



# **Sustainable and FAIR Software in Research – A RDMO Catalogue for Software Management Plans**

Workshop “Machine-actionable Software Management Plans”,  
ZB Med, Cologne, 31<sup>st</sup> May 2023

Dr. Yves Vincent Grossmann  
[grossmann@mpdl.mpg.de](mailto:grossmann@mpdl.mpg.de)  
Max Planck Digital Library

# Agenda

1. Short Introduction
2. Added Values with SMPs
3. Realisation with RDMO

# Short Introduction

# Speaker

- Yves Vincent Grossmann
- <https://orcid.org/0000-0002-2880-8947>
- Research Data Management Officer since October 2020 at the Max Planck Digital Library
- a.o. Service Lead for [Edmond](#), [Labfolder](#) and [RDMO](#)
- Contact: [grossmann@mpdl.mpg.de](mailto:grossmann@mpdl.mpg.de)

# Max Planck Society

- 85 research institutes
  - also Italy, Luxembourg, the Netherlands and USA (Florida)
- Max Planck Digital Library and local libraries at institutes
- Three data centres (DKRZ, GWDG, MPCDF)
- General administration in Munich
- Around 24,000 employees
- Budget of €2 billion in 2020
- 30 Nobel Laureates



# Max Planck Digital Library

- Amalienstrasse 33, 80799 Munich, Germany
- Information services since 2007, independent service unit since 2012
- Around 80 employees from software development, library, science management and administration
- [www.mpdl.mpg.de](http://www.mpdl.mpg.de)
- Collections department responsible for research data, research software and more



The only bookshelf in the MPDL

# Added Values with SMPs

# SMP Definition by DINI/nestor AG FD

„Laut Definition der DINI/nestor AG Forschungsdaten beinhaltet ein Softwaremanagementplan (SMP) allgemeine und technische Informationen zum Softwareprojekt, Angaben zur Qualitätssicherung, zum Release und zur öffentlichen Verfügbarkeit sowie rechtliche und ethische Aspekte, die die Software betreffen.

Der SMP fasst Informationen zusammen, die die Erstellung, Dokumentation, Speicherung, Versionierung, Lizenzierung, Archivierung und/oder Veröffentlichung der in einem Projekt erzeugten oder verwendeten Software hinreichend beschreiben und dokumentieren.

Dazugehörige Hardware und notwendige andere Ressourcen, aber auch damit verbundene weitere Software und Softwarebibliotheken, Text- und Datenpublikationen sind ebenfalls zu beschreiben und stellen eine Besonderheit des SMP dar.

Zweck eines SMPs ist zunächst die Nachvollziehbarkeit sowie ggf. die langfristige Nutzbarkeit der Software (zur direkten Anwendung sowie zur Weiterverarbeitung) zu unterstützen und den Support der Nutzer\*innen bei Rückfragen zu erleichtern. Der SMP dient folglich auch der Qualitätssicherung (vgl. hierzu FAIR4RS Principles).“



# Researchers Writing Research Software

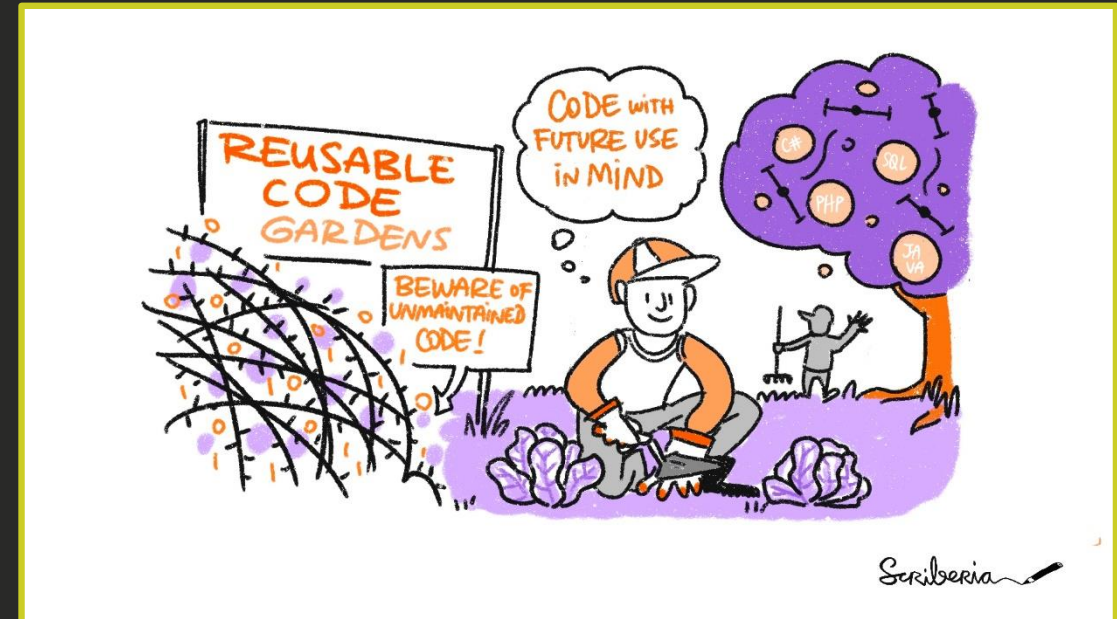
## Observations:

- Usually not trained but self-educated developers
- Functionality before documentation before sustainability
- First text publication, then nothing for a long time, and then maybe data publication and software publication
- Software is often handed over from one PhD student to the next
- ...

