



Open Software in astro and particle physics - the *Open-source Software and Service Repository*

Journées Données Ouvertes IN2P3

December 2024, Thomas Vuillaume



Who am I (to talk about software quality) ?

Astrophysics background

Turned data scientist

Research Engineer at LAPP since 2021

Interests: data analysis, machine learning & software development

Who am I (to talk about software quality) ?

Astrophysics background

Turned data scientist

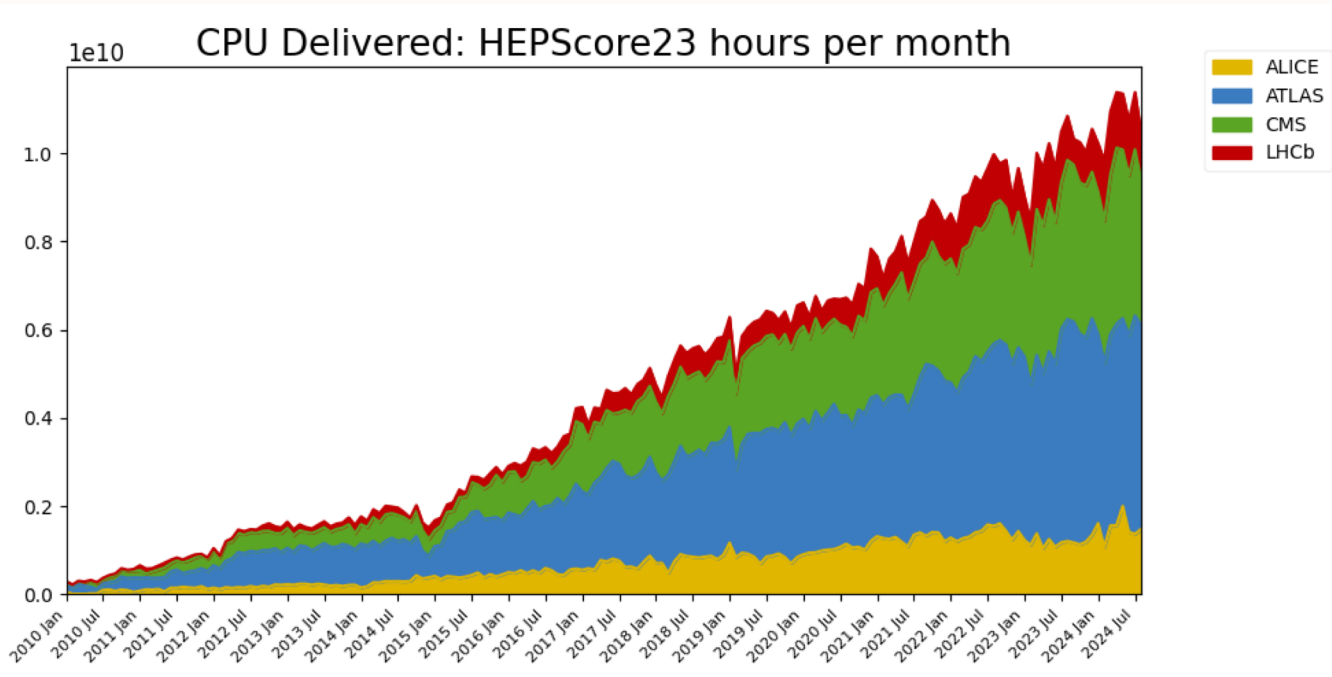
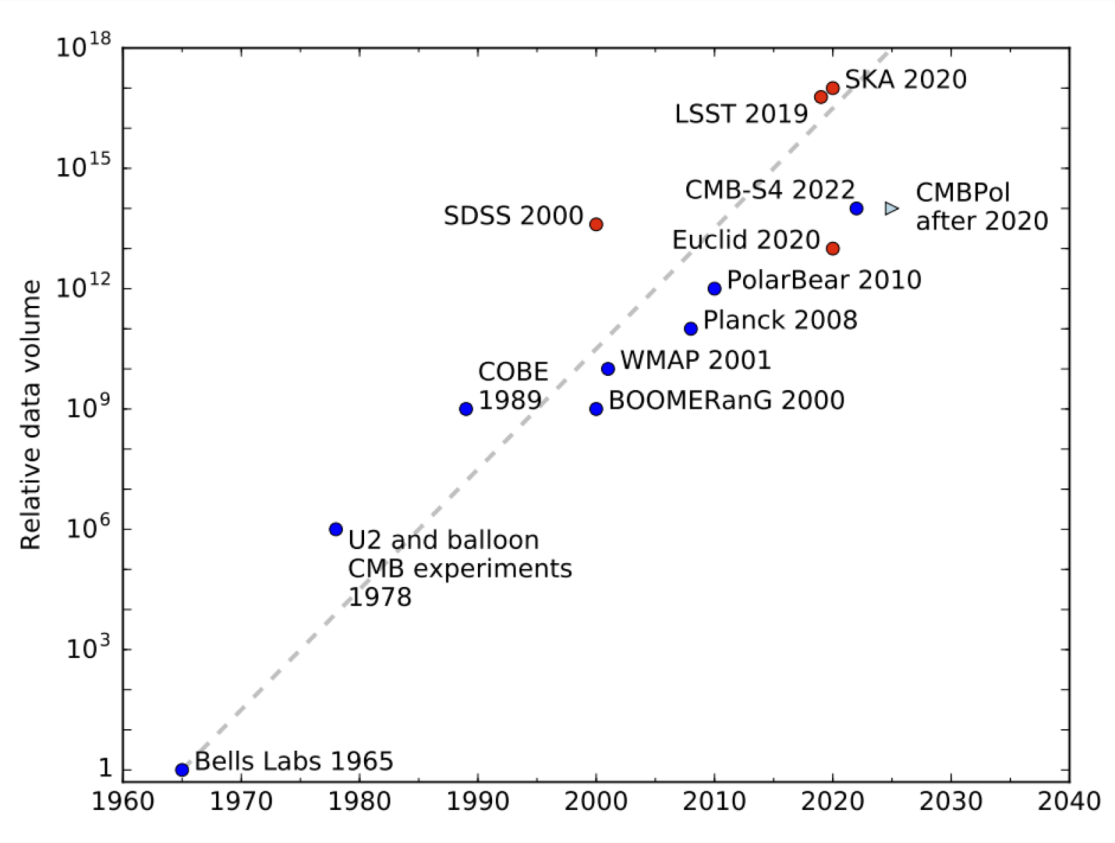
Research Engineer at LAPP since 2021



I know what
bad software is;
I write it

Interests: data analysis, machine learning & software development

Context: A new paradigm - data-driven science

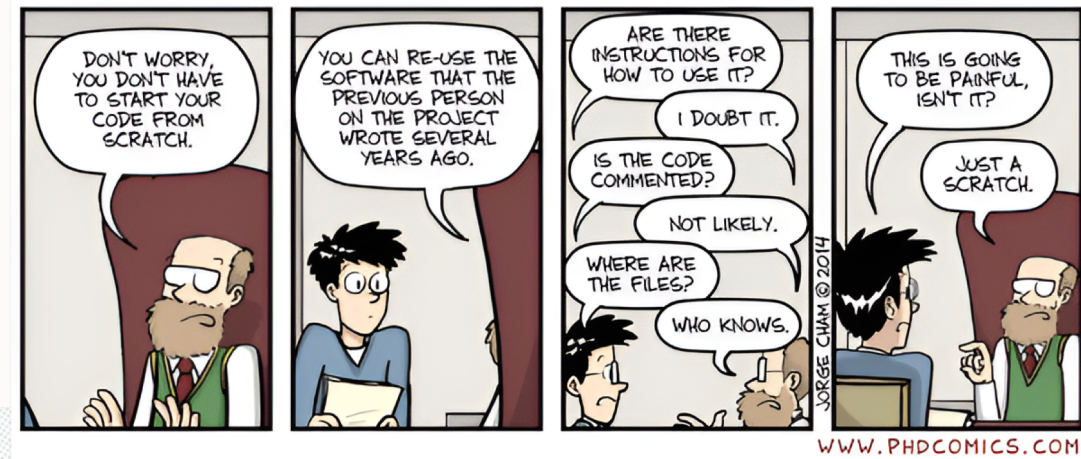


<https://wlcg.web.cern.ch/using-wlcg/monitoring-visualisation/monthly-stats>

DOI:10.3390/universe2040023

Software is not recognized as first-class output

1. Software is not shared and reused
 - waste of time, energy and resources
 - **reproducibility crisis**

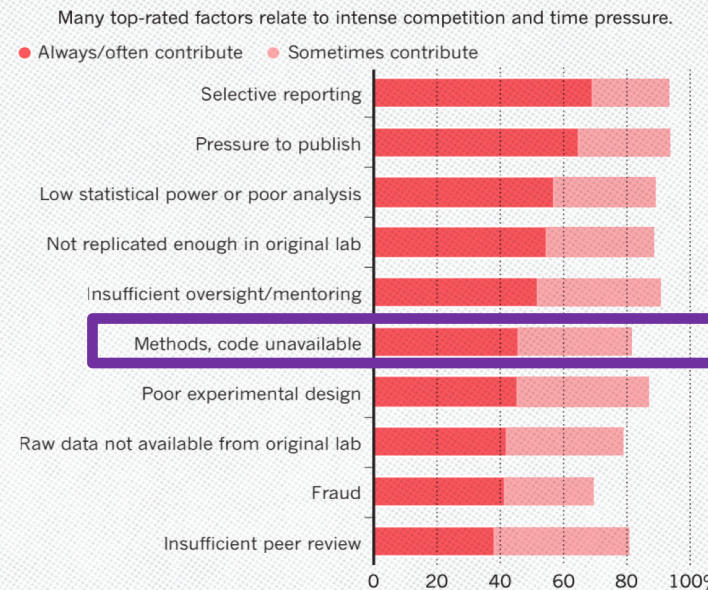


<https://phdcomics.com/comics/archive.php?comicid=1689>

IS THERE A REPRODUCIBILITY CRISIS?



