The Research Software Engineer (RSE):

Who is that?

And what skills do they have to help you?

Talk held at EuroTUG 2024, June 24, 2024

by J. Ph. Thiele

Leibniz
Association

Many scientists and engineers spend much of their lives writing, debugging and maintaining software,

but only a handful have ever been taught how to do this effectively:

after a couple of introductory courses,

they are left to rediscover (or reinvent) the rest of programming on their own.

The result?

Most spend far too much time wrestling with software,

instead of doing research,

but have no idea how reliable or efficient their programs are. [1]

[1] Wilson, Greg. "Software carpentry: getting scientists to write better code by making them more productive." Computing in science & engineering 8.6 (2006)

Software:

Source code, documentation, tests, executables and all other artefacts that are created by humans during the development process that are necessary to understand its purpose.

Research Software:

Foundational algorithms, the software itself, as well as scripts and computational workflows that were created during the research process or for a research purpose, across all domains of research.

(R)esearch (S)oftware (E)ngineer:

People who create or improve research software and/or the structures that the software interacts with in the computational environment of a research domain.

They are highly skilled team members who may also choose to conduct their own research as part of their role.

Some history

- 2010: Software Sustainability Institute founded
  by Universities of Edinburgh, Southhampton, Oxford and Manchester
- 2012: Collaboration's workshop in Oxford
  - Identified career problems for software developers in academia
  - Many people doing the same things with different job descriptions
    -> Need a common job name: Research Software Engineer
- 2016: First international RSE conference in Manchester
- 2018: German de-RSE e.V. is founded
- 2019: First international deRSE conference in Potsdam
- 2023: Gesellschaft für Informatik starts special interest group on RSE with de-RSE e.V.

Where do you get (trained) RSEs?

- Currently, mostly researchers who trained themselves for more technical role
- Integration of software engineers from outside academia often difficult
  - Competition from industry
  - Research software often not matching with their way of developing software

-> Skill gap! Relying on 'accidents' not sustainable
- Need for specialized training
  - What should be taught?
  - How much of that would be useful for domain researchers?
- Questions asked at community workshop in 2023
  - Who would benefit from RSEng?
  - What are essential topics?
  - Are there categories/specializations of RSEs?
  - How do we reach 'students'?
- Community workshop lead to teachingRSE Project
  - Paper: Foundational competencies and responsibilities of an RSE (arxiv:2311.11457)
  - Website: de-rse.org/learn-and-teach

Software/technical skills

- Adapting to the software life cycle
- Creating documented code building blocks
- Building distributable software
- Using software repositories
- Software behavious and analysis

Research skills

- Conducting and leading research
- Understanding the research cycle
- Software re-use
- Software publication and citation
- Using domain repositories

Communication skills

- Working in a team
- Teaching
- Project management
- Interaction with users and stakeholders

How much do you need to know?

Multiple dimensions
- Career level (also within domain)
- RSE specializations
- Structure of your project team/organization

Specializations
- ${DOMAIN}-RSE
- Open Science RSE
- HPC RSE
- Training/Teaching RSE
- Project/community manager RSE
- ...

Want to participate?

- In RSE Community in general
  - Look for a local chapter
  - Mailing list, matrix chat
  - Come to the next conference(s)
    - RSECon24 in Newcastle, Sep 03-05 (in person and online)
    - US-RSE'24 in Albuquerque, Oct 15-17
    - deRSE 2025 likely in Karlsruhe, co-located with SE25 of Gesellschaft für Informatik
- TeachingRSE Project
  - GitHub org: https://github.com/the-teachingRSE-project/
  - Weekly work meetings, Mondays @10 A.M. (talk/write to me)

