3rd NFDI4Energy Conference: Defining Energy Research Data Management Together
Proceedings to the 3rd NFDI4Energy Conference - Full Papers
https://doi.org/10.52825/ocp.v9i.3279

Published: 23 Mar. 2026

Enabling FAIR Energy Research Software

Stephan Ferenz[1,2,*], Oliver Werth[2], and Astrid Nieße[1,2]

[1]Carl von Ossietzky Universität Oldenburg, Oldenburg, Germany
[2]OFFIS - Institute for Information Technology, Oldenburg, Germany
*Correspondence: Stephan Ferenz, stephan.ferenz@uol.de

Abstract. Energy Research Software (ERS) is a central cornerstone to facilitate energy research, e.g., by enabling data analytics or simulations of energy systems. However, most ERS does not comply with the FAIR principles (Findable, Accessible, Interoperable, Reusable (FAIR)) and, therefore, is often not reused. To enable FAIR ERS different domain-specific standards are required, e.g., for metadata. Additionally, services can support researchers to make their ERS FAIR. A good understanding of possible helpful and required standards and services is an important first step towards their creation. Therefore, we provide a general definition of ERS and present a vision which services and standards are needed to enable FAIR ERS. We analyze the current state of these services and standards in the energy domain and provide an outlook to further developments.

Keywords: Energy Research Software, Software Management Plans, Software Journals, Research Software Engineering

1. Introduction

In energy research, self-designed research software is a fundamental tool for multiple purposes like the visualization of processes and values, for example, of power quality [1], the simulation of specific devices like fuel cells [2], the (co-)simulation of smart grids [3], or the analysis of transition pathways [4]. Thereby, Energy Research Software (ERS) contributes significantly to energy research and serves as the foundation for most research outcomes in this domain. Thus, the quality of ERS directly influences the quality of results in energy research and its effectivity.

In general, research software faces multiple challenges. For example, missing documentation of research software (e.g., with respect how to use the software, relevant assumptions) often hinders the reproducibility of research which is one of the key elements of science [5]. Also, it is often not possible to build on existing research software, e.g., because the right research software to build on can not be found or is not open source. Additionally, building on existing research software often contains the risk of including unmaintained software [6]. Overall, these issues have led to parallel developments of similar tools, e.g., 63 frameworks exist for optimization-based energy system modeling having highly overlapping features [7]. Often new tools are developed without reusing existing ones. Thereby, the research process is slowed down because a lot of time is spent on (re)developing research software instead of doing research [8].

In 2022, Chue Hong et al. [9] introduced the Findable, Accessible, Interoperable, Reusable for Research Software (FAIR4RS) principles to address these challenges. They adapted the FAIR principles from Wilkinson et al. [10] which mainly focus on research data. The FAIR4RS principles provide high-level guidance on how research software should be treated to allow high reusability and to follow good scientific practice. Each principle contains multiple subprinciples including diverse aspects such as metadata, interfaces, registering software, and versioning.

Generally, the adoption of the FAIR principles is a complex process [11]. In case of FAIR4RS, a lot of supporting resources are still missing, e.g., services to provide Persistent Identifiers (PIDs), software registries, metrics, and others [12]. While some of these required resources are non-domain specific, also domain-specific efforts are also needed, e.g., for metadata standards [9].

Applying the FAIR4RS principles to ERS can lead to better reusability and quality of ERS and, therefore, better energy research. However, the adoption of the FAIR4RS principles in the energy domain is currently at the beginning and only little resources to support them have been developed so far, e.g., the framework and model factsheets of the Open Energy Platform (OEP)[1]. Hence, the focus of this work is on the next relevant steps to enable the adoption of the FAIR4RS principles in energy research. This leads to our research question: What resources are required to enable FAIR ERS?

With this paper, we contribute to this field as follows:

- We provide a general definition of ERS (section 2).
- We present a vision how ERS can be treated FAIRly in a typical research project cycle (section 3) based on the definition, literature, and existing approaches from other domains. This vision includes multiple resources which can support FAIR ERS.
- We analyze the current state of maturity and availability of the outlined resources in the energy domain (section 4).

We conclude this paper with an outlook on further required research on ERS in section 5.

