Deutsche
Forschungsgemeinschaft

Handling of Research Software in the DFG's Funding Activities

October 2024

Project ‘Digital Turn in the Sciences and Humanities’

Dr. Matthias Katerbow (matthias.katerbow@dfg.de)

Dr. Florian Mannseicher (florian.mannseicher@dfg.de)

Dr. Saskia Metzler (saskia.metzler@dfg.de)

Dr. Stefan Karcher (stefan.karcher@dfg.de)

Table of contents

General information and context

Preliminary remarks

The provision, findability and accessibility of research software is an essential prerequisite for quality-assured, software-supported research and its reproducibility, also enabling the use of existing software for further research. The Alliance of Science Organisations in Germany previously addressed the development, use and provision of research software in 2018 in its Recommendations on the Development, Use and Provision of Research Software. Building on this, the following notes and guidelines provide essential information relevant to proposal submission, review and evaluation of research projects involving the (further) development of research software.

In view of the diverse possibilities in which research software can be used and developed in all areas of science and humanities and the growing need for funding in the areas of development, archiving, findability and reuse of research software, the following remarks are to be understood as interdisciplinary recommendations. As guiding principles, they concern the planning of research projects (section 1), the resulting consequences for proposal submission (section 2), review (section 3) and the formation of structures at universities and research institutions (section 4).

Research software in an international and institutional context

The German Research Foundation (DFG) is committed to the development of common standards for research software at an international level. The following sections are therefore based on the principles and guidelines of international initiatives in whose development the DFG is involved.

Efforts to coordinate development processes and funding procedures and to set common standards are being driven in particular by Science Europe, with its recommendations for research funding organisations to develop and align policies on research software, but also by the Research Software Alliance (ReSA), which the DFG joined in July 2023, with the Amsterdam Declaration on Funding Research Software Sustainability acting as an important foundation. These activities are embedded in the recommendations of international consortia and committees that take into account aspects of research software development such as the FAIR Principles for research software (FAIR4RS) and the ReSA Software Policies. Software development also plays a major role in the Open Science Movement (UNESCO Recommendation on Open Science (November 2021) and the Coalition for Advancing Research Assessment (CoARA)).

Further handouts have been created that already present best practices and suggestions on various aspects of research software development for specific disciplines within various research institutions and in the context of the growing number of research software communities and initiatives. These include regional and institutional guidelines issued by universities as well as documents issued by non-university research institutions that are to be taken into account in the research process. The recommendations and notes in the following sections supplement and support these.

1 Guiding principles of the DFG in the development of research software

Preamble

Research software fulfils important functions within digital research practice: it enables research and data utilisation as well as the reproducibility of scientific results. Publishing the source code of research software also ensures the visibility and availability of creative and scientific development. In this way, research software creates important impulses for advancing research and for strengthening scientific collaborations and networks. Against the background of the diverse possibilities in which research software can be used and developed in all areas of science and the growing need for funding in the areas of development, archiving, findability and reuse of research software, the DFG draws on these principles to point out key aspects that should be reflected upon and considered when planning research projects.

"What is research software?"

An internationally recognised definition of research software is:

> *"Research Software includes source code files, algorithms, scripts, computational workflows and executables that were created during the research process or for a research purpose." (Gruenpeter et al. 2021: "Defining Research Software: a controversial discussion".*
> *Zenodo: https://doi.org/10.5281/zenodo.5504016*

From the DFG's perspective, research software is software created during the research process or for a research purpose. It includes, for example, source code, scripts and executable files. It is used, among other things, for the collection, analysis, simulation, processing, presentation or use of observation and measurement data or digitised text, image, film, sound sources, objects, etc. as well as for the generation of scientific models, the control of scientific instrumentation or for process optimisation.

The definition of research software formulated here covers a broad spectrum of software projects that vary in terms of subject, topic and quality. Research software can differ accordingly in its scope, purpose, context and properties. In most cases, research software also includes aspects that play a role in its development, such as technical documentation, user manuals, parameterisation, management plans and digital notebooks.