WHAT FACTORS CONTRIBUTE TO IRREPRODUCIBLE RESEARCH?



1,500 scientists lift the lid on reproducibility, <https://doi.org/10.1038/533452a>

DOP21 IN2P3, 16/12/2024

Software is not recognized as first-class output

2. Research Software Engineers are not recognized
 - short-term contracts often related to specific projects
 - metrics mismatch (they don't publish scientific papers)
 - talents loss to industry

<https://www.fz-juelich.de/en/rse/about/what-is-a-research-software-engineer>

<https://invenia.github.io/blog/2020/07/07/software-engineering/>

<https://researchcomputing.princeton.edu/news/2021/building-career-path-research-software-engineers>

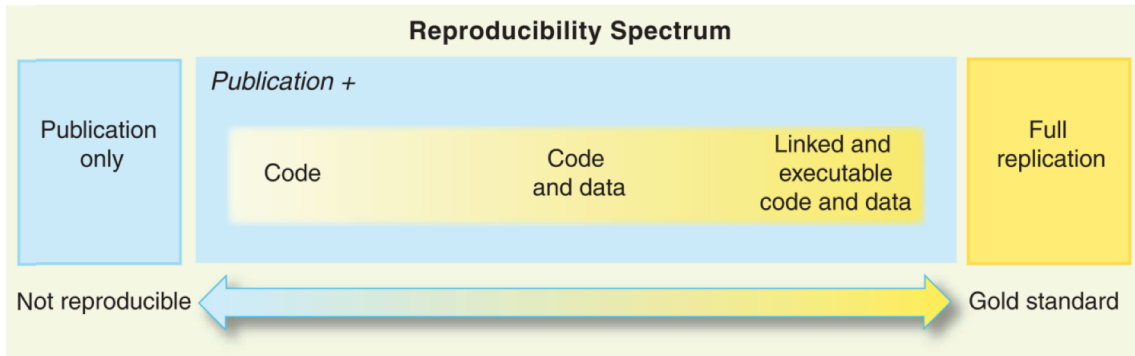
<https://www.software.ac.uk/blog/why-research-software-engineers-should-have-permanent-contracts>

So, we have two opposing considerations:

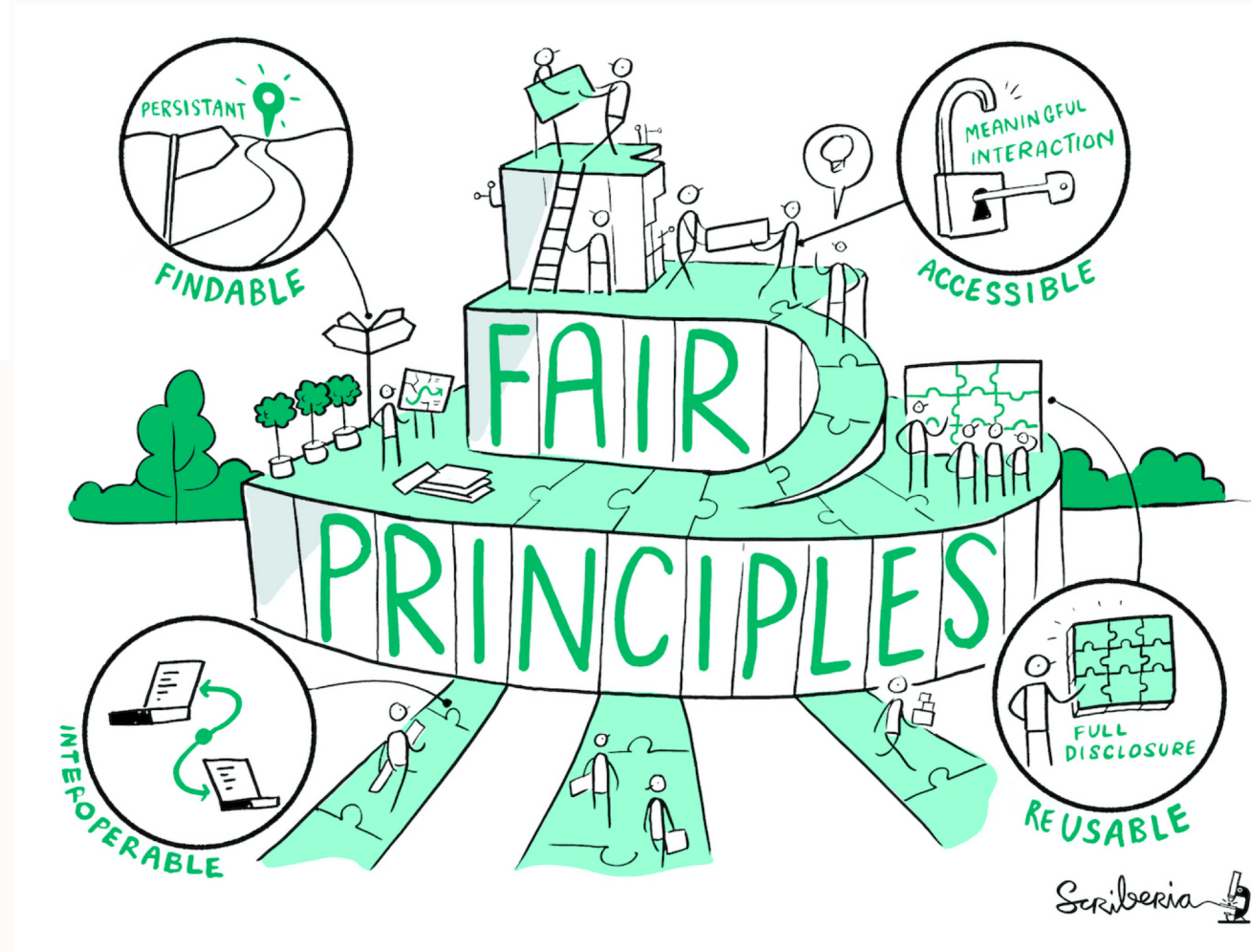
1. Software importance is **increasing**
2. Software importance is **not recognized** (enough)

What can we do about it ?

1. Publish Open & FAIR software



- Findable
- Accessible
- Interoperable
- Reusable



Barker, M., Chue Hong, N.P., Katz, D.S. *et al.* Introducing the FAIR Principles for research software. *Sci Data* 9, 622 (2022). <https://doi.org/10.1038/s41597-022-01710-x>

The Turing Way Community, & Scriberia. (2023). Illustrations from The Turing Way: Shared under CC-BY 4.0 for reuse. Zenodo. <https://doi.org/10.5281/zenodo.8169292>

Open-source Software and Service Repository

- A trusted software repository
- Community Centered
ESCAPE = Particle Physics and Astronomy Cluster in EOSC
- FAIR
- Long-term
- Curated



 New upload

[Records](#) [Requests](#) [Members](#) [Settings](#) [Curation policy](#) [About](#)


48 results found

Sort by
December 11, 2023 (0.11.3) Software Open

MOC Lib Rust, MOCCLi, MOCWasm and MOCSet

Pineau, Francois-Xavier  Baumann, Matthieu

Rust implementation of the IVOA MOC standard (MOC Lib Rust); associated command line tool (MOCCLi) and Javascript/WebAssembly wrapper to manipulate MOCs in Web Browsers (MOCWasm).

Uploaded on December 20, 2023

6 more versions exist for this record

 184  29

December 4, 2023 (v0.13.1) Software Open

cds-astro/mocpy: Release v0.13.1

Matthieu Baumann; Manon Marchand; Francois-Xavier Pineau; and 6 others

What's Changed Mostly maintenance to support astropy 6.0 and python 3.12 while maintaining support for python 3.8 These points have changed internal behaviour, or documentation: Add missing return statement in private abstract class AbstractMOC in <https://github.com/cds-astro/mocpy/pull/112> The deprecated method write now calls save intern...

