



FAIRsFAIR
Fostering Fair Data Practices in Europe

FAIRsFAIR roadshow in Nederland: FAIR and Software

Morane Gruenpeter (INRIA) & Gerard Coen (DANS)
Tuesday 2nd March, 2021
11:00–13:00 (CET)



FAIRsFAIR "Fostering FAIR Data Practices in Europe" has received funding from the European Union's Horizon 2020 project call H2020-INFRAEOSC-2018-2020 Grant agreement 831558



Our agenda today

Software in Academia

Research initiatives regarding software:

- FAIRsFAIR - *FAIR Software & Services*
- LCRDM - *Software archiving Task Group*

Pragmatic advice for following the

FAIR Software Route

& using the Software Heritage Archive

Presenters

Morane Gruenpeter

Inria  Software Heritage
THE GREAT LIBRARY OF SOURCE CODE



@moraneottilia

Gerard Coen

Data Archiving and Networked Services

DANS



@gerardcoen

Software is both for humans & machine

Full width

Home Development Documentation **Donate** login

Archive

Software Heritage

Features

Search

Downloads

Save code now

Help

52

THE MASTER IGNITION ROUTINE IS DESIGNED FOR USE BY THE FOLLOWING LEM PROGRAMS: P12, P40, P42, P61, P63.

53

IT PERFORMS ALL FUNCTIONS IMMEDIATELY ASSOCIATED WITH APS OR DPS IGNITION: IN PARTICULAR, EVERYTHING LYING

54

BETWEEN THE PRE-IGNITION TIME CHECK -- ARE WE WITHIN 45 SECONDS OF TIG? -- AND TIG + 26 SECONDS, WHEN DPS

55

PROGRAMS THROTTLE UP.

56

#

57

VARIATIONS AMONG PROGRAMS ARE ACCOMMODATED BY MEANS OF TABLES CONTAINING CONSTANTS (FOR AVEGEXIT, FOR

58

WAITLIST, FOR PINBALL) AND TCF INSTRUCTIONS. USERS PLACE THE ADRES OF THE HEAD OF THE APPROPRIATE TABLE

59

(OF P61TABLE FOR P61LM, FOR EXAMPLE) IN ERASABLE REGISTER 'WHICH' (E4). THE IGNITION ROUTINE THEN INDEXES BY

60

WHICH TO OBTAIN OR EXECUTE THE PROPER TABLE ENTRY. THE IGNITION ROUTINE IS INITIATED BY A TCF BURNBABY,

61

THROUGH BANKJUMP IF NECESSARY. THERE IS NO RETURN.

62

#

63

THE MASTER IGNITION ROUTINE WAS CONCEIVED AND EXECUTED, AND (NOTA BENE) IS MAINTAINED BY ADLER AND EYLES.

64

#

65

#

66

HONI SOIT QUI MAL Y PENSE

67

#

68

69

TABLES FOR THE IGNITION ROUTINE

70

71

#

72

NOLI SE TANGERE

73

74

P12TABLE VN 0674 # (0)

75

TCF ULLGN0T # (1)

76

TCF COMFAIL3 # (2)

77

TCF G0CUTOFF # (3)

78

TCF TASKOVER # (4)

79

TCF P12SP0T # (5)

80

DEC 0 # (6) NO ULLAGE

81

EBANK= WHICH

82

2CADR SERVEXIT # (7)

83

84

TCF DISPCNG # (11)

85

TCF WAITABIT # (12)

86

TCF P12IGN # (13)

87

88

P40TABLE VN 0640 # (0)

89

TCF ULLGN0T # (1)

90

TCF COMFAIL4 # (2)

91

TCF G0POST # (3)

92

TCF TASKOVER # (4)

93

TCF P40SP0T # (5)

