Ensure Research Software Quality

David, Mario (LIP) on behalf of the task force subgroup3:

Colom, Miguel (ENS Paris-Saclay); Garijo, Daniel (UPM); Castro, Leyla Jael (ZB MED); Louvet, Violaine (CNRS); Ronchieri, Elisabetta (INFN/CNAF); Torquati, Massimo (UNIPI); del Caño, Laura (CSIC/CNB); Leong, Cerlane (CSCS); Van den Bossche, Maxime (KU Leuven); Campos, Isabel (CSIC/IFCA); Di Cosmo, Roberto (INRIA)

Introduction

Framework of EOSC Association:

- Task Forces created to study and report about:
  - Open Science
  - Open Data
  - **Quality for Research Software** → "***Infrastructure for Quality Research Software***":
    - Subdivided into 3 sub groups
- This presentation is about sub group 3: "***Ensure Research Software Quality***"
  - 1st deliverable "Review of Software Quality Attributes and Characteristics": https://zenodo.org/record/8221384

"Research Software is commonly used to refer to software used and/or generated in a research context, including and not limited to scientific, non-scientific, commercial, academic and non-academic research." - Gruenpeter, Morane, et al.; Defining Research Software: a controversial discussion (Version 1). Zenodo. https://doi.org/10.5281/zenodo.5504016

Objectives

- Improve the quality of RS, both from the technical and organizational point of view for RS in general and in particular the software used in the services offered through EOSC.
- Identify Quality Attributes that are appropriate for RS and do a **recommendation**:
  - (Depends on the type and levels of RS - see next slide).

Research Software levels

Correspond to the four user stories defined in subgroup 1 (***Research Software Lifecycle***) of this Task Force.

https://docs.google.com/document/d/13TzhxNpGLtbFWYOmWSKkEHFC22ICmVpB/

1 - **Individual** creating research software for own use (e.g. a PhD student): easy to implement, good practice for research software at any level.

2 - **Team** creating an application or workflow for use within the team: easy to implement, good practice for research software at any level, useful for some basic coordination when more than one person participates.

3 and 4 - **OSS**: **A team / community** developing (possibly broadly applicable) open source research software or service platform: Open Source Software in general, all other cases.

Research Software stack AND types

Research Software stack taken from:

- K. Hinsen, "Dealing With Software Collapse," in Computing in Science & Engineering, vol. 21, no. 3, pp.104-108, 1 May-June 2019, doi: 10.1109/MCSE.2019.2900945.
- Defines 4 stack levels

RS stack	RS stack definition	RS type
4 - Project specific code	*"Software written by scientists for a specific research project. It can take various forms including scripts, notebooks, and workflows, but also special-purpose libraries and utilities."*	• Library • Analysis script, workflows • Services and platforms
3 - Domain specific tools	*"Tools and libraries that implement models and methods which are developed and used by specific communities. Gromacs, MMTK, Amber."*	• Library • Application (such as Monte-Carlo simulation) • Services and platforms
2 - Scientific infrastructure	*"Infrastructure created specifically for scientific computing, but not for any particular application domain: mathematical libraries such as BLAS, LAPACK, or SciPy, scientific data management tools such as HDF5."*	• Library • Framework • Services and platforms
1 - Non-scientific infrastructure	*"Compilers and interpreters, libraries for data management; gcc, python…"*	

Review of Software Quality models: Methodology

Follow the methodology for the survey, proposed by Kitchenham and Charters (B. Kitchenham and S. Charters.Guidelines for performing systematic literature reviews in software engineering. 2007) which has the following steps:

1. Source selection and search: Searched in the Scopus dataset, including the top five journals in Software Engineering related to software (https://research.com/journals-rankings/computer-science/software-programming) and articles of the "International Conference on Software Engineering". The search included the keywords "software quality" in the title of the target publications.
    a. ***Added documents and web resources that the Task Force subgroup considered relevant.***
2. Excluded journals not in the Software Engineering domain. Excluded articles not written in English.
3. Selection procedure: Skim article titles and abstracts. The process was performed by 2-3 people. Final list was agreed upon by the group through discussion about the relevance of the paper or document and analysis if that paper contains or proposes Software Quality attributes.
4. Review process: After following the selection procedure, we ended up with **19** articles, which have been reviewed in this survey. Some of the articles refer to the ISO/IEC 25010:2011(E) or to its precursor ISO/IEC 9126, have been grouped together.

Review of Software Quality models: Results

Obtained **272** results. Additional filtering, excluded:

- Papers with no abstracts.
- Proceedings/workshop summary.
- Those which did not seem related by browsing the abstract and title.
- Removed those papers that did not seem to propose quality models (e.g., if they talk about practices).

