

Categorizing Research Software for RSE Research

Wilhelm (Willi) Hasselbring

Software Engineering, Kiel University, Germany
<http://se.informatik.uni-kiel.de>

School of Electronics & Computer Science, University of Southampton, UK
<https://www.southampton.ac.uk/research/areas/electronics-computer-science>

IRSER Community Meetup, 16th January 2024



Kiel University
Christian-Albrechts-Universität zu Kiel



Context, so far: German Special Interest Group “Research Software Engineering”

Interdisciplinary forum for:

- Software Engineering Researchers
- Research Software Engineers



<https://fg-rse.gi.de/> (German)

Task Forces:

- Categories of Research Software ←
- RSE Advocacy Strategy
- RSE Community Events
- RSE Online Community
- RSE Research
- RSE Software Development Guidelines
- RSE State of Nation Report

Research Software

RDA FAIR for Research Software (FAIR4RS) WG [Chue Hong et al. 2022] :

- Research software includes source code files, algorithms, scripts, computational workflows, and executables that are created **during** the research process or **for** a research purpose.
- Software components (e.g., operating systems, programming languages, libraries, etc.) that are used for research but were not created during or with a clear research intent should be considered **‘software in research’** and not **‘research software’**.

Research software should be **FAIR** [Hasselbring et al. 2020b, Lamprecht et al. 2020] and **open** [Hasselbring et al. 2020a].



