

## FAIR Principles for Research Software (FAIR4RS Principles) Draft Document for Community Input

The commenting period on this document has now closed. Responses to comments are being added as the FAIR4RS principles editors address them.

This public document is for community consultation on the draft of the FAIR4RS Principles developed by the [FAIR4RS](#) Working Group between April 2021 to May 2021. We recommend you [become a member of the FAIR4RS Working Group](#) before collaborating on this document.

This document presents a first draft of the FAIR4RS Principles, and includes: a) the aims of the principles and previous related work; b) notes on how these principles were developed; c) the draft principles; and, d) challenges to implementation and adoption.

### How to contribute

This document will remain open for collaborators to respond to the options and questions posed, propose revisions to the text, or add comments from 17 May 2021 until 30 May 2021.

Please add your name to the list of contributors on the next page.

If adding new information, please make sure to cite your sources in the [References section](#) at the end of the document.

As an online global and diverse community, we expect professional behaviour. Your contributions are valued by the community. We ask that you help others feel equally valued and welcomed by treating others with the respect and professionalism with which you would like to be treated. Please adhere to the [RDA Code of Conduct](#).

### What happens next

Following the two week consultation process in May, the feedback received will be acted upon to produce a revised version of the document. This will be submitted for formal Community Review, expected to take place between 8 June 2021 and 8 July 2021.

## Contributors

If you make a contribution (by providing new or revised text, or feedback) and you would like to be credited for it, please add your details to this table.

If you have collaborated before please also check that your information is correct in [Appendix C](#)

Full name	Your role, organisation, location, discipline, and ORCID
Malin Sandström	Community Engagement Officer, INCF, Stockholm Sweden, Neuroinformatics & Computational Neuroscience ORCID ID: 0000-0002-8464-2494
Nadica Miljković	Associate Professor, University of Belgrade - School of Electrical Engineering, ORCID ID: 0000-0002-3933-6076
Susanna-Assunta Sansone	Associate Director, Oxford e-Research Centre; Associate Professor, Dep of Engineering Science; Group Leader, Data Readiness, University of Oxford, UK. 0000-0001-5306-5690
Allen Lee	Associate Research Professional, School of Complex Adaptive Systems, Arizona State University; Co-director, CoMSES.Net ORCID: 0000-0002-6523-6079
<a href="#">Alexander Struck</a>	
<a href="#">Esther Plomp</a>	
<a href="#">Patricia Herterich</a>	
<a href="#">Nick Lynch</a>	
<a href="#">Limor Peer</a>	
<a href="#">Nadica Miljkovic</a>	
<a href="#">Anna-Lena lamprecht</a>	
<a href="#">Axel Loewe</a>	
<a href="#">Mathieu Servillat</a>	
<a href="#">Morane Gruenpeter</a>	

**Commented [1]:** As noted by email

**Commented [2]:** Check if contribution to be recognised in deposited version.

**Commented [3]:** Check if contribution to be recognised in deposited version.

**Commented [4]:** Check if contribution to be recognised in deposited version.

**Commented [5]:** already added

<a href="#">Neil Chue Hong</a>	
<a href="#">Paula Andrea Martinez</a>	
<a href="#">Tom Honeyman</a>	

# First Draft of the FAIR for Research Software Principles (FAIR4RS Principles)

## Authors

1. Editors
2. Drafting Group + Steering Committee
3. Significant Contributors (regular and significant contributions to WG meetings, authored significant text)
4. Contributors (provided significant feedback or input on draft documents, significant contributions to subgroups)
5. FAIR4RS WG (as an entity)

*The author list and ordering is determined by the contributions each author has made to the document. The full list of contributors and their contributions can be found in [Appendix C - Contributor List](#).*

*If you make a contribution (by providing new or revised text, or feedback) and you would like to be credited for it, please add your details to the table at the **start** of this document.*

## Abstract

To be completed

Date	Version Number	Description	Editor(s)
17/5/2021	0.1	First draft for review by FAIR4RS WG	Neil Chue Hong, Michelle Barker

# Table of Contents

<b>Introduction</b>	6
<b>Aims</b>	6
<b>Previous work</b>	7
<b>Development of the FAIR4RS principles</b>	8
<b>Draft FAIR principles for research software</b>	9
Findable	11
Accessible	14
Interoperable	16
Reusable	17
<b>Challenges to implementation</b>	19
<b>Challenges to adoption</b>	20
<b>Acknowledgements</b>	21
<b>References</b>	21
<b>Appendices</b>	23
Appendix A - Additional Figures	23
Appendix B - Comparison of FAIR principles	25
Appendix C - Contributor List	29

## Introduction

This document presents the first draft of the application of the [FAIR principles](#) to research software, for review by the research software community. This work is an outcome of the FAIR for Research Software (FAIR4RS) Working Group (WG) that convened in 2020 with the aim of developing community-endorsed FAIR principles for research software, which is shortened to the FAIR4RS principles in the remainder of this document.

Formatted: Font: Not Italic

Formatted: Font: Not Italic

This document begins with an explanation of the history of this work. The draft FAIR4RS principles are then provided, alongside comparison with the equivalent FAIR principle if appropriate, and explanation of any changes. Further details are provided on choices made and why, and alternative options that were not utilized. Finally, this document concludes with discussion on the challenges in using and implementing these principles, such as gaps in existing infrastructure and standards that make it hard to follow the principles, and work being undertaken to develop an implementation roadmap to address these.

Extensive community consultation is needed to ensure that the principles developed support the widest possible range of use cases. Feedback is sought on this document from stakeholders including people who use research software, write and/or maintain research software, create/implement policy around research software and other research outputs, manage infrastructure that supports usage and/or development of research software and/or other research outputs, fund research software and/or other research outputs, and others with an interest in the FAIR4RS principles. This feedback will be used to continue to evolve the principles, which will be submitted for formal community review by mid-2021.

## Aims

The [FAIR4RS WG](#)<sup>1</sup> is jointly convened by the Research Software Alliance (ReSA), Future Of Research Communications and E-Scholarship (FORCE11) and the Research Data Alliance (RDA). The FAIR4RS WG is a global and interdisciplinary community composed of 200479+ people who have an interest in the application of FAIR principles to research software and other research outputs, such as software users, software developers and maintainers, policy makers, infrastructure support staff and funders.

Commented [6]: we keep growing

Commented [7]: Change is accepted in next version

The FAIR4RS WG aims to define and publish community-endorsed FAIR4RS principles by mid-2021, followed by adoption guidelines and use cases, and this document is part of this process. This effort requires extensive consultation with the research community to create the principles and encourage their adoption. The resulting adoption and implementation of FAIR4RS principles will create significant outcomes for many stakeholders, ranging from increased

---

<sup>1</sup> FAIR4RS Working Group: <https://www.rd-alliance.org/groups/fair-4-research-software-fair4rs-wg>

research reproducibility for research organizations, to clarity for funders around their own requirements for software investments, and guidelines for publishers on sharing requirements.