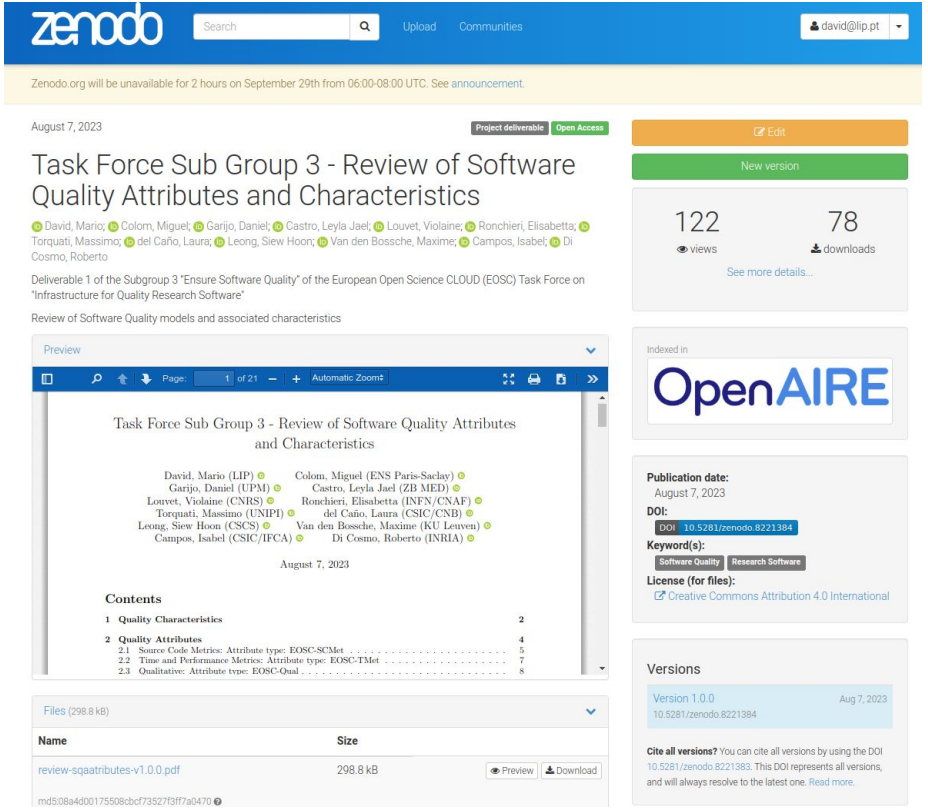
147 papers after filtering + **4** documents that were not published as paper but considered relevant.

We ended up with **19** articles after full review by 2-3 persons.

FCT Fundação para a Ciência e a Tecnologia | Lisb@20[20] | COMPETE 2020 | PORTUGAL 2020 | UNIÃO EUROPEIA

1st Deliverable of the Task Force Subgroup

- Published in zenodo: https://zenodo.org/record/8221384
- Contains:
  - Quality Characteristics
  - Quality Attributes: from all identified Quality models.
    - Include information about the papers and documents selection and review process.



The Software Quality Characteristics

Most are from ISO/IEC 25010:2011(E)

There are a total of 25 characteristics, examples are:

- **Functional suitability**: Degree to which a product or system provides functions that meet stated and implied needs when used under specified conditions.
- **Availability**: Degree to which a system, product or component is operational and accessible when required for use.
- **Reliability**: Degree to which a system, product or component performs specific functions under specified conditions for a specified period of time
- **Performance**: Performance relative to the amount of resources used under stated conditions.
- **Fault tolerance**: Degree to which a system, product or component operates as intended despite the presence of hardware or software faults.
- **Security**: Degree to which a product or system 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 authorisation.

