FAIR principles for research software

Software Source Code IG

research data sharing without barriers
rd-alliance.org

18th March 2020 - RDA 15th Plenary Meeting - Day 1 Melbourne Australia

Contributors

- Michelle Barker, Research Software Alliance (ReSA), Australia
- Salvador Capella-Gutierrez, Barcelona Supercomputing Center (BSC), ELIXIR, Spain
- Neil Chue Hong, Software Sustainability Institute / University of Edinburgh, UK
- Martin Fenner, DataCite, Germany
- Leyla Garcia, ZB MED Information Centre for Life Sciences, Germany
- Carole Goble, Software Sustainability Institute UK / University of Manchester / ELIXIR, UK

continues...

Join our collaborative notes and add your name: http://tiny.cc/FAIR4ResSof

Contributors

- Morane Gruenpeter, Inria, Software Heritage, France
- Daniel S. Katz, University of Illinois at Urbana-Champaign, USA
- Mateusz Kuzak, eScience Center, The Netherlands
- Anna-Lena Lamprecht, Utrecht University, The Netherlands
- Paula Martinez, National Imaging Facility (NIF),The University of Queensland (UQ), Australia
- Fotis Psomopoulos, Institute of Applied Biosciences (INAB), Centre for Research and Technology Hellas (CERTH), Greece
- Mingfang Wu, Australian Research Data Commons, Australia

Join our collaborative notes and add your name: http://tiny.cc/FAIR4ResSof

Agenda

1. Software Source Code Interest Group (SSC IG) update
2. Updates on related projects FAIR for Research Software, Versioning and Citation
3. Activity - split into 4 different groups for discussions (depending on number of people on call):
4. Discussion
   a. Be part of this new group, you are needed.
   b. Why do we care about FAIR for research software?
   c. What should the scope for a FAIR for research software WG be?
   d. Moving forward with a FAIR4RS WG across RDA, ReSA and FORCE11

Introduction

The Software Source Code IG

Co-chairs:

- Neil Chue Hong
- Julia Collins
- Roberto Di Cosmo
- Mingfang Wu

Objectives:

A forum for discussing **research software** inside RDA

https://www.rd-alliance.org/groups/software-source-code-ig

Chronology:

RDA P10, Montreal 9/2017 motivations, survey of ontologies, metadata use cases

RDA P11, Berlin 3/2018   identification of gaps in metadata

RDA P13, Philadelphia 4/2019   **FAIR for Software Source Code** https://bit.ly/2Tt8C5L

FORCE11-2019, Edinburgh 10/2019   full day hackathon on **Research Software**

Summary from P13 session: How to make RS FAIR

Findable	identifier/identification, citation, good documentation, metadata (controlled vocabulary, licence, language, contributor), repository for software
Accessible	stable, incentive, reward, licence, citation, access libraries and packages
Interoperable	apply to software metadata (vocabulary, dependency, operating environment), compatible b/w different versions, archived in containers
Reuse	dependency, description, documentation, metadata, license
Other properties	**T**rusted, **S**ustainable, **P**reservable

Definition: Research Software

Research Software is defined by (Hettrick et al., 2014) as :

*"software that is used to **generate**, **process** or **analyse** results that you intend to appear in a publication (either in a journal, conference paper, monograph, book or thesis)"*

Software plays a triple role in research, as stated in (The Committee for Open Science's Free Software and Open Source Project Group, 2019):

- It is a tool
- It is a research result
- And it is a research object

Research Software in Open Science

FAIR doesn't necessarily mean *open*, but as stated in ([Mons et al., 2017](#)) requires clarity and transparency to **A**ccess and **R**euse conditions.