2. Defining Energy Research Software

Energy Research Software (ERS) is the central object of this paper and, therefore, we provide a precise definition in this section. The term combines two concepts that are first clarified separately: "research software" and "energy research".

The definition of the term *research software* has been the subject of an intensive debate. Gruenpeter et al. summarized this debate and proposed an exclusive definition that focuses on all software created for or during scientific investigations:

> "Research software includes source code files, algorithms, scripts, computational workflows and other executables that were created during the research process or for a research purpose." [13]

While *energy research* is an often used term, it is less well defined in the literature [14]. It is generally understood to comprise investigations on technologies for energy supply and/or end use [14]. This includes especially technologies for energy systems, like conversion and transmission of energy. Nowadays, energy research is often defined in a more mission-oriented way, aiming at sustainability, net-zero emissions, and/or the energy transition, for example in the 8th German Energy Research Program [15].

By bringing these components together, we define ERS as:

[1] https://openenergyplatform.org/factsheets/models/add/ and https://openenergyplatform.org/factsheets/frameworks/add/, last access 2025-11-27

Definition 1 *Energy Research Software (ERS) is software that was created during energy research and/or specially to support energy research. Energy research aims to understand, analyze, improve, and/or design energy systems or components specifically for energy systems.*

As general research software, ERS spans a wide spectrum of technical complexity. At the simplest end are single-purpose scripts and small tools. More elaborate projects can consist of libraries, modular frameworks, or full-scale software suites that integrate many different components and which might come with user interfaces.

Functionally-wise, ERS can visualize measurements or simulation results, analyze data from laboratory experiments or field deployments, generate synthetic data for testing or scenario analysis. Also, ERS can implement mathematical models of individual energy components (such as heat pumps, batteries, or converters) or of whole energy system configurations. It can support the analysis of specific designs, e.g., including design optimizations or control strategies which may also be implemented as prototypes. Additionally, ERS can act as a framework that indirectly supports energy research. Regarding other software, ERS can manage other ERS and/or enable orchestrating of other ERS, like co-simulation frameworks. Regarding data, ERS can facilitate the organization and management of research data in energy research. Regarding hardware, ERS can enable hardware-in-the-loop experiments.

By fulfilling any of these purposes, ERS enables energy researchers to analyze current energy systems or its components and to explore new solutions for energy systems, thereby constituting an essential component of modern energy research.

3. Vision for FAIRly Treating Energy Research Software

This section focuses on what can enable and support FAIR ERS. To make their ERS FAIR, researchers must treat their ERS differently than today at many stages of a typical research project. This new way of handling ERS can be supported by a set of dedicated resources like standards, recommendations, and services. By simplifying the FAIR handling of ERS, those resources become key enablers for FAIR ERS. In order to identify the required resources, this section presents a vision of how ERS could be managed in a FAIR way in future projects. Thereby, we extend existing work on research software cycles in energy research like in [8].

The vision aligns the FAIR4RS principles with a typical research project cycle, illustrating where each resource can be applied. The typical cycle of a research project is divided into five distinct phases: (1) starting with the analysis of existing research software, (2) planning for new software developments, (3) actual software development, (4) activities to make software findable, and, finally, (5) ensuring the software fulfills FAIR4RS principles to allow reuse. Each phase poses unique requirements and challenges that should be addressed while adhering to the FAIR4RS principles. Therefore, relevant resources are proposed to support the different activities in each phase. While the vision targets ERS, it includes examples from other domains to showcase how it could be for energy research. The overall cycle is shown in Figure 1.

In the first phase (1), researchers will start with searching for relevant ERS by using a domain-specific **software registry**. Software registries allow software discoverability and improve research transparency [16]. They only include metadata while software repositories also store the source code [16]. The FAIR4RS principles recommend software to be registered in a software registry [9]. An example for an established software registry is bio.tools[2] [17] in the life science domain.