Uploaded on December 4, 2023

5 more versions exist for this record

 158  17


December 4, 2023 (v2.0.0) Software Open


eOSSR

Enrique Garcia; Thomas Vuillaume

The ESCAPE OSSR library The eOSSR is the Python library to programmatically manage the ESCAPE OSSR. In particular, it includes: an API to access the Zenodo and the OSSR, retrieve records and publish content functions to

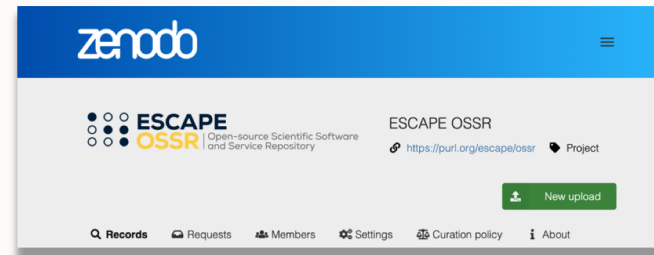
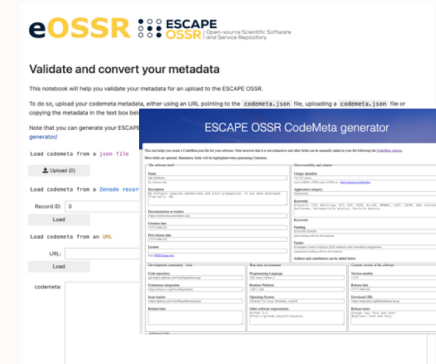
Zenodo as

- FAIR centered
- long-term archive
- software citability (DOI)
- widely accepted and used
- don't reinvent the 
- integrates with other services
- community management

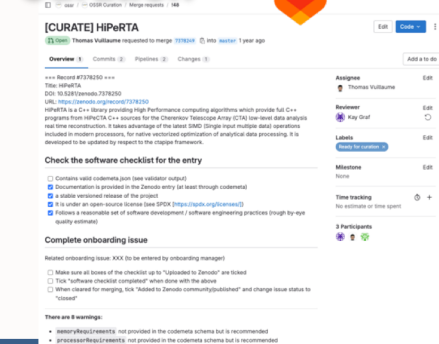
 *escape2020* community



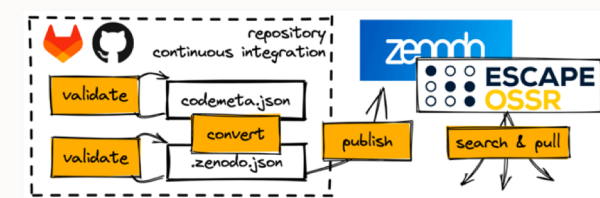
The OSSR galaxy



Curation



eOSSR

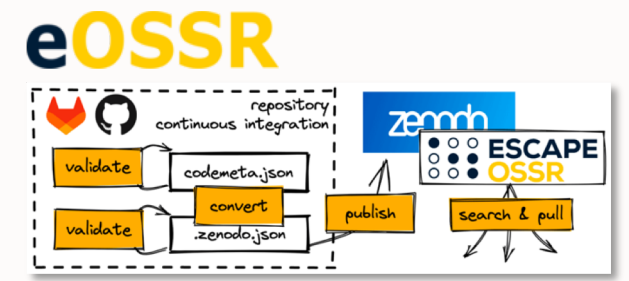
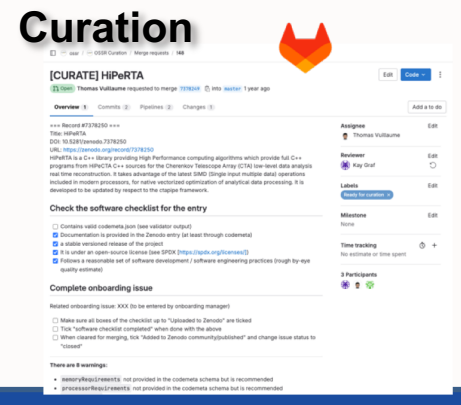
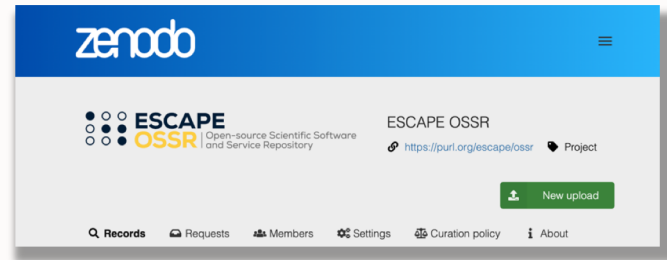
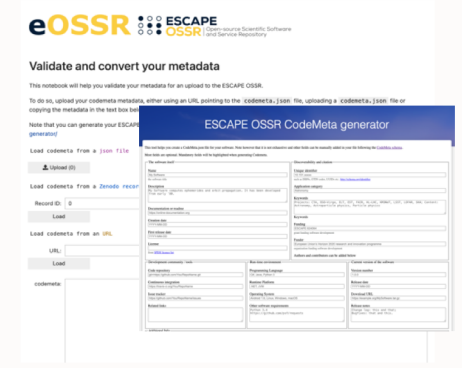


The OSSR galaxy - the software provider path

OSSR website

- Information
- Onboarding process

<http://purl.org/escape/ossr>

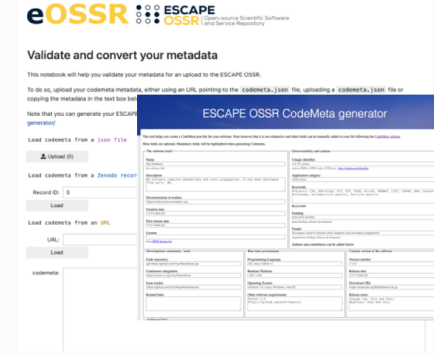
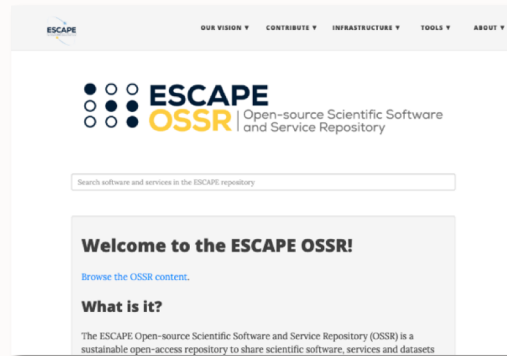




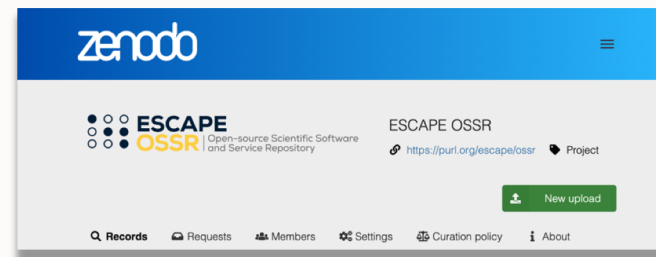
The OSSR galaxy

OSSR website

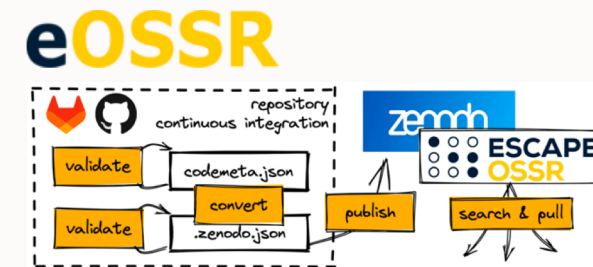
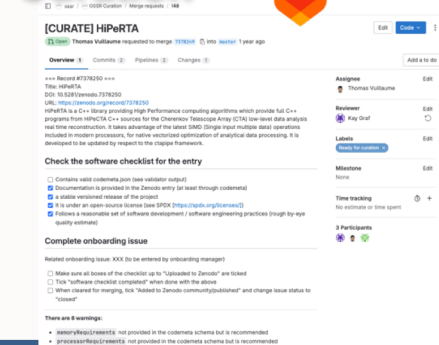
- Information
 - Onboarding process
- <http://purl.org/escape/ossr>



Tools to help RSEs generate the right metadata for their software



Curation

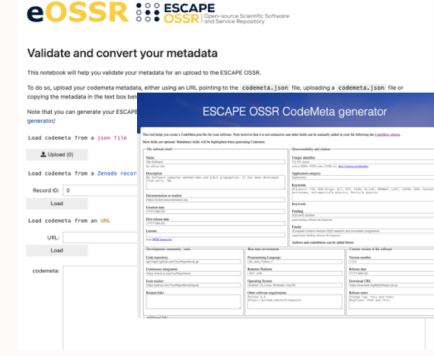




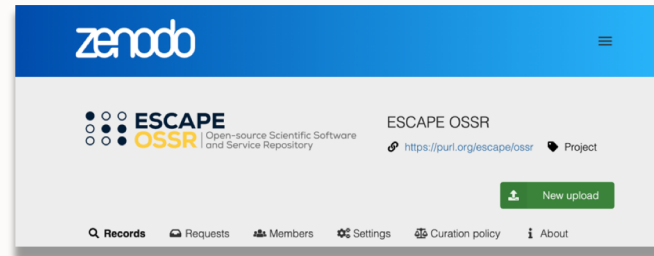
The OSSR galaxy

OSSR website

- Information
 - Onboarding process
- <http://purl.org/escape/ossr>



Tools to help RSEs generate the right metadata for their software



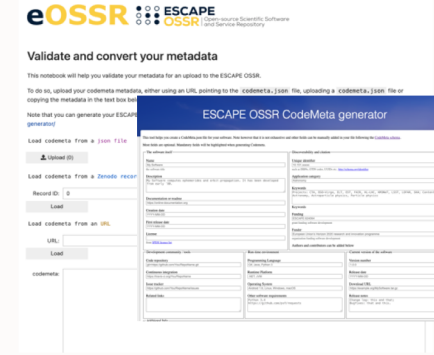
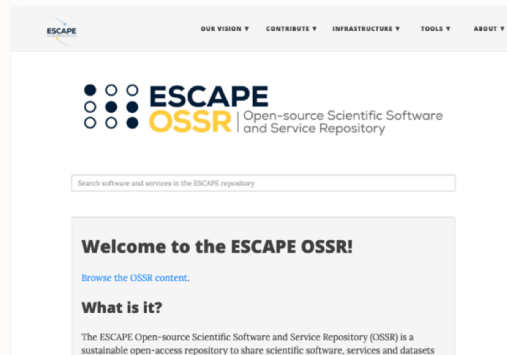
A python library to communicate with the OSSR