The Quality Attributes: aggregation

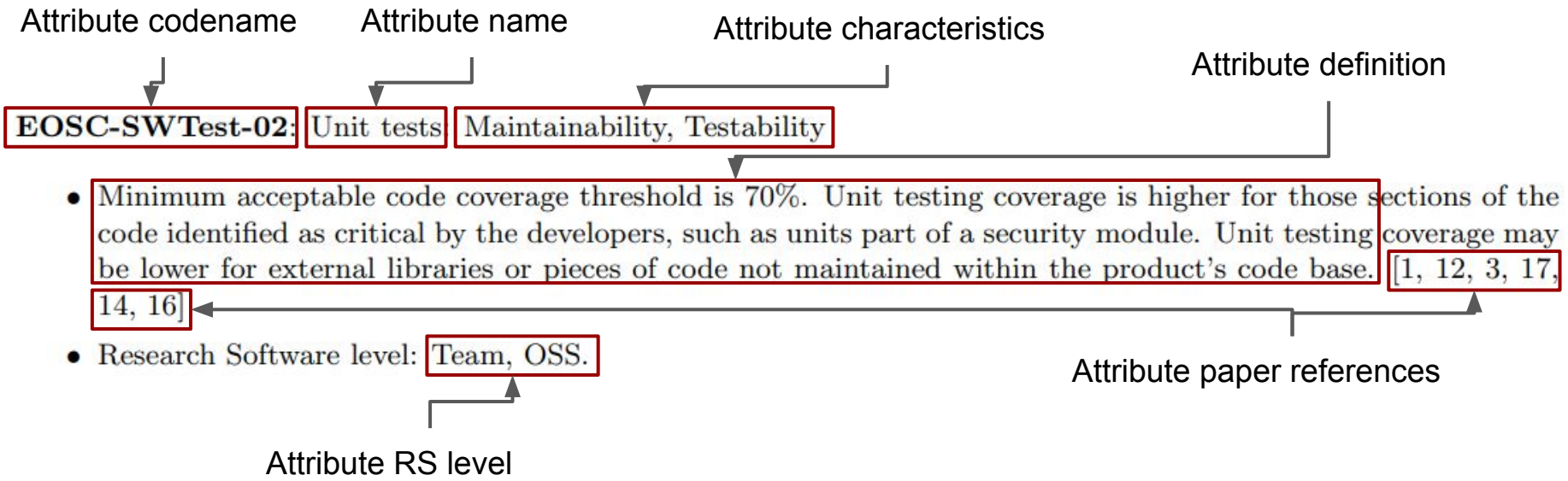
- Overall, a total of **241 Quality Attributes** were gathered in a table.
- A fraction of these attributes were aggregated into a single attribute:
  - When they were from different sources but with the same or very similar meaning.
- End up with **132 Quality Attributes** after aggregation:
  - All paper sources have been referenced.

The Quality Attributes: grouping

The metrics and attributes where subdivided into six categories and for each metric or attribute, a new codename was created by the EOSC Task Force:

1. Source Code Metrics **EOSC-SCMet**: 20 metrics.
2. Time Metrics **EOSC-TMet**: 11 metrics.
3. Qualitative Attributes **EOSC-Qual**: 30 attributes.
4. DevOps - SW release and management Attributes **EOSC-SWRelMan**: 34 attributes.
5. DevOps - Testing Attributes **EOSC-SWTest**: 25 attributes.
6. Service Operability Attributes **EOSC-SrvOps**: 12 attributes.

The Quality Attributes in 1st deliverable explained



https://zenodo.org/record/8221384

Source Code Metrics EOSC-SCMet

Metrics on the source code

EOSC-TF_Codename	EOSC-TF_Name	Characteristics	RS_level
EOSC-SCMet-01	Size of the software application	Maintainability	OSS
EOSC-SCMet-02	\% of redundant code	Maintainability, Modifiability	OSS
EOSC-SCMet-03	\# Lines of code	Maintainability	Individual, Team, OSS
EOSC-SCMet-04	\% of assertions	Maintainability	OSS
EOSC-SCMet-05	Cyclomatic Complexity	Maintainability	OSS
EOSC-SCMet-06	Cyclomatic Complexity test/source ratio	Maintainability	OSS
EOSC-SCMet-07	Number of arguments	Maintainability	Team, OSS
EOSC-SCMet-08	Number of function calls	Maintainability	Team, OSS
EOSC-SCMet-09	Modularity	Maintainability, Functional suitability	Individual, Team, OSS
EOSC-SCMet-10	Number of comments	Modifiability	Individual, Team, OSS

Time metrics EOSC-TMet

Metrics related to time or periods time

EOSC-TF_Codename	EOSC-TF_Name	Characteristics	RS_level
EOSC-TMet-01	Effort required for changes	Reliability, Supportability	Team, OSS
EOSC-TMet-02	\# Resolved bugs	Supportability	Team, OSS
EOSC-TMet-03	\# Open bugs	Supportability	Team, OSS
EOSC-TMet-04	Defect rates	Maintainability	OSS
EOSC-TMet-05	Integrity	Integrity, Maintainability	OSS
EOSC-TMet-06	Maintainability	Maintainability	OSS
EOSC-TMet-07	Adaptability	Reusability. Adaptability	OSS
EOSC-TMet-08	\# Registered users	Operability	OSS
EOSC-TMet-09	\# Active users	Operability	OSS

Qualitative Attributes EOSC-Qual

Qualitative attributes are obtained in general through surveys to or some manual analysis: SW developers, SW administrators, users. (Are not fit, or not possible to automatize).