[2] https://bio.tools/, last access 2025-11-27

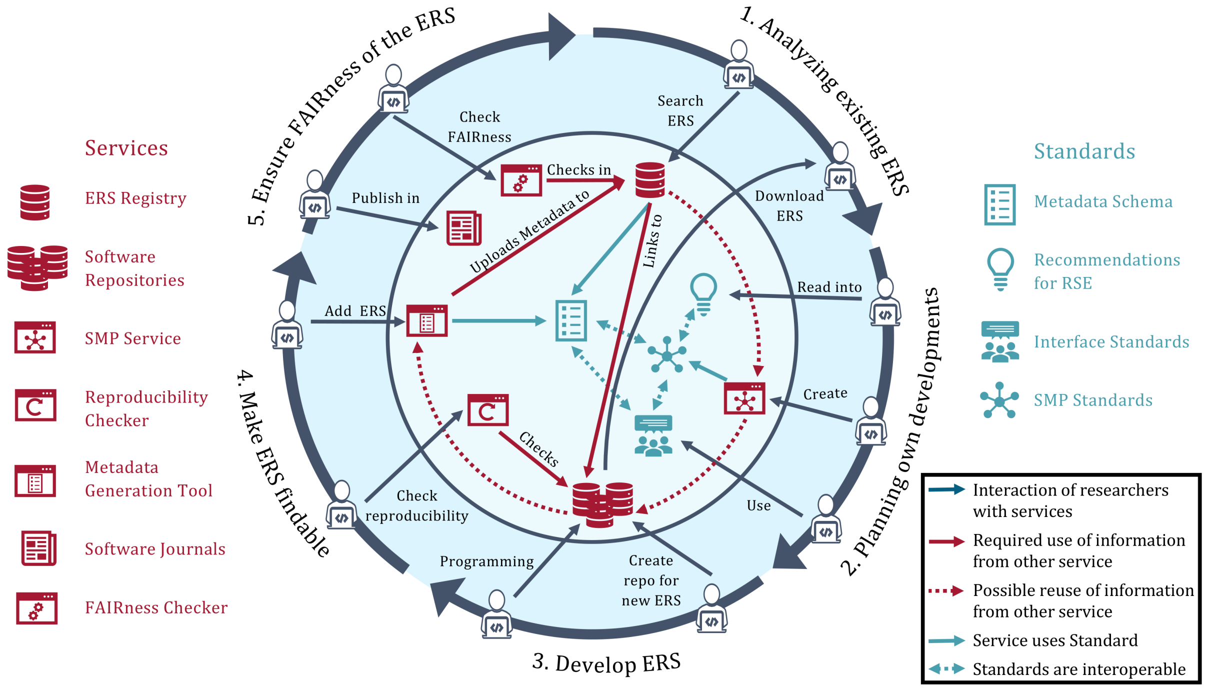

Figure 1. *Vision for a research cycle enabling FAIR ERS*

The software registry includes metadata about different ERS like its title or its license. Providing metadata is an important key to enable good findability and reusability [9]. The metadata should follow a **metadata schema** which standardizes the properties in the metadata and makes them easier to be compared and used by machines. The metadata schema should be a domain-relevant community standard [9] and should be interoperable with other schemas [16]. For example, bio.tools uses the specific schema *biotoolsschema* [18]. When the researchers find a fitting software in the registry, the registry links to the software repository (e.g., on GitHub). The researchers can download the software and try it.

Often, the existing software will not contain all functionalities needed by the researchers for their own research. If that is the case, the researchers will plan some additional development in the second phase (2). Since research software engineers are not commonly established in research institutions the researchers are often responsible for the software development. Since they are often not trained in Research Software Engineering (RSE) [19], [20], they should read **recommendations** summarizing the most important aspects on RSE in their domain. The recommendations are an important instrument to successfully integrate the FAIR4RS principles into the routine practice of software development and use in research projects. Examples for relevant recommendations papers already exist in life science, e.g., by List et al. [21] who recommend how to deal with challenges like reproducibility, open-source development, and other.

Based on the recommendations, the researchers will plan their own development. Therefore, they will first write a Software Management Plan (SMP) which helps to plan and structure the development [22]. The SMP is based on a domain-specific **SMP standard**. The researchers will use a specific service supporting them in writing the SMP, a **SMP service**. The service will allow some automation, as proposed by Alves et al. [22], e.g., by reusing software metadata as a starting point. Therefore, the SMP should be based on the same metadata schema allowing high interoperability. When planning their new software, the researchers will also plan the usage of domain-specific **interface standards**, especially for data exchange. These standards ensure interoperability with other ERS as emphasized by the first interoperability principles of FAIR4RS [9].