## Previous work

It is useful to understand the origins of the concept of FAIR when considering its application to software, and the FAIR4RS WG builds on previous efforts for both FAIR in general, and FAIR research software.

The concept of FAIR originated in the Netherlands during the 2014 Lorentz Workshop "Jointly Designing a Data FAIRport", where participants formulated the FAIR data vision to optimize data sharing and reuse by humans and machines. This vision supports existing communities that try to realize and enable a situation where valuable scientific data is 'FAIR' in the sense of being Findable, Accessible, Interoperable and Reusable. The resulting Guiding Principles can be viewed and commented on ([FORCE11, 2015](#)) and resulted in the publication of The FAIR Guiding Principles for scientific data management and stewardship ([Wilkinson et al., 2016](#)).

The FAIR principles were ~~originally~~ intended to be ~~applicable~~~~applied not only~~ to ~~many kinds of digital assets, not just datasets, but to other digital objects as well~~. A number of ~~research communities and groups resources and events~~ have been considering how to apply aspects of FAIR to research software since 2017. Community produced outcomes before February 2020 can be found in the Software Source Code identification Interest Group's [Wiki FAIR4Software reading resources](#)<sup>2</sup>. Newer resources can be found in the [FAIR4RS collection on Zenodo](#) and the literature review completed by the FAIR4RS subgroup ([FAIR4RS WG, 2021](#)). General work on FAIR ~~in~~ has also recognized the need to incorporate other digital objects, and ~~some recent works~~ have specifically highlighted the need for inclusion of software (e.g., [European Commission, 2018](#), [European Commission & EOSC Executive Board, 2020](#)).

The FAIR4RS WG coordinated four subgroups from July 2020 to March 2021 to provide outputs to support the development of the FAIR4RS principles:

- [A fresh look at FAIR for Research Software](#) examined the FAIR principles in the context of research software from scratch, not based on pre-existing work ([Katz, Gruenpeter & Honeyman, 2021](#)).
- [FAIR work in other contexts](#) examined efforts to apply FAIR principles to different forms including workflows, notebooks and training material, to provide insights for the definition and implementation of FAIR principles for research software.
- [Defining Research Software: a controversial discussion](#) reviews existing definitions of research software in order to provide the overall context of the subgroup outputs ([Gruenpeter et al., 2021](#)).

<sup>2</sup> FAIR4Software Reading Resources: <https://www.rd-alliance.org/group/software-source-code-ig/wiki/fair4software-reading-materials>

**Commented [8]:** maybe rather 'from the outset' or something like that ('originally' sounds to me as if it has changed in the meantime)

**Commented [9]:** +1  
"even then" could be another option

**Commented [10]:** +1

**Commented [11]:** Suggestion is accepted in next version, along with Allen Lee's suggested rephrasing of this sentence.

**Commented [12]:** is it worth adding RDA COVID-19 Recommendations and Guidelines on Data Sharing <https://www.rd-alliance.org/group/rda-covid19-rda-covid19-omics-rda-covid-19-epidemiology-rda-covid19-clinical-rda-covid19-0>

**Commented [13]:** I wasn't sure about this - I've relooked through the document, and it doesn't include a specific observation that the FAIR principles need to be extended to include software (as the other two references do).

- [Review of new research related to FAIR Software](#) reviewed new research around FAIR software that has come out since the release of “Towards FAIR principles for research software” ([Lamprecht et al., 2020](#)) and reviewed the principles set out in that paper.

The work of the subgroups was brought together and presented for consultation by the wider FAIR4RS community ([Katz, Chue Hong, Barker & Gruenpeter, 2021](#)).

## Development of the FAIR4RS principles

This section explains the context within which the FAIR4RS principles should be understood, based on the input received by the FAIR4RS WG and subgroups, as part of the first community consultation, and the subsequent discussion and development. In proposing this draft of the FAIR principles for research software, the intent and methods of the FAIR principles were taken as the starting point: to “maximize the added value gained by contemporary, formal scholarly digital publishing” and “to ensure transparency, reproducibility, and reusability.” Importantly, it is acknowledged that the foundational principles of *Findable, Accessible, Interoperable, and Reusable* may need to be reinterpreted to both stay true to the goals of the FAIR principles but also ensure that they are applicable to software. It is also recognized that the goal of the FAIR principles is to support both human-driven and machine-driven activities.

[The Principles are aspirational, and](#) FAIR is not binary, nor can it be compared numerically. Something is not FAIR or unFAIR. The number of principles that something implements is not an indicator of how FAIR something is. For example, in a perfectly FAIR world, [third-party code](#) that FAIR software has dependencies on would ideally be FAIR as well. But, because [software consists of large stacks of interdependent components](#), any definition of metrics and indicators of FAIR for software can only be made in the context of specific components with which it is designed to work.

The definition of software can include source code, executables or [other forms that make sense, for example, containerized environments](#). Often the source code is the most useful form to understand the software, and to make the software FAIR.

FAIR should be applied to things which are [externalized](#), not internal only. These are variously referred to in other literature as research objects, research outputs, research artefacts, research assets, digital research objects, digital assets, digital research artefacts, non-data assets, scholarly outputs, digital objects, etc. All facets [or manifestations of research software](#) should be [considered](#) FAIR, including [software as a tool, a research outcome or result, or being the object of research](#). [To illustrate this point: researchers utilize software as a means to achieve data-driven results, while others invent a new algorithmic implementation and again other researchers are interested in the performance or security of software.](#)

The [achievement of FAIR software](#) should not be considered synonymous with long-term preservation of software. Software has a wide range of useful lifetimes, and the findability, accessibility, interoperability and reusability will degrade over time if the software is not

**Commented [14]:** This is a direct quotation from the 2016 FAIR Guiding Principles paper, so I propose keeping the hyphen.

**Commented [15]:** ok then

**Commented [16]:** Good spot!

**Commented [17]:** I am not sure I understand what this means, but this is not unique to software. I mean, many datasets are compiled from a number of datasets components, of different types, and their FAIRness will most probably vary too.

**Commented [18]:** Here, we were trying to move away from one line of thought that came up in the first consultation that suggested that software could only be FAIR if its dependencies were FAIR but this leads to a ...