# Re-Usability of Research Software

Conscious handling is increasing the likeliness of re-use:

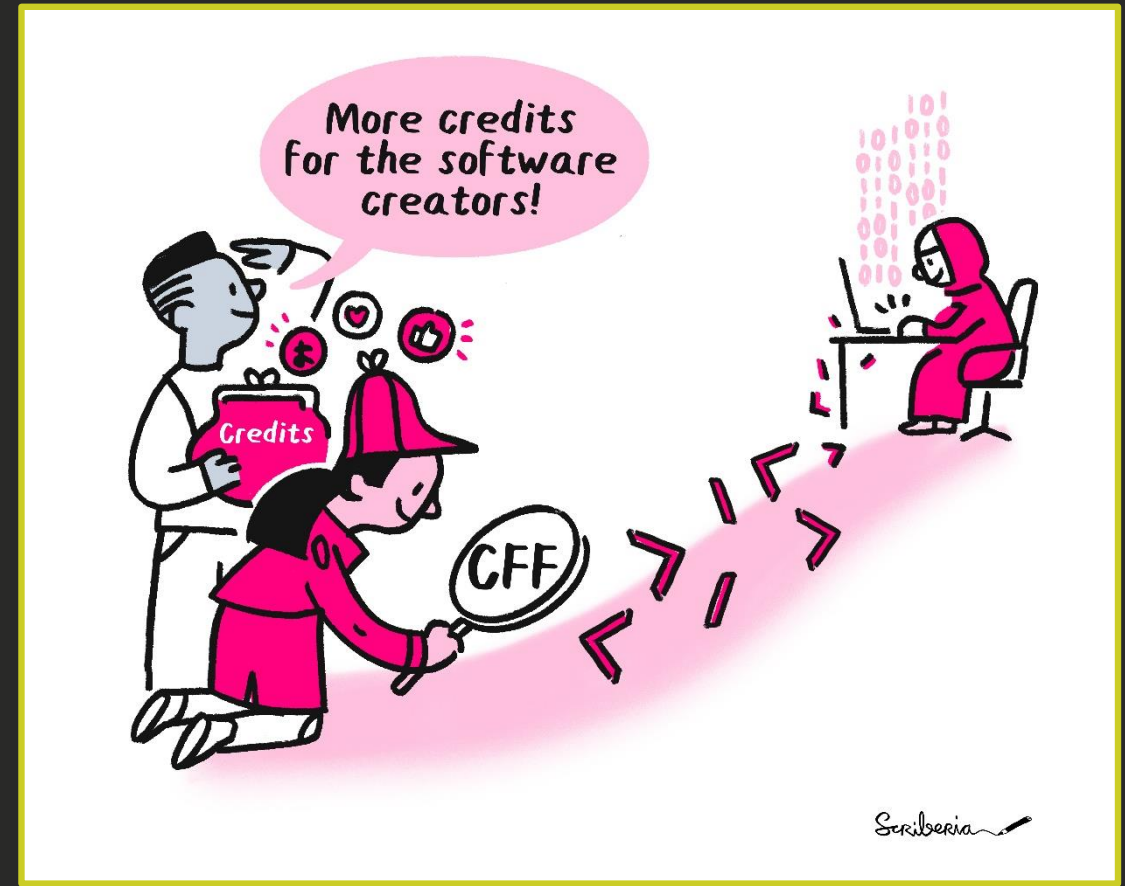
- Increased probability of publication
- Explicit licensing
- Clear code structure and reflected use of third-party libraries
- A targeted approach to archiving
- ...



The Turing Way Community, & Scriberia. (2022). Illustrations from The Turing Way: Shared under CC-BY 4.0 for reuse. Zenodo. <https://doi.org/10.5281/zenodo.6821117>.

# Recognition of Research Software

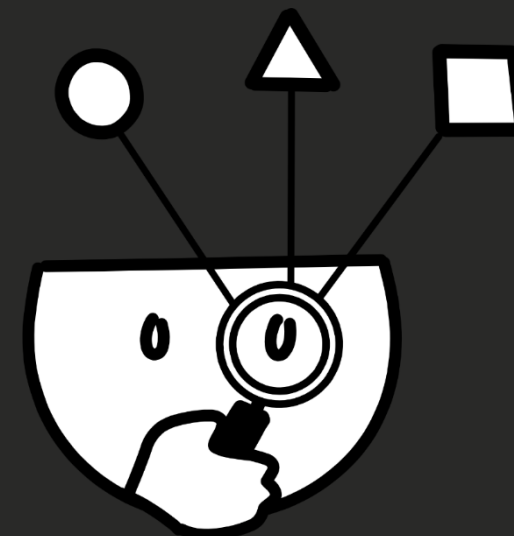
- Disciplinary credits for software publication
- Credits/funding by funders
- Institution-wide credits for development
- Policies needed
  - Normative framework for software publication
  - Endorsement of software publication



The Turing Way Community, & Scriberia. (2022). Illustrations from The Turing Way: Shared under CC-BY 4.0 for reuse. Zenodo. <https://doi.org/10.5281/zenodo.6821117>.