In the development phase (3), the researchers will first create their own **repository** or decide to further use an existing one, e.g., when adding functionalities to an existing software. To enable version control for their development, the researchers will use git, e.g., by creating or using a repository on GitHub or GitLab. Ideally, the SMP service will support the creation of the new repository and will allow to reuse the information from the SMP. By sharing their developments early, the researchers will allow others to collaborate. Afterwards, they will develop their software to their needs.

In the next phase (4), the researchers want to publish their work after finalizing their research. Therefore, they will first check the reproducibility of their research. For this step, they will use a tool which simplifies and automates checking the reproducibility, a **reproducibility checker**.

Also, the researchers want to make their developed software findable. Therefore, they will first create relevant comprehensive and standardized metadata to add the software to the registry. To simplify this process, they will be supported by a **metadata generation tool** which extracts as many information as possible from the software repository and other available sources. The tool should ensure that the metadata follow the metadata schema. It should also allow the researchers to review the metadata and to add additional information. The tool should also be able to directly upload the metadata to a software registry and to the repository of the software.

In the last phase (5), the researchers want to increase the visibility and FAIRness of the software. Therefore, the researchers will publish their software in a **software journal**. Software journals offer a legitimate publication medium to ensure peer-review of the software components [23]. The software is normally presented in short articles which can be properly cited [23]. An example for a software journal is the Journal of Open Source Software (JOSS) [23].

Finally, to see if further improvements to the FAIRness can be made, the researchers will check the FAIRness of their software with a **FAIRness checker**. The service should work with the link to the entry in the software registry and should verify to which extend the software fulfills the FAIR4RS principles. A first approach for such a service are FAIRsoft which works on metadata of a software [24] and FAIRSECO [25].

For new research projects, the researchers will start the cycle again. Most steps stay relevant in every iteration, while some steps like reading the recommendations get less relevant.

Overall, four domain-specific standards and recommendations are identified that can significantly contribute towards making the ERS FAIR: a metadata standard, RSE recommendations, interface standards, and a SMP standard. Additionally, seven services were highlighted which can support the adaption of the FAIR4RS principles for ERS.

4. Current State of Services and Standards for FAIR Energy Research Software

In this section, we provide a comprehensive overview of the current state of the proposed resources that support FAIR research software in the energy domain. We conduct a thorough analysis of existing initiatives and approaches, highlighting their strengths and weaknesses. A summary is presented in Table 1.

4.1 Standards

With respect to the proposed standards, there is a need for domain-specific approaches that cater to the unique requirements of the energy domain. Currently, the energy domain lacks solutions to all four aspects which are recognized as standard by enough stakeholders, highlighting a

Table 1. *Overview on the current state of the envisioned services and standards in the energy domain including the need for domain-specific solutions*

		Domain-specific solution required?	Current Status	Next Steps
Standards	Metadata Schema	Yes	Available approach	Usage in tools, standardization
	RSE Recommendations	Yes	Available approach	Standardization through usage
	Interface Standards	Yes	Unsystematic approaches	Systematization, derivation of common standards
	SMP Standard	Yes	Available approach	Standardization through usage
Services	ERS Registry	Yes	First approaches	Integration of a metadata schema
	Software Repositories	No	Available services	-
	SMP Service	Partly	Available service	Implementation of more features
	Reproducibility Checker	Unclear	Open Topic	First approaches
	Metadata Generation Tool	Partly	Existing software	Connection to a registry, more extraction sources
	Software Journals	No	Available service	-
	FAIRness Checker	Unclear	Existing software	More harmonization of quality indicators, integration with registries

significant gap in the existing infrastructure. However, there are several initiatives and approaches being developed to address this shortfall, offering promising prospects for the future.

Regarding the metadata schema, Ferenz et al. introduced *ERSmeta*, a formalized schema developed based on a rigorous requirement analysis [26]. Although *ERSmeta* is a promising approach, it currently lacks the status of a standard and is not yet widely adopted. Moreover, there are no tools that utilize this metadata schema, highlighting the need for further development and implementation to establish it as a widely accepted standard.