EOSC-TF_Codename	EOSC-TF_Name	Characteristics	RS_level
EOSC-Qual-02	Complexity of architecture	Maintainability. Reusability	Team, OSS
EOSC-Qual-03	Complexity of a use case	Maintainability. Reusability. Usability	OSS
EOSC-Qual-04	Sustainable community	Supportability	Team, OSS
EOSC-Qual-05	User satisfaction	Attractiveness	OSS
EOSC-Qual-06	Usability	Usability	OSS
EOSC-Qual-07	Reliability	Reliability	OSS
EOSC-Qual-08	Efficiency	Time behavior, Performance	Individual, Team, OSS
EOSC-Qual-09	Portability	Portability	Team, OSS

DevOps - SW release and management EOSC-SWRelMan

Attributes related to SW release and management, based largely in DevOps, most can be automated for verification.

EOSC-TF_Codename	EOSC-TF_Name	Characteristics	RS_level
EOSC-SWRelMan-01	Open source	Supportability, Maintainability, Availability	Individual, Team, OSS
EOSC-SWRelMan-02	Version Control System (VCS)	Supportability, Maintainability	Individual, Team, OSS
EOSC-SWRelMan-03	Source code hosting	Supportability, Maintainability	Team, OSS
EOSC-SWRelMan-04	Working state version	Maintainability	Team, OSS
EOSC-SWRelMan-05	Changes branches	Maintainability	Team, OSS
EOSC-SWRelMan-06	Good patching practice	Maintainability	OSS
EOSC-SWRelMan-07	Support	Maintainability, Operability	Team, OSS
EOSC-SWRelMan-08	Code review	Maintainability	Team, OSS
EOSC-SWRelMan-09	Semantic Versioning	Maintainability	Individual, Team, OSS

DevOps - Testing Attributes EOSC-SWTest

Attributes related to SW testing, most can be automatically verified, follows DevOps approach.

EOSC-TF_Codename	EOSC-TF_Name	Characteristics	RS_level
EOSC-SWTest-01	Code style	Maintainability, Testability	Individual, Team, OSS
EOSC-SWTest-02	Unit tests	Maintainability, Testability	Team, OSS
EOSC-SWTest-03	Test doubles	Functional suitability, Testability	Team, OSS
EOSC-SWTest-04	Test-Driven Development (TDD)	Functional suitability, Maintainability, Testability	OSS
EOSC-SWTest-05	API testing	Functional suitability, Testability	Team, OSS
EOSC-SWTest-06	Integration testing	Functional suitability, Testability, Interoperability	OSS
EOSC-SWTest-07	Functional testing	Functional suitability, Testability	Team, OSS
EOSC-SWTest-08	Performance testing	Functional suitability, Testability	OSS
EOSC-SWTest-09	Stress testing	Functional suitability, Testability	OSS
EOSC-SWTest-12	Open Web Application Security Project (OWASP)	Security	OSS
EOSC-SWTest-13	Static Application Security Testing (SAST)	Security	OSS

Service Operability Attributes EOSC-SrvOps

Attributes related to a service in operation.

EOSC-TF_Codename	EOSC-TF_Name	Characteristics	RS_level
EOSC-SrvOps-01	Acceptable Usage Policy (AUP)	Supportability	OSS
EOSC-SrvOps-02	Access Policy and Terms of Use	Supportability	OSS
EOSC-SrvOps-03	Privacy policy	Supportability	OSS
EOSC-SrvOps-04	Operational Level Agreement (OLA)	Supportability	OSS
EOSC-SrvOps-05	Service Level Agreement (SLA)	Supportability	OSS
EOSC-SrvOps-06	Monitoring service public endpoints	Availability, Reliability	OSS
EOSC-SrvOps-07	Monitoring service public APIs	Availability, Reliability	OSS
EOSC-SrvOps-08	Monitoring service Web Interface	Availability, Reliability	OSS
EOSC-SrvOps-09	Monitoring security public endpoints and APIs	Availability, Reliability	OSS

Summary and next steps

- 1st delivery done: contains Quality Attributes divided into 6 categories, associated with Quality Characteristics, published in Zenodo.
- A significant fraction - 55% - of the quality attributes are from EOSC-Synergy's common baselines, http://hdl.handle.net/10261/160086 and https://digital.csic.es/handle/10261/214441, some of them also in other references:
  - 4 (of the 11) of the Time metrics EOSC-TMet
  - 31 (of the 34) of the DevOps - SW release and management EOSC-SWRelMan
  - 25 (of the 25) of the DevOps - Testing Attributes EOSC-SWTest
  - 12 (of the 12) of the Service Operability Attributes EOSC-SrvOps
- Next step is produce a document with recommendations of Quality Attributes for RS.
  - Detailed Table of Contents shown in the next slide.

Detailed Table of Contents