**Commented [19]:** it implies the other component are not relevant to FAIRness, which seem in ...

**Commented [20]:** imho, irrelevance of the other forms is not implied here, but I think the view is too narrow: ...

**Commented [21]:** Yes, this is a little bit confusing - in shortening the original text, some of the meaning was ...

**Commented [22]:** to software? I'm not entirely following this sentence

**Commented [23]:** Yeah, I agree. I'm guessing that they mean associated or referenced objects/materials?

**Commented [24]:** While I think I grasp the intended meaning I'd prefer something like 'published (to some ...

**Commented [25]:** The intent was indeed "made external" in the context of differentiating between ...

**Commented [26]:** not sure if the statement would be clearer when dropping the "considered"

**Commented [27]:** I'm not sure what the sentence actually means, to be honest.

**Commented [28]:** I'm with Axel here and suggest to drop 'considered'.

**Commented [29]:** yes the whole para seems out of place here, and indeed not sure what it means.

**Commented [30]:** Thanks for pointing this out - this paragraph has been rewritten in the next version to ...

**Commented [31]:** I am really struggling to understand this whole para and its message, sorry

**Commented [32]:** Agreed. I think it's trying to convey something along the lines of "The FAIR principles ...

**Commented [33]:** Thanks - this has been rewritten to make this clearer in the next version.

**Commented [34]:** do you mean: making a software FAIR does not mean preserving it long term?

**Commented [35]:** What we meant was that making something FAIR doesn't require it to be preserved for ...



maintained. The application of the FAIR principles to software is still important for many other reasons and, as a by-product, "record keeping" (at a level below digital curation for preservation) can still be achieved.

The application of the FAIR principles is the responsibility of the owners (who are often the creators) of the software, not the users. We want to emphasize the concept of FAIR at the first mile, at the source, because -as for data- those producing the software are best placed to ensure they provide all the necessary information to make their work as FAIRer as possible, and get credited for doing so. These draft FAIR4RS principles can be applied to any software used in research, but there is no value yet in trying to define where the "boundary" might be, e.g. closed vs open, commercial vs not-for-profit, research software vs academic software vs non-academic software.

## Draft FAIR principles for research software

In this section, each of the draft FAIR4RS principles is proposed and explained. First, the foundational principle (F, A, I and R) is described, followed by the numbered guiding principles used to interpret the foundational principle.

- Text in **bold** is the draft text for the principle.
- Text in *italics* is the draft narrative text explaining the intent behind the phrasing of the principles and providing guidance for how they should be interpreted.
- Text that is underlined are options to be discussed.

A key challenge of defining the FAIR principles for research software is the balance between general principles vs actionable principles. As for the FAIR Principles, also our principles are aspirational. For this reason indicators are needed, not just to document the level of FAIRness of the software in a given moment in time, but also to provide a clear and measurable path to progressively improve their FAIRness. How a principle can be implemented in an actionable way will be described in the guidance that will be produced as the next step of the FAIR4RS WG schedule.

Therefore, the draft FAIR4RS principles that follow may include options and questions that generally fall into two categories:

1. Finding the balance between general, more abstract, principles that capture the ethos of FAIR vs specific principles that point to the means of implementation
2. Sticking closely to the FAIR data principles or reinterpreting the FAIR ethos for software

**Commented [36]:** By whom? long-term preservation and record keeping is often not the focus of the original creators; library and information professionals can help achieve these goals with curation.

**Commented [37]:** Thanks for this comment. Here we were also considering shorter-term maintenance, often done by the original creators. When this is stopped then the FAIRness (particularly I and R) of the software will degrade.

**Commented [38]:** Not sure if I understand the meaning of "still" here

**Commented [39]:** same as above

**Commented [40]:** While the first two sentences make sense to me, the third does not yet fit in. Maybe we don't need the third sentence? Or am I missing an important point made here?

**Commented [41]:** We're going back to the originator ...

**Commented [42]:** although responsibility can be ...

**Commented [43]:** I have added a piece of text in the ...

**Commented [44]:** imo the paper is missing words or ...

**Commented [45]:** The intention is that specific ...

**Commented [46]:** this sentence seems to be out of ...

**Commented [47]:** first time that the word draft is used

**Commented [48]:** Good spot - this is an oversight in ...

**Commented [49]:** Two comments here. ...

**Commented [50]:** In answer to your comments: ...

**Commented [51]:** verb missing here?

**Commented [52]:** Yes - I think either the "where" cou ...

**Commented [53]:** Paragraph starts with this already.

**Commented [54]:** Agreed. This will be removed in the ...

**Commented [55]:** e.g. users will not deposit a softwa ...

**Commented [56]:** Perhaps we make it even clearer b ...

**Commented [57]:** I found this sentence unclear, I ...

**Commented [58]:** +1 remove

**Commented [59]:** I think this is a repetition of a ...

**Commented [60]:** actually this is not unique to ...

**Commented [61]:** Agreed - it's included here becaus ...

**Commented [62]:** I think this paragraph flows better ...

**Commented [63]:** The new version has been ...

**Commented [64]:** In the sections below, what ...

**Commented [65]:** +1 While I don't think the FAIR4RS ...

**Commented [66]:** The intent is that Appendix B ...



## Findable

**F: The software, and its associated metadata, should be easy to find for both humans and computers.**

*Machine-readable metadata are essential for automatic discovery of software and this metadata should meet domain-relevant community standards.*

### F1. Software is assigned a globally unique and persistent identifier.

*The use of globally unique and persistent identifiers enable adherence to many of the other FAIR4RS principles by removing ambiguity (for humans and machines) around what software (or part of it) is being referenced. Software differs from data because of the increased complexity around granularity (the "level of detail being implemented") and versioning (the "changes between implementations"), and how identifiers are applied to these and relate to each.*

*Granularity levels for software are shown in Figure 1 in Appendix A. However the principles do not prescribe which granularity levels should be assigned identifiers, as this is likely to be implementation-specific. Nevertheless, it is important to acknowledge the relationship between the different granularity levels and the types of identifier most suited for each case.*

*Versioning for software is generally more complex than for data. It is important to understand how one version of a piece of software relates to another, particularly as metadata such as authors or name/title may change. Although most identifier systems support relations that can be used to implement this functionality, some repositories do not yet, and this should not discourage the application of identifiers to software because of a requirement to support versioning.*

Option: A key difference between software and data is the way that relationships between versions (especially releases) and levels of granularity are represented (e.g. between a project and a release). Possible additional sub-principles (suggestions for rewording are accepted) are:

- **F1.1. Software is assigned multiple, distinct identifiers relating to different granularity levels and components of the software**
- **F1.2. Different versions of the same software are assigned distinct identifiers**

The explanatory paragraph would be updated to reflect that the relationship between granularity levels and versions should be embodied by a relationship between the identifiers ideally via the metadata related to the identifier.