Regarding the recommendations, Ferenz et al. published a set of recommendations for developing FAIR ERS [27]. These recommendations are available on GitHub for further discussions and refinement through the community. While these recommendations present a good first approach, they can only achieve the status of a standard through usage and adaptation over time.

While there are some general approaches to interface standards, such as the FMI standard[3], there is currently a lack of standardized interfaces in the energy research domain. However, some frameworks, such as oemof[4] and PyPSA [28], have developed clear interfaces that are used by their larger user community. But, these interfaces are not yet standardized on a larger scope. A practical way to describe these interfaces may be derived from the work of Kuckertz et al. [29]. Still, a large effort is required to collect and systemize the existing interfaces in order to derive and promote interfaces that can be used for multiple frameworks and in multiple ERS.

Regarding SMP templates, NFDI4Energy is currently developing a SMP template that will be available in their RDMO instance[5]. This template will provide a standardized approach to creating SMPs for ERS.

[3] https://fmi-standard.org/, last access 2025-11-27
[4] https://oemof.org/, last access 2025-11-27
[5] https://rdmo.nfdi4energy.org/projects/, last access 2025-11-27

4.2 Services

Some services do not require a domain-specific adaptation to energy research, such as software repositories. When examining the approaches, we also differentiate between the provided service and the underlying software solution, as the software may already be developed but not yet available as a service.

There are several approaches for ERS registries, including the OEP model and framework factsheets[1], the NFDI4Energy community in nfdi.software[6], and the openmod-tracker[7]. However, these approaches currently lack the use of a formalized metadata schema, such as *ERSmeta*, which is necessary to achieve FAIR ERS.

Most researchers in the energy domain already utilize general-purpose software repositories, such as GitHub[8] or GitLab[9], often hosted on-premise. While there may be some limited advantages to domain-specific solutions, such as improved metadata representation, these benefits do not outweigh the effort required to develop and maintain domain-specific repositories.

Regarding the service for SMPs, the tool RDMO [30] already supports SMPs. However, its current limitations include the lack of support for metadata exchange with registries and repositories and the lack of support for machine-actionable SMPs. The later is currently being targeted by DMP4NFDI [31]. While the service needs to be able to include domain-specific templates, the required efforts for the SMP service are mainly non-domain specific. Nevertheless, further development of the tooling is still needed to simplify this process for researchers.

Regarding the reproducibility checker, existing approaches primarily rely on human intervention to perform the reproducibility check, such as CODECHECK [32] or the use of notebooks in R or Python to simplify the testing of reproducibility.

Regarding the creation of metadata, two tools currently support the combination of extracting metadata and allowing curation: the Auto CodeMeta Generator[10] [33] and SMECS [34]. Notably, SMECS includes the energy-specific metadata schema, *ERSmeta*. In contrast to the Auto CodeMeta Generator, SMECS is currently not available as an online service. However, both tools lack the ability to directly push the created metadata to a software registry. Furthermore, the amount of extracting sources can be further increased in both tools. SMECS is further developed in the project ConnOSS [35].

There are already multiple software journals that contain ERS, such as the Journal of Open Source Software (JOSS)[11], SoftwareX[12], and the Journal of Open Research Software (JORS)[13]. While domain-specific software journals may offer some advantages, such as better fitting reviewers and improved findability, the current journals are already adopted in energy research and provide suitable reviewing mechanisms for ERS.

Regarding the FAIRness checkers for research software, FAIRsoft [24] is a pioneering approach that can check the metadata of a research software against the FAIR4RS principles to a certain extent. However, it is currently only available locally. The discussion of suitable (and possible automated) metrics for the FAIR4RS principles is still ongoing and it remains unclear whether a FAIRness Checker requires domain-specific components.