# Added Value for Institutions

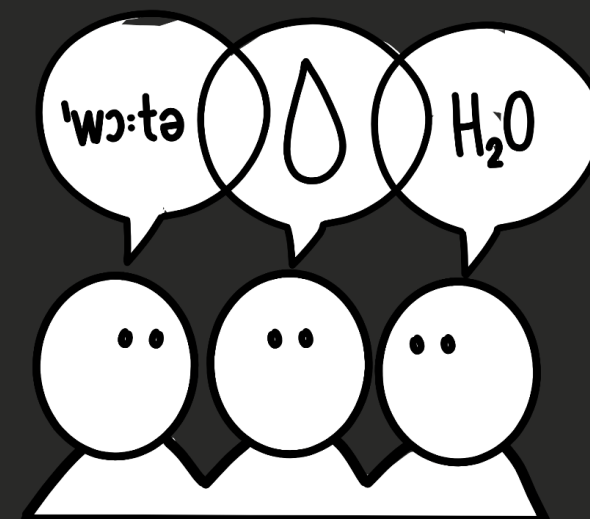
- Support for developers
- Getting an overview on starting/running software projects
- More reasonable planning of demands (e.g. storage, software licenses)
- Better quality management
- Easier archiving
- Easier re-use
- ...



<https://doi.org/10.5281/zenodo.3674561>

# Why should I write an SMP?

- **For myself!**
- Together with IT/Scientific Computing Unit/... to better design a software project
- For a funding application
- For internal planning
- For sustainability and a possible publication/archiving (good scientific practice)
- Quality assurance
- ...

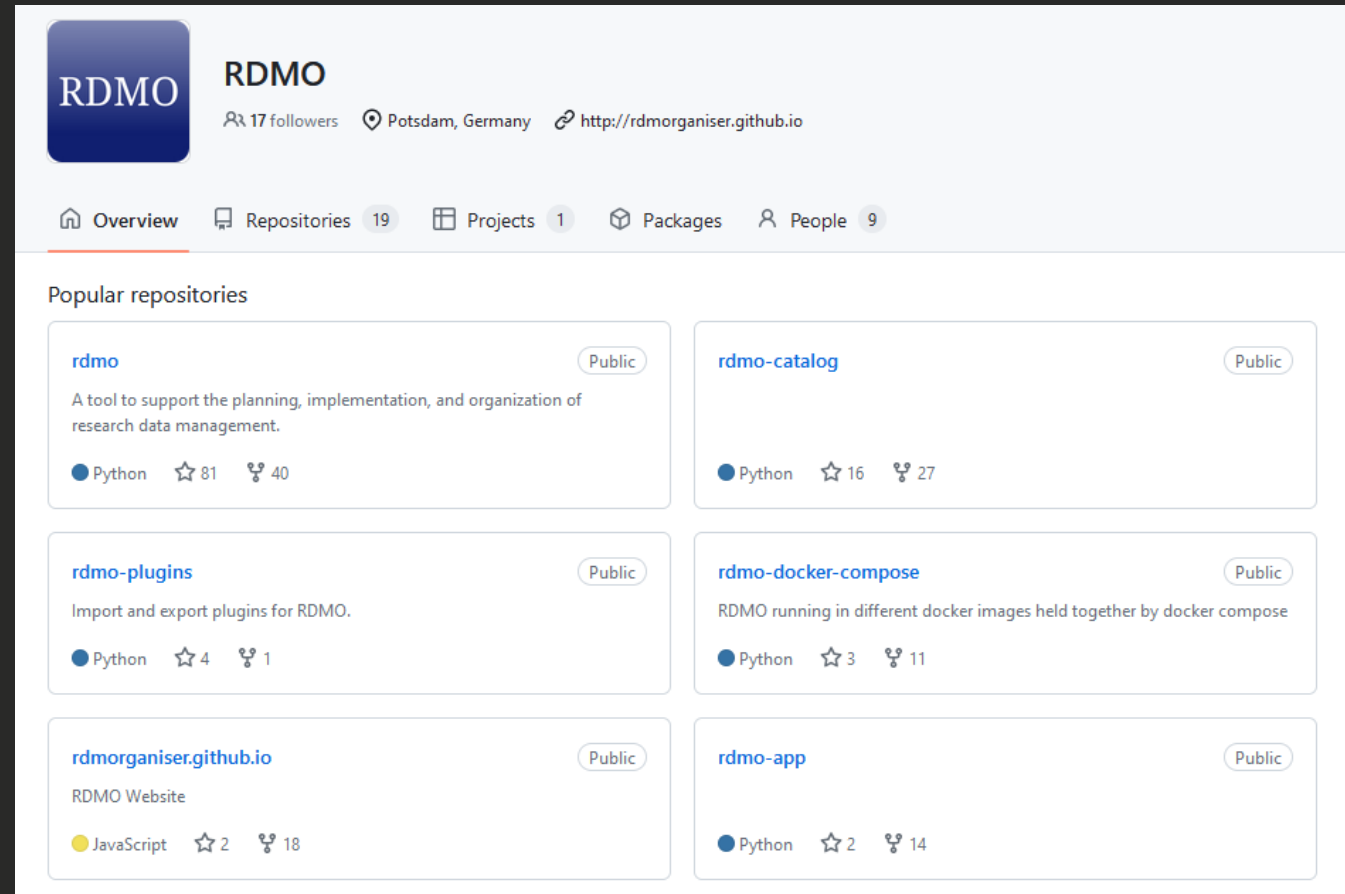


<https://doi.org/10.5281/zenodo.3674561>

# Realisation with RDMO

# RDMO

- Open Source Software
- Started with an DFG grant in 2015
- Vivid community
- Widely used software for DMP service at German-speaking institutions
- Many different catalogues for DMPs, i.e. Horizon Europe, DFG, Science Europe