Options: If this alternative wording is accepted, a related consideration is whether the definition of versioning in the FAIR principles should:

1. Ignore software engineering practices around the management and versioning of software artefacts as out of scope

**Commented [67]:** examples? Such as codemeta, software ontology?

**Commented [68]:** There was not consensus on how to choose specific examples, given that each community often has different selections of standards. It is hoped that the accompanying guidance documents will be able to go into more detail and provide various examples.

**Commented [69]:** Sorry, but I respectfully disagree, because the very same issue apply to datasets: long (unresolved) debates about which components a dataset describe get a PID and why: e.g. each sample in a lab experiment, each protocol file, each assay, ...

**Commented [70]:** I would agree with that. The new version is reworded so that it simply notes the ...

**Commented [71]:** If we aspire to assign PIDs and metadata to "(sub)modules" according to fig.1, the ...

**Commented [72]:** same issue for data

**Commented [73]:** note that no one is stopped to assign multiple PIDs to different components of a dataset, ...

**Commented [74]:** phrasing as principle, not a (technical) possibility

**Commented [75]:** I agree with the reasoning behind that change. Perhaps we need to slightly chan ...

**Commented [76]:** @Anna: do you believe this sub-principle should be added?

**Commented [77]:** +1 to change to 'is' and +1 to add both F1.1 and F1.2

**Commented [78]:** in F1.2 "same software" is appealing but lacks definition. I suggest removal of F1.2 as argu ...

**Commented [79]:** I think the subprinciples are not absolutely necessary (could also be explained in the ...

**Commented [80]:** Thanks - these two subprinciples will be added along with explanatory text.

**Commented [81]:** it would be ideal to have one identifier that links all of the versions and its identifiers

**Commented [82]:** Does that suggest an F1.3 which states that?

**Commented [83]:** F1.1 implies that you can have an identifier at the highest conceptual level. I think ...

**Commented [84]:** +1 on the implementation!

**Commented [85]:** +1 on implementation but include best practice on parent to versions?

**Commented [86]:** +1 for Nick's suggestion

**Commented [87]:** to Neil's question, we can add "preferably linked to the same software project in the ...

**Commented [88]:** I agree with Patricia, and all of the comments about implementation. F1.2 is implied in ...

**Commented [89]:** +1 Paula and Patricia

2. Acknowledge the practices but note them as an implementation detail that should be addressed in guidance on implementing the principles
3. Incorporate as a change in wording of this principle to acknowledge software versions as a fundamental characteristic of software that must be included for the FAIR principles to work for software (e.g. around semantic versioning)

## F2. Software is described with rich metadata.

*Metadata should be used to support search, discoverability (including indexing) and reusability. This metadata should itself be FAIR, follow community standards and use controlled vocabularies. The FAIR4RS principles do not define which standards should be used, as this is better captured in guidance for implementing the principles coming out of each community.*

Question: does the following alternative wording make the scope and intent of the F2 principle clearer?

- **F2. Software is described with rich metadata to support search and discoverability.**

Question: Is there any metadata (or class of metadata) relating to findability that is common to all software that should be included explicitly in this principle or a sub-principle, or in the explanatory text? E.g. descriptive metadata.

## F3. Metadata clearly and explicitly include the identifier of the software they describe.

*The association between the metadata (wherever it is stored, F4) and the software should be made explicit by mentioning the software's globally unique and persistent identifier in the metadata. For software to be findable, metadata is not required to include identifiers for all of its dependencies. Principles I2 and R2 describe how references to dependencies make software interoperable and reusable.*

## F4. Software is published in a searchable resource.

*Metadata about the software must be sufficient to enable the software to be listed or indexed in a registry or catalog such that it can be found.*

Question: As phrased, adherence to F4 is not entirely in the hands of the owners of the software - there may be no suitable registry or catalog, or the administrators of the registry or catalog

Commented [90]: +1

Commented [91]: @Paula: can you expand on your ...

Commented [92]: +1 F1.2 is sufficient without going ...

Commented [93]: +1 Paula choice; one thing is the ...

Commented [94]: examples?

Commented [95]: Perhaps they are referring to ...

Commented [96]: As noted above, we intend the ...

Commented [97]: yes, to me at least.

Commented [98]: Do we lose anything important in t ...

Commented [99]: I would say rich metadata also ...

Commented [100]: I agree with Malin, it supports ...

Commented [101]: that's true, but this is about the ...

Commented [102]: +1 for Anna-Lena's reasoning

Commented [103]: +1 to Anna-Lena (we don't loose ...

Commented [104]: It appears the consensus is for th ...

Commented [105]: Include administrative ...

Commented [106]: That seems more relevant for re- ...

Commented [107]: To answer the question: I don't ...

Commented [108]: agreed, we no not need to add n ...

Commented [109]: +1 Patricia: most people are not ...

Commented [110]: These words do not add much to ...

Commented [111]: I think that the "explicitly" is an ...

Commented [112]: references?

Commented [113]: I think this refers to persistent ...

Commented [114]: the I2. Software includes qualified ...

Commented [115]: however recommended

Commented [116]: imho, we want to express that the ...

Commented [117]: imho, too specific. You could also ...

Commented [118]: Here, the intent was to not restric ...

Commented [119]: This reads as we want to express ...

Commented [120]: I'd like to offer a suggested text ...

Commented [121]: maybe also: Publish software ...

Commented [122]: +1 Paula's second comment, and ...

Commented [123]: I suggest to remove the second ...

Commented [124]: I am not sure it is worth you ...

Commented [125]: I'll suggest to remove the sentenc ...

Commented [126]: This is true; there are many ...

Commented [127]: if we used "publish" instead of ...

Commented [128]: There is a challenge here in term ...

could refuse the software. Is the explanatory text sufficient to show how the principle can be applied in these cases, by enabling e.g. indexing by a search engine, or does the text of principle F4 require rewording.

**Commented [129]:** but this still require that the software is described somewhere on the web with a schema.org annotation.....

**Commented [130]:** the text is clear and sufficient

**Commented [131]:** I suggest adding the advantage of having the software in a discipline software registry - making more findable in the intended community. In any case, the text is good and clear.

**Commented [132]:** yes in any explanatory text adding advantages will motivate implementation. The advantages of software in a registry is to provide a fast and convenient way to find and retrieve software, and increase the use of the software while promoting the Rs : replicability, reproducibility and reuse..