[6] https://nfdi.software/communities/nfdi4energy/software, last access 2025-11-27
[7] https://openmod-tracker.org/, last access 2025-11-27
[8] https://github.com/, last access 2025-11-27
[9] https://about.gitlab.com/, last access 2025-11-27
[10] https://autocodemeta.linkeddata.es/, last access 2025-11-27
[11] https://joss.theoj.org/, last access 2025-11-27
[12] https://www.sciencedirect.com/journal/softwarex, last access 2025-11-27
[13] https://openresearchsoftware.metajnl.com/, last access 2025-11-27

5. Summary and Outlook

Within this paper, we have provided a general definition of ERS which is a crucial step towards creating a nomenclature as well as an understanding about the needs and requirements regarding research software in the energy domain. While this definition is already helpful, a more fine-grained categorization of ERS may be required to better specify the needs and the required services. The multi-dimensional categorization of research software by Hasselbring et al. [36] can be a good starting point, as it differentiates between research infrastructure software, technology research software, and modeling, simulation, and data analytics. A categorization can be particularly useful for the recommendations and the interface standards which would benefit from a more detailed and nuanced understanding of the different types of ERS.

As we outlined in this paper, there are still many challenges to be addressed in terms of providing standards and services that support FAIR ERS. While progress has been made in this area, there is still much work to be done, particularly in terms of developing community-based standards. We believe that providing these resources is only one side of the equation and that general awareness and education about research software (engineering) in the energy domain, as well as about FAIR4RS, are also crucial for their adoption.

In addition, we also see the support from research institutions and funding agencies as essential for the successful adoption of the FAIR4RS principles in the energy domain. Research institutions should provide guidance and support for researchers developing and using FAIR ERS. Funding agencies should prioritize funding for FAIR ERS and require SMPs for research projects.

In conclusion, the development of standards and services for FAIR ERS is an ongoing process that needs the sustained efforts of researchers, research software engineers, and additional stakeholders in energy research. We hope that this paper will make a meaningful contribution to this effort and serve as a foundation for future work in this area.

Data availability statement

There is no data supporting this article.

Author contributions

Stephan Ferenz: Conceptualization, Writing - Original Draft, Visualization
Oliver Werth, Astrid Nieße: Conceptualization, Writing - Review & Editing

Competing interests

The authors declare that they have no competing interests.

Funding

The authors would like to thank the German Federal Government, the German State Governments, and the Joint Science Conference (GWK) for their funding and support as part of the NFDI4Energy consortium. The work was partially funded by the German Research Foundation (DFG) – 501865131 within the German National Research Data Infrastructure (NFDI, www.nfdi.de).

References

[1] A. J. Christe, S. I. Negrashov, P. M. Johnson, D. Nakahodo, D. Badke, and D. Aghalarpour, "OPQ Version 2: An Architecture for Distributed, Real-Time, High Performance Power Data Acquisition, Analysis, and Visualization", in *2017 IEEE 7th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*, Jul. 2017, pp. 1415–1419. DOI: 10.1109/CYBER.2017.8446111.

[2] W. Wu, J. S. Partridge, and R. W. G. Bucknall, "Simulation of a stabilised control strategy for PEM fuel cell and supercapacitor hybrid propulsion system for a city bus", *International Journal of Hydrogen Energy*, vol. 43, no. 42, pp. 19 763–19 777, Oct. 2018, ISSN: 0360-3199. DOI: 10.1016/j.ijhydene.2018.09.004.

[3] C. Steinbrink et al., "CPES Testing with mosaik: Co-Simulation Planning, Execution and Analysis", *Applied Sciences*, vol. 9, no. 5, p. 923, Jan. 2019. DOI: 10.3390/app9050923.

[4] K. Löffler, K. Hainsch, T. Burandt, P.-Y. Oei, C. Kemfert, and C. Von Hirschhausen, "Designing a Model for the Global Energy System—GENeSYS-MOD: An Application of the Open-Source Energy Modeling System (OSeMOSYS)", *Energies*, vol. 10, no. 10, p. 1468, Oct. 2017. DOI: 10.3390/en10101468.

[5] W. Hasselbring, L. Carr, S. Hettrick, H. Packer, and T. Tiropanis, "From FAIR research data toward FAIR and open research software", *it - Information Technology*, vol. 62, no. 1, pp. 39–47, Feb. 2020, ISSN: 1611-2776, 2196-7032. DOI: 10.1515/itit-2019-0040.