The screenshot shows the GitHub profile for RDMO. The profile header includes the RDMO logo, the name 'RDMO', 17 followers, location 'Potsdam, Germany', and the website 'http://rdmorganiser.github.io'. Below the header is a navigation bar with 'Overview' (selected), 'Repositories 19', 'Projects 1', 'Packages', and 'People 9'. The main content area is titled 'Popular repositories' and displays six repository cards:

Repository Name	Language	Stars	Forks	License
rdmo	Python	81	40	Public
rdmo-catalog	Python	16	27	Public
rdmo-plugins	Python	4	1	Public
rdmo-docker-compose	Python	3	11	Public
rdmorganiser.github.io	JavaScript	2	18	Public
rdmo-app	Python	2	14	Public

<https://github.com/rdmorganiser>

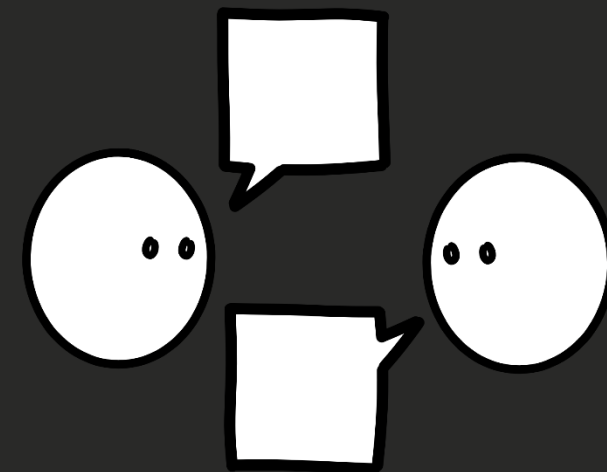
# Project Data

- Max Planck Digital Library, Collections Division
  - Mainly Yves Vincent Grossmann, Michael Franke and Jan Matthiesen
- From July 2022 to December 2022
- RDMO as technical framework
- Outcome: CC0-Push of an SMP catalogue for the RDMO Community



# Project Intensions

- Offer writing SMPs as a service soon
- Direct output, no paper
- Develop explicit knowledge on research software
- Contribute to a general discussion



<https://doi.org/10.5281/zenodo.2581783>

# SMP as RDMO Catalogue

- Title: „Software Management Plan for Researchers“
- in German & Englisch
- 49 questions in total
- Available at <https://github.com/rdmorganiser/rdmo-catalog> with a CC0 license
- FAIR4RS Viewer as add-on

# Audience

- Researchers
- IT staff, scientific core unit, to consult researchers
- Funding acquisition staff
- PIs, research coordination
- ...



<https://doi.org/10.5281/zenodo.2581783>

# Catalogue Structure

1. General
  - Stakeholders, project management, ...
2. Technical
  - Code, Infrastructure, Security/Safety, ...
3. Quality Assurance
  - Testing, Documentation, ...
4. Release and Publish
  - Releases, metadata, re-use, ...
5. Legal and Ethics
  - Copyright, licenses, dual use, ...

# Screenshots

RDMO for MPG Back to project Language Michael Franke

My Projects / SMP test / General

## Topic

**What is the title of the software project?**

The title of the software can, of course, still change or be specified in the course of a project. Nevertheless, it makes sense to give the project a specific name at the beginning in order to facilitate further (internal) communication about it.

Best Software Ever (BSE)

**Which research field(s) does this software belong to?**

The list of disciplines follows the [subject classification of the DFG \(German Research Foundation\)](#). Please enter the items line by line. You can add items using the green button and remove them using the blue cross (x).

Engineering Sciences / Process Engineering, Technical Chemistry

Add item

**What is the intended use of the software? How will your software contribute to research?**

The intention for developing software in a scientific context can be manifold. The same applies to the specific application of the software and the contribution to scientific knowledge. Depending on the scientific community, these points can also differ significantly from each other.

We offer here some recommended reading that may be helpful in answering the questions:

- Anzt et al.(2021): deRSE Position Paper: "An environment for sustainable research software in Germany and beyond: Current state, open challenges, and call for action", <https://doi.org/10.12688/f1000research.23224.2>.
- Gardner et al. (2022): Sustained software development, not number of citations or journal choice, is indicative of accurate bioinformatic software, In: Genome Biology 23, 56, <https://doi.org/10.1186/s13059-022-02625-x>.
- Katerbow, & Feulner (2018): Recommendations on the development, use and provision of Research Software, <https://doi.org/10.5281/zenodo.1172988>.
- Lee, et al. (2021): Barely sufficient practices in scientific computing, In: Patterns, 2(2), 100206, <https://doi.org/10.1016/j.patter.2021.100206>.

## Overview

Project: SMP test  
 Catalog: Software Management Plan for Researcher  
 Back to my projects

## Progress

Back Skip

## Navigation

Please note that using the navigation will discard any unsaved input.

Entries with Ⓞ might be skipped based on your input.

General  
 → Topic  
 Software Project Partner(s)  
 Software Project Schedule  
 Software Project Management  
 Software Development Requi...

Technical  
 Quality Assurance  
 Release and Publish  
 Legal and Ethics

General Topics

RDMO for MPG Back to project Language Michael Franke

My Projects / SMP test / Technical

## Code

**Which programming language(s) do you plan to use?**

The software languages go hand in hand with different methods, focuses and required skill levels. Here you can document which language(s) you choose and why. Examples of software languages can be Java, Python, R, etc.

At this point it is also useful to document whether technical standards are relevant for the development of software. Examples include [IEC Standard 62304](#) for software in medical applications, or [ECSS-E-ST-40C](#) for software in space applications, or [DICOM](#) for the exchange of medical image data.

