# The Software Management Plan

**Stéphanie Cheviron**



# What is a SMP?

→ A document describing how a software project is developed, maintained, and organised.

→ The goal of an SMP is to ensure that the software is usable and maintainable in the long term.

→ An SMP is written by the developers, maintainers, and/or other stakeholders of a software project.

Martinez-Ortiz, C., Martinez Lavanchy, P., Sesink, L., Olivier, B. G., Meakin, J., de Jong, M., & Cruz, M. (2023). Practical guide to Software Management Plans (1.1). Zenodo. https://doi.org/10.5281/zenodo.7589725

# What is a research software?

*Research software includes **source code files, algorithms, scripts, computational workflows and executables** that were created during the research process or for a research purpose.*

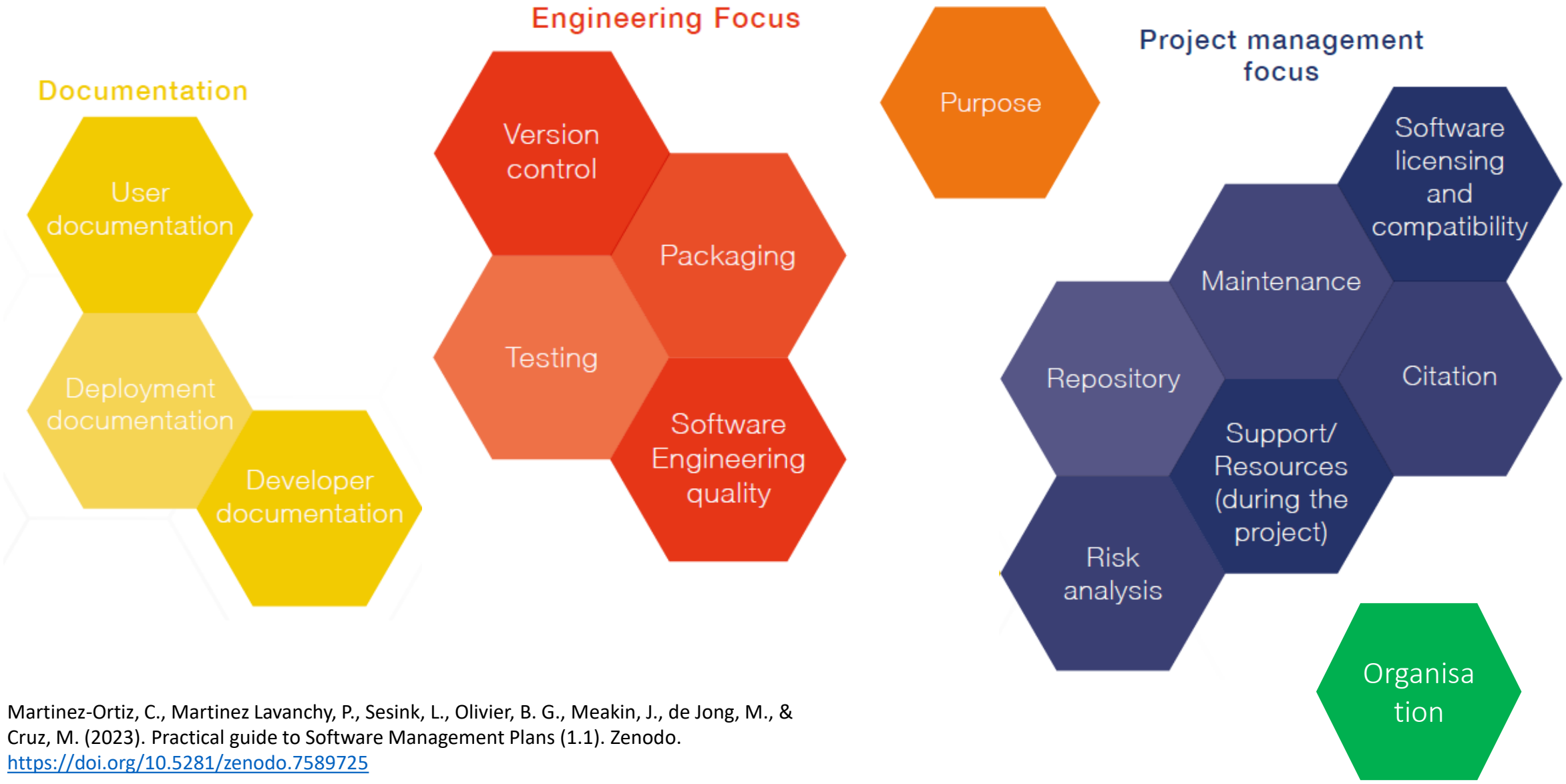
***Software components** (e.g., operating systems, libraries, dependencies, packages, scripts, etc.) that are **used for research but were not created during** or with a clear research intent should be considered **software in research** and not Research Software.*