[6] H. Anzt et al., "An environment for sustainable research software in Germany and beyond: Current state, open challenges, and call for action", *F1000Research*, vol. 9, p. 295, Jan. 2021, ISSN: 2046-1402. DOI: 10.12688/f1000research.23224.2.

[7] M. Hoffmann et al., "A review of mixed-integer linear formulations for framework-based energy system models", *Advances in Applied Energy*, vol. 16, p. 100 190, Dec. 2024, ISSN: 2666-7924. DOI: 10.1016/j.adapen.2024.100190.

[8] S. Ferenz and A. Nieße, "Towards Improved Findability of Energy Research Software by Introducing a Metadata-based Registry", *ing.grid*, vol. 1, no. 2, Nov. 2023, ISSN: 2941-1300. DOI: 10.48694/inggrid.3837.

[9] N. P. Chue Hong et al., "FAIR Principles for Research Software (FAIR4RS Principles)", FAIR4RS WG, Tech. Rep., May 2022, p. 32. DOI: 10.15497/RDA00068.

[10] M. D. Wilkinson et al., "The FAIR Guiding Principles for scientific data management and stewardship", *Scientific Data*, vol. 3, no. 1, pp. 1–9, Mar. 2016, ISSN: 2052-4463. DOI: 10.1038/sdata.2016.18.

[11] F. Belliard, A. M. Maineri, E. Plomp, A. F. R. Padilla, J. Sun, and M. Z. Jeddi, "Ten simple rules for starting FAIR discussions in your community", *PLOS Computational Biology*, vol. 19, no. 12, e1011668, Dec. 2023, ISSN: 1553-7358. DOI: 10.1371/journal.pcbi.1011668.

[12] M. Barker, "FAIR4RS Roadmap Report", Tech. Rep., Feb. 2022. DOI: 10.5281/zenodo.6239373.

[13] M. Gruenpeter et al., "Defining Research Software: A controversial discussion", Zenodo, Tech. Rep., Sep. 2021. DOI: 10.5281/zenodo.5504016.

[14] P. J. Runci and J. J. Dooley, "Research and Development Trends for Energy", in *Encyclopedia of Energy*, C. J. Cleveland, Ed., New York: Elsevier, Jan. 2004, pp. 443–449, ISBN: 978-0-12-176480-7. DOI: 10.1016/B0-12-176480-X/00477-0.

[15] Bundesministerium für Wirtschaft und Klimaschutz (BMWK), "8. Energieforschungsprogramm zur angewandten Energieforschung – Forschungsmissionen für die Energiewende", Tech. Rep., Oct. 2023. Accessed: Sep. 5, 2025. [Online]. Available: https://www.bundeswirtschaftsministerium.de/Redaktion/DE/Publikationen/Energie/8-energieforschungsprogramm-zur-angewandten-energieforschung.html.

[16] D. Garijo et al., "Nine best practices for research software registries and repositories", *PeerJ Computer Science*, vol. 8, e1023, Aug. 2022, ISSN: 2376-5992. DOI: 10.7717/peerj-cs.1023.

[17] J. Ison et al., "The bio.tools registry of software tools and data resources for the life sciences", *Genome Biology*, vol. 20, no. 1, p. 164, Aug. 2019, ISSN: 1474-760X. DOI: 10.1186/s13059-019-1772-6.

[18] J. Ison et al., "BiotoolsSchema: A formalized schema for bioinformatics software description", *GigaScience*, vol. 10, no. 1, 2021, ISSN: 2047-217X. DOI: 10.1093/gigascience/giaa157.

[19] A. Johanson and W. Hasselbring, "Software Engineering for Computational Science: Past, Present, Future", *Computing in Science & Engineering*, vol. 20, no. 2, pp. 90–109, Mar. 2018, ISSN: 1558-366X. DOI: 10.1109/MCSE.2018.021651343.

[20] S. Druskat, M. Felderer, and C. Haupt, "Software engineering, research software and requirements engineering", in *Softwaretechnik-Trends Band 44, Heft 1*, Gesellschaft für Informatik e.V., 2024. Accessed: Jan. 2, 2025. [Online]. Available: https://dl.gi.de/handle/20.500.12116/44043.