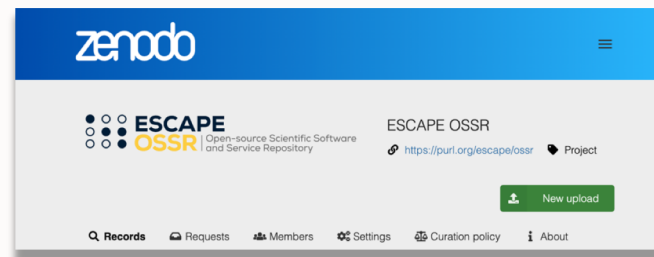
The OSSR galaxy

OSSR website

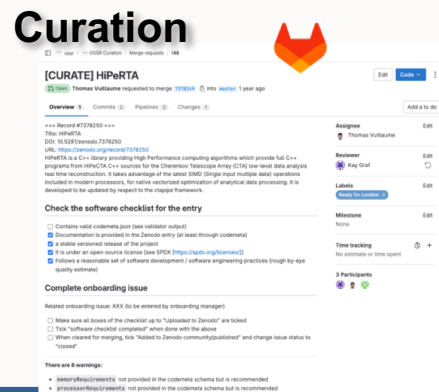
- Information
 - Onboarding process
- <http://purl.org/escape/ossr>



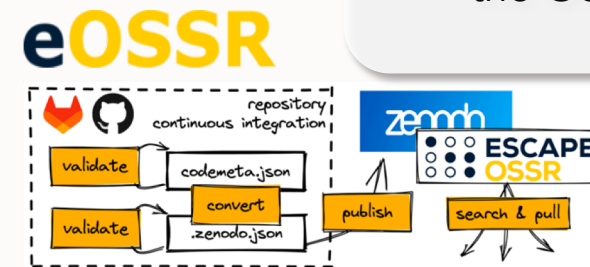
Tools to help RSEs generate the right metadata for their software



A python library to communicate with the OSSR



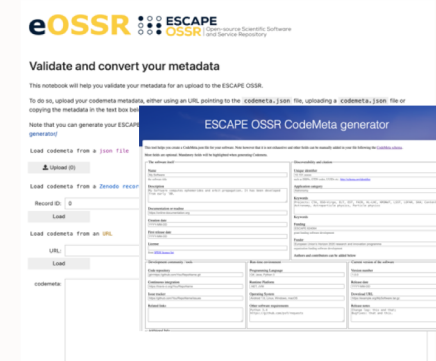
Tools to automatically publish software from GitHub or GitLab



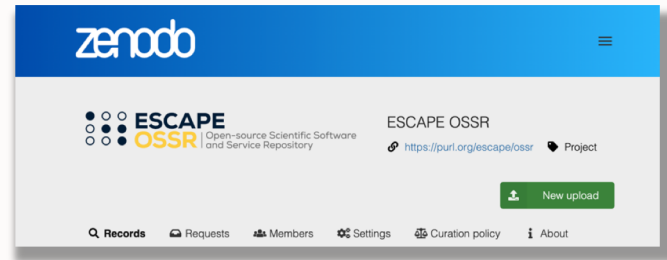
The OSSR galaxy

OSSR website

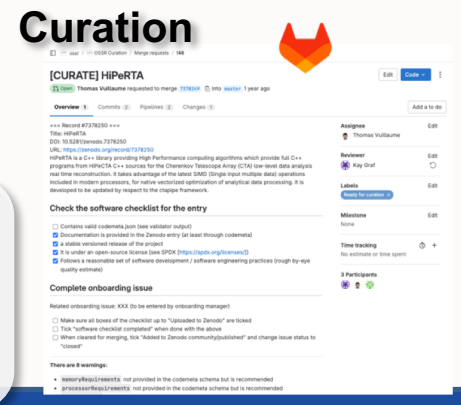
- Information
 - Onboarding process
- <http://purl.org/escape/ossr>



Tools to help RSEs generate the right metadata for their software



A python library to communicate with the OSSR

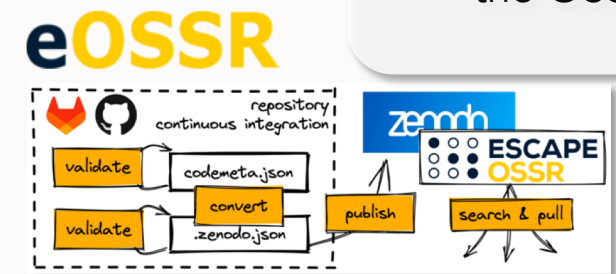


A curation platform to review the requests

OSSR accepted ✓



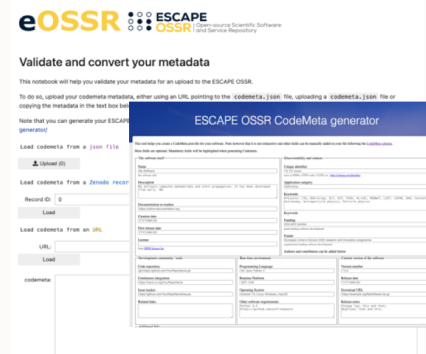
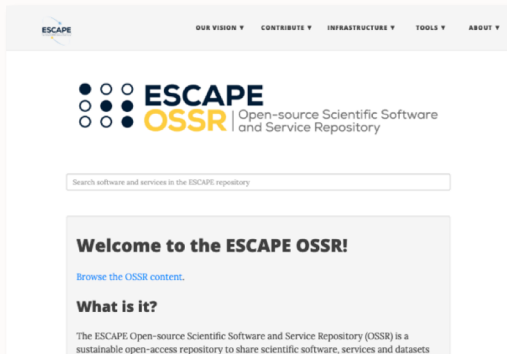
Tools to automatically publish software from GitHub or GitLab



The OSSR galaxy

OSSR website

- Information
 - Onboarding process
- <http://purl.org/escape/ossr>



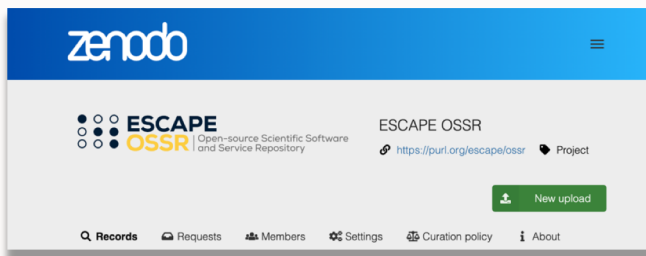
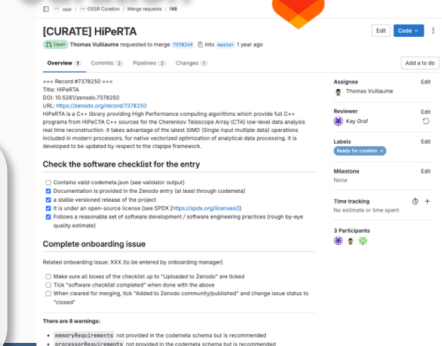
Tools to help RSEs generate the right metadata for their software



A connection to other services



Curation



A python library to communicate with the OSSR

A curation platform to review the requests



Tools to automatically publish software from GitHub or GitLab



2. Improve software quality

- Programming schools at LAPP since 2017

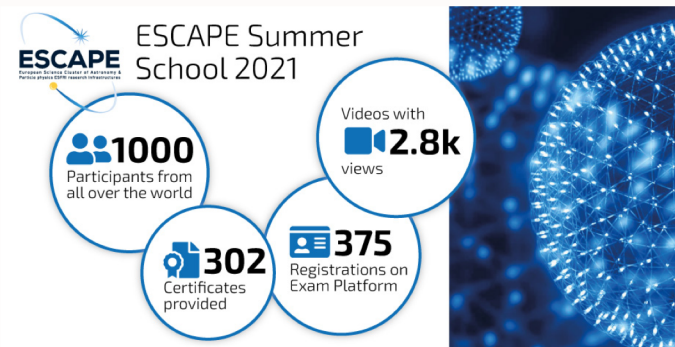


Code development for physicists:

- Coding environment and good code practices
- Version control and collaborative development
- Debugging and profiling
- Python packaging
- Scientific libraries for data science and analysis
- Machine learning

All courses open-source, recorded and available online

DOI [10.5281/zenodo.5093909](https://doi.org/10.5281/zenodo.5093909)



- HPC
- Heterogeneous architectures
- Code optimisation
- 12 satellite sites

2. Improve software quality

EVERSE  eosc | EVERSE

Paving the way towards a

European Virtual Institute for Research Software Excellence

Pilots & Drivers



Environmental Sciences: *Integration of Science Cluster ENVRI through ENVRI-HUB*

- Integrate EVERSE framework into the ENVRI-HUB Knowledge base and Virtual Research Environment
- Apply to the development of the Essential Climate Variable computing program and cloud workflows



Life Sciences: *Integration of Science Cluster EOSC-Life through ELIXIR*

- Make RO-Crate actionable by incorporating the five safes concept into WfExS for secure and federated workflow orchestration
- Use of community-led standards for materialising research software packaged using container technologies and mobilising encrypted data whenever needed



Astronomy and particle physics: *Integration of Science Cluster ESCAPE through the Dark Matter Test Science Project*

- ML for scientific data compression (standalone code, python)
- A Common Tracking Software
- Choose an ATLAS trigger algorithm as an option for the collaboration



Proton and neutron science: *Integration of Science Cluster PaNOSC through LEAPS/LENS*

Transition software to high performance computing (HPC) and heterogeneous computing architectures



Social sciences: *Integration of Science Cluster SSHOC*

Develop a multilanguage textual analysis pipeline of tools that use a combination of open source tools and own code to create an integrated SotA tool capable of deploying locally or as a service

Paving the way towards a European **Virtual Institute** for **Research Software Excellence**

EVERSE aims to create a framework for research software and code excellence, collaboratively designed and championed by the research communities, in pursuit of building a European network of Research Software Quality and setting the foundations of a future Virtual Institute for Research Software Excellence

ensure **research software curation, quality, preservation and adoption of best practices**, by the Communities, for the Communities, build on collaboration with the five EOSC Science Clusters

adopt a **three-tier model for research software**, i.e., analysis code, prototype tools and research software infrastructure, which captures the varying complexity of research software and its development, and can be used as a basis for research software excellence

credit and recognition for both developers and software are essential components of our strategy to promote sustainable software practices


Paving the way towards a European **Virtual Institute** for **Research Software Excellence**

EVERSE aims to create a framework for research software and code excellence, collaboratively designed and championed by the research communities, in pursuit of building a European network of Research Software Quality and setting the foundations of a future Virtual Institute for Research Software Excellence

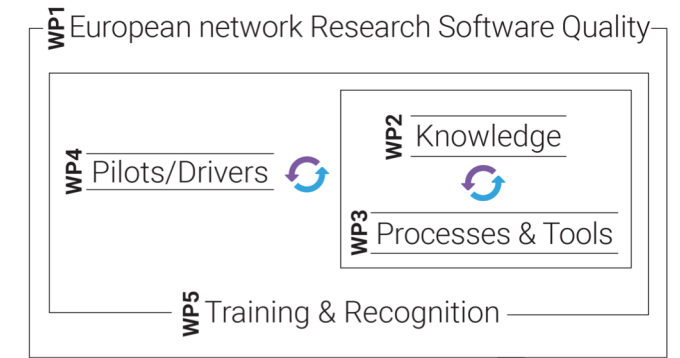
ensure **research software curation, quality, preservation and adoption of best practices**, by the Communities, for the Communities, build on collaboration with the five EOSC Science Clusters

adopt a **three-tier model for research software**, i.e., analysis code, prototype tools and research software infrastructure, which captures the varying complexity of research software and its development, and can be used as a basis for research software excellence

credit and recognition for both developers and software are essential components of our strategy to promote sustainable software practices

Mar/2024  Feb/2027 (36 months)
 15 Beneficiaries, 1 Associated partner & 2 Affiliated entities
 Coordinated by CERTH

Project objectives



Provide a **framework** that will ensure appropriate **recognition, reward, and career development** for researchers and RSEs who implement research software and code quality assurance practices and policies

Leverage existing tools and resources to support the evaluation, verification and improvement of research software and code quality, based on **existing practices and standards** across research communities represented by the five EOSC Science Clusters.

Establish a **sustainable and collaborative ecosystem of stakeholders** across the research communities associated with the five **EOSC Science Clusters** to ensure research software and code quality assurance and support the advancement of reliable and reproducible research.

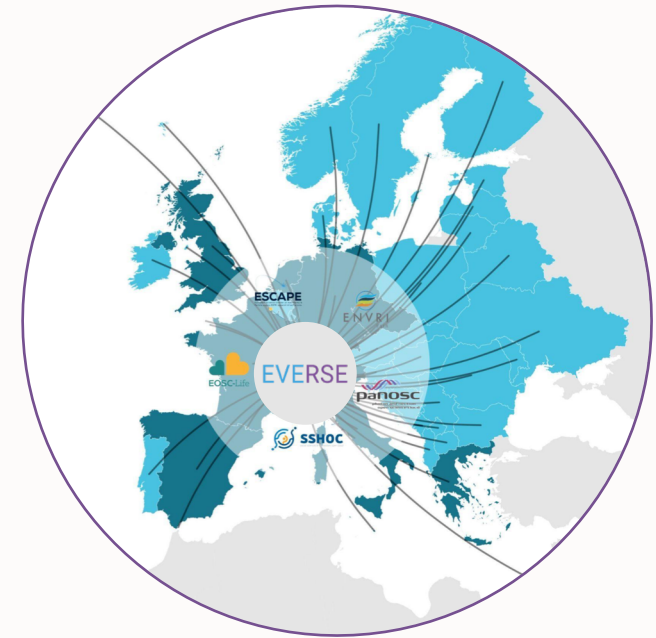
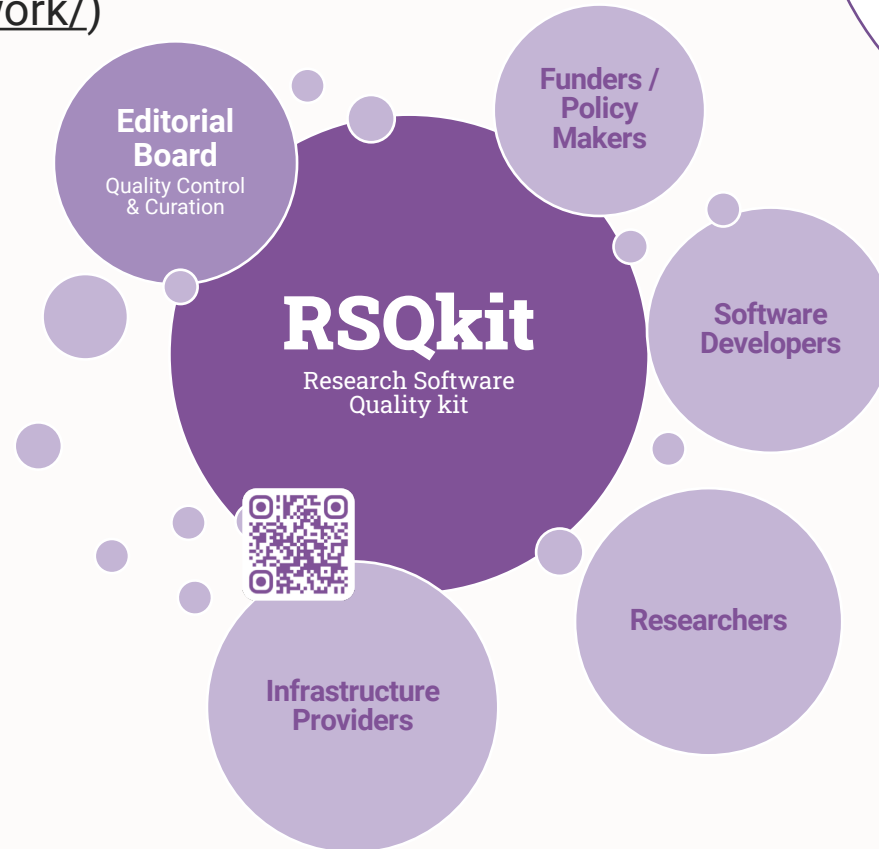
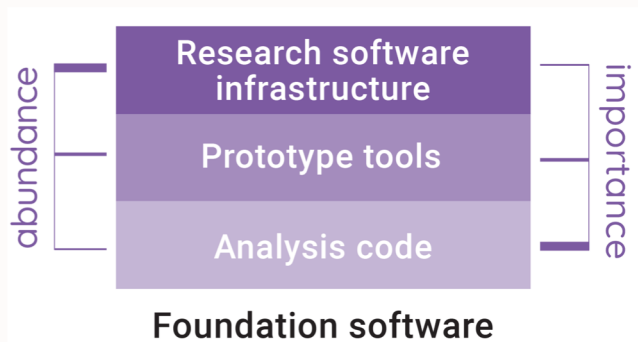
Build a **collaborative, community-led structure** for evaluating, verifying, and improving the quality of research software and code, by **actively involving** researchers, software developers, and other stakeholders in the research community.