"Based on the FAIR software metrics/indicators, communities will be able to agree on degrees of **FAIRness** *that the different kinds of software should comply to, in order to reflect their Open Science ideals."* ([Lamprecht et al., 2019](#)).
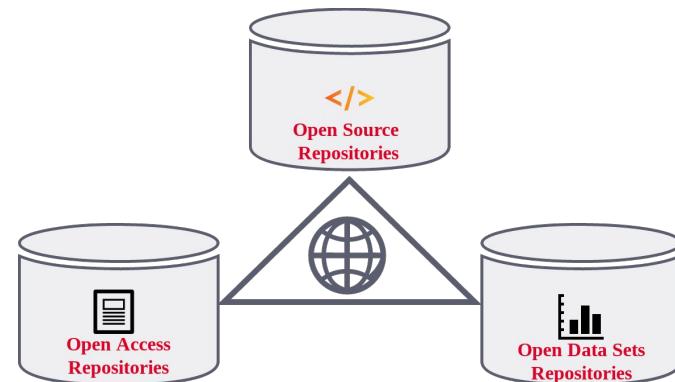
Three pillars of Open Science Gruenpeter, Software Heritage CC-By 4.0 2019

Update community on efforts around FAIR for Research Software

Related Projects - FORCE11 Software Citation

FORCE11 Software Citation Implementation Working Group
(co-chairs: N. Chue Hong, M. Fenner, D. S. Katz)

Following-on from FORCE11 Software Citation Working Group and the Software Citation Principles it developed

Objective: Produce concrete guidelines for software citation, and implement them within the scholarly research community (software developers, repositories and registries, journals and conference and publishers, indexers, institutions)

A community with monthly calls to discuss challenges and progress in implementing software citation, with task forces for

- **CodeMeta** - standardizing metadata for software, moving towards merging into schema.org
- **Guidance** - developing documents for developers, authors, and reviewer
- **Journals** - coordinating editors and publishers to simplify and implement guidance
- **Repositories** - developing best practices document for handling software
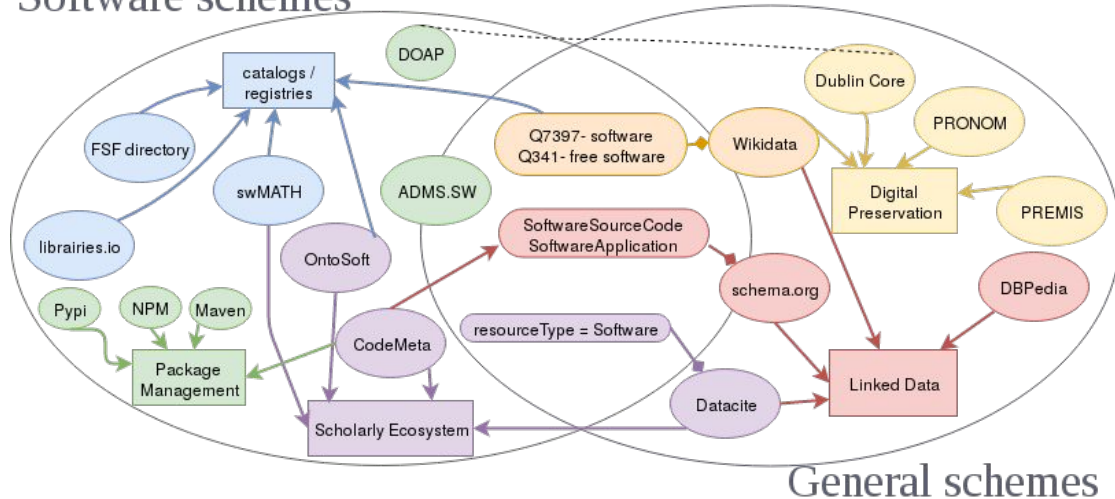
Related Projects - CodeMeta

CodeMeta initiative

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

The CodeMeta Task Force

- Part of the FORCE11 SCIWG
- **Goal**: express all codemeta properties using schema.org

Taken from RDA P11 SSC IG:
https://www.rd-alliance.org/sites/default/files/2018-03-22-R
DA-IG_0.pdf

Related Projects - Software Source Code Identification WG

Software Source Code Identification WG

Joint RDA & FORCE11 WG which spawned from RDA's Software Source Code IG & FORCE11's SCIWG to address this common challenge: how to identify source code

Objectives:

- bring together people involved/interested in software identification
- produce concrete recommendations for the academic community
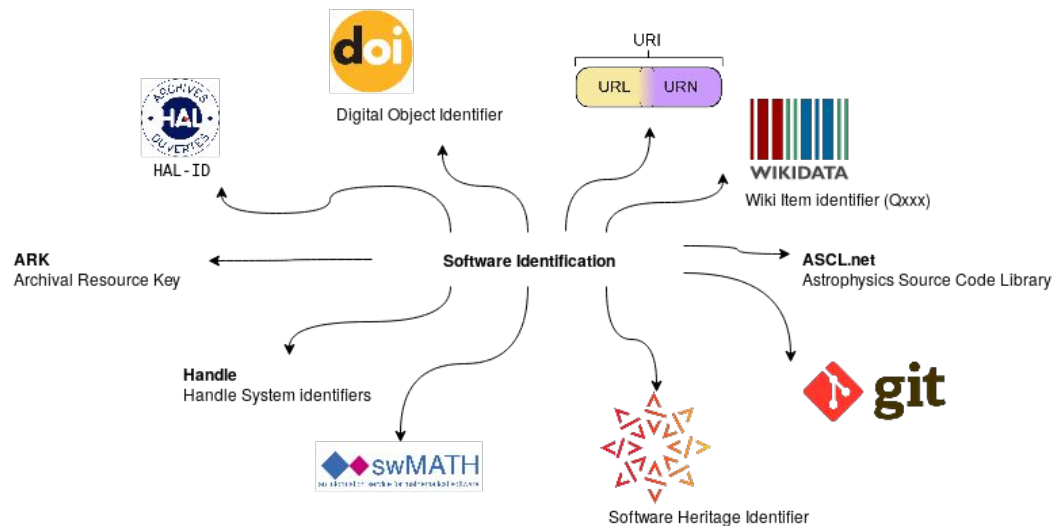
Remote meeting will be announced in the following days

Related Projects - ReSA

Research Software Alliance (ReSA)

Vision: Research software: recognised and valued as a fundamental and vital component of research worldwide

Mission: To bring research software communities together to collaborate on the advancement of research software.

<RΘSA>
Research Software Alliance

Task Forces:

- Software landscape analysis
- Evidence for the importance of research software
- Register of research software funding opportunities

RDA
RESEARCH DATA ALLIANCE

FAIR software initiatives and outcomes

"Applying FAIR Principles to Software" at the 2017 Workshop on Sustainable Software Sustainability (WOSSS17)

"Making Software FAIR" at the DTL Communities@Work 2018 Conference

"FAIRness assessment for software" at the 2018 DBCLS/NBDC BioHackathon

"Sharing Your Software – What is FAIR?" at the 2018 American Geophysical Union (AGU) Fall Meeting

Top 10 FAIR Data & Software Global Sprint, including "10 easy things to make your software FAIR" 2019

"FAIR principles for Software" at 2019 Workshop on Sustainable Software Sustainability (WOSSS19)

"FAIR Software" Birds of a Feather meeting at deRSE 2019

"Five recommendations for FAIR software" at NL-RSE 2019

TIB Training workshops on FAIR Data and Software 2018 - 2019

Towards FAIR principles for research software 2019 DOI: 10.3233/DS-190026

FAIR Computational Workflows 2020 DOI: 10.1162/dint_a_00033

FAIRsFAIR T2.4: FAIR assessment for research software

From FAIR research data toward FAIR and open research software

Lorentz Workshop 9-13 March 2020 (Automated Workflow Composition in the Life Sciences)

BRDI NAS Washington 16-17 March 2020

https://www.rd-alliance.org/fair-principles-research-software
https://www.rd-alliance.org/group/software-source-code-ig/wiki/fair4software-reading-materials

FAIR for Research Software

Motivation

- **FAIR principles**
  - Basis for reproducible open science → but more beyond FAIR is needed here
  - Mainly addressing data but aimed to any sort of digital objects → software is a digital object BUT
- **Research software is not (just) data (Katz et al., 2016; Lamprecht et al., 2019)**