Which technology or process is used for versioning?

Versioning code during development is strongly recommended. A widely used software versioning tool is Git. Various Max Planck Institutes maintain their own hosting services for git repositories such as Gitlab and Github.:

- Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen: <https://gitlab.gwdg.de>
- Max Planck Computing and Data Facility: <https://gitlab.mpcdf.mpg.de>
- Max Planck Institute for Molecular Genetics: <https://github.molgen.mpg.de> (as Max Planck research you can get an account via their helpdesk ([helpdesk@molgen.mpg.de](mailto:helpdesk@molgen.mpg.de)))

## Overview

Project: SMP test  
 Catalog: Software Management Plan for Researcher  
 Back to my projects

## Progress

Back Skip

## Navigation

Please note that using the navigation will discard any unsaved input.

Entries with Ⓞ might be skipped based on your input.

General  
 Technical  
 → Code  
 Third Party Components and ...  
 Infrastructure  
 Preservation  
 Security  
 Quality Assurance  
 Release and Publish  
 Legal and Ethics

Code and Programming

# Screenshots

RDMO for MPG Back to project Management Admin Language Yves Vincent Grossmann

## Software Project Management

**Which software development process is defined? How will process roles be assigned?**

For this question, it is useful to first describe the type of project management. Examples here would be agile methods such as Scrum. But a more classical approach like e.g. the "waterfall" model is also possible. Ultimately, it is crucial that the chosen method fits the goal and is communicated to and works for the people involved.

Once a methodological approach has been established, it may be useful to define different roles for the software project. Examples would be "Developer", "Tester", "Scrum Master" etc. At the same time, it is advisable to already draft a specification document for the project with requirements, usage scenarios and other specifications. The requirements from the user's point of view should be clear so that the conception and specification process by the developers can start on this basis.

For guidance, these texts may be of interest:

- Smith, Srinivasan and Shankar (2019): Debunking the Myth That Upfront Requirements Are Infeasible for Scientific Computing Software, IEEE/ACM 14th International Workshop on Software Engineering for Science (SE4Science), pp. 33-40, <https://doi.org/10.1109/SE4Science.2019.00011>.
- Marques and Milewicz (2022): User Stories in Scientific Software Development, Better Scientific Software Blog, [https://bssw.io/blog\\_posts/user-stories-in-scientific-software-development](https://bssw.io/blog_posts/user-stories-in-scientific-software-development).

**How do you track the different tasks and use cases?**

The development of software goes hand in hand with a high degree of complexity. In order to keep track of the diverse requirements, tasks and problems, it is advisable to organise these points in a clear structure. At the same time, this enhances the traceability of developments.

Examples of applications are stand-alone ticket systems such as Jira or Redmine. However, integrated systems can also be used just as well, such as provided in GitHub/GitLab (see for example the [Carpentries Lesson](#)). Working with a (digital) Kanban board can also support software development.

**Will there be a specification document (briefly) outlining the most important requirements?**

The description of the requirements can be one of the cornerstones for a successful software. It is therefore advisable to invest time resources into this right from the start.

Yes  No

Project Management

RDMO for MPG Back to project Language Michael Franke

My Projects / SMP test / Legal and Ethics

## License

**Under what kind of license(s) will the software be published?**

There are good arguments for assigning a license. [Guideline 13 in the DFG Code of Conduct "Guidelines for Safeguarding Good Scientific Practice"](#) even explicitly states that if "self-developed research software is to be made available to third parties, an appropriate licence is provided". FAIR4RS R1.1 also emphasises the same. Please also consider the compatibility with already integrated third-party libraries.

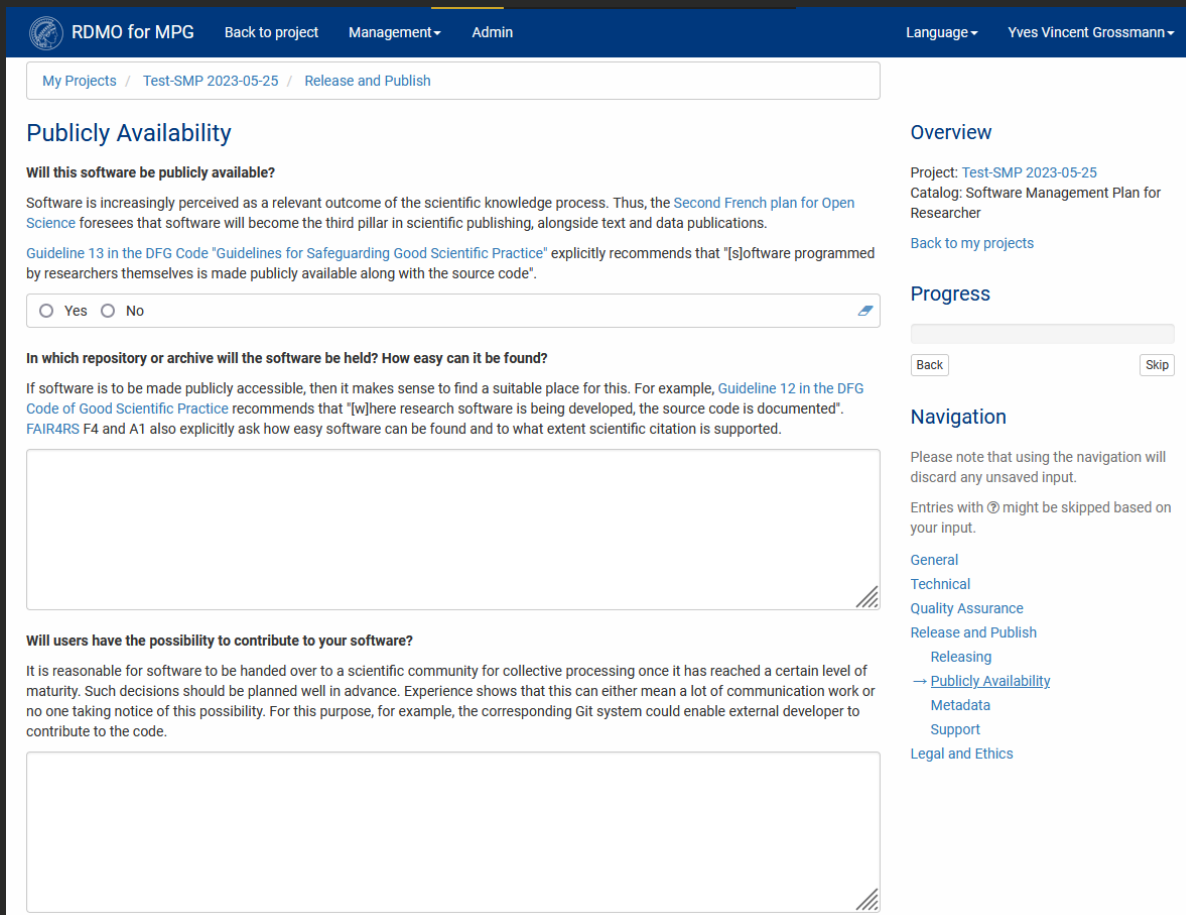
The following websites give you an initial overview of software and licences:

- <https://choosealicense.com>
- <https://github.com/readme/guides/open-source-licensing>
- <https://opensource.org/licenses>
- <https://tldrlegal.com>

Back Skip Save Save and proceed

License

# Screenshots



RDMO for MPG Back to project Management Admin Language Yves Vincent Grossmann

My Projects / Test-SMP 2023-05-25 / Release and Publish

## Publicly Availability

**Will this software be publicly available?**

Software is increasingly perceived as a relevant outcome of the scientific knowledge process. Thus, the [Second French plan for Open Science](#) foresees that software will become the third pillar in scientific publishing, alongside text and data publications.

[Guideline 13 in the DFG Code "Guidelines for Safeguarding Good Scientific Practice"](#) explicitly recommends that "[s]oftware programmed by researchers themselves is made publicly available along with the source code".

Yes  No

**In which repository or archive will the software be held? How easy can it be found?**

If software is to be made publicly accessible, then it makes sense to find a suitable place for this. For example, [Guideline 12 in the DFG Code of Good Scientific Practice](#) recommends that "[w]here research software is being developed, the source code is documented". [FAIR4RS F4](#) and [A1](#) also explicitly ask how easy software can be found and to what extent scientific citation is supported.

**Will users have the possibility to contribute to your software?**

It is reasonable for software to be handed over to a scientific community for collective processing once it has reached a certain level of maturity. Such decisions should be planned well in advance. Experience shows that this can either mean a lot of communication work or no one taking notice of this possibility. For this purpose, for example, the corresponding Git system could enable external developer to contribute to the code.

**Overview**

Project: [Test-SMP 2023-05-25](#)  
 Catalog: Software Management Plan for Researcher

[Back to my projects](#)

**Progress**

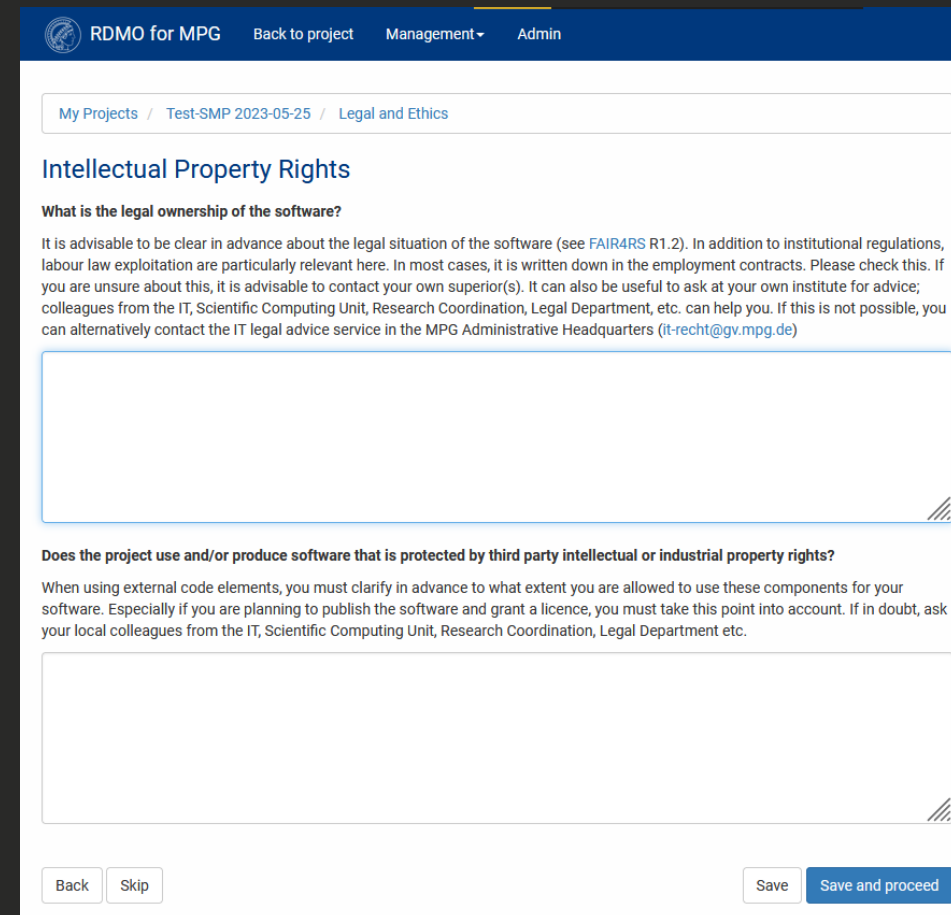
**Navigation**

Please note that using the navigation will discard any unsaved input.