Establishing a Community

How to contribute to, and engage with EVERSE

Elements of EVERSE

- The Network (<https://eiverse.software/network/>)
- RSQkit (<https://eiverse.software/RSQKit/>)
- Software Reference model
- Training
- Recognition framework



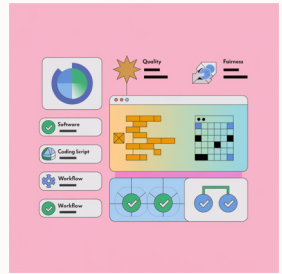
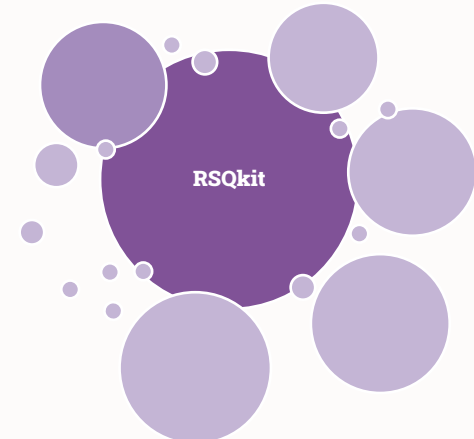
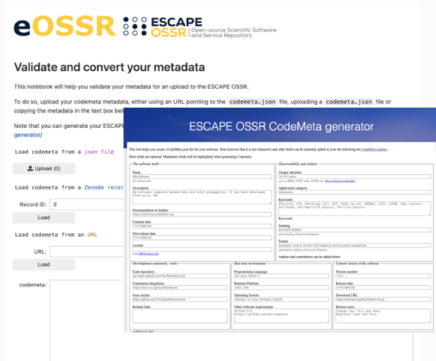
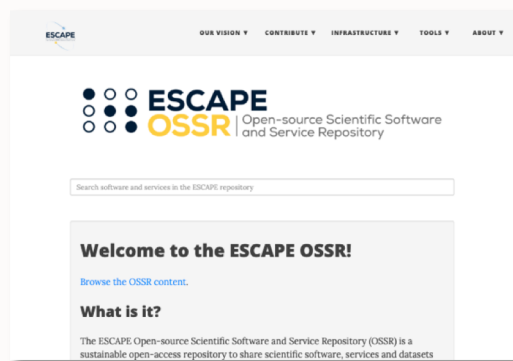
Join Us



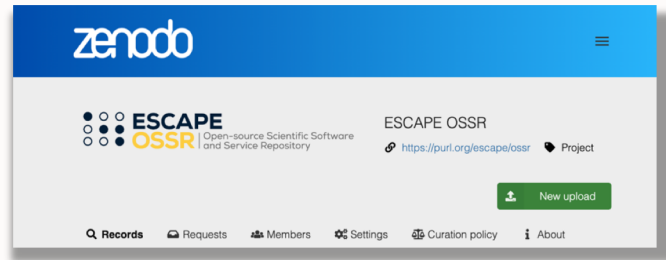
Any individual or organization that agrees with our vision statement is welcome to join the network



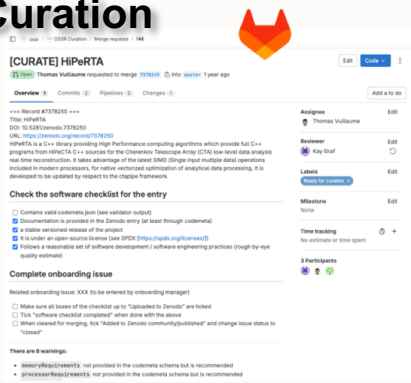
Recognition for software providers



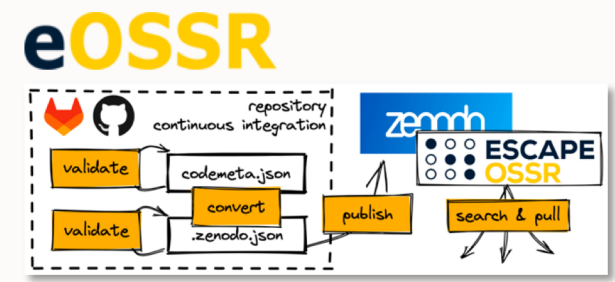
Dashboards to measure globally the software quality and its evolution



Curation



Integrated pipelines to measure and improve software quality



CONCLUSION

- OSSR establishes a community-based approach to publish **curated** software
- With EVERSE, it provides a **framework** to improve **software quality and recognition**
- **Contribute** your software now!

<http://purl.org/escape/ossr>

Vuillaume T, Al-Turany M, Füßling M et al. The ESCAPE Open-source Software and Service Repository, Open Res Europe 2023, <https://doi.org/10.12688/openreseurope.15692.2>

BACK-UP SLIDES

Software metadata

Software metadata are the implementation of FAIR principles

- Findable, Interoperable
- They should be part of the software and not defined or retained by an external service



OSSR uses [CodeMeta](#)

- Universal metadata schema to describe software
 - Not limited or linked to a specific service
 - Increasing adoption
 - Integration with other services
- A **codemeta.json** file with a number of required keys is mandatory to submit software to the OSSR. The file comes with the source code, at the root of the repository.

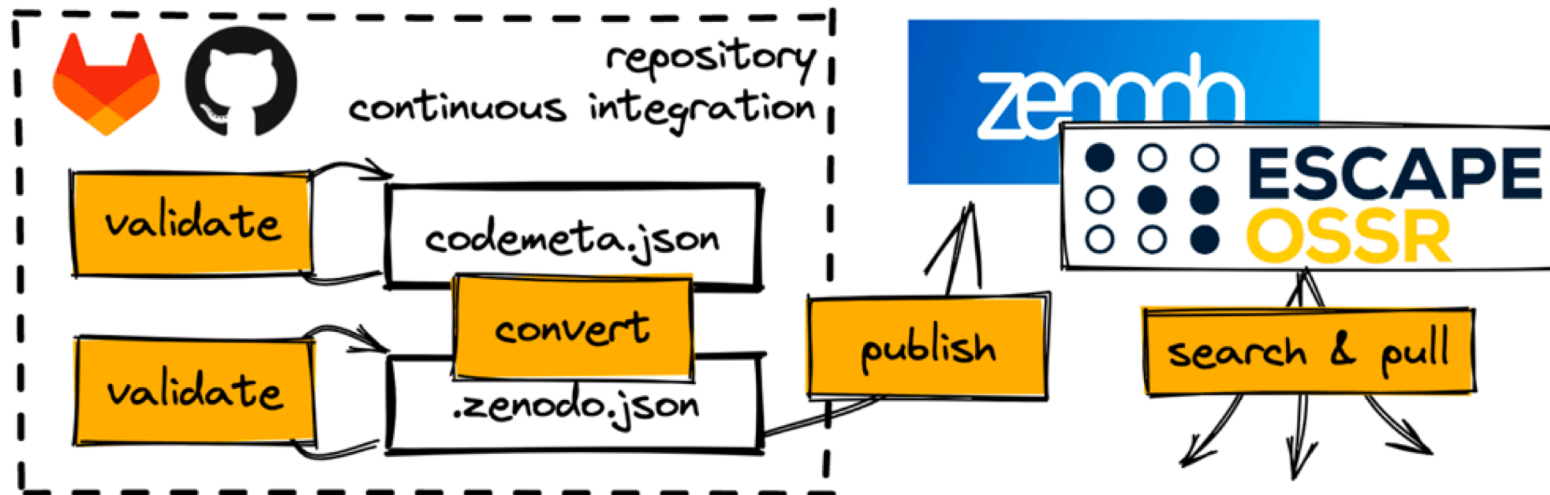
- The eOSSR is the OSSR Python library

- Connects to Zenodo API to handle:

- records: search, download, upload, publish, submit...
- communities: list records, list and handle submissions

- Handles OSSR metadata:

- Defines required one
- Converts from CodeMeta to Zenodo schema
- Validates **codemeta.json** file





Online tools: metadata generator, converter & validator



Validate and convert your metadata

This notebook will help you validate your metadata for an upload to the ESCAPE OSSR.

To do so, upload your codemeta metadata, either using an URL pointing to the `codemeta.json` file, uploading a `codemeta.json` file or copying the metadata in the text box below.

Note that you can generate your ESCAPE codemeta file using the online generator: <https://escape2020.pages.in2p3.fr/wp3/codemeta-generator/>

Load codemeta from a `json` file

Upload (0)

Load codemeta from a `Zenodo record ID`

Record ID:

Load

Load codemeta from an `URL`

URL:

Load

codemeta:

ESCAPE OSSR CodeMeta generator

This tool helps you create a CodeMeta.json file for your software. Note however that it is not exhaustive and other fields can be manually added in your file following the [CodeMeta schema](#). Most fields are optional. Mandatory fields will be highlighted when generating CodeMeta.

