eosc | EVERSE

S3 School
Sustainable Scientific Software School

EVERSE

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

Original slides by Fotis Psomopoulos

Adapted and presented by Thomas Vuillaume at the S³ School, January 2026

Funded by the European Union

13 | 10 | 2025 by Fotis Psomopoulos (INAB|CERTH)

eosc | EVERSE

We all know reproducibility is hard — and increasingly, it depends on the **quality and sustainability of our software**. But how do we help researchers improve that in a structured, community-driven way?

Funded by
the European Union

EVERSE

Paving the way towards a **E**uropean **V**irtual Institut**e** for **R**esearch **S**oftware **E**xcellence

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 and BSC

eosc | EVERSE

Partners, associates, and affiliated entities

Pilots & Drivers

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

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

Life Sciences: *Integration of Science Cluster Life Science RI 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

Photon and neutron science: *Integration of Science Cluster PaNOSC through LEAPS/LENS*
Transition software to high performance computing (HPC) and heterogeneous computing architectures

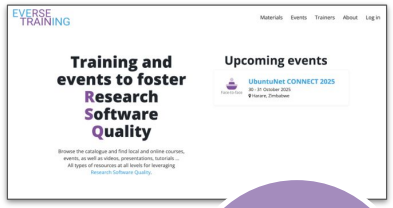
Social sciences: *Integration of Science Cluster SSHOC*
Develop a multilanguage textual analysis pipeline of tools that use a combination of open source tools and own code to create an integrated SotA tool capable of deploying locally or as a service

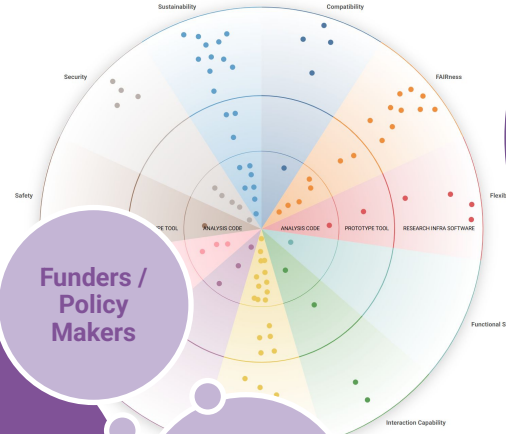
Building a Network of Research Software Competence Centres

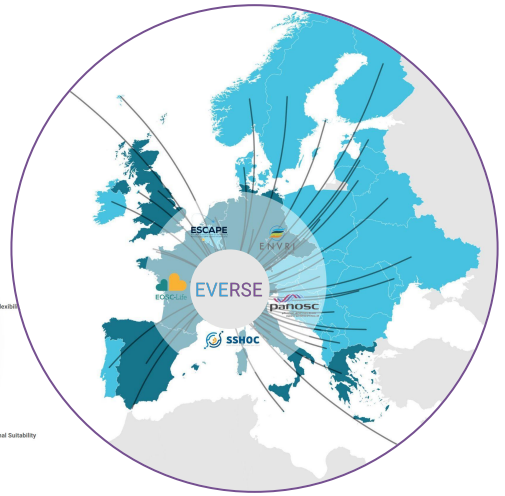
Paving the way towards a **E**uropean **V**irtual Institut**e** for **R**esearch **S**oftware **E**xcellence