Similarities	In between	Differences
- Not commonly cited - All FAIR metadata principles apply - Multiple versions can exist	- Can be built on top of (but dependencies on software are more complex) - Depend on hardware and software (data for display and production, software for that + execution) - Licenses are different (data is not a creative work, software is)	- Software is more volatile and quickly changing, it is "alive" (evolves and requires maintenance) - Reuse comes in different flavors (re-run/execute, reuse, repeat, reproduce, extend) - Can be connected via workflows

FAIR and Software

Findability	
F1	(Meta)data are assigned a globally unique and persistent identifier.
F2	Data are described with rich metadata.
F3	Metadata clearly and explicitly include the identifier of the data it describes.
F4	(Meta)data are registered or indexed in a searchable resource.

Accessibility	
A1	(Meta)data are retrievable by their identifier using a standardized communications protocol.
A1.1	The protocol is open, free, and universally implementable.
A1.2	The protocol allows for an authentication and authorization procedure, where necessary.
A2	Metadata are accessible, even when the data are no longer available.

8.2.6

Major version   Minor version   Patch

- We have versions and releases
- We have specialized registries

RDA
RESEARCH DATA ALLIANCE

FAIR and Software

Interoperability	
I1	(Meta)data use a formal, accessible, shared, and broadly applicable language for knowledge representation.
I2	(Meta)data use vocabularies that follow FAIR principles.
I3	(Meta)data include qualified references to other (meta)data.

Reusability	
R1	(Meta)data are richly described with a plurality of accurate and relevant attributes.
R1.1	(Meta)data are released with a clear and accessible data usage license.
R1.2	(Meta)data are associated with detailed provenance.
R1.3	(Meta)data meet domain-relevant community standards.

- Vocabularies work well for software metadata
- Metadata should target machine readability and data exchange
- Software product, i.e., data should be FAIRly described
- And still, what is an interoperable software? Are we talking about workflows, containers or so?

- Software dependencies and their licenses should be covered
- We also need software documentation (including use case examples)
- Re-run, reuse, repeat, what are we talking about?

RDA
RESEARCH DATA ALLIANCE

Discussion
Software is not data

notes:
ice breaker activity, 5 minutes to share your thoughts

Let's start

Full room discussion or in groups depending on how many people. 25' and 10' wrap up

- Introduce yourself to your neighbours (name, affiliation, why do research software interest you?)
- How can the FAIR principles be applicable to software ?
  - Should we apply them? Why ? What strategies?
  - What can this community do to improve FAIRness for research software?
- Parallel discussion:

Metadata	Indicators/ Metrics	Versioning	FAIR scope
What controlled vocabularies are currently available to describe software and its associated metadata?	What aspects should be taken into account in order to create indicators to measure FAIRness in software?	Software versioning and how to handle releases? A discussion about software provenance.	Why do we care about FAIR for software? What is the higher-level goal of FAIR? Is FAIR by itself going to help us get there?

research data sharing without barriers
rd-alliance.org

RDA
RESEARCH DATA ALLIANCE

Why FAIR for Research Software?

- Do we need a Working Group on FAIR for Research Software?
- What would the scope of the WG be?
  - Source code, Executables, Containers, All.
  - Domain specific, Scientific, All research software
  - What is out of scope?
- What other RDA WGs or IGs should we collaborate or connect to?
- What kind of outputs can this WG produce?
  - Recommendations? Guidance?
  - Assessment tool?

Let's work collaboratively on the notes: http://tiny.cc/FAIR4ResSof

Collecting information

Let's work collaboratively on the notes: http://tiny.cc/FAIR4ResSof

- Names, institutions, emails, level of involvement (1-3 High to low)
- Next steps
- What else should be added to the Wiki of materials?
    - https://www.rd-alliance.org/group/software-source-code-ig/wiki/fair4software-reading-materials

Task force aims

- Agree what's in scope for FAIR for software
- Define FAIR for software - what are the sub-elements of each letter
- Potentially define metrics of how this is measured, or even stages of adoption (minimal, good, best) for each subelement and each letter
- Coordinate community consultation to reach agreement of FAIR for software principles
- Identify implementation approaches for the principles
- Identify funding mechanisms to support this work

What else?