Findable



Accessible

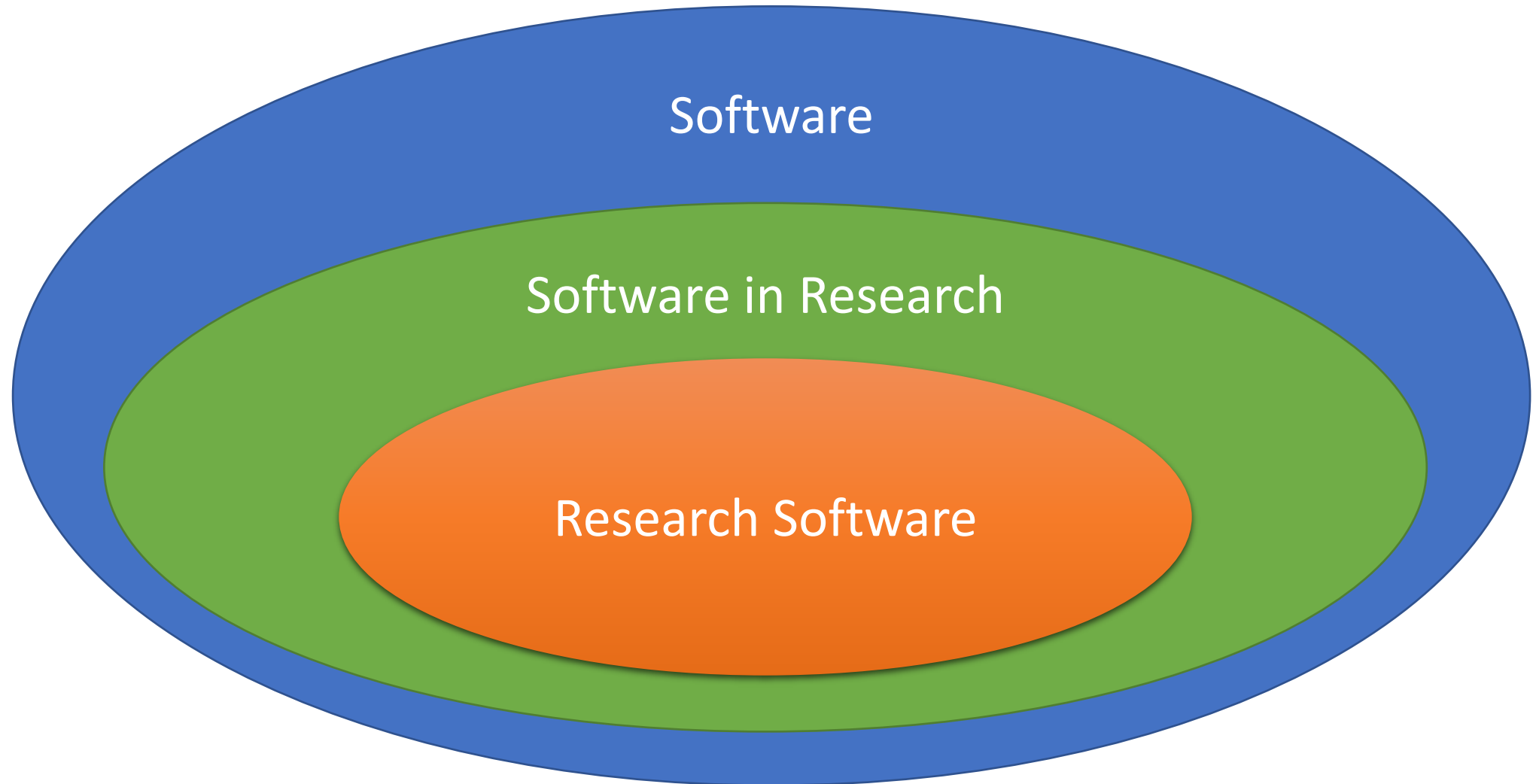


Interoperable

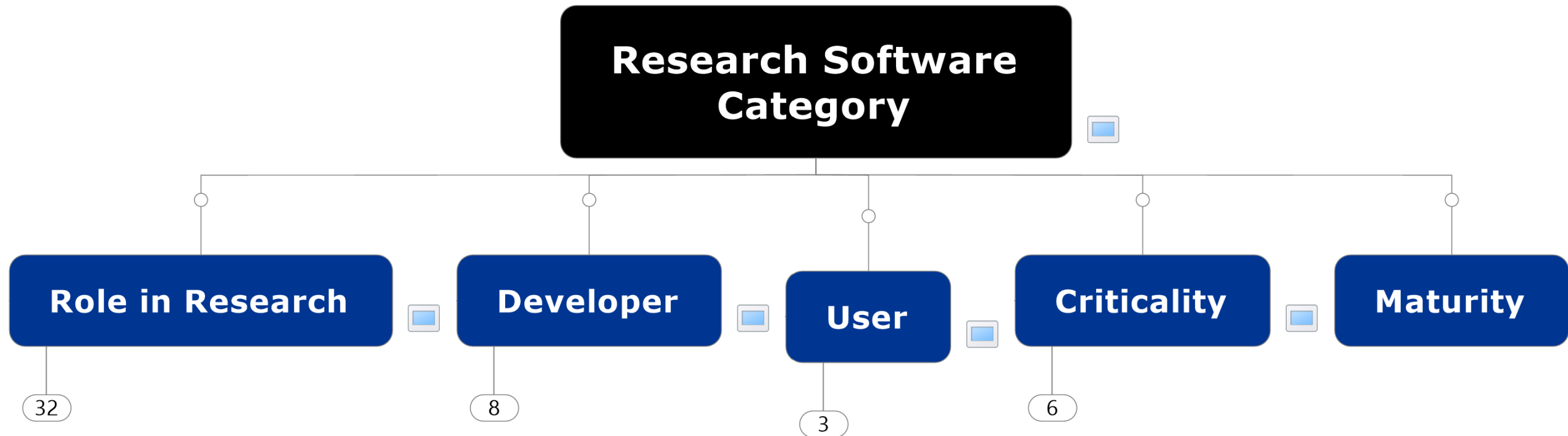


Reusable

Software Segmentation



Category Dimensions



Roles of Research Software

Research software mainly falls into one of the following role categories (and sometimes combinations):

- 1. Modeling, Simulation and Data Analytics** of, e.g., physical, chemical, social, or biological processes in spatio-temporal contexts.
 - Numerical and agent-based modeling and simulation (in silico experiments)
 - Data science and data engineering
 - Data-driven modeling
 - Integrative analysis (data assimilation and decision analysis)
 - Scientific visualization
- 2. Control and Monitoring Software** for complex physical or chemical experiments and instruments, including many forms of sensor-based data collection.
- 3. Proof-of-Concept Software** in science and engineering research.
- 4. Research Infrastructure Software**, such as research data and software management systems.

These categories have varying quality requirements!