Entries with © might be skipped based on your input.

- General
- Technical
- Quality Assurance
- Release and Publish
  - Releasing
  - **Publicly Availability**
  - Metadata
  - Support
- Legal and Ethics

Publicly Availability



RDMO for MPG Back to project Management Admin

My Projects / Test-SMP 2023-05-25 / Legal and Ethics

## Intellectual Property Rights

**What is the legal ownership of the software?**

It is advisable to be clear in advance about the legal situation of the software (see [FAIR4RS R1.2](#)). In addition to institutional regulations, labour law exploitation are particularly relevant here. In most cases, it is written down in the employment contracts. Please check this. If you are unsure about this, it is advisable to contact your own superior(s). It can also be useful to ask at your own institute for advice; colleagues from the IT, Scientific Computing Unit, Research Coordination, Legal Department, etc. can help you. If this is not possible, you can alternatively contact the IT legal advice service in the MPG Administrative Headquarters ([it-recht@gv.mpg.de](mailto:it-recht@gv.mpg.de))

**Does the project use and/or produce software that is protected by third party intellectual or industrial property rights?**

When using external code elements, you must clarify in advance to what extent you are allowed to use these components for your software. Especially if you are planning to publish the software and grant a licence, you must take this point into account. If in doubt, ask your local colleagues from the IT, Scientific Computing Unit, Research Coordination, Legal Department etc.

Intellectual Property Rights

# Questions and Supporting Information as .docx

- Only in English
- CC0 → easily adaptable
- <https://doi.org/10.17617/2.3481986>

## Template “Software Management Plan for Researcher”

This catalogue is for the management of scientific software projects. It supports scientists in the development and project organisation of software developments through fifty questions in different topic blocks.

This questionnaire was developed by the Max Planck Digital Library in the Collections area in late summer and autumn 2022. It can be used to create a software management plan in RDMO. Of course, we are always happy to receive feedback: [rdmo@medl.mpg.de](mailto:rdmo@medl.mpg.de).

The catalogue is available as RDMO catalogue under the CC0 licence (<https://creativecommons.org/licenses/by-sa/4.0/>) (<https://github.com/rdmo/rdmo-catalogue>), so that it can be freely (re)used. Changes and adaptations by the RDMO community are of course welcome and can easily be made via <https://github.com/rdmo/rdmo-catalogue>.

