

# The Shared Language of Crowds: A Crowd-Sourced Approach to Mapping Research Software Engineering and Software Engineering Research Terminology

Timo Kehrer  
University of Bern  
Bern, Switzerland  
timo.kehrer@unibe.ch

Robert Haines  
University of Manchester  
Manchester, United Kingdom  
robert.haines@manchester.ac.uk

Guido Juckeland  
Helmholtz-Zentrum Dresden  
Rossendorf  
Dresden, Germany  
g.juckeland@hzdr.de

Shurui Zhou  
University of Toronto  
Toronto, Ontario, Canada  
shurui@ece.utoronto.ca

David E. Bernholdt  
Oak Ridge National Laboratory  
Oak Ridge, Tennessee, United States  
bernholdtde@ornl.gov

## Abstract

Research software engineers (RSEs) often have little or no formal training in software engineering, acquiring their understanding of software engineering through experience, self-study, and interactions with colleagues. Anecdotally, RSEs often use different terminology than is commonly used in the software engineering research (SER) community for similar concepts. This has the potential to create challenges in communication between the two communities. As an outcome of the Dagstuhl Seminar 24161, we developed an approach to explore the terminologies used by the RSE and SER communities, which uses crowd-sourcing to build out a website presenting a “mapping of terms” between the groups. In this work, we will briefly recap our approach and the design of the website and describe some of our initial experiences gathering input from members of both communities, offering a look “behind the scenes” of the development of the term mapping content.

### ACM Reference Format:

Timo Kehrer, Robert Haines, Guido Juckeland, Shurui Zhou, and David E. Bernholdt. 2026. The Shared Language of Crowds: A Crowd-Sourced Approach to Mapping Research Software Engineering and Software Engineering Research Terminology. In *1st International Workshop on Software Engineering and Research Software (SERS '26)*, April 12–18, 2026, Rio de Janeiro, Brazil. ACM, New York, NY, USA, 2 pages. <https://doi.org/10.1145/3786172.3788358>

## 1 Motivation and Background

Experience shows that different areas of science and technology often develop distinctive terminologies as a result of limited interactions with other communities. This can pose challenges to multidisciplinary collaborations – teams need to spend time to learn each other’s terminology, and may not even realize that they are using a given term in different ways, or that different terms actually refer to the same underlying concept.

In this work, we consider two communities: the traditional software engineering research (SER) community, which systematically investigates software engineering practices, processes, tools, and methods (hereafter referred to as *SE fundamentals*) and the research software engineering (RSE) community, in which most participants have developed an informal understanding of SE fundamentals through their own and the shared experience of others in their community. Although both communities have a common interest in understanding and practicing the development of high-quality software, historically, the level of interaction between these two communities appears to be rather limited.

As participants in the Dagstuhl Seminar 24161 “Research Software Engineering: Bridging Knowledge Gaps”<sup>1</sup> [1], our working group decided to dig deeper to try to understand the extent to which the RSE and SER communities are or are not using common terminology as starting point to help identify gaps where RSE experience may provide useful inputs to software engineering research, or where results from the SER community might benefit developers of research software if they were more widely known and used.

## 2 Approach

Our approach involves developing mappings between terminology used in the SER community and that used for comparable concepts in the RSE community, along with concise descriptions of the underlying concept from each community’s perspective. Terms may be the same or different between the two communities, and multiple related terms can be grouped together. We also capture coarse estimates of (a) the level of *awareness* of the SE fundamental in the RSE community and (b) the level of *usage*. From the SER perspective, we capture coarse estimates of software engineering research activities to improve the awareness or usage by RSEs, along with a brief description of the opportunities identified. Early on, we realized the necessity to have a systematic and widely-recognized source of terminology and made the decision to use the *Guide to the Software Engineering Body of Knowledge* (SWEBOK) [3] as our initial source of “ground truth” for SER terminology. The RSE community currently lacks anything comparable, but we expect that the



This work is licensed under a Creative Commons Attribution 4.0 International License. *SERS '26, Rio de Janeiro, Brazil*  
© 2026 Copyright held by the owner/author(s).  
ACM ISBN 979-8-4007-2404-6/2026/04  
<https://doi.org/10.1145/3786172.3788358>

<sup>1</sup><https://www.dagstuhl.de/seminars/seminar-calendar/seminar-details/24161>, <https://ser-rse-bridge.github.io/>

work in the community to define educational pathways, common training materials, and other shared resources will eventually lead to comparable references from the RSE perspective, which we can incorporate into our term mapping effort as they become available.