Definition from FAIR for Research Software Working Group (a joint initiative of RDA, ReSA and FORCE11)

# Why is it important?

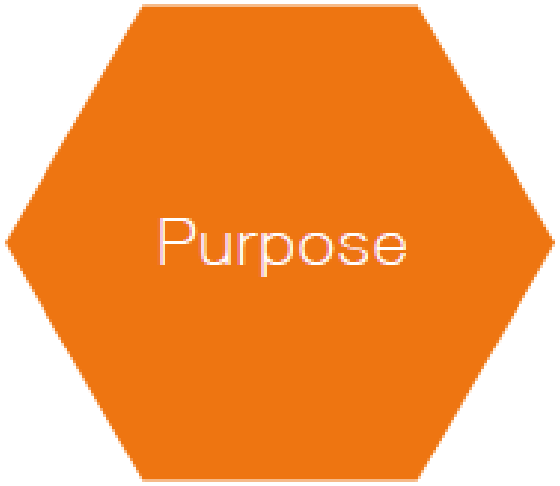
- It explains why the software is necessary and beneficial for the research communities.
- By adding context, it make the research software reusable and sustainable.
- It's a **project management tool** (just like DMPs) → it helps you plan for the necessary resources
- All research software, **whether open or closed source** (for example, in case of security concerns or commercial interests), can benefit from using SMPs.

# What is in a SMP?




Martinez-Ortiz, C., Martinez Lavanchy, P., Sesink, L., Olivier, B. G., Meakin, J., de Jong, M., & Cruz, M. (2023). Practical guide to Software Management Plans (1.1). Zenodo. https://doi.org/10.5281/zenodo.7589725

# Purpose




> What is the software used for?

> Who is it designed for?

Source: Practical guide to Software Management Plans (1.1).
https://doi.org/10.5281/zenodo.7589725

