ZDIN
ZUKUNFTSLABOR ENERGIE
Zentrum für digitale Innovationen Niedersachsen

RESEARCH SOFTWARE ENGINEERING –

HOW DO WE TEACH THE NECESSARY SKILLS TO INTERDISCIPLINARY TEAMS?

Experiences 2020 - 2024

Stephan Ferenz, Astrid Nieße

Carl von Ossietzky Universität Oldenburg

MOTIVATION – ZUKUNFTSLABOR ENERGIE

ZDIN ZUKUNFTSLABOR ENERGIE
Zentrum für digitale Innovationen Niedersachsen

Speaker: Prof. Dr.-Ing. Astrid Nieße (Computer Science)

Coordination: Laura Niemann

Involved expertise: Power Systems Engineering, Electrical Engineering, Economics, …

Carl von Ossietzky Universität Oldenburg

Prof. Dr. Astrid Nieße
Department of Computing Science
Digitalized Energy Systems

DLR Deutsches Zentrum für Luft- und Raumfahrt
Institut für Vernetzte Energiesysteme

Prof. Dr. Carsten Agert
Institute of Networked Energy Systems

University of Applied Sciences HOCHSCHULE EMDEN·LEER

Prof. Dr. Johannes Rolink
Faculty of Engineering, Department Electrical Engineering and Computer Science

Leibniz Universität Hannover

Prof. Dr. Michael H. Breitner
Institute of Information Systems Research

OFFIS

Prof. Dr. Sebastian Lehnhoff
R&D Division Energy

Ostfalia Hochschule für angewandte Wissenschaften

Prof. Dr. Lars Kühl
Institute for Energy Optimized Systems

Technische Universität Braunschweig

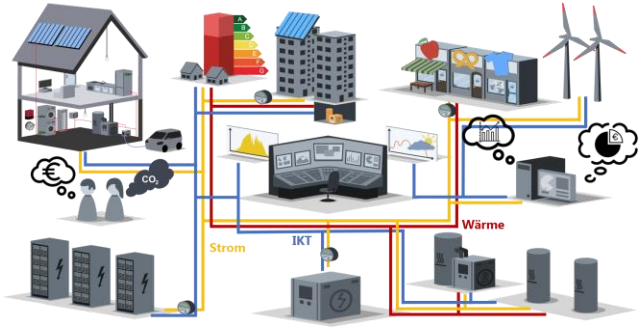
Prof. Dr. Bernd Engel
elenia Institute for High Voltage Technology and Power Systems

ZUKUNFTSLABOR ENERGIE

Main Objectives of ZLE

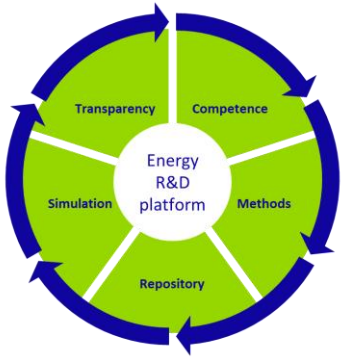
Investigation and development of digitized energy systems

- Investigation of neighborhood scenarios
- Modeling of user interactions
- Investigation of (cyber) resilience



Digitization of energy system research and development

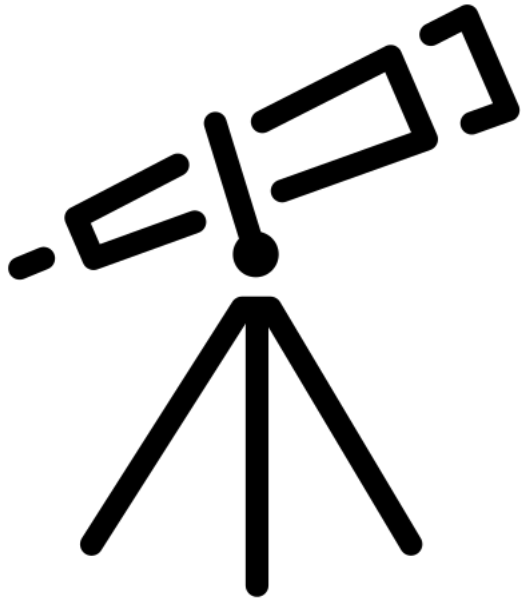
- Development of a collaborative R&D platform for digitalized energy systems
- Networking Energy System Research
- Strengthening the transfer of practical experience