1. Introduction

2. Competencies of an RSE

3. How WIAS handles research software
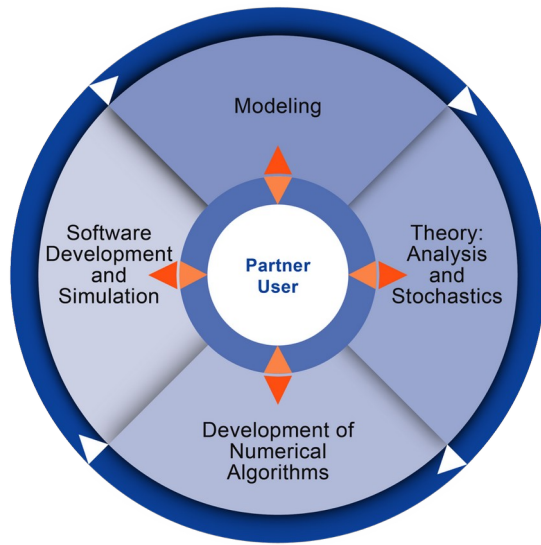
- WIAS research: solve complex mathematical problems in science and industry
- Modeling, analysis, numerical methods, research software development, simulation
- Numerical computations and research software create research data used to evaluate models and algorithms

Research Software – from Matlab scripts to dedicated simulation code ubiquitous in
- Knowledge driven research
- Application driven research
- Transfer of results to **industry** and **other fields of science**

A Research Story

1. Ph.D. student X develops software S v1.0 and produces data D
2. PostDoc Y finds out that they can also use S
3. X and Y collaborate, resulting in S v2.0, a new paper is published using S
4. New results and S help to raise third-party funded project
5. X leaves institute after Ph.D., used 25% of their time for S
6. Y leaves for University of P, would like to continue with S

How to ensure the work on software and resulting data is not lost?

Make S and D findable and citeable

- X and Y can cite S in their CVs.
- Users of S can provide proper reference and attribution

Make S and D accessible and reusable

- Development of S continues as cooperation with University of P
- Other researchers can build upon and improve S and D
- Potential for new research questions and cooperations for X and Y

**Research Software and data are part of the scientific achievements
of the institute AND of the individual scientists**

- Research software and data are results of scientific work
- Recognition as scientific achievements is necessary
- **Good Scientific Practice demands proper handling of software and data**
- Reward increased effort of researchers by increased visibility and efficiency

$F_{indable}$ $A_{ccessible}$ $I_{nteroperable}$ $R_{eusable}$
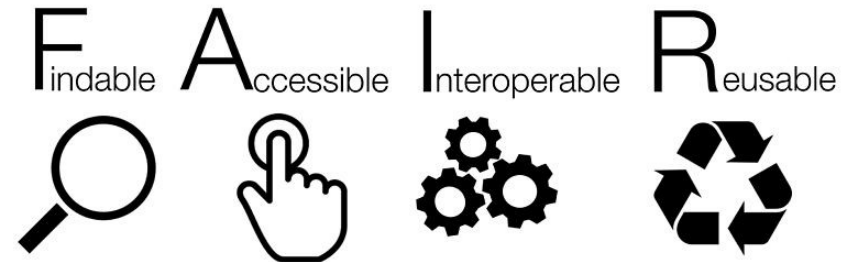
- Metadata, easy to find, retrievable by standard protocols
- Interaction through metadata, APIs
- Useable (can be executed) or reusable (can be incorporated into other projects)

Classification of research software
DLR Software guidelines

Application Class 0

- Personal use, code accompanying publications

Application Class 1 (Software by WIAS researchers)

- Use and/or further development by others, „small" size
- @WIAS: listed along with preprints

Application Class 2

- Use and/or further development by others, „medium or large" size
- Long term development, R&D contracts

Application Class 3

- „Mission critical" to group or institute, obligations to third parties
- „product level", commercial licensing, spin-offs

WIAS Software™ (AC 2&3)

- Listed on homepage and in annual report
- Dedicated GitHub organization github.com/WIAS-BERLIN for open source software
- swMATH listings

DLR Guidelines: https://zenodo.org/records/1344612

Support for scientists

Infrastructure

- Classical IT services (workplace equipment, compute server, mail, etc.)
- GitLab with continuous integration services
- RDMO: research data management organizer for data management plans, etc.
- Reposis: Presentation and archiving of publications, software and data

Training and qualification

- WIAS Software & Data Seminar
- Coffee Lectures (~10 minutes) on interesting tools, developments, etc.
- Communities on relevant topics (FEniCS, Julia, ML)
- Featured events at WIAS days and Leibniz-MMS network.

Support

- Research Software Engineer and Steward (me)
- Commission on Software and Data
- Experienced developers