Permalinks

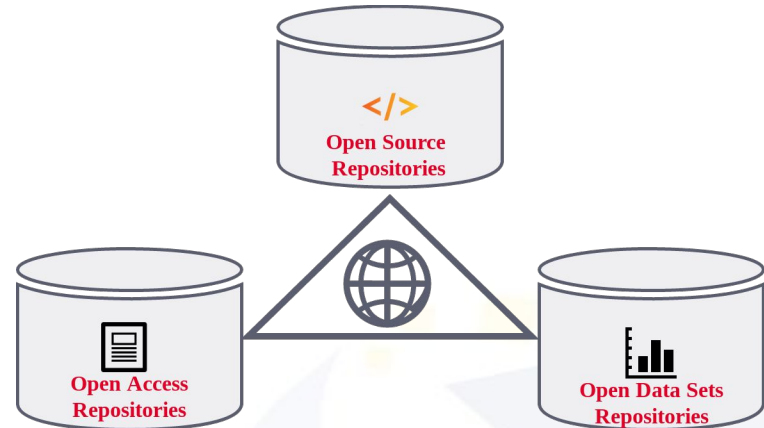
*“Programs must be written for people to **read**, and only incidentally for machines to **execute**.”*

Harold Abelson, 1985
Structure and Interpretation
of Computer Programs (1st ed.)

Software in Research: A pillar of Open Science

Multiple facets, it can be seen as:

- a **tool**
- a research **outcome** or result
- **the object** of research



*Three pillars of Open Science
Gruenpeter, Software Heritage CC-BY 4.0 2019*

Why are we here? A plurality of needs

Researchers

- **Archive and reference** software used and created in articles
- **Find** useful software
- **Get credit** for developed software
- **Verify/reproduce/improve** results

Research Organizations

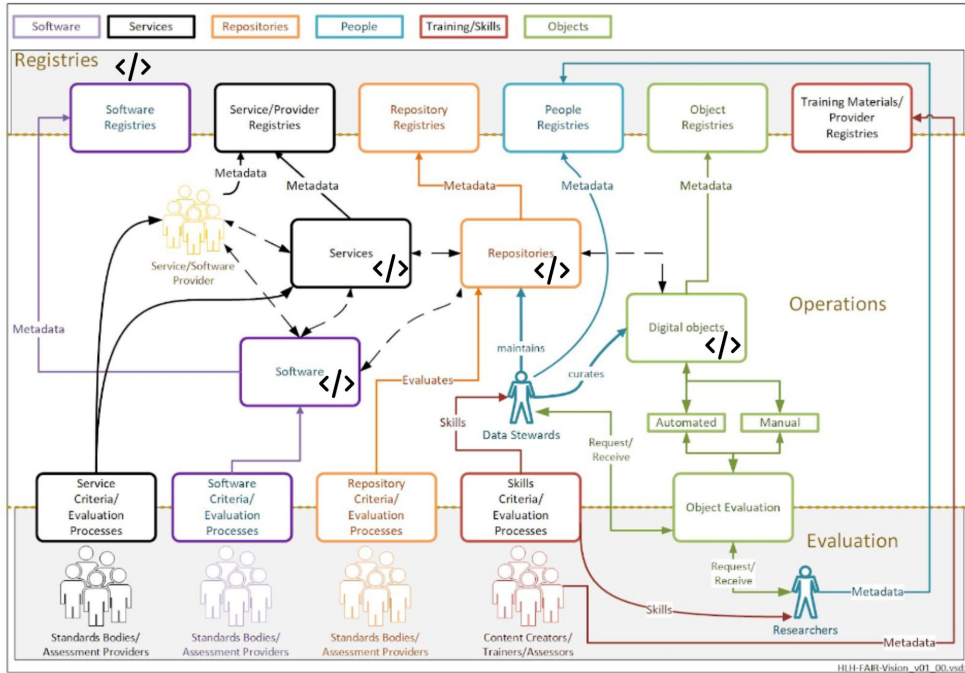
Know its **Software assets** for:

- Technology transfer
- Impact metrics
- Strategy

Laboratories /Teams

- **Track** software contributions
- **Produce** reports
- **Maintain** web page

Software in the FAIR ecosystem



“Central to the realisation of FAIR are **FAIR Digital Objects**, which may represent data, **software** or other research resources. These digital objects must be accompanied by persistent identifiers, metadata and contextual documentation to enable discovery, citation and reuse. Data should also be accompanied by the **code** used to process and analyse the data.”