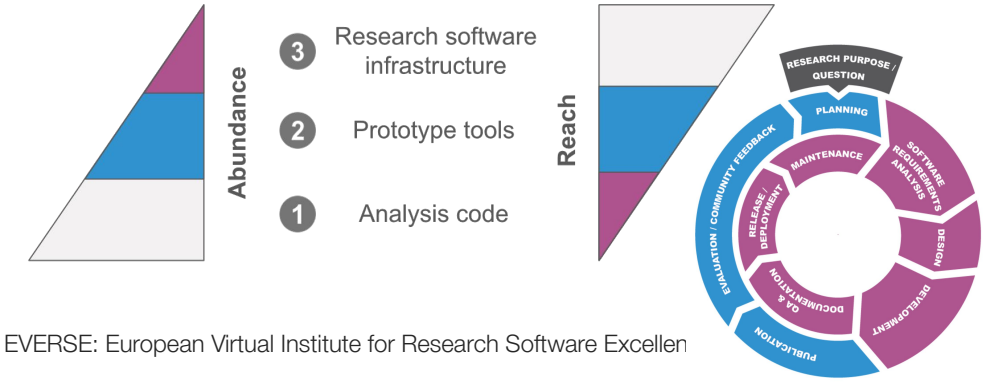
Outputs of EVERSE

- Network (everse.software/network/)
- RSQKit (everse.software/RSQKit/)
  - Three-Tier Model of Research Software
  - Research Software Lifecycle
  - Quality Dimensions & Indicators
- TechRadar (everse.software/TechRadar/)
- Training (EVERSE Training)
- Recognition framework









Join Us

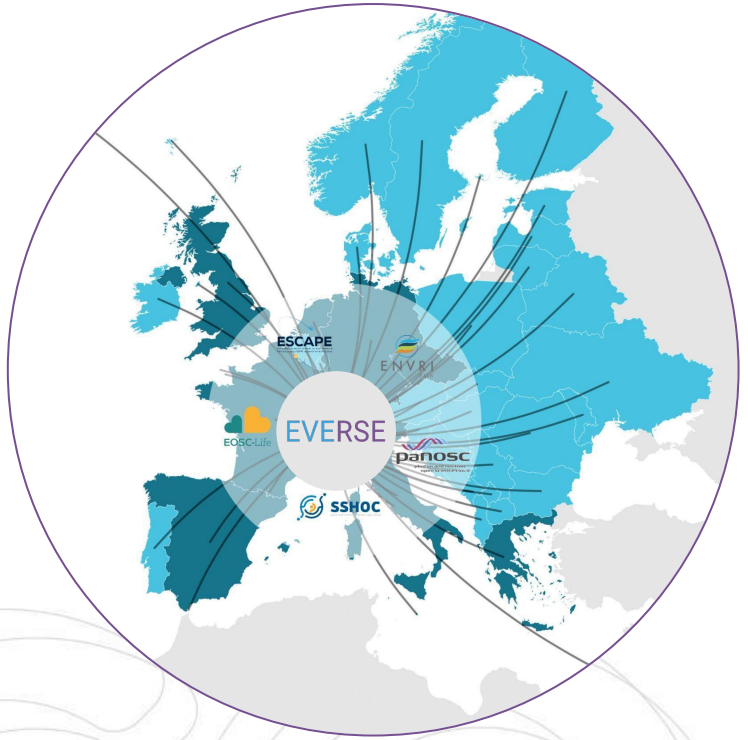
Any individual or organization that agrees with our vision statement is welcome to join the network

Building a Network of Research Software Competence Centres

Paving the way towards a **E**uropean **V**irtual Institut**e** for **R**esearch **S**oftware **E**xcellence

Network of Research Software Quality (everse.software/network/)

- The EVERSE project is committed to improving the quality of software in European research in the international context. **EVERSE is launching a Network of Research Software Quality to allow us to work with individuals and partner organisations** to achieve this common goal.
- **Establish a Community of Practice to improve the quality of research software in Europe and beyond**. EVERSE and partners work to provide standard and documented practices around tools and training for software developers, researchers who code and research software engineers and service providers.