In principle, the use and (further) development of research software follows the DFG Guidelines for Safeguarding Good Research Practice and the DFG's ethical, legal, ecological and structural standards. These must be taken into consideration in the (further) development of research software beyond the actual research tasks, for example in the responsible use of natural resources with regard to the utilisation of computing power and instrumentation procurement (further information: Sustainability Guide for Research Processes in the DFG context). In addition, when assembling project teams and establishing communities, measures can be taken and processes established that promote gender equality, create a diverse working environment and enable inclusion, e. g. with regard to inclusive community management or the composition of development teams (further information: Equity and Diversity in the DFG context).

The following guiding principles additionally define key aspects that should be taken into consideration in the (further) development of research software as notes and guidelines in the research process. They are based on the observation that the (further) development of research software is subject to a trade-off between the diversity of use cases and research objectives on the one hand and standardisation in the software development process on the other. In particular, the orientation towards common, interdisciplinary standards in the (further) development of research software therefore enables the co-operative applicability and sustainable usability of research software. Accordingly, application of the guiding principles in practice should also be geared to the individual state or orientation of a research software.

1. Software development and standards

The development and application of research software should follow best practice standards for software development. In the development process, these software development standards depend on the purpose of the software (e. g. simulation, control of instrumentation, data analysis, etc.). It is therefore an essential element of project planning to define all steps of software development in advance in accordance with disciplinary standards, e. g. for use cases, authorship, versioning, licensing, reuse, security, etc. Clear evaluation and development criteria should be defined in order to enable an agile development process.

2. Software quality

Appropriate research software quality is an important basis for the development and use of research software in practice. In addition to general software quality criteria in the field of software engineering, the quality of research software should be defined on

the basis of discipline-specific criteria. In addition to the FAIR4RS Principles tailored to research software, there are also frameworks from the field of software engineering that allow a description of the quality of (research) software (test coverage criteria, analytical quality assurance, interface and tutorial documentation). Quality standards for research software should therefore refer to these broader frameworks. Preference should be given to those frameworks that potentially allow development services to be appropriately rewarded.

3. Accessibility and documentation

As the use and development of research software is part of the research process, accessibility and documentation should ensure the traceability of research results in the methodological sense. Making research software accessible is an important quality assurance element of the research process, as research methods can then be retraced and the verification of findings can be guaranteed. Source code, workflows and the functionality of research software should therefore be documented in a comprehensible manner and made available in order to ensure verifiability and reproducibility in the research process.

4. Citability and reusability

In addition to its pursuit in the research process itself, the development of research software should enable utilisation in other research projects. Research projects should examine the extent to which (a) existing research software can be reused, (b) this software can be further developed and improved, or (c) the development of new software is necessary which is then made available to the research community for further use. When reusing existing research software, the rights of use should be checked (usually via the licence). In order to ensure the successful reuse of research software that is to be (further) developed in-house, care should be taken to ensure the greatest possible machine findability and openness of the licence for scientific use.

5. Software sustainability

The sustainability of research software should not only be ensured through its accessibility but also through plans for maintenance, further development, findability and interoperability. In order to ensure the long-term usability of research software, infrastructures and existing repositories should be used to secure research software in the long term, but structures should also be established to enable the active participation of the

user and development community. In some cases, the creation of sunsetting concepts at the end of the development process can also be useful in order to transfer results to subsequent research software projects, e. g. individual software modules or active communities.

2 Information for applicants: the (further) development of research software as part of project funding

The (further) development of research software affects different levels of the research process within projects. Research software can, for example, be used or (further) developed as a research method or as part of a research infrastructure, or as an independent research object. For all uses of research software, the DFG's guiding principles for the development of research software provide practical advice that should be taken into consideration during project planning and addressed when submitting a proposal.

Documentation and software management plans

Subject-specific best-practice examples can provide useful guidance when creating software and its documentation. Depending on the maturity or category of the research software, it may be advisable to set out key steps in the development process in a software management plan, from defining requirements to covering aspects of software evaluation and maintenance. It will often be advisable to do this during the project planning stage.

It is not necessary to submit a software management plan as part of the funding proposal. However, the proposal should indicate and justify the fundamental decisions made regarding the type of software as well as the development process, accessibility and maintenance of the software.

New or further development of research software

The proposal should address the reasons for a decision to develop new research software or further develop existing research software (e. g. the possibilities of or obstacles to using existing software). In addition, the proposal should describe how compliance with subject-specific development standards will be achieved. The personnel required for software development should be explicitly named and their tasks and qualification requirements described.