Rec. 16: Apply FAIR broadly: “FAIR should be applied broadly to all objects (including metadata, identifiers, **software** and DMPs) that are essential to the practice of research, and should inform metrics relating directly to these objects.”

Turning FAIR into reality (2018)

Ecosystem components, to highlight the software roles in the Ecosystem, the symbol `</>` was added (Original diagram 3 from L’Hours & Von Stein, 2020)

Software is not just another type of data

Recommendation n°5 :

*Recognise that FAIR guidelines will require **translation for other digital objects** and support such efforts.*

2019: 'Six Recommendations for Implementation of FAIR Practice'

([FAIR Practice TF, 2020](#))

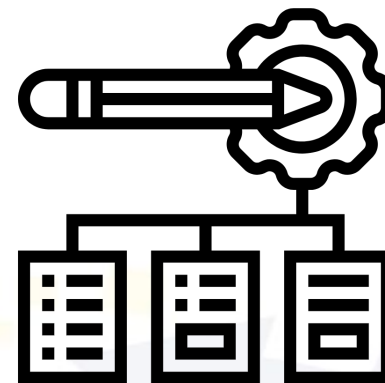
Recommendation n° 2 :

*Make sure **the specific nature of software** is recognized and not considered as “just data” particularly in the context of discussion about the notion of FAIR data.*

2019: the Opportunity Note by the French national Committee for Open Science's Free Software and Open Source Project Group
([Clément-Fontaine, 2019](#))

FAIRsFAIR Assessment report on 'FAIRness of software'

1. **State-of-the-art overview** of current solutions, challenges and practices in research software
2. **Literature review** on the application of FAIR principles to research software
3. **10 recommendations** *for the creation of* FAIR guiding principles for research software



LCRDM (National Coordination Point Research Data Management)

Supported by **SURF**

DOI: 10.5281/zenodo.4543569



researcher



support staff



funder

Authors and Contributors:

Saskia van Eeuwijk (*Initiator task group*), Miriam Roelofs (*Chair task group*),
Tom Bakker, Maria Cruz, Vera Sarkol, Barbara Vreede

Bart Aben, Patrick Aerts, Gerard Coen, Bob van Dijk, Peter Hinrich, Lena Karvovskaya, Meta Keijzer,
Jacko Koster, Jason Maassen, Jan Rijnders, Alfons Schroten, Laurents Sesink,
Chris van der Togt, Jurgen Vinju, Petra de Willigen, **Nina Noordzij**



CREATE AWARENESS

INSTITUTIONS/ INSTITUTE LEADERS	FACULTIES / RESEARCH GROUP LEADERS	RESEARCHERS
Introduce programmes to highlight stories of successfully reused research software to demonstrate its advantages	Show your researchers the advantages of reusing software by using examples of good practice	Stay informed about relevant findings, developments and recommendations from e.g. FAIR Software, relevant RDA initiatives, resa and NL-RSE

National organisations in the digital research infrastructure (eScience Center, DANS, SURF) should take on a more active and visible role in supporting research institutions in their ambitions to develop relevant and high-quality support capacity within research groups and in digital competence centres.





CREATE AWARENESS

RESEARCHERS



Stay informed about relevant findings, developments and recommendations from e.g. FAIR Software, relevant RDA initiatives, reSA and NL-RSE



researcher



<ReSA>

Research Software Alliance



DEVELOP POLICIES

RESEARCH FUNDERS	INSTITUTIONS/ INSTITUTE LEADERS	FACULTIES / RESEARCH GROUP LEADERS	RESEARCHERS
Require software developed within funded projects to be as open as possible, as closed as necessary	Develop and implement research software policies and guidelines, including archiving and licensing	Familiarise your faculty/ research group with research software policies and guidelines, including archiving and licensing	Familiarise yourself with research software policies and guidelines, including archiving and licensing