A framework to think research software

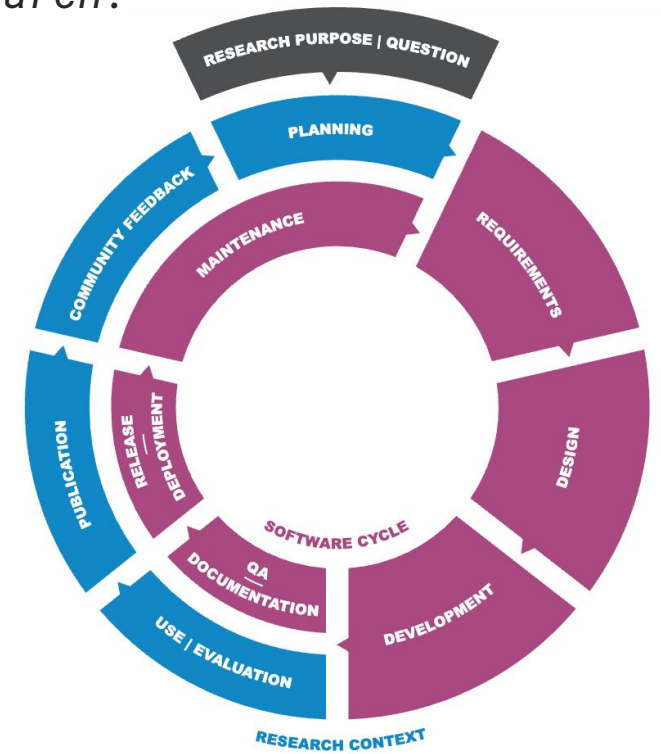
A definition of research software (FAIR4RS RDA working group)

- *Well identified software that is part of the research discovery process, which might require specialised domain knowledge and is by itself a contribution to science and research.*
- *Software that was developed with the intention of being part of research.*

⇒ **Research Software as a first-class citizen** for the scientific endeavours

Not all software has the same level of importance.

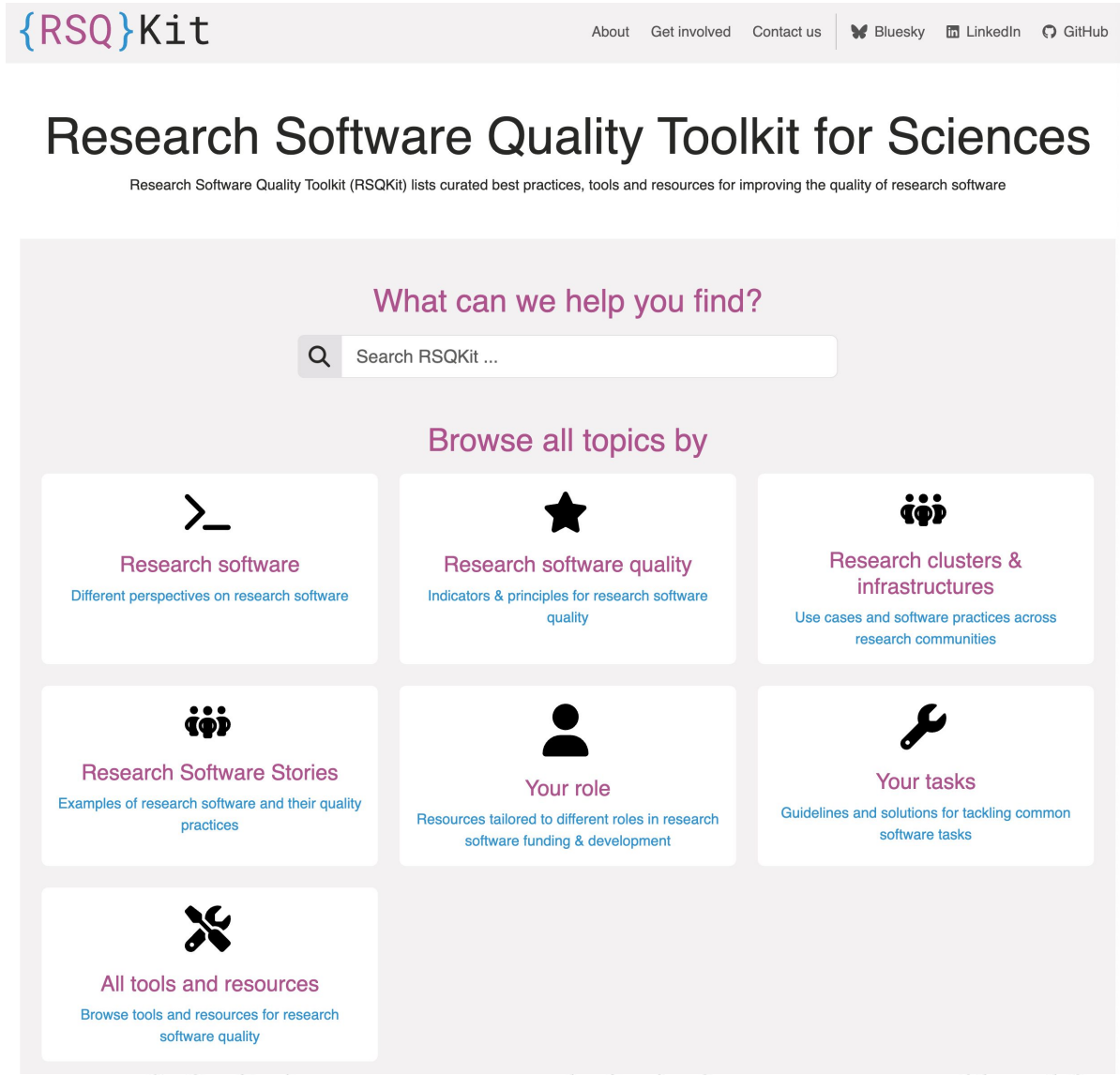
The software tier categorization.



