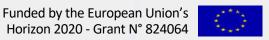






- Current onboarding process
 - How to improve it
 - Records adequacy to policy changes
- Policy
 - Computing environment
 - FAIR principles and requirements enforcement
- The future of OSSR
 - Onboarding and curation process
 - **Evolution of the OSSR**





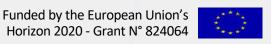


Current process

- See
 - ossr-pages: https://escape2020.pages.in2p3.fr/wp3/ossr-pages/
 - Jutta's summary and minutes: https://indico.in2p3.fr/event/25282/

Two open requests at the moment:

- https://zenodo.org/record/5592836#.YaDCVvHML0o
- https://zenodo.org/record/5549470#.YaDCWPHML0o







Improve that process without policy changes

- Final acceptation by a single curator in Zenodo (technical limitation)
- Still, curation could (should) be a common effort
 - 1. Set-up a group of WP3 curators
 - 2. Curation process, two ways:
 - They all have a look to the code and checklist (during and after onboarding process)
 - · give feedback by email to me who contact back the authors if needed
 - once green light --> I accept the software in the escape2020 community
 - Automatised way through open reviews
 - Use Zenodo and GitLab API
 - GitLab CI checks for new addition requests to the OSSR
 - If any, open a merge request for each new record, with all information to that record
 - Review happens openly in that merge request (benefiting from all the nice process provided by gitlab = discuss, assignements, minimum numbers of approval...)
 - Once accepted, merge request is closed CI automatically accepts the record in the OSSR
 - · Working POC under dev

My proposal: Do 1. right now for the open requests







Policy changes

•What to do with software in the OSSR that were added before we file our current policy and don't comply with it?

What happens when our policy evolves?

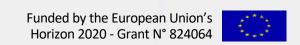






Policy changes: Computing environment

- Providing an environment
 - Not in our official requirements! (point raised during ESAP call)
 - Note: I think for most recent records, it was more or less checked (?)
- It should be in the requirements in order to provide FAIR software
- problem: what is a valid requirement file?
 - There are many ways, which are language dependent
 - If a env file is provided, shall we check for its validity?
 - To what extend should it be accurate? (fixing all dependencies exact versions?)
- >> The responsability of providing a working environment should fall back to the software provider (can/should we help?)
- >> **WE** should provide guidelines of what is a good environment in general (or limit ourselves to docker and provide docker recipe guidelines)







Policy changes: enforce requirements?

- Should all records (software and datasets) in the OSSR meet the FAIR principles?
 - Findable
 - Accessible ? yes if we strictly refuse embargoed or closed records
 - Interoperable
 - Reusable