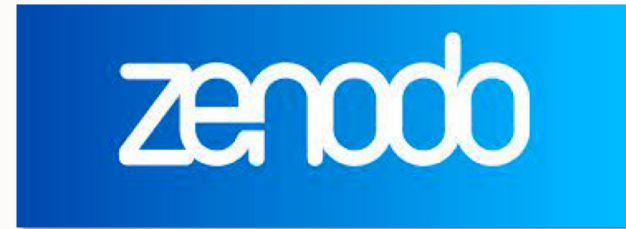
The software itself Name <input type="text" value="My Software"/> <small>the software title</small> Description <input type="text" value="My Software computes ephemerides and orbit propagation. It has been developed from early '80."/> Documentation or readme <input type="text" value="https://online-documentation.org"/> Creation date <input type="text" value="YYYY-MM-DD"/> First release date <input type="text" value="YYYY-MM-DD"/> License <input type="text"/> <small>from SPDX license list</small>	Discoverability and citation Unique identifier <input type="text" value="10.151.xxxxx"/> <small>such as ISBNs, GTIN codes, UUIDs etc. http://schema.org/identifier</small> Application category <input type="text" value="Astronomy"/> Keywords <input type="text" value="Projects: CTA, EGO-Virgo, ELT, EST, FAIR, HL-LHC, KM3NeT, LSST, LOFAR, SKA; Content: Astronomy, Astroparticle physics, Particle physics"/> Keywords <input type="text"/> Funding <input type="text" value="ESCAPE 824064"/> <small>grant funding software development</small> Funder <input type="text" value="European Union's Horizon 2020 research and innovation programme"/> <small>organization funding software development</small> Authors and contributors can be added below	Development community / tools Code repository <input type="text" value="git+https://github.com/You/RepoName.git"/> Continuous integration <input type="text" value="https://travis-ci.org/You/RepoName"/> Issue tracker <input type="text" value="https://github.com/You/RepoName/issues"/> Related links <input type="text"/>	Run-time environment Programming Language <input type="text" value="C#, Java, Python 3"/> Runtime Platform <input type="text" value=".NET, JVM"/> Operating System <input type="text" value="Android 1.6, Linux, Windows, macOS"/> Other software requirements <input type="text" value="Python 3.4"/> <small>https://github.com/psf/requests</small>	Current version of the software Version number <input type="text" value="1.0.0"/> Release date <input type="text" value="YYYY-MM-DD"/> Download URL <input type="text" value="https://example.org/MySoftware.tar.gz"/> Release notes <input type="text" value="Change log: this and that; Bugfixes: that and this."/>
---	---	---	---	--

- Help software developers to provide valid and complete metadata
- Get that first working version of **codemeta.json**
- Test things out

Gitlab to Zenodo




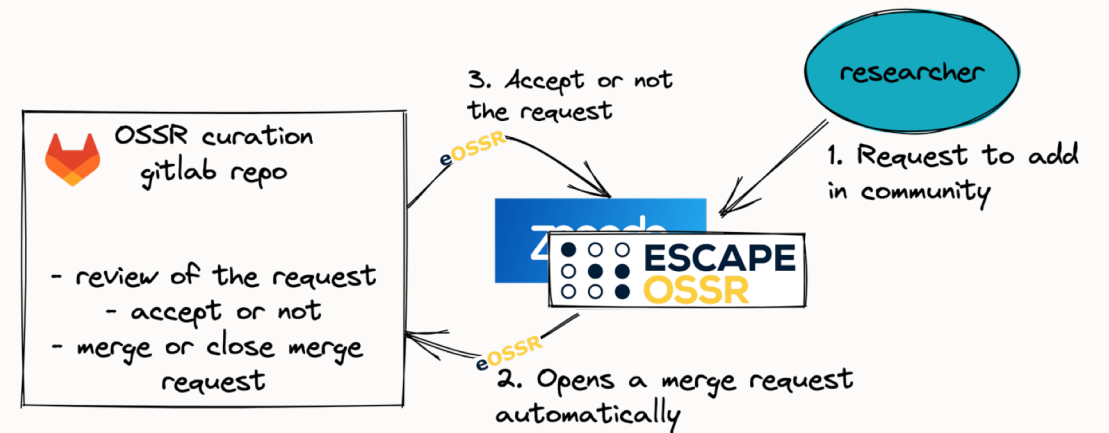
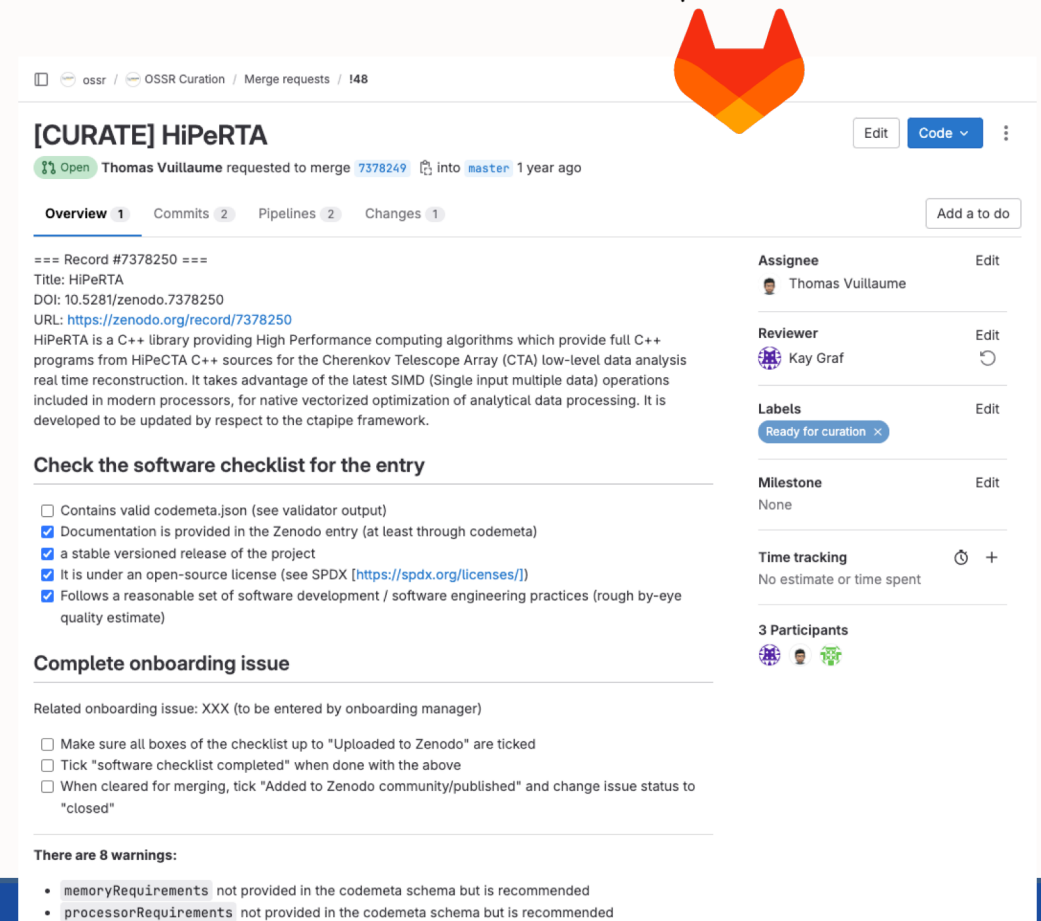
GitLab



- Zenodo has an efficient GitHub integration, but no GitLab integration
- Many ESFRIs use their own Gitlab instance
- We provide a simple gitlab-ci snippet
 - to publish your software to Zenodo / OSSR, e.g. when making a release in gitlab
 - using metadata provided in ***codemeta.json***

Software curation

- The OSSR is a curated software repository
 - implementation of the FAIR principles
 - good code practices
 - software quality
 - do not review scientific results  science paper
- Curation happens in a dedicated gitlab repository
 - completely open
 - automated checks
 - discussion between reviewers and providers
- Curation provides
 - Trust in the repository and provided content
 - Recognition for software providers

The screenshot shows a GitLab merge request for the repository 'ossr / OSSR Curation'. The merge request is titled '[CURATE] HiPeRTA' and was requested by Thomas Vuillaume to merge branch '7378249' into 'master' 1 year ago. The merge request is currently open and has 1 commit, 2 pipelines, and 1 change.

The merge request details include:

- Title:** HiPeRTA
- DOI:** 10.5281/zenodo.7378250
- URL:** <https://zenodo.org/record/7378250>

The description states: "HiPeRTA is a C++ library providing High Performance computing algorithms which provide full C++ programs from HiPeCTA C++ sources for the Cherenkov Telescope Array (CTA) low-level data analysis real time reconstruction. It takes advantage of the latest SIMD (Single input multiple data) operations included in modern processors, for native vectorized optimization of analytical data processing. It is developed to be updated by respect to the ctape framework."

The merge request includes a checklist for the entry:

- Contains valid codemeta.json (see validator output)
- Documentation is provided in the Zenodo entry (at least through codemeta)
- a stable versioned release of the project
- It is under an open-source license (see SPDX <https://spdx.org/licenses/>)
- Follows a reasonable set of software development / software engineering practices (rough by-eye quality estimate)

The merge request also includes a section for "Complete onboarding issue" with related onboarding issue: XXX (to be entered by onboarding manager).