The software lifecyle.

- The Research Software Quality Kit:
- Task driven guide – how to improve you research software quality (RSQ)
- Training related to improving RSQ
- Exemplars of good practice – Research Software Stories
- Roles involved in research software
- Research clusters and Infrastructures that make & use research software
- A model for understanding the research software process for different types of research software

https://everse.software/RSQKit/



{RSQ}Kit A product of EVERSE

EVERSE: European Virtual Institute for Research Software Excellence

Dimensions & Indicators

everse.software/indicators/website/indicators.html

- Which software quality dimensions / categories
- Which software quality indicators are these associated with
- What do these apply to
  - Best practices
  - Tools
  - Training
  - Assessment pipelines

Research Software Quality Dimensions

A research software quality dimension represents criteria relevant for assessing software quality (e.g., Documentation, Performance, etc.).

Search in table...

Name	Description	Source
Compatibility	Degree to which a product, system or component can exchange information with other products, systems or components, and/or perform its required functions while sharing the same common environment and resources.	ISO/IEC 25010 standard
FAIRness	FAIRness refers to the degree to which research software adheres to the FAIR principles: Findable, Accessible, Interoperable, and Reusable. These principles, adapted for research software, aim to enhance the discoverability, accessibility, interoperability, and reusability of software, thereby maximizing its value and impact in scientific research.	Introducing the FAIR Principles for research software
Flexibility	Degree to which a product can be adapted to changes in its requirements, contexts of use or system environment.	ISO/IEC 25010 standard
Functional suitability	Represents the degree to which a product or system provides functions that meet stated and implied needs when used under specified conditions	ISO/IEC 25010 standard
Interaction Capability	Degree to which a product or system can be interacted with by specified users to exchange information via the user interface to complete specific tasks in a variety of contexts of use.	ISO/IEC 25010 standard
Maintainability	This characteristic represents the degree of effectiveness and efficiency with which a product or system can be modified to improve it, correct it or adapt it to changes in environment, and in requirements.	ISO/IEC 25010 standard
Performance Efficiency	This characteristic represents the degree to which a product performs its functions within specified time and throughput parameters and is efficient in the use of resources (such as CPU, memory, storage, network devices, energy, materials...) under specified conditions.	ISO/IEC 25010 standard
Reliability	Degree to which a system, product or component performs specified functions under specified conditions for a specified period of time.	ISO/IEC 25010 standard
Safety	This characteristic represents the degree to which a product under defined conditions to avoid a state in which human life, health, property, or the environment is endangered.	ISO/IEC 25010 standard
Security	Degree to which a product or system defends against attack patterns by malicious actors and protects information and data so that persons or other products or systems have the degree of data access appropriate to their types and levels of authorization.	ISO/IEC 25010 standard
Sustainability	The capacity of the software to endure. In other words, sustainability means that the software will continue to be available in the future, on new platforms, meeting new needs.	Defining Software Sustainability

Fairly stable list

Research Software Quality Indicators

A research software quality indicator represents a specific software aspect that can be measured (e.g., FAIRness, test coverage, documentation coverage, etc.).