DEVELOP POLICIES



INSTITUTIONS/ INSTITUTE LEADERS

Develop and implement research software policies and guidelines, including archiving and licensing



support staff



**Develop and implement
policies and guidelines**



STIMULATE CONSIDERED REUSE OF SOFTWARE

INSTITUTIONS / INSTITUTE LEADERS	FACULTIES / RESEARCH GROUP LEADERS	RESEARCHERS
Facilitate paragraph on software management e.g. in the institutional DMP-tool Organise formal registration of research software (using pids) as a scholarly output in research information systems, similar to the registration of publications	Encourage researchers to consider which part of their research software will be suitable for reuse, and which part needs to be archived to ensure reproducibility of published results Encourage researchers to prefer the use of existing and stable software developed and supported by others over developing software from scratch	Increase visibility by enabling software citation. Deposit developed software on platforms like GitHub or Gitlab. Archive it on Zenodo or Figshare for long-term preservation and for obtaining a PID Increase your research efficiency by using software already developed by other researchers



STIMULATE CONSIDERED REUSE OF SOFTWARE



researcher



**Enable citations
of software**



RESEARCHERS

Increase visibility by enabling software citation. Deposit developed software on platforms like GitHub or Gitlab. Archive it on Zenodo or Figshare for long-term preservation and for obtaining a PID

Increase your research efficiency by using software already developed by other researchers



PROVIDE TRAINING & SUPPORT

INSTITUTIONS/ INSTITUTE LEADERS	FACULTIES / RESEARCH GROUP LEADERS	RESEARCHERS
Facilitate effective and efficient support in the field of software development, sustainability and licensing Provide research software engineering support Include guidelines and training for software development in the curriculum	Train your researchers in good practices for the development of sustainable software Hire research software engineers as part of the research team	Participate in training courses on software development Make use of research software engineering expertise



PROVIDE TRAINING & SUPPORT



support staff



Enable researchers to
develop software skills



FACULTIES / RESEARCH GROUP LEADERS

Train your researchers in
good practices for the
development of
sustainable software

Hire research software
engineers as part of
the research team



ADJUST ACADEMIC EVALUATION

RESEARCH FUNDERS	INSTITUTIONS/ INSTITUTE LEADERS	FACULTIES /RESEARCH GROUP LEADERS
Acknowledge research software as a principal form of academic output	Reward software development capabilities in your evaluation procedures, according to the new Strategy Evaluation Protocol (SEP) guidelines	Include software citations and other relevant metrics when evaluating researchers



 ADJUST ACADEMIC EVALUATION**FACULTIES /RESEARCH
GROUP LEADERS**

Include software
citations and other
relevant metrics when
evaluating researchers

**support staff**

**Enable citations
of software**



ALLOCATE FUNDING

RESEARCH FUNDERS	INSTITUTIONS/ INSTITUTE LEADERS	FACULTIES /RESEARCH GROUP LEADERS	RESEARCHERS
Explicitly address sustainable software and sustainable software development in funding instruments (e.g. in conditions and evaluation criteria) Require a paragraph on financial needs for software sustainability Develop funding instruments for covering the cost of software archiving after the project has finished	Gather information about the financial needs and required infrastructures for long-term archiving and the hosting of sustainable software	Include the cost of archiving software after the research has been completed in your funding proposals	Include the cost of archiving software after your research has been completed in your funding proposals



ALLOCATE FUNDING



funder



**Acknowledge software
and create funding lines**



RESEARCH FUNDERS

Explicitly address sustainable software and sustainable software development in funding instruments (e.g. in conditions and evaluation criteria)

Require a paragraph on financial needs for software sustainability

Develop funding instruments for covering the cost of software archiving after the project has finished

POLL #3: Which stakeholder has the biggest responsibility to enable FAIR, and sustainability, for software?