# Engineering


**Version control:** Which version control system are you using?

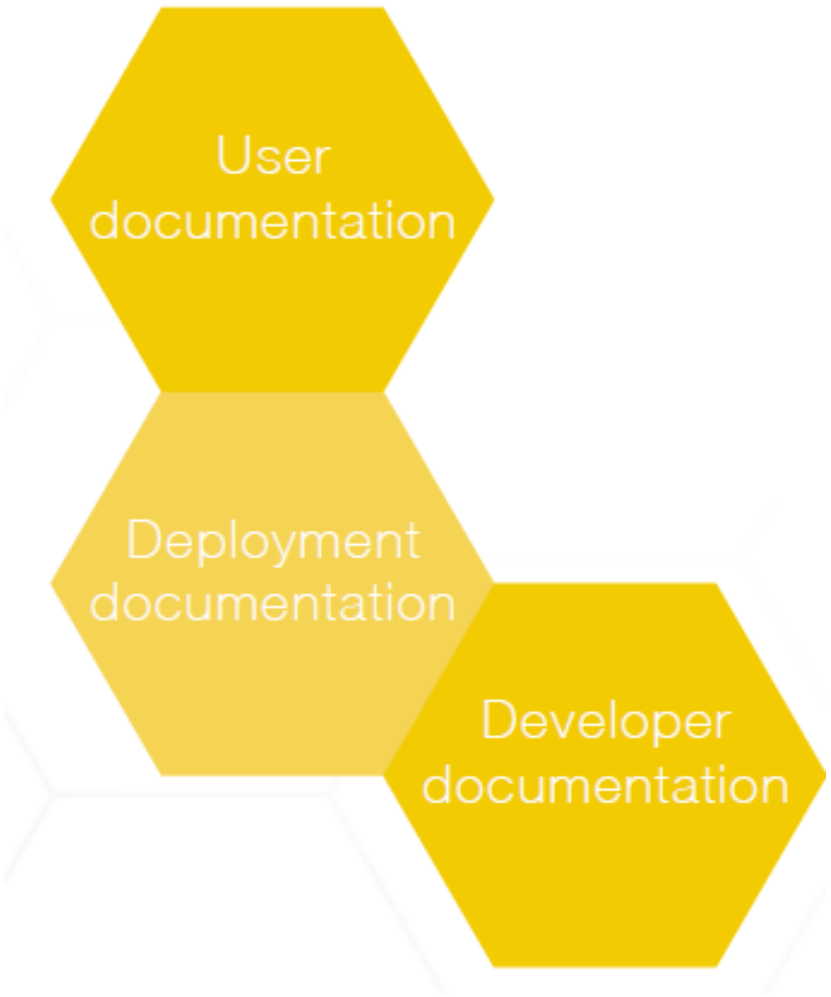
**Packaging:** Do you use package managers to allow users to install/deploy your software? What package format?

**Testing:** What types of tests are you using to ensure your software continues to work as intended? (unit, functional, integration, linting, typing, regression, etc.)

**Software Engineering quality:** What measures were/are in place to ensure code quality?

Source: Practical guide to Software Management Plans (1.1). https://doi.org/10.5281/zenodo.7589725

# Documentation




**User documentation:** How can the software be used?

**Deployment documentation:** How can the software be installed?

**Developer documentation:** How can the software be modified and tested?

Source: Practical guide to Software Management Plans (1.1). https://doi.org/10.5281/zenodo.7589725

# Project management


Source: Practical guide to Software Management Plans (1.1). https://doi.org/10.5281/zenodo.7589725

**Software licensing and compatibility:** What is the license of your software?

**Maintenance:** What type of maintenance is in place? Whenever suitable, develop a retirement strategy for your software.

**Citation:** Did you add information indicating how your software should be cited?

**Repository:** In which repository will you deposit releases of your software? Does it provide a PID?

**Support (during the project):** What are the resources necessary to reach the level of service provided by your software? HR, training, hardware, etc.

**Risk analysis:** Are there other factors that could have an impact on your software? (compliance with privacy policies, security considerations, portability, etc.)

# Organisation


> Who is in charge of the software (governance, consortium, etc.) ?

> List the team members. Who is responsible for what?

> Who is in charge of the SMP?

# A tool to write a SMP

DMP OPIDoR

Based on **DMP Roadmap** by
Digital Curation Centre and the University of
California Curation Centre, and customized by INIST-CNRS for French researchers

Several DMPs templates (ANR, Horizon Europe, ERC,…) and **2 SMP templates :**

> Research Software Management Plan template
(PRESOFT project)
> SSI Software Management Plan template