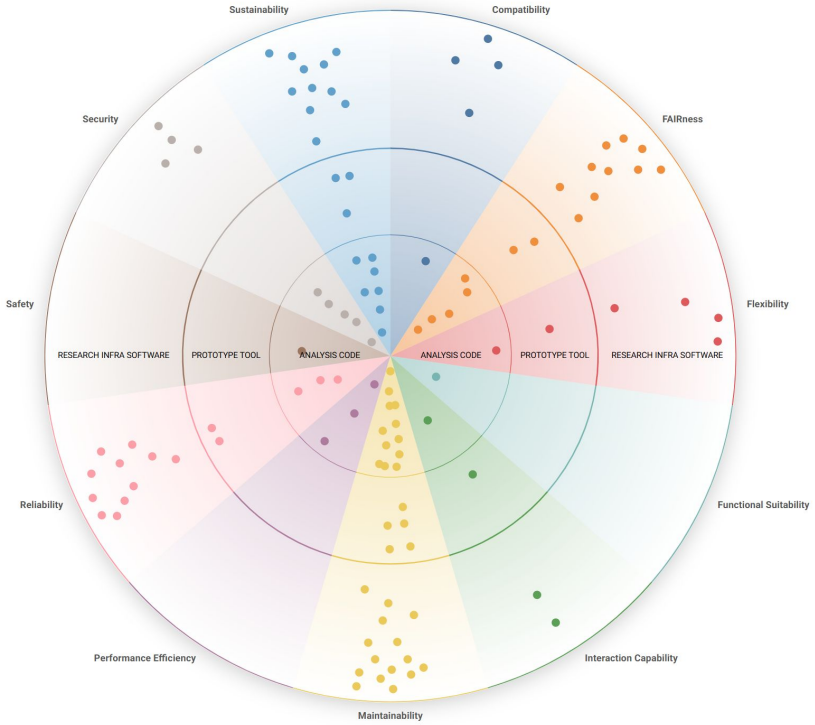
Search in table...

Name	Abbreviation	Keywords	Quality Dimension	Description	Source
CodeMeta completeness	codemeta_completeness	codemeta, metadata, fair	fairness	Indicator that checks the completeness of a codemeta.json file, i.e., the percentage of properties that are filled with metadata. This indicator does not assess the quality of the metadata fields available.	no source available
No critical vulnerabilities	no_critical_vulnerability	security, critical, vulnerability	security	Checks if reported critical vulnerabilities have been fixed	OpenSSF Best practice Critical Vulnerability Fixed
No leaked credentials	no_leaked_credentials	security, credential, leak, secret, password	security	Checks if hardcoded secrets like passwords, API keys, and tokens is stored in the public git repository	OpenSSF Best practice No Leaked Credentials
Software follows versioning standards	versioning_standards_use	version, versioning, fair, releases	fairness	This indicator aims to determine if the version (or versions) of a software tool follows an established community convention like semantic versioning (SemVer) or calendar versioning (CalVer).	D5.2 - Metrics for automated FAIR software assessment in a disciplinary context
Software has ci/cd workflows in its repository	repository_workflows	workflows, github, fair, repository	maintainability	This indicator tries aims determine if a software project makes use of workflows to automate processes like testing and deployment.	OpenSSF Scorecard: CI-Tests
Software has continuous integration tests	has_ci-tests	continuous, integration, test	maintainability	This indicator aims to determine if the project runs tests before pull requests are merged.	OpenSSF Scorecard: CI-Tests
Software has dependency management solution	dependency_management	sustaninability, dependencies	sustainability security compatibility	Reviews how external libraries and dependencies are managed to ensure compatibility and security.	Software Quality Indicators from BioHackathon 2024 (Sheet: Sustanability, Row: Su7) RSMD checklist
Software has descriptive	descriptive_metadata	metadata, software,	fairness	This indicator aims to determine if a software component comes	D5.2 - Metrics for automated

More being added actively ….