- Research Funders
- Research Infrastructures
- Research Institutions / Institute Leaders
- Faculties / Research Groups leaders
- Researchers

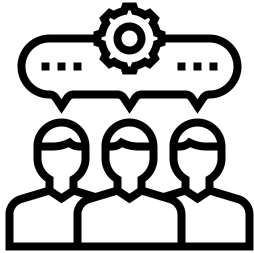




support staff



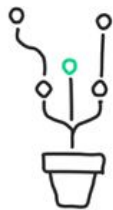
researcher



funder

Collaboration is
vital
for enabling both
FAIR Software
and Open Science

#1
USE A PUBLICLY
ACCESSIBLE
REPOSITORY
WITH VERSION
CONTROL



#2
ADD A **LICENSE**



#3
REGISTER YOUR
CODE IN A
COMMUNITY
REGISTRY



#4
ENABLE **CITATION**
OF THE
SOFTWARE



#5
USE SOFTWARE
QUALITY
CHECKLIST



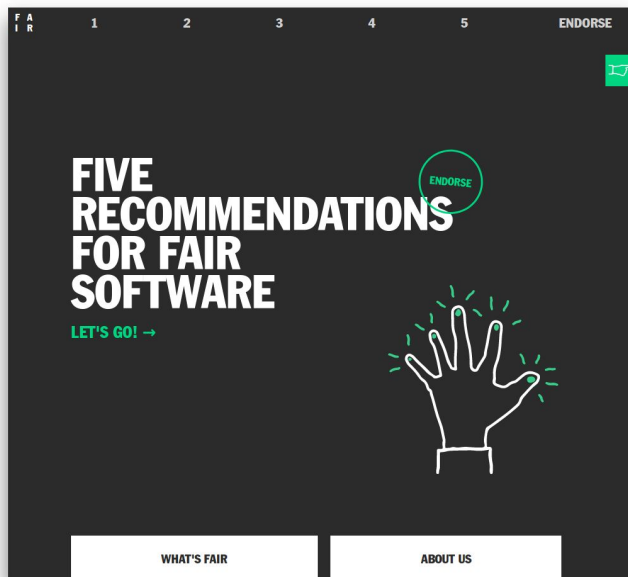
<https://fair-software.eu/>

Data Archiving and Networked Services

DANS

netherlands

& eScience center



Research
Software
Directory



Software Heritage

- 1 Prepare your public repository
README, AUTHORS & LICENSE files
- 2 Save your code
<http://save.softwareheritage.org/>
- 3 Reference your work
(full repository, specific version or code fragment)

Collect, preserve and **share** all software source code
Preserving our heritage, enabling **better software** and **better research** for all



Save your whole repository in just one click!

[Home](#) [Archive](#) [Development](#) [Documentation](#)[Donate](#)

Beta version

 Software Heritage
Archive

Archive Access



Browse



Web API

Features



Search



Vault



Save code now

Miscellaneous



Help

 **Save code now**

- **Origin type** the type of version control system the software origin is using

Currently, the only supported type is **git**, for origins using Git.

Soon, the following origin types will also be available to save into the archive:

- **hg** for origins using Mercurial
- **svn** for origins using Subversion

- **Origin url** the url of the remote repository for the software origin.

In order to avoid saving errors from Software Heritage, you should provide the correct feedback as given by the provider hosting the software origin.

Origin type

Origin url

git

Submit

Reference the code with a SWHID

≡  Browse archived content for origin <https://github.com/Unipisa/TAUmus> 

Beta version



Visits  Snapshot date: 08 October 2019, 17:49 UTC  Branches (3)  Releases (0)

Branch: **refs/heads/SourceCode** 8c34070 / SUBROUTINE_CALMUS.FOR

Raw File Select Language Actions

SUBROUTINE_CALMUS.FOR

Permalinks

1 SUBROUTINE CALMUS

2 COMMON FR,T ,R,S,INIZ,IF

3 *ALR,DURA,NOA,IP,KONT

4 DIMENSION NNN(1700),I(10),NN

5 1 FR (5000) ,T(5000) NPLLOD

6 REAL KFT(8)

7 READ(5,10)N,N1,N2

8 10 FORMAT(3I4)

9 N=4

10 LN=1

11 KK=0

12 KONT=0

13 KKONT=0

14 L=0

15 LL=0

16 KP=0

17 8 K=0

18 11 K=K+1

19 IF(K.LE.N)GO TO 12

20 LN=LN+1

21 IF(LN.GT.5)GO TO 50

22 GO TO 8

To reference or cite the objects present in the Software Heritage archive, permalinks based on persistent identifiers must be used instead of copying and pasting the url from the address bar of the browser (as there is no guarantee the current URI scheme will remain the same over time).