We are pursuing a crowd-sourcing strategy to gather inputs for this mapping of terms – initially through meetings with members of both communities, and hopefully as awareness of and experience with the effort spreads, through direct interaction with the web platform we have developed to support this activity. An “editorial board”, currently consisting of the working group members, will provide oversight of the content and resolve differences of opinion where necessary.

### 3 Collaboration Platform

We have developed a simple web-based platform to gather inputs and present the evolving term mapping results to the public. The latter are presented through a website<sup>2</sup>, built with the Jekyll static site generation tool. The main page is based on the table of contents of the SWEBOOK, referring to the SWEBOOK section numbers and headings. Links are provided to pages that capture individual term mappings.

Individual term pages systematically present the information discussed in the previous section, based on a YAML schema [2]. Each page of the site supports a discussion capability, which is one mechanism for community members to provide input. The entire site is hosted in a repository on GitHub<sup>3</sup>, allowing the use of the usual “issue” and “pull request” mechanisms as other means of community input. The GitHub Pages service is used to host the website, and the Discussions capability is used as a backend for the webpage discussion capability.

The overall design and implementation of the collaboration platform were based on the desire for simplicity and to utilize tools that are likely to be familiar and comfortable to members of both target communities. The use of version control allows for examination of the evolution of the term mappings over time and with further input.

### 4 Initial Crowd-Sourcing Experience

So far, we have conducted several crowdsourcing sessions with participants from the Dagstuhl seminar to pilot our approach and the platform. In each session, we had participants who identified variously as software engineering researchers or research software engineers. Members of the working group who were able to attend the session also joined in the discussions rather than acting strictly as observers.

After an introduction to the goals, the approach, and the platform, we let the participants decide which terms they wanted to discuss. Some of the discussions focused on the structure of the SWEBOOK and the variable granularity with which it seems useful to extract terminology. For example, in SWEBOOK 4.0, the chapter on testing lists a variety of testing strategies that participants felt were worth treating as individual terms, while a single section at the same level on non-functional testing, named within it (not called out as separate sections), includes numerous terms, which participants

concluded could usefully be grouped into four separate term pages. Discussions identified terms with common understandings in both communities, others where terms would be understood somewhat differently between the communities, and others where a common concept might be recognized by different terms.

Participants also provided useful feedback on the website and on the approach, including suggestions for some of the efforts that they felt might develop into sources of systematic terminology in the RSE community, such as The Turing Way<sup>4</sup> and the Research Software Quality Toolkit for Sciences (RSQKit)<sup>5</sup>

In the workshop presentation, we plan to expand on our crowd-sourcing experience, as time permits.

### 5 Future Work

We envision additional crowd-sourcing sessions, including engagement through various institutional and national RSE groups and organizations, and outreach via prominent SER venues, such as the International Conference on Software Engineering, ACM SIGSOFT, and others. In time, we hope that the site will become widely enough known that practitioners from either community will visit and contribute regularly. The term mapping effort was also envisioned, from the beginning, to provide a touchstone to inspire new software engineering research studies, cross-community education and training, and other activities that will help bridge the gaps between the SER and RSE communities.

### Acknowledgments

This material is based in part upon work supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Next-Generation Scientific Software Technologies program, under contract number DE-AC05-00OR22725 to Oak Ridge National Laboratory.

Zhou’s work was supported in part by the Alfred P. Sloan Foundation (G-2022-19472).

### References

- [1] Stephan Druskat, Lars Grunske, Caroline Jay, and Daniel S. Katz. 2024. Research Software Engineering: Bridging Knowledge Gaps (Dagstuhl Seminar 24161). *Dagstuhl Reports* 14, 4 (2024), 42–53. doi:10.4230/DagRep.14.4.42
- [2] Timo Kehrer, Robert Haines, Guido Juckeland, Shurui Zhou, and David E. Bernholdt. 2025. Do Research Software Engineers and Software Engineering Researchers Speak the Same Language? *Computing in Science and Engineering* 27, 2 (April–June 2025), 18–26. doi:10.1109/MCSE.2025.3557236 Preprint available online at <https://arxiv.org/abs/2507.02665>.
- [3] H. Washizaki (Ed.). 2024. *Guide to the Software Engineering Body of Knowledge (SWEBOOK Guide), Version 4.0* (4th ed.). IEEE Computer Society, Washington, DC, USA. <https://www.swebok.org>.

<sup>2</sup><https://ser-rse-bridge.github.io/mapping-of-terms/>

<sup>3</sup><https://github.com/ser-rse-bridge/mapping-of-terms>

<sup>4</sup><https://book.the-turing-way.org/>

<sup>5</sup><https://everse.software/RSQKit/>