Software Management Plan for Researcher <https://doi.org/10.17617/2.3481986>

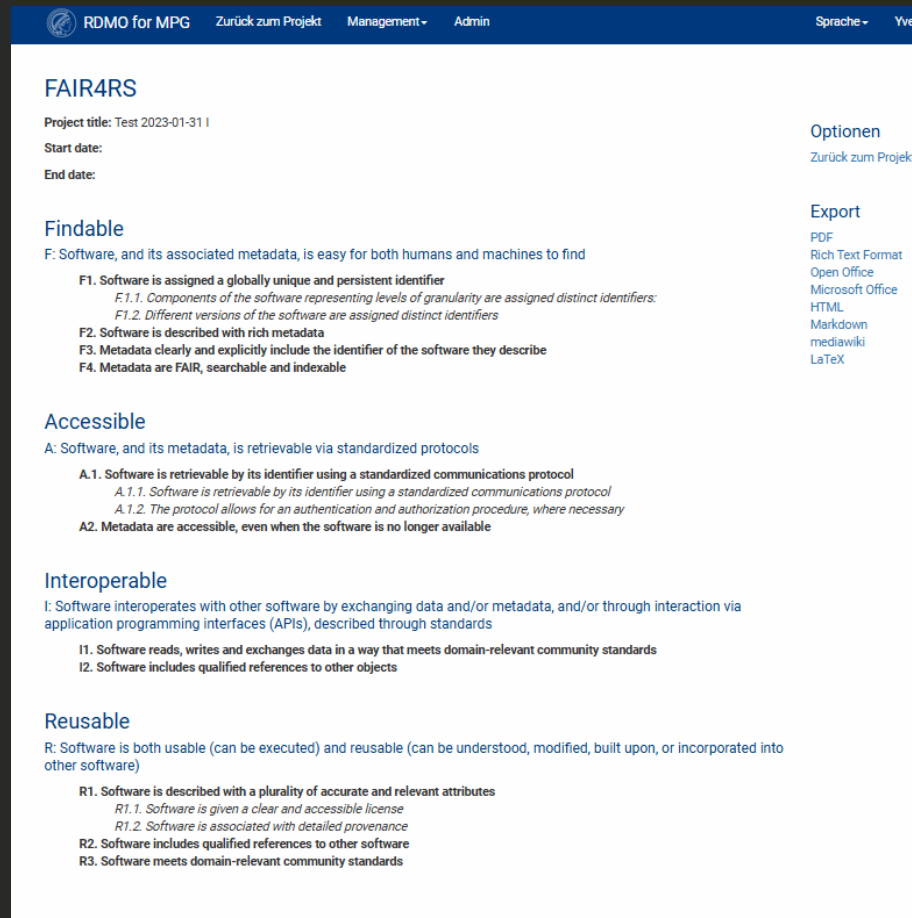
### Table of Content

<b>General</b> 4	<b>Security</b> 11
<b>Topic</b> 4	Which measures or provisions are in place to ensure software security? 11
What is the aim of the software project? 4	What Measures Do You Take to Mitigate Risks in Relation to Software Development? 11
In which research field(s) does this software belong to? 4	<b>Quality Assurance</b> 12
What is the intended use of the software? How will your software contribute to research? 4	<b>Governance and Defined Processes</b> 12
Are you sure that no suitable exists with the functionality you are planning? 4	Do you have a governance model for the software development? 12
<b>Software Project Partner(s)</b> 5	Do you apply specific coding standards? How do you take care about code quality control? 12
Who are the project participants that deal with this software? 5	<b>Documentation</b> 12
Is there existing (technical/scientific) resources that will serve as specific funding for the software development? 5	How is software documentation created? 12
<b>Software Project Schedule</b> 5	Where will the documentation be stored or made available? Which language will be used? 12
When does the software project start? 5	<b>Testing</b> 13
When does the software project end? 5	Which software test strategy are you going to use? Which types of tests are planned for the project? 13
<b>Software Project Management</b> 5	How is testing and test documentation organised? 13
How do you track the different tasks and milestones? 5	<b>Release and Publish</b> 14
Will there be a specification document (briefly) outlining the most important requirements? 5	<b>Relicensing</b> 14
<b>Software Development Requirements</b> 6	Are there defined release procedures for the software? 14
Are there institutional requirements for software development? 6	What is the decision process for releasing? How often will a software version be released? 14
Are there requirements regarding the software development from other parties? 6	<b>Publicly Availability</b> 14
<b>Technical</b> 8	Will this software be publicly available? 14
<b>Code</b> 8	In which repository or archive will the software be kept? How long can it be found? 14
In which programming language(s) do you plan to use? 8	Will users have the possibility to contribute to your software? 14
In which technology or process is used for development? 8	Is (Open) Peer Review planned for the software? 15
<b>Third Party Components and Libraries</b> 8	<b>Metadata</b> 15
Which external software components will be used? What dependencies on software libraries do exist? How do you document that? 8	How do you assign metadata for your software? 15
What licenses are on the third-party software components? 8	Do you give a persistent identifier for your software? 15
What is the process to keep track of the external software components? Can critical dependencies be identified or replaced? 8	<b>Support</b> 15
Do you plan to use third party web services? 8	How do you organise the support and feedback process with other users? 15
<b>Infrastructure</b> 9	Does your Software Management Plan refer to the Software Data Management Plan? 15
What infrastructure resources are needed? To what extent? 9	Do you intend to make your software management plan publicly available (host)? 15
Is there already existing infrastructure for the software development? Where is the infrastructure located? 9	<b>Legal and Ethics</b> 17
Are there technical aspects where competences are still lacking, so that support would be helpful? 9	<b>Intellectual Property Rights</b> 17
<b>Retention/Archiving</b> 10	What is the legal ownership of the software? 17
How long should the software remain usable? 10	Does the project use existing software that is protected by third party intellectual or industrial property rights? 17
Does this software have to be preserved for a longer term? 10	<b>License</b> 17
Where will the software be stored? Does the storage place have a clear preservation policy? 10	Under what kind of license(s) will the software be published? 17
	<b>Open Use</b> 17
	Can the software also be used for military purposes? 17



# FAIR4RS Viewer

- A view of the answers structured by the FAIR4RS facets
- Quick assessment of a software's FAIRness
- Exportable to many formats
- Developed by Jan Matthiesen (MPDL)
- Currently at re-work



The screenshot shows the FAIR4RS Viewer interface. At the top, there is a navigation bar with "RDMO for MPG", "Zurück zum Projekt", "Management", and "Admin". On the right, there are links for "Sprache" and "Yes". The main content area is titled "FAIR4RS" and displays project information: "Project title: Test 2023-01-31", "Start date:", and "End date:". To the right of this information are links for "Optionen" and "Zurück zum Projekt". Below the project information, the interface is organized into facets: "Findable", "Accessible", "Interoperable", and "Reusable". Each facet has a brief description and a list of specific criteria (F1-F4, A1-A2, I1-I2, R1-R3) with their respective sub-points. On the far right, there is an "Export" section with links for PDF, Rich Text Format, Open Office, Microsoft Office, HTML, Markdown, mediawiki, and LaTeX.

# Take Home Messages

1. Develop explicit knowledge on a software project is easily possible with an SMP
2. SMP template in RDMO is openly available and freely customisable for your own institute
3. Our RDMO template is a try to offer a SMP service quite early at the discussion

# Test SMP on RDMO

Like to Test SMP on RDMO?

→ Write us an email to [rdmo@mpdl.mpg.de](mailto:rdmo@mpdl.mpg.de) for an account

# Thanks

to the MPDL Collections Team, the [UAG-DMP](#) of the DINI/nestor-AG Forschungsdaten and to the RDMO Community

# Thanks for listening!

[rdmo@mpdl.mpg.de](mailto:rdmo@mpdl.mpg.de)