Select below a type of object currently browsed in order to display its associated persistent identifier and permalink.

content directory revision snapshot

✳ archived repository ✳ archived **swh:1:cnt:137968d1b65eabd6390647f95119f2eb24704962**

swh:1:cnt:137968d1b65eabd6390647f95119f2eb24704962;origin=https://github.com/Unipisa/TAUmus

☒ Add origin info ☒ Add selected lines info

Copy identifier

Copy permalink

A call for Ambassadors Software Heritage

We are looking for **enthusiastic organizations** and **individuals** to volunteer as ambassadors to help grow the Software Heritage community.

Become a **Software Heritage ambassador** to support this effort in your area by:

- **Presenting Software Heritage**, the services it provides, and its *long term mission* to key persons in your area
- **Sharing information** with relevant mailing lists, venues and contacts
- Posting Software Heritage **news** on blogs and social media
- **Identifying code hosting platforms** in your area that it is **urgent and important** to archive
- Updating the Software Heritage team with **your community's feedback**



How to get involved today?

Join the **FAIR for Research Software (FAIR4RS) Working Group**

Defining FAIR Principles for Research Software

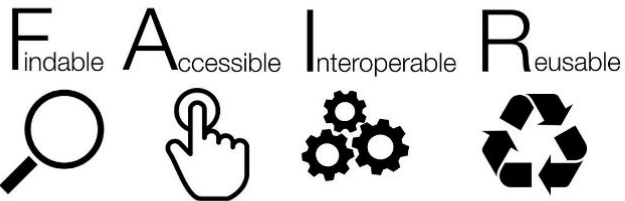
First subgroup 1 output January 2021:

[A Fresh Look at FAIR for Research Software](#)

Steering committee:

Morane Gruenpeter, Paula A. Martinez, Carlos Martinez, Michelle Barker, Daniel S. Katz, Leyla Garcia, Neil Chue Hong, Fotis Psomopoulos and Jennifer Harrow





Building bridge between communities

Software development communities
&
Research Software communities



Created by Juliet Fish
from Noun Project



Designed by Nickolas Point
from Noun Project



Software Heritage
THE GREAT LIBRARY OF SOURCE CODE

Thank you!

Questions?

*Special thanks to all of our colleagues for their slides
and materials used for this presentation!*



@moraneottilia



Software Heritage
THE GREAT LIBRARY OF SOURCE CODE



@gerardcoen

Data Archiving and Networked Services
DANS



CodeMeta initiative

- A subset of schema.org
- An academic community discussing software metadata
- A crosswalk table - mapping the metadata landscape

An open source tool to create
codemeta.json files - Contributed to the
community by



Software Heritage
THE GREAT LIBRARY OF SOURCE CODE

CodeMeta generator

Most fields are optional. Mandatory fields will be highlighted when generating Codemeta.

The software itself

Name

My Software

the software title

Description

My Software computes ephemerides and orbit propagation. It has been developed from early '80.