There are 8 warnings:

- `memoryRequirements` not provided in the codemeta schema but is recommended
- `processorRequirements` not provided in the codemeta schema but is recommended

The merge request is assigned to Thomas Vuillaume and reviewed by Kay Graf. The merge request is labeled "Ready for curation" and has 3 participants.

Integration with other services



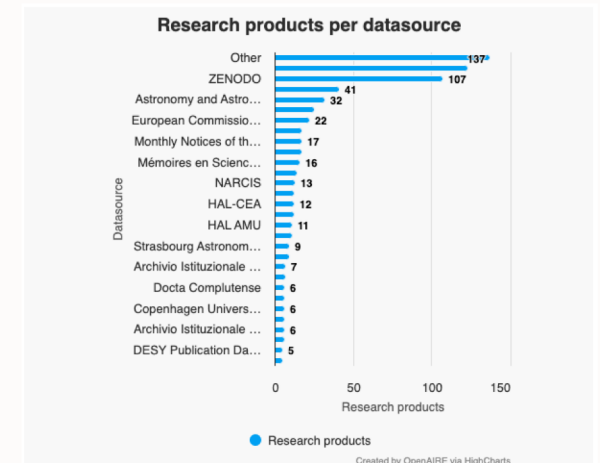
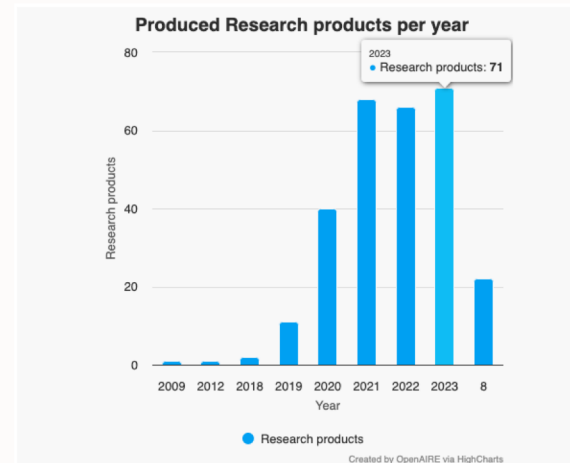
- connects to other services
- analyze data
- search and pull software from the OSSR



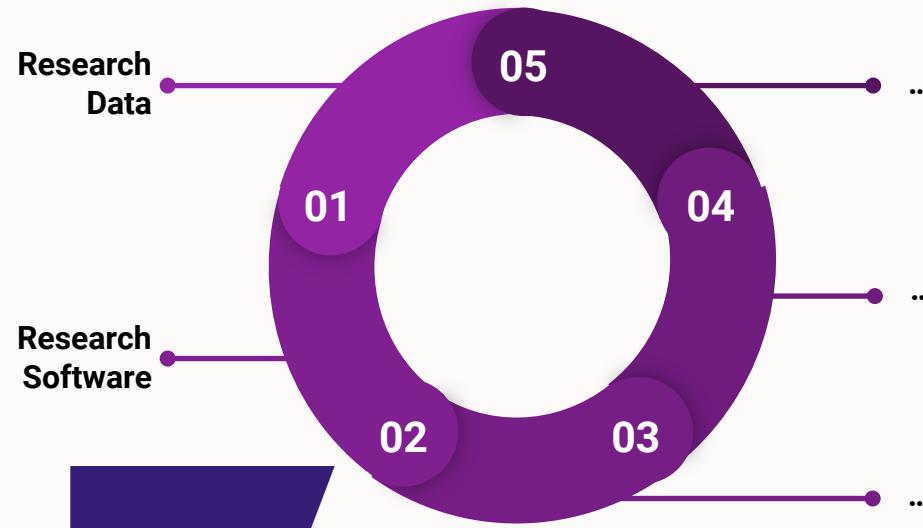
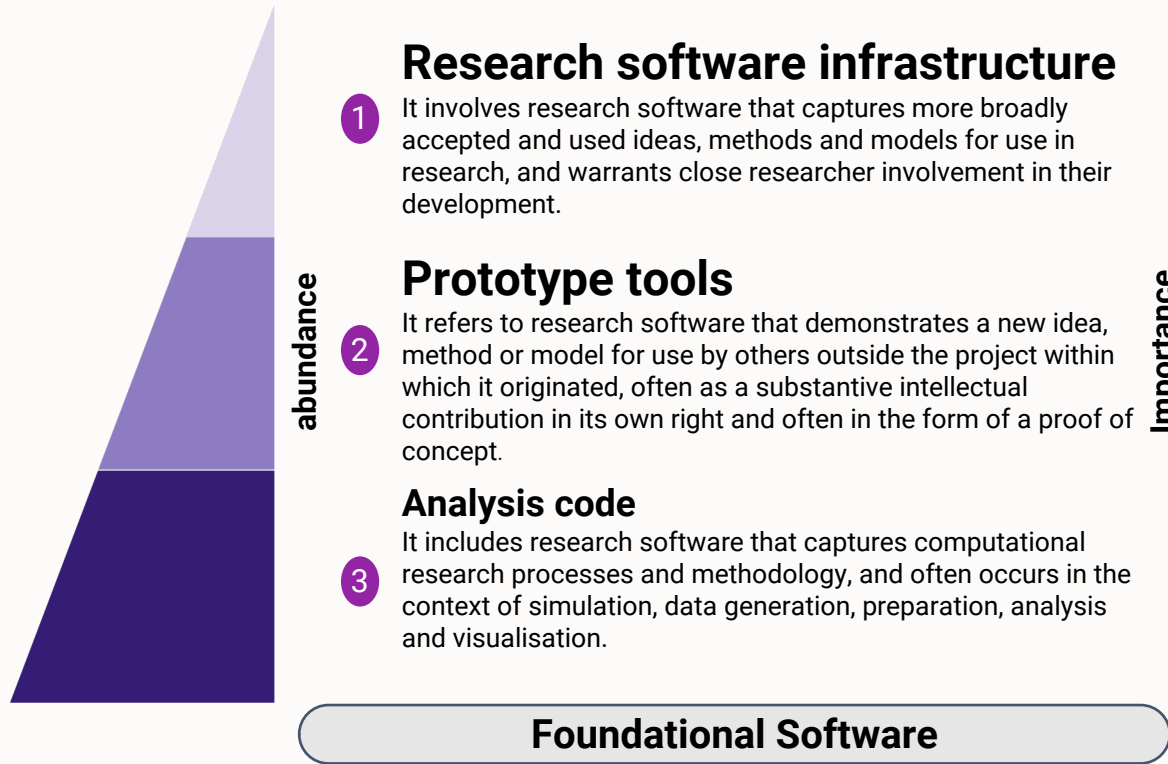
- EOSC integration
- Provides integrated statistics
- Connects with other data sources

The screenshot displays the ESCAPE interface with several service cards:

- WSRT-Aperif**: Data from the Aperif survey include imaging and time-domain data. The time-domain products consist of high-time resolution time-series data in the FITS format. The imaging data products include the new observations in the measurement and FITS standard format. In addition, processed data products are available, including calibration tables, calibrated visibilities, multi-frequency synthesis, continuum images, polarization images and cubes, and polarized neutral hydrogen HII line and weak radio. Full details of these data will be provided in upcoming papers by Lisenauer et al. 2020, Adams et al. 2020.
- ASTRON VO**: The Virtual Observatory defines a set of standards that can be used to download astronomical data. The ASTRON VO contains several image surveys which use images in the FITS format. Since the VO is currently under development, more data types will be available in the future.
- Zooniverse**: The Zooniverse is the world's largest and most popular platform for people-powered research. This research is made possible by volunteers – more than a million people around the world who come together to assist professional researchers. Our goal is to enable research that would not be possible, or practical, otherwise. Zooniverse research results in new discoveries, datasets useful to the wider research community and many publications.
- Virtual Observatory (VO)**: The Virtual Observatory offers a set of standards that can be used to download astronomical data.
- CTAO**: Cherenkov Telescope Array Observatory (CTAO) will be the world's first ground-based gamma-ray observatory with telescopes located in both northern and southern hemispheres. CTAO will observe at very high energies, with a diameter from 20 Gm to 300 Gm and a collection area that will exceed one million square meters.
- RUCIO**: Built on more than a decade of experience, Rucio serves the data needs of modern scientific experiments. Large amounts of data, countless numbers of file, heterogeneous storage systems, globally distributed data centres, monitoring and analysis. All coming together in modular solutions to fit your needs.
- Science Data Centre (SDC)**: Science Data Centre (SDC) is a service of Leibniz-Institute for Astrophysics IPHT. Its primary purpose is to provide a common platform for the solar community with solar data produced by a heterogeneous group of scientific instruments. The SDC hosts data from the ground-based solar telescope GREGOR on Tenerife and is planned to host data from the world's largest solar telescope DKIST of the National Solar Observatory USA.
- ZENODO**: Zenodo Built and developed by researchers, to ensure that everyone can join in Open Science.



Research Software as a first class citizen for the scientific endeavours



Not all software has the same level of importance