Example for Category 1 (Modeling and simulation): Modularization of Earth-system simulation software as basis for domain-specific languages



Software Modularization



How to

- improve maintainability, stability, reusability, reproducibility, ... ?
- enable scalable execution in the Cloud?
- parallelize for high performance computing?
- test for higher quality?
- achieve higher flexibility?

[Johanson & Hasselbring 2017, Jung et al. 2021, 2022a, 2022b]

OceanDSL

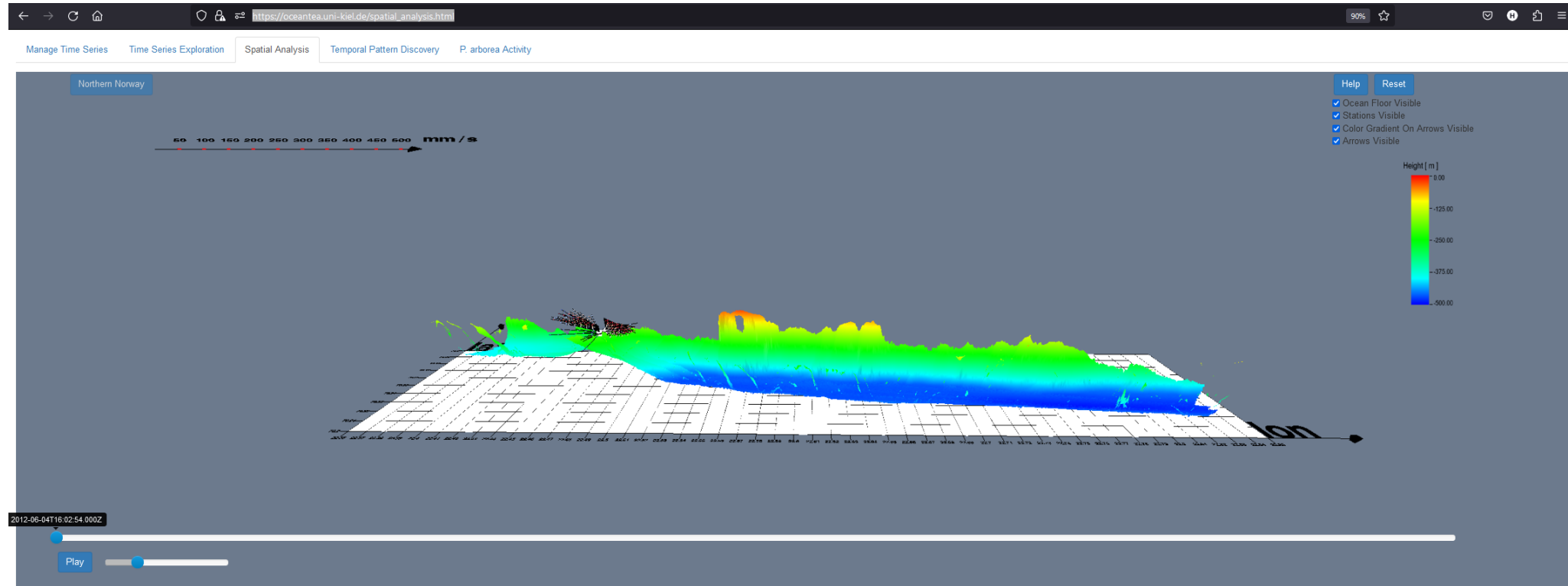
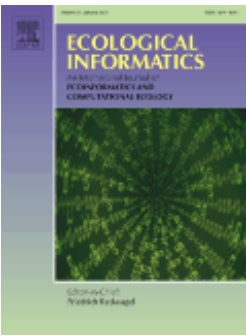
Funded by

DFG

Deutsche
Forschungsgemeinschaft

German Research Foundation

Example for Category 1 (Data analytics): OceanTEA: Analyzing Ocean Observation Data



Paper on the analysis results: [Johanson et al. 2017]