## Accessible

**A: The software, and its metadata, must be retrievable via standardized protocols.**

*In the 2016 FAIR guiding principles, accessibility translates into retrievability. However, for software, there are extra accessibility concerns. Additionally, because software by necessity requires the use of standardized communications protocols to operate, some of the FAIR data principles may be considered commonly understood and implemented for software.*

Options: The principle of “accessible” in the FAIR4RS principles should be:

1. Narrowly scoped to just the ability to “retrieve”, as currently defined in the 2016 FAIR guiding principles
2. Extended to include the definition used in software engineering of the ability to access software regardless of impairment, but only applied to protocols required to retrieve the software and its metadata
3. Expanded to include the usability of software regardless of impairment (which could also be addressed in “reusability”)
4. Expanded to include elements of accessibility that are defined by open access / open science e.g. fee-free / gratis access to software

Application of options 2, 3 or 4 would result in changes to the FAIR4RS introductory definition for the foundational principle of accessible, as well as the guiding principles on accessibility below.

Question: Some other software engineering best practices like encapsulation / abstraction may be better considered under interoperable or reusable than accessible. Which practices should be:

1. Included under accessibility?
2. Included under interoperability?
3. Included under reusability?
4. Excluded because they are not directly relevant to implementing FAIR?

**A1. Software is retrievable by its (resolvable) identifier using a standardized communications protocol.**

*Different types of software have different methods for access. For instance, software that is only available in source code form may be downloaded from a repository before being compiled locally, whereas software hosted as a service on a remote server may be accessed without retrieving it. This principle states that software should not require specialised or proprietary tools or communication methods to obtain.*

There are different ways of scoping accessibility of software. Principally, the FAIR4RS must decide what forms of software are important to be accessible, and whether “standard” includes protocols which involve a well-defined and understood but manual process.

**Commented [133]:** This is hard, all options have their appeal. :) I think I'd prefer going for option 1. As others have said, it's the most concise way to go, and most aligned with the original FAIR principles. Definitely address 4., though, and state the importance of OS and that there are less hurdles to it than for (often sensitive) data. 2. and 3. are important in general, but would take the discussion quite far from the original FAIR work. Mention this somewhere in the document to show awareness, but maybe explicitly leave for future work/discussion.

**Commented [134]:** Option 1 is the consensus choice.

**Commented [135]:** I would go for option 1 because it is more concise. Extra accessibility concerns should not change the core of the principle I think.

**Commented [136]:** +1

**Commented [137]:** +1 as this is clear and simple, hence will be met. I always think in terms of ...

**Commented [138]:** +1

**Commented [139]:** This is the consensus choice.

**Commented [140]:** I am for this suggestion

**Commented [141]:** I'm missing open source in this option, or was that intended?

**Commented [142]:** this can be hinted in the explanatory text, but not editing the principle.

**Commented [143]:** FAIR is not open, so I am not sure this is a useful addition.

**Commented [144]:** +1 SAS. A binary with sufficient metadata could be FAIR. Requiring open source code ...

**Commented [145]:** Open will not be added here, but instead included in the challenges to implementation ...

**Commented [146]:** Design for the simple case first

**Commented [147]:** don't repeat yourself, so allow enough documentation for reusability

**Commented [148]:** encapsulation

**Commented [149]:** Indeed, these FAIR4RS should not include good SE practices, but complement them. Or ...

**Commented [150]:** Agreed. Good SE practices are important and relevant to several of the principles, but ...

**Commented [151]:** Explanatory notes have been added to say that software engineering practices are ...

**Commented [152]:** testing

**Commented [153]:** Strictly speaking, an object is not retrievable by a DOI but the DOI must be resolved by ...

**Commented [154]:** I think this is adequately implied in the principle - if it can't be resolved it can't be retrieved ...



Options: The A1 principle of “accessible” should be written to include:

1. Software for which the source code is available and retrievable
2. Software for which the source code or executable is retrievable
3. Instances of software running as services or platforms
4. Software in any form

Options: The A1 principle can also be interpreted more generally to include “manual” protocols to negotiate access to software such as “request by email” or “NDA must be signed before software is provided”. This should be:

1. Explicitly included
2. Explicitly excluded
3. Addressed as being suboptimal but not explicitly excluded

#### A1.1. The protocol is open, free, and universally implementable.

It is the openness of the protocol that is important, not the implementation of the infrastructure that supports it. Here “open” means that there are no restrictions to implementing it and “free” means that there are no fees or licensing costs to implement it.

#### A1.2. The protocol allows for an authentication and authorization procedure, where necessary.

There are often conditions of access to software, for instance a requirement for payment. This procedure may be a manual one, for instance requiring the signing of a non-disclosure agreement or software having an embargo period. It may also include things such as requiring a license server to be contacted.

#### A2. Metadata are accessible, even when the software is no longer available.

Availability of software may change over time, because there is a cost to maintaining access or because the software has degraded and is no longer safely usable. The metadata describing the software is generally easier and cheaper to store, and there is value in understanding the details of the software even if it is no longer accessible.

This principle may be redundant depending on whether it is assumed that this is implied by the 2016 FAIR guiding principles or not.

Options:

1. Keep this principle so the FAIR principles for research software can stand alone without reference to the 2016 FAIR guiding principles.
2. Remove this principle as it is just replicating the principle in the 2016 FAIR guiding principles, which describe how the FAIR principles apply to metadata.

Commented [155]: 1 or 4. I would argue that being ...

Commented [156]: I like (reading) source code but I ...

Commented [157]: Option 4 is the choice that best ...

Commented [158]: Option 1: 2-4 is too ...

Commented [159]: imho, option 4 is the least ...

Commented [160]: Option 4 is the choice that best ...

Commented [161]: +1

Commented [162]: @Paula: could you expand on yo ...

Commented [163]: I'm voting here, but I ...

Commented [164]: I agree with @Morane. Proprieta ...

Commented [165]: I'm thinking to gain access in term ...

Commented [166]: +1 Morane; "source code might b ...

Commented [167]: Agree. The point with "accessibili ...

Commented [168]: If we explicitly mention "source ...

Commented [169]: Option 4 is the choice that best ...

Commented [170]: I'm with this. Software available in ...

Commented [171]: +1 ... maybe also explicate ...

Commented [172]: Option 4 is the choice that best ...

Commented [173]: to be excluded: we strive for ...

Commented [174]: It's true that FAIR emphasizes th ...

Commented [175]: In the next version, text has been ...

Commented [176]: IMO "manual" is okay, but ...

Commented [177]: +1

Commented [178]: In the next version, text has been ...

Commented [179]: a possible option; again, FAIR is ...

Commented [180]: +1

Commented [181]: @Paula: could you expand on yo ...

Commented [182]: +1 this type of access is not fair ...

Commented [183]: the FAIR Principles put specific ...

Commented [184]: In the next version, text has been ...

Commented [185]: I think it would be nice to keep ...