Creation date

YYYY-MM-DD

First release date

YYYY-MM-DD

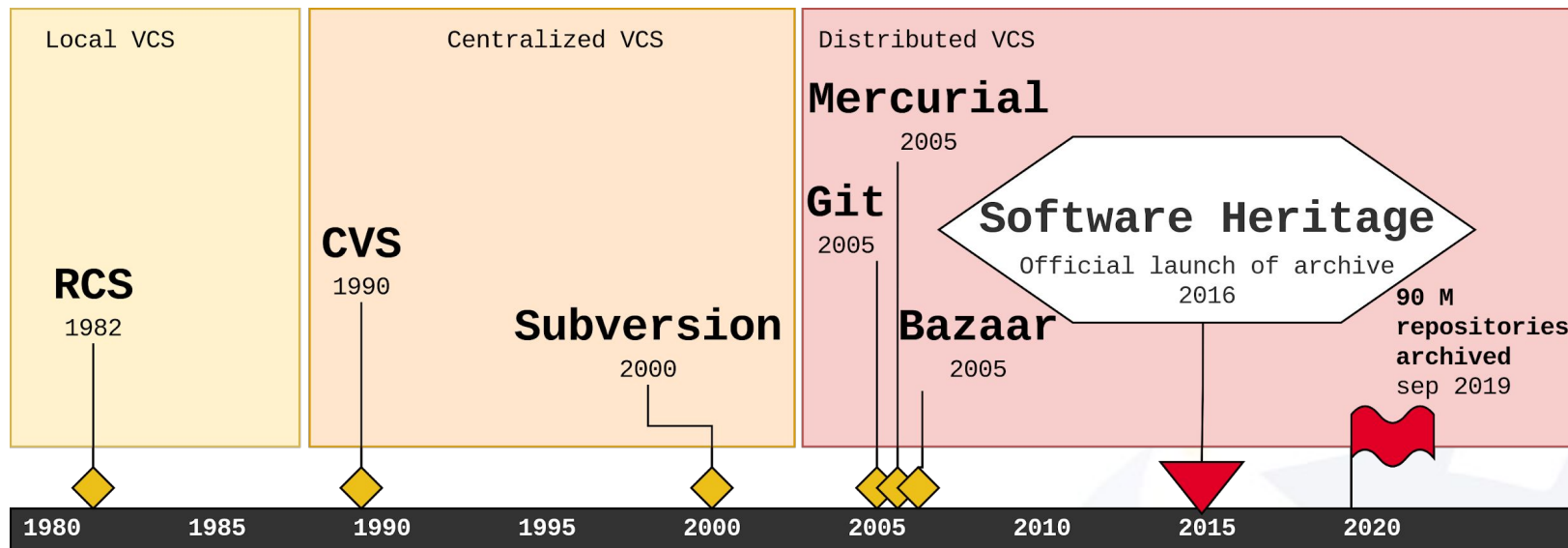
Use it directly on the CodeMeta [hosted version](#)

Contributions are welcome on the [code repository](#)

How to get involved?

1. **Participate** in the community discussions on FAIR and software
2. **Join** the [FAIR4RS](#) Working Group
 - a. Receive updates
 - b. Contribute to the subgroups work
 - c. Discuss the FAIR definition for research software
3. **Adopt** the existing infrastructures and mechanisms
4. **Spread the word** and let's start **recognizing software in academia**

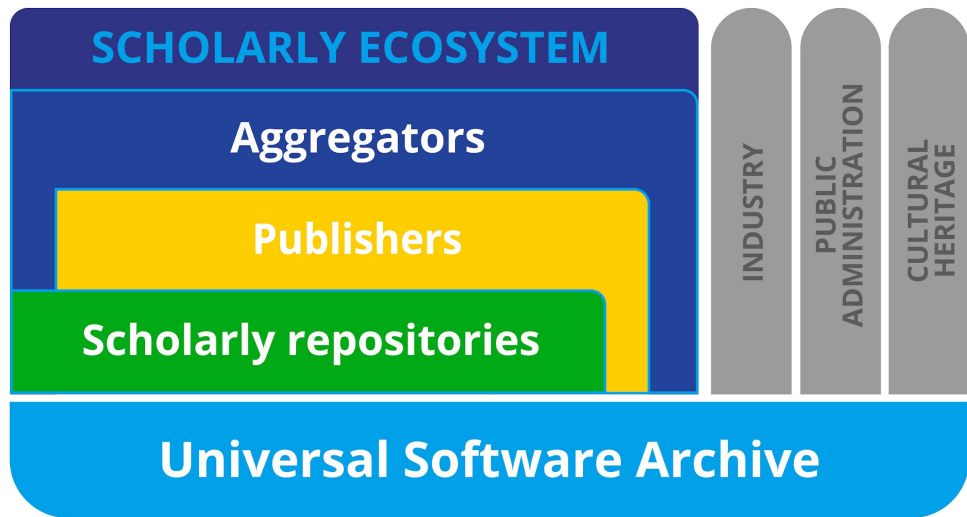
Version control system (VCS) history



- Records changes made to a (set of) source code file (s)
- Allows to operate on versions: diff/merge/fork/recover etc.
- Essential tool for software development