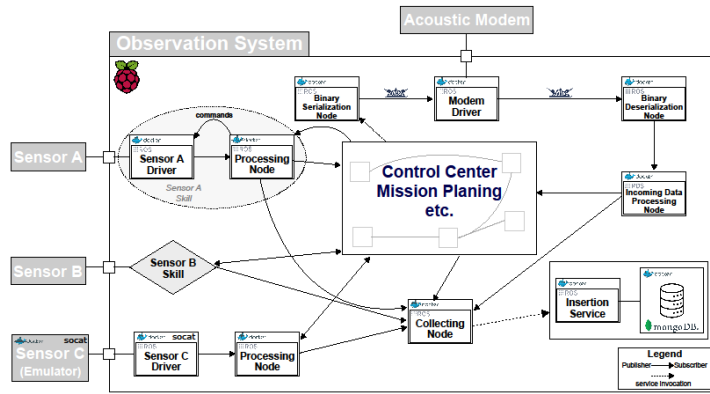
Paper on the software architecture: [Johanson et al. 2016]

Code: <https://github.com/cau-se/oceantea>

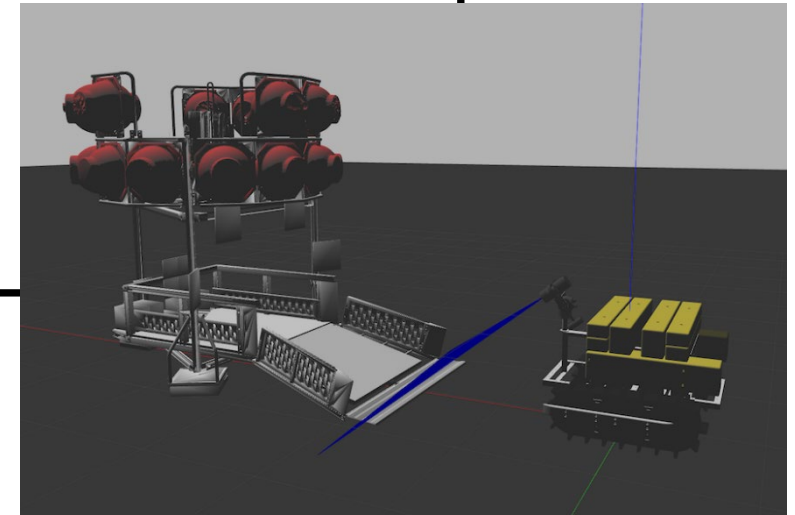


Example for Category 2 (Control & Monitoring): Software for Ocean Observation Robotics

Digital Twin
Prototype



Physical
Twin



Digital Twin

[Barbie et al. 2021]

Examples for Category 3 (Proof-of-Concept Software for Software Engineering Research)



<https://github.com/kieker-monitoring>

Kieker: A monitoring framework for software engineering research
[Hasselbring and van Hoorn 2020]



<https://github.com/ExplorViz>

ExplorViz: Research on software visualization, comprehension and collaboration
[Hasselbring et al. 2020c]



<https://github.com/cau-se/titan-ccp>

The Titan Control Center for Industrial DevOps analytics research
[Henning and Hasselbring 2021]

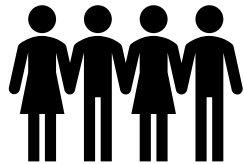
Examples for Category 4 (Infrastructure): EPrints Software for Open Access Repositories

The screenshot displays the OceanRep website interface. At the top left is the GEOMAR logo (Helmholtz Centre for Ocean Research Kiel) and at the top right is the OceanRep GEOMAR logo. A blue navigation bar contains a 'DE' language selector. The left sidebar lists navigation options: OceanRep Home, Contact, QUICK SEARCH (with a search box), Simple Search, Advanced Search, BROWSE (with sub-options for Author, Research division, Document type, Year, and Course of Study). The main content area features the title 'Categorizing Research Software for RSE Research.' and a record by 'Hasselbring, Wilhelm (2024)'. The record details include a talk at the IRSER Community Meetup, a PDF presentation, and a 'Tools' dropdown menu with 'BibTeX' selected and an 'Export' button. Metadata includes 'Document Type: Conference or Workshop Item (Talk)', 'Keywords: RSE Research', and 'Research affiliation: Kiel University > Software Engineering'. A 'Related URLs' section lists a 'Conference' link.