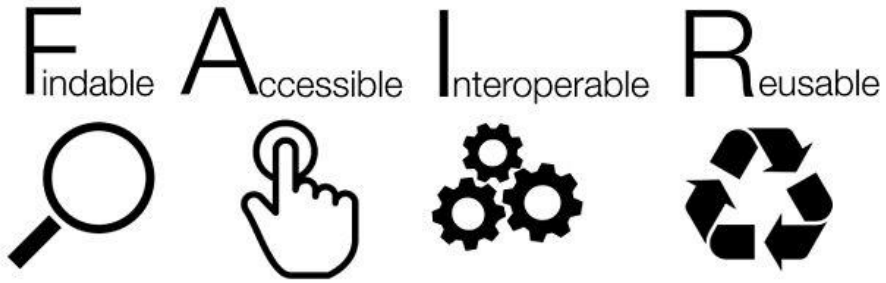
**dmp.opidor.fr**

# FAIR for Research Softwares

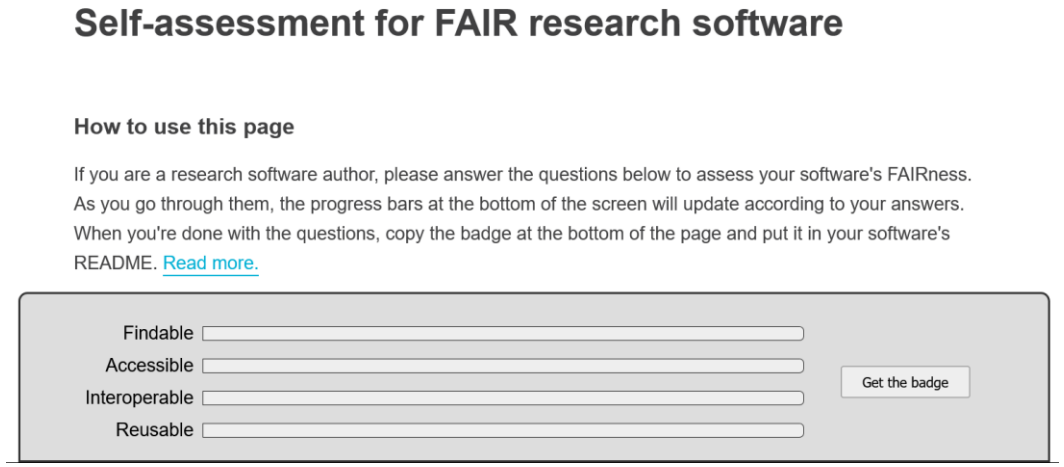
RESEARCH DATA ALLIANCE

**FAIR Principles for Research Software (FAIR4RS Principles) [10.15497/RDA00068](https://doi.org/10.15497/RDA00068)**

The FAIR for Research Software Working Group was jointly convened as a Research Data Alliance (RDA) Working Group, FORCE11 Working Group, and Research Software Alliance (ReSA) Task Force.








https://fairsoftwarechecklist.net/v0.2/

# Preservation of softwares and source codes

HAL science ouverte

https://hal.science

**Deposit** your source code files on HAL

http://doc.hal.science/deposer/deposer-le-code-source/

**Archive** your source code files and softwares on SH

Software Heritage

THE GREAT LIBRARY OF SOURCE CODE

https://www.softwareheritage.org

# Resources

> Martinez-Ortiz, C., Martinez Lavanchy, P., Sesink, L., Olivier, B. G., Meakin, J., de Jong, M., & Cruz, M. (2023). Practical guide to Software Management Plans (1.1). Zenodo. https://doi.org/10.5281/zenodo.7589725

> Zotero library about Software Management Plan : https://www.zotero.org/groups/4684302/software_management_plan_smp/library

> Citation File Format (CFF) : https://citation-file-format.github.io/

> The CodeMeta Project : https://codemeta.github.io/

**Webinar (in french) Friday 21 June (13:00 to 14:00) about CodeMeta and metadata for describing a software with Jérôme Pansanel (SCIGNE, IPHC, CNRS)**

→ **more information here**

It will be recorded and will be available on: https://printempsdeladonnee.fr/ressources/

# *Thanks for your attention!*

Service des **bibliothèques**
Université de Strasbourg

adele
HELPDESK

**Stéphanie Cheviron**
Research Data Librarian
+33 (0)3 68 85 07 93
scheviron@unistra.fr