[21] M. List, P. Ebert, and F. Albrecht, "Ten Simple Rules for Developing Usable Software in Computational Biology", *PLOS Computational Biology*, vol. 13, no. 1, e1005265, Jan. 2017, ISSN: 1553-7358. DOI: 10.1371/journal.pcbi.1005265.

[22] R. Alves et al., *ELIXIR Software Management Plan for Life Sciences*, Oct. 2021. DOI: 10.37044/osf.io/k8znb.

[23] A. Smith et al., "Journal of Open Source Software (JOSS): Design and first-year review", *PeerJ Computer Science*, vol. 2018, no. 2, 2018. DOI: 10.7717/peerj-cs.147.

[24] E. Martín del Pico, J. L. Gelpí, and S. Capella-Gutierrez, "FAIRsoft—a practical implementation of FAIR principles for research software", *Bioinformatics*, vol. 40, no. 8, btae464, Jul. 2024, ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btae464.

[25] Deekshitha, S. Farshidi, J. Maassen, R. Bakhshi, R. Van Nieuwpoort, and S. Jansen, "FAIRSECO: An Extensible Framework for Impact Measurement of Research Software", in *2023 IEEE 19th International Conference on E-Science (e-Science)*, Limassol, Cyprus: IEEE, Oct. 2023, pp. 1–10, ISBN: 979-8-3503-2223-1. DOI: 10.1109/e-Science58273.2023.10254664.

[26] S. Ferenz, O. Werth, and A. Nieße, "Requirements for a Metadata Scheme to Enable FAIR Energy Research Software", *Electronic Communications of the EASST*, vol. 83, Feb. 2025, ISSN: 1863-2122. DOI: 10.14279/eceasst.v83.2594.

[27] S. Ferenz et al., "Ten Recommendations for Engineering Research Software in Energy Research", in *Proceedings of the 16th ACM International Conference on Future and Sustainable Energy Systems*, ser. E-Energy '25, New York, NY, USA: Association for Computing Machinery, Jun. 2025, pp. 446–459, ISBN: 979-8-4007-1125-1. DOI: 10.1145/3679240.3734606.

[28] T. Brown, J. Hörsch, and D. Schlachtberger, "PyPSA: Python for Power System Analysis", *Journal of Open Research Software*, vol. 6, no. 1, Jan. 2018, ISSN: 2049-9647. DOI: 10.5334/jors.188.

[29] P. Kuckertz et al., "DataDesc: A framework for creating and sharing technical metadata for research software interfaces", *Patterns*, p. 101 064, Oct. 2024, ISSN: 2666-3899. DOI: 10.1016/j.patter.2024.101064.

[30] J. Klar et al., *Research Data Management Organiser (RDMO)*, Zenodo, Jul. 2025. DOI: 10.5281/zenodo.15807655.

[31] L. J. Castro et al., "DMP4NFDI - NFDI Basic Service for Data Management Plans. Proposal Integration Phase", May 2025. DOI: 10.5281/zenodo.15482168.

[32] D. Nüst and S. J. Eglen, *CODECHECK: An Open Science initiative for the independent execution of computations underlying research articles during peer review to improve reproducibility*, Jul. 2021. DOI: 10.12688/f1000research.51738.2.

[33] *Auto CodeMeta Generator*, Ontology Engineering Group (UPM), Jun. 2025. Accessed: Jul. 17, 2025. [Online]. Available: https://github.com/oeg-upm/auto-codemeta-generator.

[34] S. Ferenz, A. Jafarbigloo, O. Werth, and A. Nieße, *SMECS: A Software Metadata Extraction and Curation Software*, Jul. 2025. DOI: 10.48550/arXiv.2507.18159. arXiv: 2507.18159 [cs].
[35] L. J. Castro, B. Mathiak, and A. Nieße, "Connected Open Source Software - ConnOSS - Proposal", Jun. 2025. DOI: 10.5281/zenodo.15616384.
[36] W. Hasselbring et al., "Multi-Dimensional Research Software Categorization", *Computing in Science & Engineering*, pp. 1–10, 2025, ISSN: 1558-366X. DOI: 10.1109/MCSE.2025.3555023.