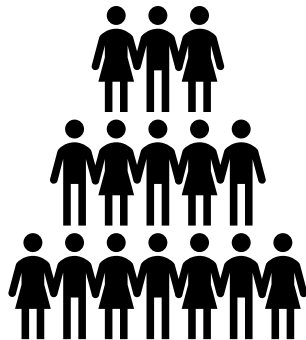
Developer Dimension: Stages of Research Software



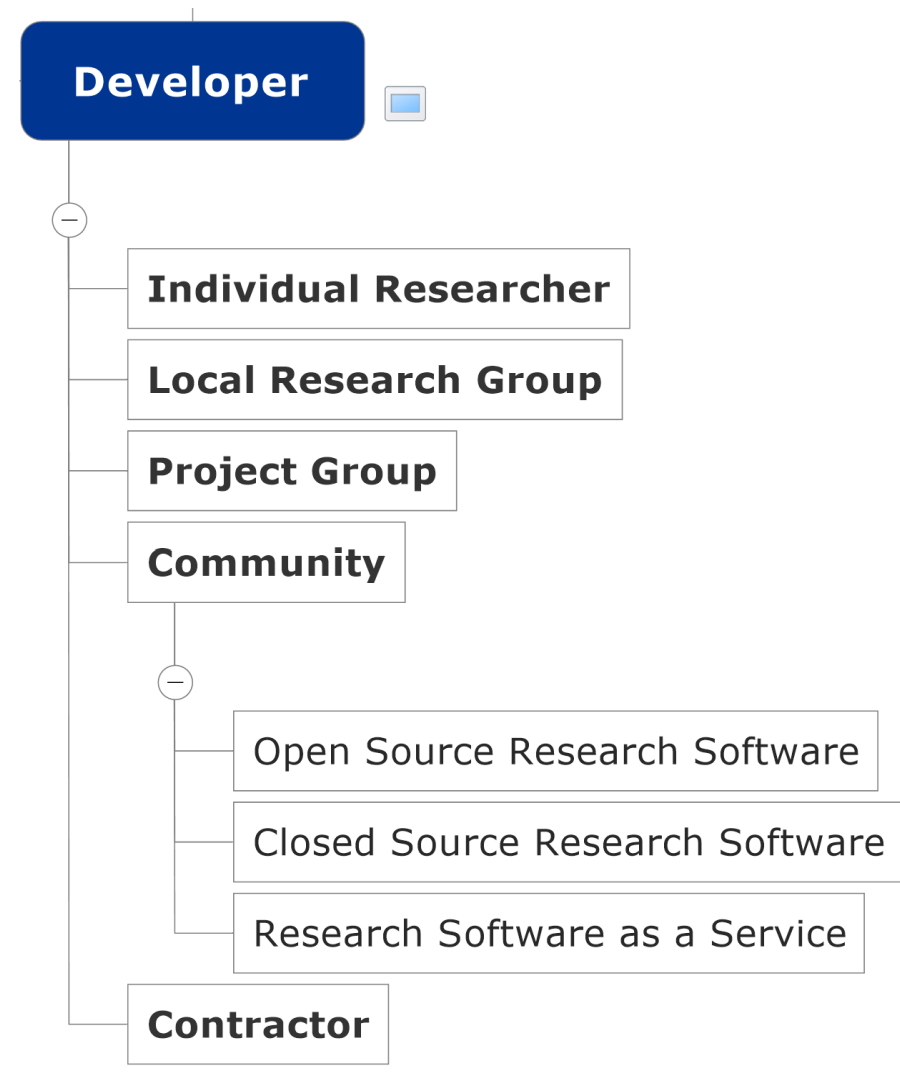
Individual Researcher



Local Research Group



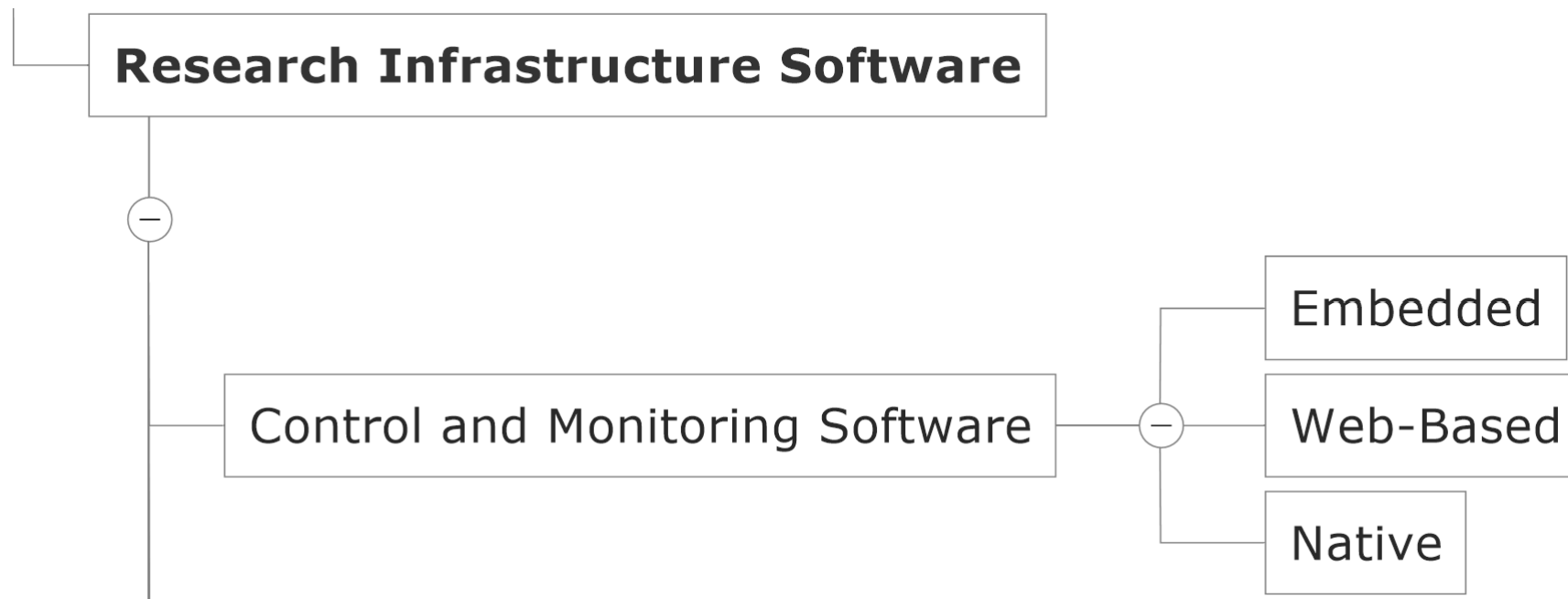
Community



What could we / others do with a Research Software Categorization?

- Assign specific quality requirements to the individual categories
- Recommend appropriate software engineering methods for the individual categories
 - This is, for instance, relevant for institutional software engineering guidelines and checklists.
 - For instance, requirements engineering may be relevant for Category 4, but not for Category 1.
 - For instance, a safety analysis may be relevant for Category 2, but not for Category 1 and 3.
 - Related: Good Practices for High-Quality Scientific Computing [Dubey 2022]
- Design appropriate teaching / education programs for the individual categories
- Rationale:
 - We need to understand what kinds of software we have to deal with, and their specific quality requirements

Open Issue: How to classify Control and Monitoring Software?



Comments welcome!