Version management

For research software that passes through various development stages in the course of a research project, the proposal should address how suitable version management is implemented in order to ensure the traceability of the research process.

Quality assurance in the research and development process

Considerations regarding objectives, milestones, etc., which can indicate project success, and targeted options for certifying the software – including subject-specific certification

where available – should be included in the project planning as a further element to underpin sound quality assurance for software development. The project-specific decisions must be outlined in the proposal submission.

Integration, use, licensing of third-party code and citation practice

When citing research software, in addition to the usual information ("title of the software", "year", etc.), it is also necessary to state the "author(s)", "URL/DOI" and "version number". The recommendation here is that applicants should draw on the usual professional standards and ensure machine citation.

The use of third-party code, i. e. integration of software packages into the applicant's own software, etc. must be documented transparently in accordance with the applicable licence terms and in particular with regard to research integrity.

When submitting a proposal, care must be taken to ensure that any rights of use are obtained and that the use of third-party code is cited in accordance with the guidelines of good research practice.

Publications

When publishing research software and software documentation, preference should be given to publication channels that follow the Open Science or FAIR4RS principles, providing there are no legal or other reasons to the contrary. If decisions regarding documentation, licensing and publication models are made during project planning, this should be stated when the proposal is submitted.

Rights and authorship of research software engineers

Before starting a research project, it must be clarified and, if necessary, contractually regulated with employees, students or external service providers to what extent their development contribution constitutes their own co-authorship and exploitation rights. A distinction should be drawn between development tasks that relate to normal work processes (e. g. installation of modules) and those that constitute an independent scientific achievement (e. g. project contributions in qualification and final theses). The development work done by project staff and third parties should be identified in a suitable manner (e. g. in the software documentation) and their role in the project should be described in the proposal.

Networking and building development and user communities

The (further) development of research software and ensuring its subsequent use is often dependent on close cooperation and the participation of a development and user community. It is therefore advisable to specify in the proposal the extent to which participation in

existing national and/or international communities or the establishment of a user or development community is planned as part of the project and which objectives are being pursued.

Support options in funding programmes

Individual and collaborative projects

For the (further) development of research software that is used to address a research question and in a specific research context, funding can be requested for personnel, material costs, etc. accordingly. Funding can also be requested to outsource development tasks to third parties (e. g. central service units or external service providers), to increase ecological sustainability (e. g. CO_2 compensation or higher acquisition costs for ecologically sustainable hardware), and to set up measures in the area of equal opportunities and diversity (equal opportunities module).

Depending on the relevant job description for academic staff in all academic fields, personnel costs can be applied for up to a maximum of 100%. Staff positions do not have to be linked to an academic qualification (usually a doctorate or post-doctoral lecturing qualification). When submitting a proposal, it must therefore be precisely stated (a) which work tasks are to be fulfilled by the academic staff, (b) in which personnel cost category (E9-E14) the employees are classified based on the job description, and (c) whether the assigned tasks involve a qualification opportunity.

Funding for community-building can be applied for under the "Project-specific workshops" and "Public relations" modules.

In collaborative projects, central funding (e. g. in the coordination fund) can be applied for to develop software that creates a recognisable added value for the individual projects through the development and use of software.

Proposals must be submitted in accordance with the requirements of the respective funding programme or in accordance with the Proposal Preparation Instructions – Project Proposals (54.01):

www.dfg.de/formulare/54_01

Infrastructure

Through its Research Software Infrastructures programme in the field of Scientific Library Services and Information Systems (LIS), the DFG supports science in establishing structures for developing, making available, finding and securing research software. It is important to ensure that the infrastructure to be developed does not depend on a

specific research question but rather supports researchers across all locations in using research software. The LIS funding programme e-Research Technologies offers funding in particular for the expansion of comprehensively matured research software, virtual research environments and digital research platforms. The funding programme Coordinating Roles and Responsibilities in Information Infrastructures (VIGO) is recommended for the development of solutions to tackle specific challenges in the establishment, expansion or long-term safeguarding of research-related information infrastructure.

