

KDE and GNOME: KUnity, Presentations, GitLab and Flatpaks

By *Roy Schestowitz*

Created *12/02/2020 - 5:17am*

Submitted by Roy Schestowitz on Wednesday 12th of February 2020 05:17:54 AM Filed under [KDE](#) [1] [GNOME](#) [2]

- [A year as a KDE developer | The KUnity Setup](#) [3]

It has been more than a year that I had push rights for all the KDE repositories. So this is an obligatory anniversary post.

I got introduced to Linux while searching for development environments that came with all sorts of compilers & interpreters by default and I don't have to manually install those stuff. It was 2012 as far as I remember, Ubuntu 12.04 just came out and it was the first solution suggested by the search engines. Though the unavailability of a proper internet connection meant, that I had to wait a couple more years when one of my friends downloaded a copy of Ubuntu 14.04 for me.

- [FOSDEM and Plasma Mobile Sprint 2020](#) [4]

This was the 20th anniversary for FOSDEM, I first attended 15 years ago, but this year was the first time I actually managed to present a talk there. The subject was, unsurprisingly, KDE Itinerary. You can find the slides and the video recording on the corresponding FOSDEM talk page.

KDE had a very busy presence at FOSDEM, Plasma Mobile draw a lot of attention, as did efforts for truly free and user-controlled mobile platforms in general. I'm particularly happy

seeing the cross-community collaborations going on in that space.

FOSDEM is a great place to connect and coordinate with other communities, and by now that's probably one of the main reasons for me to attend. The collaboration with Nextcloud on integrating itinerary extraction into Nextcloud Hub started there last year for example.

FOSDEM two years ago saw the first successful flight using a KDE Itinerary rendered boarding pass, this year we had the first ever Thalys ride with a ticket presented in KDE Itinerary. We also made a bit of progress with decoding Thalys binary barcodes, more samples would help a lot here though.

•

[Presentations Archive](#) [5]

Some time ago I ran across remark-cmake, a CMake framework for building remark.js-based presentations. Since I'm a sucker for CMake I started using it, even if my presentations are rarely big-and-complicated enough to warrant a build-system.

Since then I've submitted a few pull-requests to remark-cmake, but also given eight (8) presentations using that framework at four (4) different conferences in four countries on two continents. Current scheduling suggests that one more continent and at least four more talks will be added before summer.

•

[Clean and linear history with GitLab](#) [6]

Many GNOME projects still use clean and linear commit history without merge commits even after porting to GitLab. That means that each commit represents one comprehensive feature or bug fix and there are not any side branches. I am not about to discuss the pros and cons of this approach here, you can find many and many posts on this topic on the internet. I would like to explain some common issues for newcomers when using GitLab forks.

To make some contribution, one has to create a fork of some repository, push desired changes in a new branch and create a merge request to the original project. Please be sure that the "Allows commits from members who can merge to the target branch" checkbox is checked when creating the merge request (or later using the "Edit" button on the top of the page). This is needed to simplify the consequent workflow for the contributor (and maintainers as well). One of the reasons, why this is needed is the fact, that the changes need to be often rebased before they can be merged (to ensure the linear history). Maintainers can't do this when this feature is not enabled and have to ask contributors to do so. Another reason is that the maintainers can do some changes when the contributor needs help or doesn't have time to do the changes itself.

•

[Robocode and others](#) [7]

As expressed in a previous post, I prefer to spend my free time with my kids than with technology (for technology I already have my job). However, when there is an exception to that, I do like to do some sort of smaller projects, like `porting` stuff to Flatpak.

I did my share of Debian and RPM packaging in the past, and honestly I have never enjoyed it (for a number of reasons not really interesting for this post). But `flatpaking` stuff is completely different to me. Maybe it's my early involvement with it, or maybe it's my admiration for how it's designed, but the feeling when making a Flatpak is of reward, rather than a chore.

[KDE GNOME](#)

Source URL: <http://www.tuxmachines.org/node/133979>

Links:

- [1] <http://www.tuxmachines.org/taxonomy/term/108>
- [2] <http://www.tuxmachines.org/taxonomy/term/146>
- [3] https://www.hellozee.dev/a_year_as_a_kde_developer/
- [4] <https://www.volkerkrause.eu/2020/02/11/fosdem-2020-plasma-mobile-sprint.html>
- [5] <https://euroquis.nl/bla%20bla/2020/02/10/presentations.html>
- [6] <http://ondrej.holych.net/clean-and-linear-history-with-gitlab/>
- [7] <https://www.joaquimrocha.com/2020/02/12/robocode-and-others/>