Commented [186]: Agree with Patricia, also with the ...

Commented [187]: +1

Commented [188]: +1

Commented [189]: To complement Esther's point the ...

Commented [190]: In the next version, text has been ...

Commented [191]: the access protocol?

Commented [192]: This is probably adequately impli ...

Commented [193]: I'd consider the resolver of the ...

Commented [194]: I've added some text to include th ...

Commented [195]: a license usually regulates (re)us ...

Commented [196]: More broadly, many licenses do ...

Commented [197]: This expresses access to third- ...

Commented [198]: I think this is, as the owner of the ...

Commented [199]: Yes, but it has to be stored ...

Commented [200]: With software, metadata may als ...

## Interoperable

**I: The software interoperates with other software through exchanging data and/or metadata, and/or through interaction via application programming interfaces (APIs).**

*The definitions of interoperability and reusability as defined by the FAIR data principles overlap when applied to software. To differentiate between the two, interoperability is limited to being concerned with the capacity to exchange data between independent software. As an example, the sense of "integrated" that applies to data (where two pieces of data combine to form a new third thing) does not apply in the same way to software where, in a sense, all software is "integrated" with, or depends on, other software (and this concept is more sensibly placed under reusability, in R2). Software also has "agency": software calls on other software. Two independent pieces of software can be said to interoperate when the capability exists in both to read and write or otherwise exchange the same formats.*

**I1. Software reads, writes and exchanges data in a way that meets domain-relevant community standards.**

*Software interoperates through the exchange of data. This includes the use of data types and formats that are formally described using (preferably) controlled vocabularies, to facilitate machine readability and data exchange.*

**I2. Software includes qualified references to other objects.**

*Some software includes references to external data objects required to execute the software. To be fully FAIR, the data would ideally be FAIR as well, and references to external data fully qualified. Qualified references should be to metadata and data, as well as to non-digital objects that have a virtual presence in digital systems (e.g., samples, reagents, etc.), which with the software interoperate. These qualified references should be described using identifiers or controlled vocabularies. "Qualified" means specifying the authoritative source for an identifier or vocabulary item, possibly including a resolvable reference to further information about the source.*

*Furthermore, research software may include links to research papers through permanent identifiers.*

**Commented [212]:** is it just about the mean (API) or the content that this exposes? I would say both, so 'and'.

**Commented [213]:** There are examples of software that interact and interoperate with other software purely through the use of files, which is why the "or" was originally added. However, we did not want to simply say through exchanging data or metadata, as although it could be argued that an API is simply an exchange of data between two pieces of software, it has a well-understood meaning.

**Commented [214]:** Is this sufficient? What if the API is not open or specified/documented? Or just implemented in a really niche language, not a broadly used one?

**Commented [215]:** I would argue that implementation in a niche language doesn't affect interoperability

**Commented [216]:** agree.

**Commented [217]:** +1

**Commented [218]:** A sentence has been added to the

**Commented [219]:** define it, please

**Commented [220]:** Here, we mean "independent" to

**Commented [221]:** functionality or implementation? i

**Commented [222]:** "functionality" has been used in the

**Commented [223]:** examples? in this case many, sin

**Commented [224]:** +1 for adding examples. As it

**Commented [225]:** I like the abstract version. A link t

**Commented [226]:** Well community standards (inc

**Commented [227]:** Imho we should aspire to have

**Commented [228]:** I have added a link to FAIRSharin

**Commented [229]:** examples? I am not sure how

**Commented [230]:** dependencies? would indeed be

**Commented [231]:** ah, seeing now that dependencie

**Commented [232]:** This is intended to cover things

**Commented [233]:** I am confused here: how does th

**Commented [234]:** i really like this as recommendation

**Commented [235]:** in my understanding, an identifier

**Commented [236]:** and/or

**Commented [237]:** +1

**Commented [238]:** I was thinking in terms of a recent

**Commented [239]:** But a paper does not "interoperat

**Commented [240]:** Exactly! I was also referring to the

**Commented [241]:** For the next version of the



## Reusable

**R:** The software is both usable (it can be executed) and reusable (it can be understood, modified, built upon, or incorporated into other software).

*The ultimate goal of FAIR is to enable transparency, reproducibility, and reusability of research. To achieve this, software must be usable and reusable. Software should be well-described (by metadata), inspectable, documented and appropriately structured so that it can be executed, replicated, combined, reinterpreted, reimplemented, and/or used in different settings.*

*The definitions of interoperability and reusability as defined by the FAIR data principles overlap when applied to software. To differentiate between the two, reusability (implicitly including usability) focus on humans and machines, while interoperability is limited to being concerned with the relationship between a piece of software and the external software upon which it depends in order to operate (i.e., its dependencies).*

*Note that the general intent of these principles is that software is “executable in principle” - not “guaranteed to execute”.*

### **R1. Software is richly described with a plurality of accurate and relevant attributes.**

*It is easier to find and reuse software if there are many descriptive labels attached to it. Software should be described for the categories of R1.1 (license), R1.2 (provenance), and additionally address the categories of metadata that facilitate reuse. Relevant attributes can be determined by repositories, and by communities who create and reuse software. Plurality means that, where possible, multiple terms for the same, similar or overlapping concepts should be provided to enable the broadest possible reuse.*

#### **R1.1. Software reuse requires a license.**

*Software must have a license that clearly describes how it can be used and reused, ideally with conditions that are clear to humans and machines and as open as possible. This license must also be compatible with the licenses of the software’s dependencies.*

#### **R1.2. Software is associated with detailed provenance.**

*The primary provenance information for software is authorship. Although a version control system may provide detailed provenance for software, care must be taken that authorship and contributions are properly tracked.*

### **R2. Software includes qualified references to other software.**

Commented [242]: this is not achievable with software ...

Commented [243]: Also this touches on the issue ...

Commented [244]: Indeed. Sometimes OS software ...

Commented [245]: I don't see a problem here as the ...

Commented [246]: Here, the intent was to define the ...

Commented [247]: maybe this would fit in the intro ...

Commented [248]: +1

Commented [249]: +1

Commented [250]: Thanks for the suggestion - this ...

Commented [251]: don't we have this in I?

Commented [252]: imho the sentence explains ...

Commented [253]: I think this sentence is ...

Commented [254]: Sorry - this was a copy and paste ...

Commented [255]: Fantastic addition!

Commented [256]: To me, this reads like a duplicate ...

Commented [257]: This is a pertinent issue to raise. ...

Commented [258]: i think the need to emphasize the ...

Commented [259]: Indeed, also if F fails.... well, ...

Commented [260]: is plurality referred to metadata ...

Commented [261]: I take this to mean both, but ...