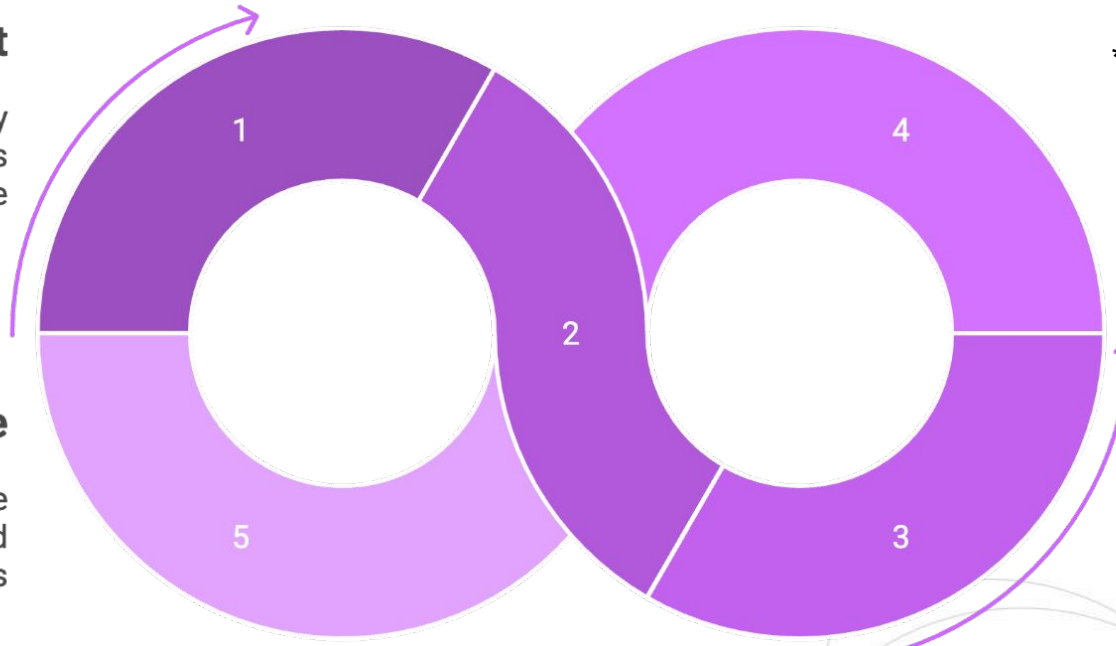
EVERSE Technology Radar

- **A platform to collect, classify, and present tools and services that help assess and enhance Research Software Quality**
- Two main elements:
  - A **catalogue of tools and services** designed to evaluate, measure, and improve the quality of research software.
  - The TechRadar, a **visual dashboard** that displays this catalogue in an accessible and intuitive way.
- TechRadar provides a structured overview of research software quality tools and services, systematically organized according to the EVERSE Quality Dimensions.



https://everse.software/TechRadar/

EVERSE Software Quality Cycle



RSQKit

- Identify role and user story
- Determine goals
- Get Knowledge

*** Quality Pipelines**

Use or create quality pipeline

TechRadar

Find relevant tools

Quality Dimensions and Indicators

- Identify relevant indicators
- Create new indicators

*** Dashverse**

Display and observe repository or aggregated metrics

*under development

EVERSE Training / Recognition framework

Used for the S3 School !