References

- [Barbie et al. 2021] Barbie, A., Pech, N., Hasselbring, W., Flögel, S., Wenzhöfer, F., Walter, M., Shchekinova, E., Busse, M., Türk, M., Hofbauer, M. und Sommer, S.: “Developing an Underwater Network of Ocean Observation Systems with Digital Twin Prototypes - A Field Report from the Baltic Sea.” IEEE Internet Computing. 2021. DOI <https://doi.org/10.1109/MIC.2021.3065245>
- [Chue Hong et al. 2022] N. P., Chue Hong, et al. (2022). FAIR Principles for Research Software version 1.0. (FAIR4RS Principles v1.0). Research Data Alliance. DOI <https://doi.org/10.15497/RDA00068>
- [Dubey 2022] A. Dubey: “Good Practices for High-Quality Scientific Computing,” in Computing in Science & Engineering, vol. 24, no. 6, pp. 72-76, Nov.-Dec. 2022. DOI <https://doi.org/10.1109/MCSE.2023.3259259>
- [Felderer et al. 2023] Felderer, M., Goedicke, M., Grunske, L., Hasselbring, W., Lamprecht, A. L. und Rumpe, B.: “Toward Research Software Engineering Research”. 2023. DOI <https://doi.org/10.5281/ZENODO.8020525>.
- [Hasselbring et al. 2020a] W. Hasselbring, L. Carr, S. Hettrick, H. Packer, T. Tiropanis: “Open Source Research Software”. In: Computer, 53 (8), pp. 84-88. 2020. DOI <https://doi.org/10.1109/MC.2020.2998235>
- [Hasselbring et al. 2020b] W. Hasselbring, L. Carr, S. Hettrick, H. Packer, T. Tiropanis: “From FAIR Research Data toward FAIR and Open Research Software”, it - Information Technology, 2020. DOI <https://doi.org/10.1515/itit-2019-0040>
- [Hasselbring et al. 2020c] Hasselbring, W., Krause, A., Zirkelbach, C.: “ExplorViz: Research on software visualization, comprehension and collaboration.” In: Software Impacts, 6, 2020. DOI <https://doi.org/10.1016/j.simpa.2020.100034>.
- [Hasselbring and van Hoorn 2020] Hasselbring, W., van Hoorn, A.: “Kieker: A monitoring framework for software engineering research.” In: Software Impacts, 5, 2020. pp. 1-5. DOI <https://doi.org/10.1016/j.simpa.2020.100019>
- [Henning and Hasselbring 2021] Henning, S., Hasselbring, W.: “The Titan Control Center for Industrial DevOps analytics research,” In: Software Impacts, 7, 2021 . DOI <https://doi.org/10.1016/j.simpa.2020.100050>

References

- [Johanson et al. 2016] A. Johanson, S. Flögel, C. Dullo, W. Hasselbring: “OceanTEA: Exploring Ocean-Derived Climate Data Using Microservices”. In: Sixth International Workshop on Climate Informatics (CI 2016), 2016, DOI <http://dx.doi.org/10.5065/D6K072N6>
- [Johanson et al. 2017] A. Johanson, S. Flögel, C. Dullo, P. Linke, W. Hasselbring: “Modeling Polyp Activity of Paragorgia arborea Using Supervised Learning”, In: Ecological Informatics, 39. pp. 109-118. 2017, DOI <https://doi.org/10.1016/j.ecoinf.2017.02.007>
- [Johanson & Hasselbring 2017] A. Johanson, W. Hasselbring: “Effectiveness and efficiency of a domain-specific language for high-performance marine ecosystem simulation: a controlled experiment”, In: Empirical Software Engineering 22 (8). pp. 2206-2236, 2017. DOI <https://doi.org/10.1007/s10664-016-9483-z>
- [Johanson & Hasselbring 2018] A. Johanson, W. Hasselbring: “Software Engineering for Computational Science: Past, Present, Future”, In: Computing in Science & Engineering, 2018. DOI <https://doi.org/10.1109/MCSE.2018.021651343>
- [Jung et al. 2021] R. Jung, S. Gundlach, S. Simonov, W. Hasselbring: “Developing Domain-Specific Languages for Ocean Modeling”. In: Software Engineering 2021 Satellite Events, <http://ceur-ws.org/Vol-2814/>
- [Jung et al. 2022a] R. Jung, S. Gundlach, W. Hasselbring: “Software development processes in ocean system modeling.” In: International Journal of Modeling, Simulation, and Scientific Computing, 13 (02). 2022, DOI <https://doi.org/10.1142/S1793962322300023>.
- [Jung et al. 2022b] R. Jung, S. Gundlach, W. Hasselbring: “Thematic domain analysis for ocean modeling.” In: Environmental Modelling & Software, 150. p. 105323. 2022, DOI <https://doi.org/10.1016/j.envsoft.2022.105323>.
- [Lamprecht et al. 2020] A.-L. Lamprecht et al.: “Towards FAIR principles for research software.” In: Data Science 3, 1 (June 2020), 37–59. DOI <https://doi.org/10.3233/ds-190026>
- [Lamprecht et al. 2022] A.-L. Lamprecht et al.: “What Do We (Not) Know About Research Software Engineering?.” In: Journal of Open Research Software, 10(1). 2022. DOI <https://doi.org/10.5334/jors.384>