layered on top of **GNOME:Factory**
 - Ensure changes in stable branch aren't lost

Update to GNOME 3.34 in Tumbleweed 1/3

- All relevant packages were updated in **GNOME:Next** during GNOME 3.33.9x releases:
 - An live image was automatically [built](#)
 - It was then tested by [openQA](#): system boot, login and GNOME desktop and applications ability start properly
- Once GNOME 3.34.0, work started to merge **GNOME:Next** into **GNOME:Factory**
 - This requires 4 eyes review by openSUSE GNOME team

Update to GNOME 3.34 in Tumbleweed 2/3

- Once GNOME 3.34.0 had landed into **GNOME:Factory**, it was then pushed to Tumbleweed:
 - Another 4 eyes review (by openSUSE check-in team)
 - Legal review
 - Staging
 - Ensure everything in Tumbleweed would still build properly
 - Ensure openQA staging tests still pass (often requires tests adaptation or needles updates)

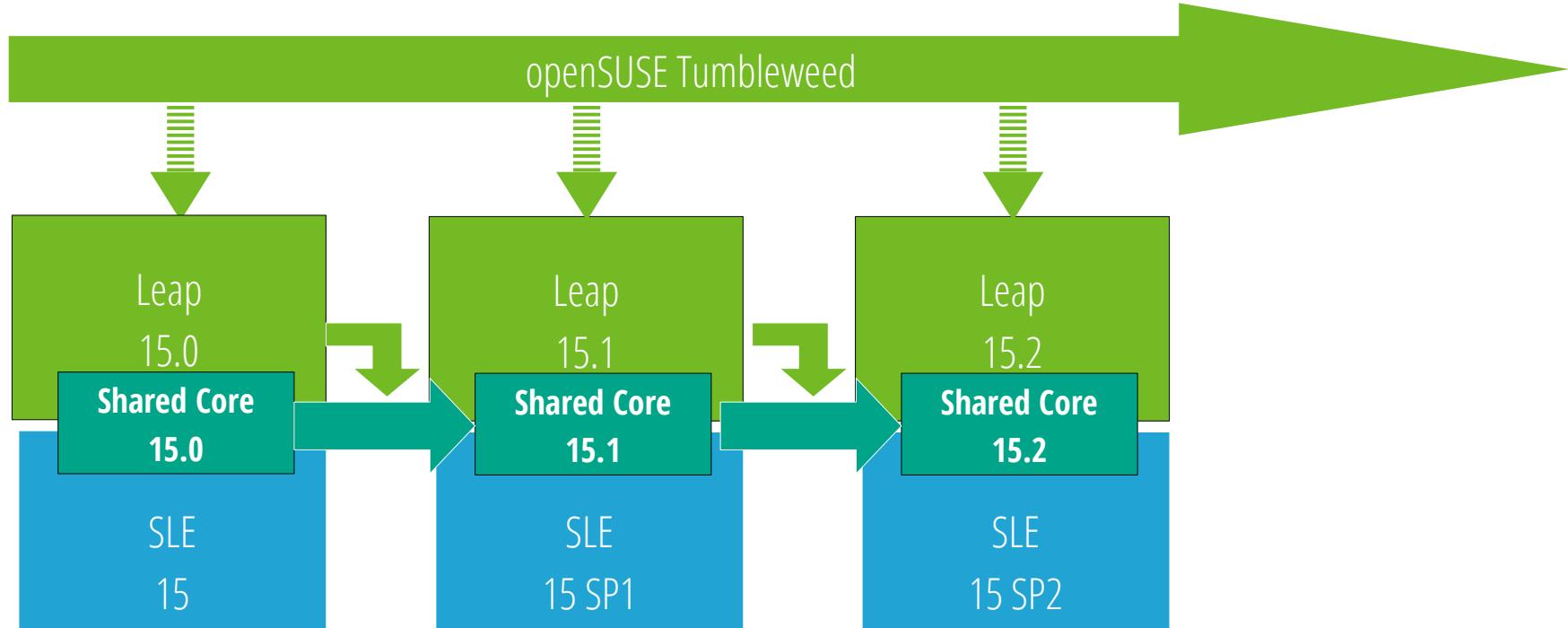
Update to GNOME 3.34 in Tumbleweed 3/3

- Once Staging was accepted, a bigger testsuite is run on the result
- If success, a new Tumbleweed snapshot is released (happen several times per week)
- GNOME 3.34.0 was available in Tumbleweed with snapshot 20191018, less than a month after GNOME 3.34.0 was release upstream (September 14, 2019). Join openSUSE GNOME team to make it even faster.

What about SLE and openSUSE Leap

openSUSE & SUSE Linux Enterprise

Developed together - Reminder



Getting GNOME in SLE 15 SP2 1/2

- Building GNOME:Next with SLE-15-SP2 repository as base
 - Discover all SLE patches which were no longer applicable (were already part of TW sources)
 - Discover all non-GNOME packages which are new to SLE or which requires updates in SLE (compared to TW)
 - Find missing or obsoletes build requirements
- Create a Staging project, with core GNOME components and add what is missing to get it building and passing openQA tests (not all packages from GNOME:Next should end-up on SLE product, the rest is available through PackageHub)

Getting GNOME in SLE 15 SP2 2/2

- Took one month to get things into acceptable state
- Once staging was accepted, we discovered other failures on aarch64 / ppc64le and s390x (some in non-GNOME dependencies such as mozjs60), staging being done on x86_64 only
- Update the rest of the GNOME applications to 3.34
- Fun fact: we forgot for a while to update some core GNOME component (dconf) but everything was still working fine
- ATM, running 3.34.2, will update soon to 3.34.3/4

Getting GNOME in openSUSE Leap 15.2 (1/2)

- Leap is based on SLE codebase, GNOME would then be inherited in Leap once accepted in SLE
- Easy ? Not really
- SLE Service Packs are layered on top of each other: unmodified packages are inherited at binary level (not rebuild)
 - Only packages modified in SLE15 SP2 are built
- openSUSE Leap is a standalone distribution
 - All packages are built (including bootstrap) for each release
 - Leap contains more packages than SLE
 - Some packages were not building anymore with GNOME 3.34 update, but this wasn't visible in SLE (binary import or not present in SLE)

Getting GNOME in openSUSE Leap 15.2 (2/2)

- Took 45 days (Christmas break included ;)
- GNOME 3.34 staging was accepted in openSUSE Leap 15.2 this week, just in time for FOSDEM !
- It will become available with next openSUSE Leap 15.2 milestone release

Summary

Thanks to OpenBuildService, openQA and our processes (Factory first policy), we were able to update a major component in our 3 distributions (including two LTS distributions) in 4 months, while maintaining quality. We could be even faster next time !

Questions ?



We adapt. You succeed.