- **Manage training resources aligned with best practices in research software engineering** - Supporting the RSQKit via training material and events collected and curated in https://everse-training.app.cern.ch/
- **Develop a framework for recognition of Trainers and RSEs** - Developed a mechanism for recognition of activities to be used in CVs based on tools developed and supported within EVERSE
- **Establish a long-term training activity, supported by the communities** - Bi-monthly series of training events put in place
- **Next**: build a community of trainers and develop guidelines on how to incorporate training into existing curricula

eosc | EVERSE

From Guidelines to Policy

EVERSE Policy Brief I
https://zenodo.org/records/17233252

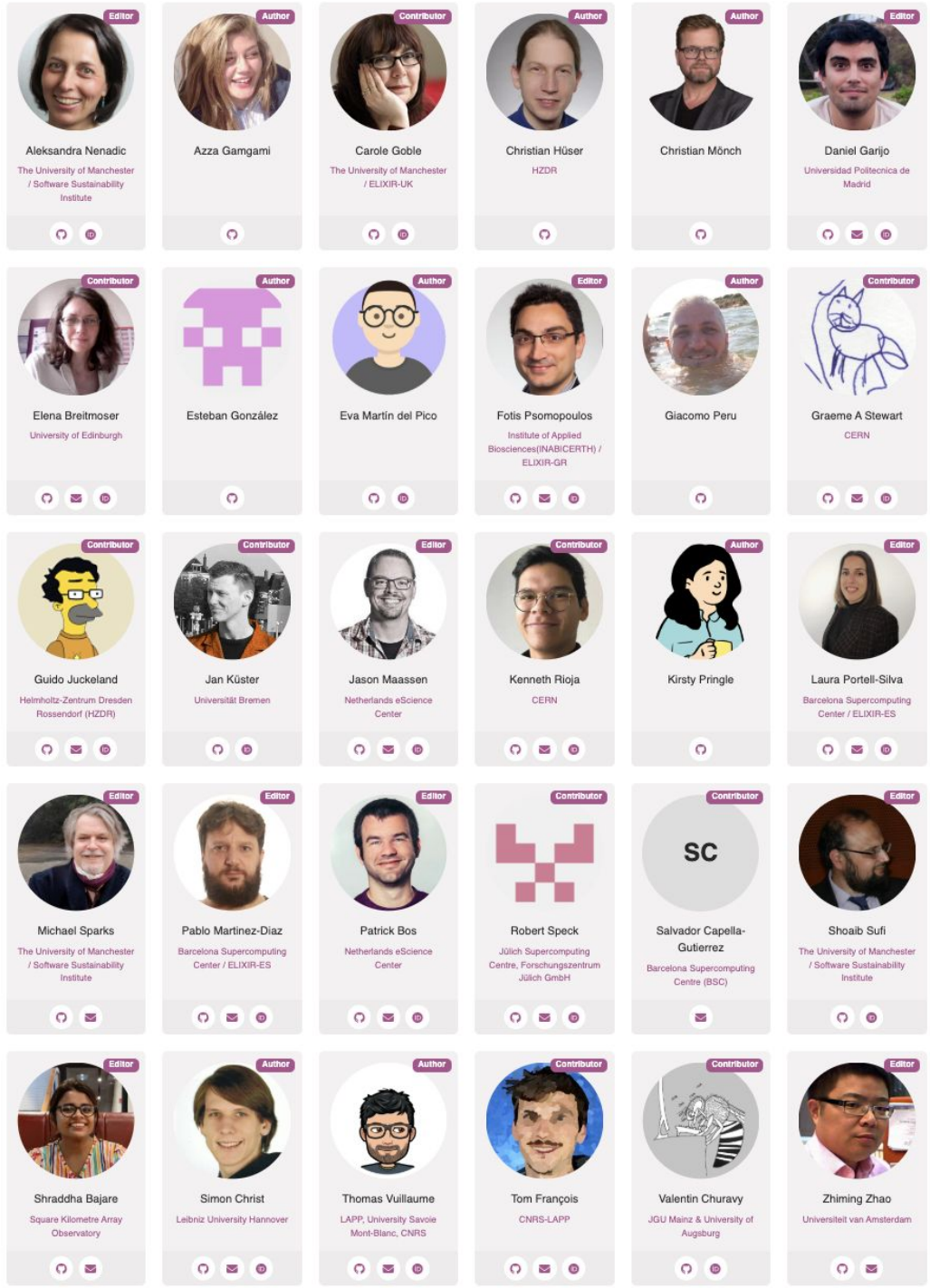
Key recommendations:

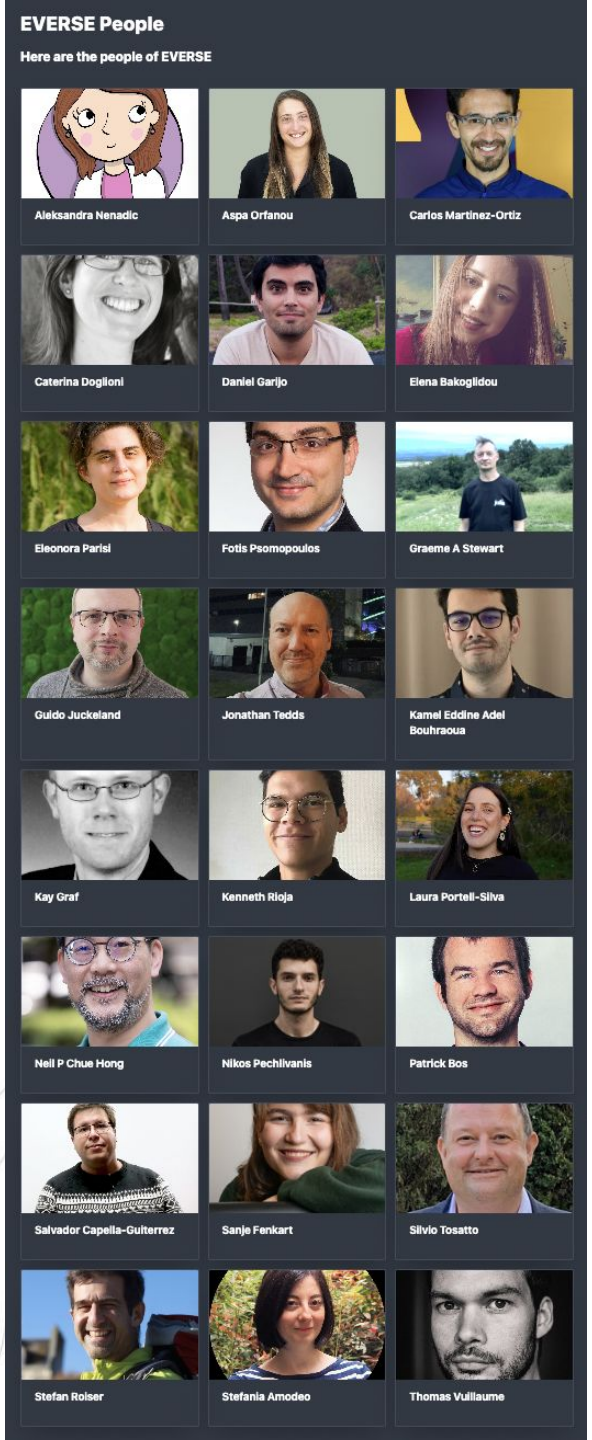
1. It is essential to establish and maintain a common European knowledge base for good practices on research software for the communities and driven by the communities.
2. We must establish and adopt a common definition of research software quality, building on the common perception of research software quality and its indicators across domains
3. We must recognize and reward contributions in research software and establish effective distribution of credit to Research Software Engineering professionals and its recognition by their employers.
4. We must establish and support capacity-building activities to consolidate the professionalism of the European practitioners in the research software field.
5. We must capitalize on the research software capacities within Europe and develop a resilient, robust and independent research software ecosystem and recognize its value as a European asset

eosc | EVERSE

Thanks

RSQKit all contributors





Thank you!

Contact: contact@everse.software

Website: https://www.everse.software/
BlueSky: https://bsky.app/profile/eosc-everse.bsky.social
LinkedIn: https://www.linkedin.com/company/eosc-everse/
FOSSTodon: https://fosstodon.org/@eosc_everse

Funded by the European Union

This project has received funding from the European Union's Horizon Europe Programme under GA 101129744 — EVERSE — HORIZON-INFRA-2023-EOSC-01-02