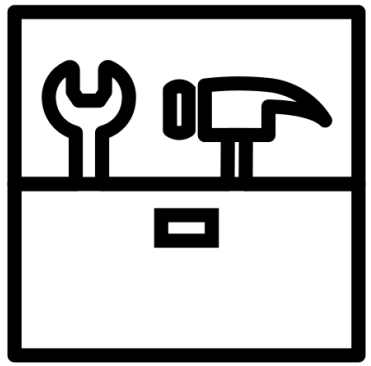


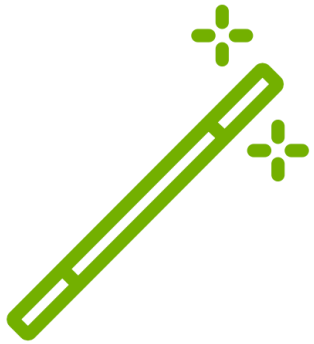
Development of simulation models and research prototypes is crucial for the overall goals of the project.

TYPICAL SETUP - GENERALIZED

Our daily work







Images by openmoji.org - CC-BY 4.0

HOW DO WE TEACH THE NECESSARY SKILLS TO INTERDISCIPLINARY TEAMS?

Integrated research software engineering is crucial for the success of the overall goals within the project!

1. Inspire the group for open software development!

➡ Discuss principles of OSS and setup an *open science declaration*.

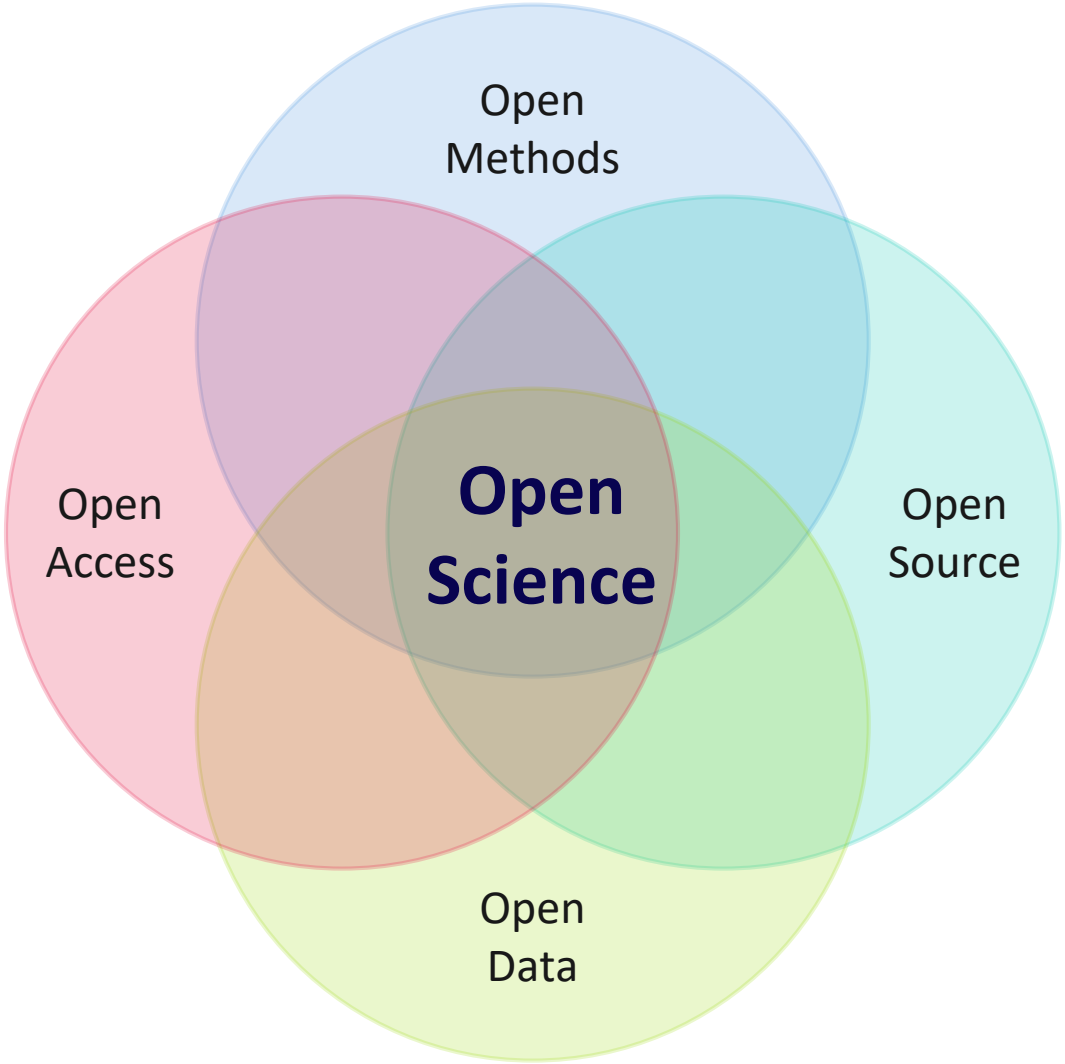
2. Teach the group the fundamental skills within short time!