Projects with research and information infrastructure components

Projects with a scientific focus can also have a strong infrastructure orientation. These projects are characterised by a close interlinking of subject-specific science and information infrastructures. For such projects, proposals can be submitted as a combination of an Individual Research Grant and funding under a programme from the area of Scientific Library Services and Information Systems (LIS):

- For example, research software can be further developed as a research method in the scientific part of the project (Individual Research Grant) and combined with the LIS funding programme for the Digitisation and Indexing of holdings or collections.
- For projects which involve both research and the (further) development of technologies, tools, procedures and methods for scientific information provision, the Individual Research Grant can be combined with the e-Research Technologies programme.

3 Information on the review of project proposals involving the (further) development and use of research software

The review of project proposals which significantly involve the (further) development of research software to address research questions usually also requires expert opinions that assess not only the scientific significance of the project but also the (further) development process and use of the research software. In addition to the usual review criteria, the following software-specific aspects should therefore be taken into account:

1. Discipline-specific regulations

The (further) development and use of research software is characterised to a large extent by the different requirements of scientific disciplines. The review should therefore determine whether the project integrates the research topic in the discourse of the discipline and whether the software development fulfils the usual disciplinary standards and possible subsequent use cases.

2. Recognising the (further) development of research software as a scientific achievement

The development of research software requires not only in-depth subject-specific knowledge in order to be able to adequately implement the requirements of the subject in research software, but also a high level of expertise in the area of software development. When recognising scientific qualifications, the skills required for subject-specific scientific software development should therefore also be explicitly considered as independent scientific achievements. In particular, aspects of the discipline-specific development process should be taken into account such as requirements definition, functionality of the research software, operationalisability of a research method, etc., which go beyond programming knowledge and require a high level of scientific expertise.

3. Project planning and work programme

The review should look at whether the (further) development of research software follows the standards of software development, is subject to the logic of the subject-specific research process, and also fulfils project-specific requirements. The objectives and work programme of a research project should be assessed accordingly according to whether

the purpose and requirements definition of the research software development and the central steps and objectives of the development process are plausibly linked to the research project.

4 Further development of the structural framework conditions (appeal to scientific communities and research institutions)

The (further) development and use of research software requires specific framework conditions that cannot be realised in the planning of an individual research project or through project-based funding. The DFG would therefore like to call on scientific communities and research institutions to promote the adaptation and establishment of infrastructures that support the (further) development and use of research software and relate to the following areas:

1. Research software engineering „as a service“

Software development can benefit considerably from the university's own central service units. These service units can support researchers in project planning, carry out development tasks and open up publication channels. The DFG therefore encourages research institutions to identify requirements for local services and offerings and to implement coordinated measures to support software development projects.

2. Change in research culture

Software development generally requires a high level of subject-specific expertise as well as transdisciplinary collaboration and communication. Research institutions should therefore strengthen structural development processes that support research conducted in groups and teams, community work, and open scientific communication. This can create an agile research software development environment that will also benefit individual research projects. In addition, training programmes should be provided to create a broad understanding and greater awareness within the scientific community of the possibilities available for researchers to use and develop research software in their own projects, thereby revealing its potential in connection with the research process and highlighting the fact that research software development itself can be recognised as a scientific achievement.

3. Standards and good practices

In order to establish uniform standards for project development and review, the DFG encourages scientific societies and communities to reflect on their use of research software and to develop subject-specific regulations and good practices for research software or to subscribe to existing recommendations. A process of understanding and negotiation

in the disciplines to identify centralised or community-relevant research software should also be initiated and actively implemented.

4. Infrastructures for research software

Infrastructure facilities such as libraries, research data centres and computing and information centres can provide nationwide support for the professional use of research software. Infrastructural facilities should therefore further develop existing local solutions and develop new innovative offerings while at the same time focusing on the greatest possible interoperability and convergence of the developed offerings with a view to creating an overall structure. This should be accompanied by the establishment of a suitable distribution of tasks within the scientific infrastructures.

Deutsche Forschungsgemeinschaft
Kennedyallee 40 • 53175 Bonn
Postal address: 53170 Bonn, Germany
Tel.: +49 228 885-1
Fax: +49 228 885-2777
postmaster@dfg.de
www.dfg.de/en