Commented [262]: rewording suggestion to focus on ...

Commented [263]: I'm not going to take the full ...

Commented [264]: Do we aspire to machine- ...

Commented [265]: +1, This is a nice addition, ...

Commented [266]: yes, we do aspire, but not require :) ...

Commented [267]: I have added to the explanatory ...

Commented [268]: If there is no license then reuse ...

Commented [269]: +1

Commented [270]: Change made in next version.

Commented [271]: Since "ideal" scenarios are ...

Commented [272]: good idea!

Commented [273]: Added as a separate sentence in ...

Commented [274]: to reflect the suggestions in the ...

Commented [275]: I am modifying this suggestion as ...

Commented [276]: strictly speaking in a legal sense.

Commented [277]: Change made in next version.

Commented [278]: Would it be OK to add an ...

Commented [279]: I don't think such an explanation f ...

Commented [280]: Some small additional words of ...

Commented [281]: Well I think we need more than ...

Commented [282]: It's important provenance to know ...

Commented [283]: In answer to Susanna's point - ...

Commented [284]: I think this misses the point of wh ...

Commented [285]: Thanks for this comment - I have ...

Commented [286]: Provenance is metadata that ...

Commented [287]: I would disagree with this - for ...

Software is rarely standalone and in most cases is built upon other software (e.g dependencies), it should include appropriate references to other software (requirements, imports, libraries, etc.) which is necessary to compile and run the software, but not in a way that meets the FAIR principles. “Qualified” here means specifying the authoritative source for an identifier, possibly including a resolvable reference to further information about the source. To follow this principle, it is desirable but not required that the other software referenced implements the FAIR4RS principles.

### R3. Software meets domain-relevant community standards

Software, including its documentation and license, should meet or rise above domain-relevant community standards and coding practices (for example choice of programming language, standards for testing, usage of file formats, etc) that enable reuse. While the FAIR4RS principles do not specify particular community standards, the intent is to ensure that practitioners are aware of what others are doing and using in the community, whilst acknowledging that community standards are (and should be) under constant development. This principle encourages collaboration with the relevant community.

**Commented [288]:** Not entirely clear to me what this part of the sentence refers to.

**Commented [289]:** i think it means that not all these references are expected to be compliant to FAIR? I would suggest to delete after the comma. Also the sentence is long already so it might become a new sentence.

**Commented [290]:** will suggest to delete, as the last sentence summarizes it To follow this principle, it is desirable but not required that the other software referenced implements the FAIR4RS principles.

**Commented [291]:** This is true for data too; when you assemble and combine datasets (maybe with your own), you have no/little control on the FAIRness of those datasets. We need to address the software developers/owners so that they ensure FAIRness at the source (at the first mile), as this is the only way to change the status quo.

**Commented [292]:** +1 for delete

**Commented [293]:** This has been deleted in the new version to avoid confusion.

**Commented [294]:** I have the impression that FAIR and FAIR4RS might not be used consistently throughout the document. So far, mostly FAIR was used even though sometimes the focus was on software, I think.

**Commented [295]:** Thanks - we'll try and do a consistency check. It's unfortunate that they're so similarly named.

**Commented [296]:** This paragraph reads like the abstract part of an intro or summary. Community standards should be considered in all FAIR4RS principles.

**Commented [297]:** This is a bit of a catch-all, however there is a specific reason that it is called out in Reusability, as there are many community standards { ...

**Commented [298]:** Happy to consider suggestions of how this could be rewritten to address your issue.

**Commented [299]:** Wondering if we should emphasize the importance of documentation for (re)use more. Not sure where that would fit best, though.

**Commented [300]:** Happy to consider suggestions of how this could be best done.

**Commented [301]:** agree, but you can point to lists and relevant registries (see comment above re FAIRsharing) that serve as educational/informative resources.

**Commented [302]:** I've added a reference to FAIRsharing here in the next version as an example of how practitioners are made aware of relevant ...

**Commented [303]:** The original intent of this sentence was to clarify that this principle was not meant to stifle innovation around the way that software was being ...

## Challenges to implementation

In this section, challenges that must be overcome to make research software FAIR are described. These represent current gaps that may make it difficult for individuals to follow the FAIR4RS principles. Additional details can be found in the related reports ([Katz, Gruenpeter & Honeyman, 2021](#); [Katz, Chue Hong, Barker & Gruenpeter, 2021](#)).

**Metadata and identifier authority.** All research software must have unique identifiers and associated metadata. How are these identifiers created? How is the metadata created, stored and maintained? Intrinsic metadata is guaranteed to be controlled by the authors but must be exposed to make the software findable. Extrinsic metadata, such as (persistent) identifiers, can be used to make the software findable but is controlled by an external authority.

**Metadata vocabularies and metadata properties.** At present, there is no community agreement on which vocabularies should be used. Vocabularies used by package managers to describe software do not capture metadata about research and there are relatively few discipline-specific vocabularies that capture metadata about software development and usage. Establishing metadata vocabularies/standards is an intensive process for which resources are limited.

**Software identifiers.** At present, there is no community agreement on the best identifiers for software, even for specific use cases such as giving software authors credit. These identifiers are mostly independent and not clearly interoperable.

**Identification target.** At present, there is no community agreement on what a software identifier should refer to, e.g. for open source software, for commercial software, for a container,

**Software structure complexity.** Software is often a complex object made up of other software, documentation, data and metadata. How do we deal with this? Where should the FAIR4RS principles be applied, and where should other interpretations of the FAIR guiding principles be applied? What should have identifiers, and how should relationships between them be described to be FAIR?

**FAIRness of related research objects.** There is still debate over whether FAIR is recursive, i.e. a digital research object is only “fully FAIR” if the objects it builds on are also FAIR. However, even if just applied to data dependencies, this might restrict the implementation of FAIR4RS principles as it would require measurable, actionable guiding principles.

**Definition of accessibility.** In software engineering, there is already a different, well-understood definition of software accessibility. Even if the meaning used in the FAIR4RS principles is well-defined and scoped, it may lead to confusion and mean the principle is not well-understood across all domains.

**Commented [304]:** how are these unique to software? Many apply to data too.

**Commented [305]:** The intention isn't to suggest that these are unique to software, but that these are challenges to the implementation and application of the FAIR principles to software.

**Commented [306]:** example?

**Commented [307]:** An example (codemeta files) has been added to the next version.

**Commented [308]:** example?

**Commented [309]:** Suggested example from Alexander Struck added to next version.

**Commented [310]:** as per request for example

**Commented [311]:** Thank you for the suggested example.

**Commented [312]:** examples?

**Commented [313]:** We have chosen not to provide examples here, because there are so few. A potential example could be the use of RRID metadata for neuroscience software: [https://neuinfo.org/rin/suggested-software?p1=SCR\\_006770](https://neuinfo.org/rin/suggested-software?p1=SCR_006770)

**Commented [314]:** As in, I have difficulties finding funding calls that would allow allocating funding to these activities..