A community taskforce, supported by:

References

- Hettrick, S., Antonioletti, M., Carr, L., Chue Hong, N., Crouch, S., De Roure, D., Emsley, I., Goble, C., Hay, A., Inupakutika, D., Jackson, M., Nenadic, A., Parkinson, T., Parsons, M.I., Pawlik, A., Peru, G., Proeme, A., Robinson, J., Sufi, S., 2014. UK Research Software Survey 2014. https://doi.org/10.5281/zenodo.14809
- Klump, J., Wyborn, L., Downs, R., Asmi, A., Wu, M., Ryder, G., & Martin, J. (2020). **Principles and best practices in data versioning for all data sets big and small**. Research Data Alliance. DOI: 10.15497/RDA00042.
- The Committee for Open Science's Free Software, Open Source Project Group, 2019. Opportunity Note: **Encouraging a wider usage of software derived from research.** URL https://www.ouvrirlascience.fr/opportunity-note-encouraging-a-wider-usage-of-software-derived-from-research (accessed 2.28.20).
- Mons, B., Neylon, C., Velterop, J., Dumontier, M., da Silva Santos, L.O.B., Wilkinson, M.D., 2017. Cloudy, increasingly FAIR; **revisiting the FAIR Data guiding principles for the European Open Science Cloud. Information Services & Use** 37, 49–56. https://doi.org/10.3233/ISU-170824
- Lamprecht, A.-L., Garcia, L., Kuzak, M., Martinez, C., Arcila, R., Pico, E.M.D., Angel, V.D.D., Sandt, S. van de, Ison, J., Martinez, P.A., McQuilton, P., Valencia, A., Harrow, J., Psomopoulos, F., Gelpi, J.L., Hong, N.C., Goble, C., Capella-Gutierrez, S., 2019. **Towards FAIR principles for research software.** Data Science. https://doi.org/10.3233/DS-190026 https://drive.google.com/file/d/1sggbc67Sp6R3zbDe5fDq8x6oDqjoiUr8/view
- Katz, D.S., Niemeyer, K.E., Smith, A.M., Anderson, W.L., Boettiger, C., Hinsen, K., Hooft, R., Hucka, M., Lee, A., Löffler, F., Pollard, T., Rios, F., 2016. **Software vs. data in the context of citation** (No. e2630v1). PeerJ Inc. https://doi.org/10.7287/peerj.preprints.2630v1
- Wilhelm Hasselbring, Leslie Carr, Simon Hettrick, Heather Packer, Thanassis Tiropanis **From FAIR research data toward FAIR and open research software** https://doi.org/10.1515/itit-2019-0040.
- fair-software.eu - endorsed by several organizations
- The Software Heritage research software guidelines https://www.softwareheritage.org/save-and-reference-research-software/
- Pierre Alliez, Roberto Di Cosmo, Benjamin Guedj, Alain Girault, Mohand-Said Hacid, Arnaud Legrand, Nicolas Rougier. *Attributing and Referencing (Research) Software: Best Practices and Outlook From Inria.* Computing in Science Engineering, 22 (1), pp. 39-52, 2020, ISSN: 1558-366X. https://dx.doi.org/10.1109/MCSE.2019.2949413
- Roberto Di Cosmo, Morane Gruenpeter, Stefano Zacchiroli. *Referencing Source Code Artifacts: a Separate Concern in Software Citation.* Computing in Science & Engineering, 2020, ISSN:1521-9615. https://dx.doi.org/10.1109/MCSE.2019.2963148
- Di Cosmo, Roberto; Gruenpeter, Morane; Marmol, Bruno; Monteil, Alain; Romary, Laurent; Sadowska, Jozefina. *Curated Archiving of Research Software Artifacts: lessons learned from the French open archive (HAL)* Slides from IDCC 2020 presentation at https://doi.org/10.5281/zenodo.3667713, full article https://hal.archives-ouvertes.fr/hal-02475835.

research data sharing without barriers
rd-alliance.org

Thanks for joining