EOSC Scholarly Infrastructures for Research Software (SIRS)

- Focus on **Software Source Code**
- Four Pillars **Archive, Reference, Describe, Credit**
- State of the Art
 - Best Practices & Open Problems
 - Cross Cutting Concerns
- The Road ahead
 - Requirements & Criteria
 - 13 Workflows / Use Cases examples
- Recommendations
 - Standards & Tools
 - Policy recommendations
 - Long term perspectives



European Commission. Directorate General for Research and Innovation. (2020). Scholarly infrastructures for research software: report from the EOSC Executive Board Working Group (WG) Architecture Task Force (TF) SIRS. Publications Office. <https://doi.org/10.2777/28598>

EOSC SIRS's Four Pillars

« the FAIR Guiding Principles for research do not fit [software source code] well, as they were not designed for it ... »

Introducing four pillars:

[Archive]

ensure software artifacts are not lost

[Reference]

ensure software artifacts can be precisely identified

[Describe]

make it easy to discover / find software artifacts

[Credit]

ensure proper credit is given to authors

A few reasons we should pay more attention to (Research) Software:

- 1) A new 2018 OECD pilot survey, the [International Survey of Scientific Authors \(ISSA\)](#), focuses on measuring the digitalisation of science. Preliminary [data](#) analysis shows that, in the Netherlands, 35% or more of **scientific publications generate new software code**. Those reporting generation of new data is just over 45%. (Based on the responses of participating researchers.)
- 2) Research software is continually acknowledged as **a key enabler of breakthrough science** such as the highly publicized [first ever image of a black hole](#); the first detection of [gravitational waves from colliding black holes](#); and significantly [reducing the time to clinical insight](#) in drug discovery - to mention just a handful of examples.
- 3) Researchers in the Netherlands (and internationally) have **acknowledged the importance of software and software sustainability** in a number of ways. The topic features prominently in many Open Science policies.

Under the FAIR Software Open Science Pillar as part of the [TU Delft Strategic Plan Open Science 2020-2024](#) 'the project will contribute by developing and facilitating various aspects of research software, including policy, infrastructure and organizational culture' (pg.7). Chapter 4 of the [Utrecht University Open Science Programme 2018-2021](#), "Sharing code and software", states that by 2021, Utrecht University will 'invest in knowledge on software sustainability to ensure software remains functional and usable for the long term' (pg.7). The Dutch [National Plan Open Science](#) seeks to include software sustainability 'to ensure the long-term storage of the software belonging to the research data itself as well' (pg.24).

And crucially, researchers themselves are actively engaging with the topic through the establishment of grassroots communities of [Research Software Engineers](#) (RSEs) and participation in dedicated events such as the upcoming [International Workshop on Software Engineering for Computational Science](#) and the [Workshop on Sustainable Software Sustainability](#) 2019 held by DANS, in collaboration with our partners at the NLeSC and the Software Sustainability Institute.

A few reasons we should pay more **attention** to (Research) Software:

- 1) Scientific publications generate new software code. Software is an output of the system of science.
 - Research software is a **key enabler of breakthrough science**.
- 2) Researchers in the **Netherlands** have acknowledged the importance of software and software sustainability. Declarations in Open Science policies and the presence of grassroots initiatives. <- Communities & increasing importance
- 3) Just as data stewardship is undergoing a process of professionalisation as part of the drive to **fully implement the FAIR principles**, the role and relevance of research software, research software engineering, and quality software policy is also receiving increased attention. The specific characteristics of software previously mentioned make it necessary to revise and extend the original FAIR data principles. <- proliferation of initiatives and recommendations regarding software and examples of national initiatives in the NL. LCRDM group + Klesc & DANS Software Route -> The still remain at a high- mid- Level and need more granularity to be useful for researchers add some pragmatic advice.