➡ Setup a workshop series for the *fundamental skills*.

ZLE OPEN SCIENCE DECLARATION

Open Methods - Exchange and sharing of methodologies to remove cognitive biases.

Open Access - Digital exchange platforms disseminate research methods and results for accessibility and transparency.



Open Source - Developed research software is available for inspection, use and modification.

Open Data - The data gathered and generated is publicly available (*). It follows the FAIR principles.

WORKSHOP SERIES

Fundamental skills for joint research software development

ZDIN ZUKUNFTSLABOR ENERGIE
Zentrum für digitale Innovationen Niedersachsen

Identify relevant skills

- Survey on research group managers within the area of interdisciplinary energy system research
- Group managers with a background in computer science / software engineering
- Focus on their experiences with interdisciplinary teams

Identified topics with sub-topics

- From requirements to software architecture
  - Use case modeling (UML)
  - Software architecture
- Open Source
- Understandable code
  - Code Quality
  - Code Review
- Version control
  - Introduction Git
  - Branches
  - Research Data Management
- Testing
  - Testing of research software
  - Pipeline
  - Unit tests

WORKSHOP CONCEPT

For all topics: Some general rules for the responsible seniors

Format

- English slides
- Workshop-character
- Ca. 10 persons
- Online
- Can be interactive as well

Content

- Flexible within topic
- Expect little prior knowledge in computer science
- Think about what would you want your non computer scientists projects partners to know?
- If possible, please prefer open content – so it is easy to publish the slides open access later on
- Python as programming language is known

Time

- 3-4 hours per Workshop
- E.g.: 3x 1h presentation + 2x 15 min breaks or 2x 1,5h presentation + 1x 15-30 min break
- All workshops were conducted separately over a year

MAIN FINDINGS AND OPEN ISSUES (1/2)

Your ideas are very welcome

(1) Junior researchers were very interested in the beginning, but …

(2) After about a year, most of the attendees did not apply the skills trained in the workshop series.

(3) Group heavily relied on more skilled colleagues.

We did not see the knowledge transfer as expected.

MAIN FINDINGS AND OPEN ISSUES (1/2)

Your ideas are very welcome

(1) Junior researchers were very interested in the beginning, but …

(2) After about a year, most of the attendees did not apply the skills trained in the workshop series.

(3) Group heavily relied on more skilled colleagues.

We did not see the knowledge transfer as expected.

- Was the program to short?
- Would an ongoing workshop / tutorial concept help? *(how to staff?)*
- Should we differentiate research software types in training as well?