**Commented [315]:** Thank you for this suggestion.

**Commented [316]:** process?

**Commented [317]:** it seems more appropriate

**Commented [318]:** they almost do not exist ;) and this is the case for datasets too.

**Commented [319]:** This could partly be addressed by running (mature/established) software identifiers through a community endorsement process, in one or more relevant communities. For instance, INCF currently has Research Resource IDs (RRIDs) under review in the INCF community endorsement process. Other communities with strong standards processes exist, they might also be relevant.

**Commented [320]:** Does software need its own special PID? I mean e.g. DOI where for other objects and now ...

**Commented [321]:** There has been a large discussion ...

**Commented [322]:** I have added part of the text from ...

**Commented [323]:** I think this is a challenge carried ...

**Commented [324]:** I've added a sentence to the next ...

**Commented [325]:** but this is literally impossible. The ...

**Commented [326]:** I agree, impossible and untractable

**Commented [327]:** I would agree, but in the earlier ...

**Definition of reusability.** In software engineering, for software to be reusable it should also be maintainable and dependable (able to be built on for other purposes). This may be captured in R3, around domain-relevant community standards, but may also require additional clarification to avoid confusion or the proliferation of many competing sets of “added letters” to FAIR4RS related to other qualities.

## The path to adoption

It will take significant effort to gain **wide-spread adoption** of the FAIR4RS principles, once finalized. The convening of the FAIR4RS\_WG across RDA, FORCE11, and ReSA will support usage of the outcomes across those communities. RDA, FORCE11, and ReSA will systematically promote the outcomes, aiming to raise awareness and facilitate a wider adoption of the FAIR4RS WG outcomes by existing and emerging initiatives. Organizations with a focus on FAIR will also be engaged, to encourage promotion of the application of FAIR to research outputs other than data.

The FAIR4RS WG’s aims also include development of **adoption guidelines and practices** to enable widespread adoption of the FAIR4RS principles across the research software community at national, disciplinary, and international levels. This will focus on the needs of a variety of stakeholders, including:

- those that will **endorse** and promote the guidelines
- those that will provide training on the guidelines
- **users of the guidelines**

The FAIR4RS WG will focus on adoption after the dissemination of the FAIR4RS principles in mid-20201, and will continue to regularly engage the community during all phases.

ReSA is also leading the FAIR4RS Roadmap to make FAIR research software a reality, with support from the Wellcome Trust. The 2018 European Commission report, “Turning FAIR into Reality” ([European Commission, 2018](#)), concludes that FAIR digital objects (including software) need to be supported by metrics, incentives, skills and FAIR services that provide persistent identifiers, metadata specifications, stewardship and repositories, actionable policies and Output Management Plans. All of these need to be created for FAIR software, to complement the significant FAIR initiatives that primarily encompass data, and to leverage the efforts already underway to enable this for software. The FAIR4RS Roadmap is identifying relevant software initiatives and equivalent FAIR data programs in areas such as: indicators, metrics, maturity models and certification; curriculums and competence centres, career profiles and reward structures; certification of FAIR services; interoperability frameworks; and policy change.

**Commented [328]:** I feel that this section's focus is more on "Ways to adoption" than "Challenges to adoption". I don't suggest to change the content but to reconsider the title.  
I like the idea of ending the report on an optimistic note (rather than a chapter of challenges to adoption after a long list of challenges to implementation).

**Commented [329]:** Thanks for the suggestion - we have used Alexander Struck's suggested change to the title.

**Commented [330]:** Do you have early adopters that could showcase 'how to do it' and the value? It is hard to go from theory to full adoption, you need lighthouses adopters that lead the way.

**Commented [331]:** Findable and Accessible principles can be mostly implemented by software registries / digital repositories that accept software deposits. Does every piece of software placed in Zenodo mostly meet these principles? I'm not sure if they require a software license for Accessible, and domain specific keywords / metadata / contributor provenance are also optional

**Commented [332]:** The SpringerNature FAIR white paper has a useful suggestion: to focus on discipline specific role models (doing FAIR in their discipline, showing the steps) and concrete discipline use cases. The earlier mentioned organizations "with a focus on FAIR" can be strategic partners there, if they can be convinced to include software focused examples.

(For instance, INCF is already planning to produce FAIR examples aimed at neuroscientists)

**Commented [333]:** Thanks for these comments - as noted later in the section, the FAIR4RS Roadmap activity is identifying potential early adopters and undertaking community engagement activities - as Malin points out, many of the organisations involved in these engagement activities are already working on discipline specific FAIR guidance, and our work is to convince them to provide software examples.

**Commented [334]:** should these go hand in hand with this work, to ground the theory into practice?

**Commented [335]:** Ideally, yes, but we believe it is too large a piece of work to do all at once - the formal review of the next version of these principles will allow there to be an incremental iteration of principles and practice.

**Commented [336]:** does this also imply funders who may "require" adherence to the guidelines in the future?

**Commented [337]:** My personal opinion is yes, similar to data. e.g. the Open Research Data pilot for EC H2020 has led to Horizon Europe requiring DMPs which outline the FAIRness of data produced.

**Commented [338]:** i guess this group needs are the most important

## Acknowledgements

This work was partially supported by the Alfred P. Sloan Foundation and the Wellcome Trust. NCH's time as editor was supported by the UKRI research councils through grant EP/S021779/1.

## References

European Commission. Directorate General for Research and Innovation. (2018). Turning FAIR into reality: final report and action plan from the European Commission expert group on FAIR data. Publications Office. <https://doi.org/10.2777/1524>

European Commission. Directorate General for Research and Innovation. & EOSC Executive Board,. (2020). Six Recommendations for implementation of FAIR practice by the FAIR in practice task force of the European open science cloud FAIR working group. Publications Office. <https://doi.org/10.2777/986252>

FAIR4RS WG. (2021). FAIR4RS Subgroup 4 - reading list of new research (Version 1.0) [Data set]. Zenodo. <http://doi.org/10.5281/zenodo.4555865>

FORCE11 FAIR Data Publishing Working Group. (2015). The FAIR Data Principles. FORCE11. <https://www.force11.org/group/fairgroup/fairprinciples>

GO FAIR. (2018). FAIR Principles. GO FAIR Initiative. <https://www.go-fair.org/fair-principles/>  
Retrieved from: <https://web.archive.org/web/20180212143802/https://www.go-fair.org/fair-principles/>

Gruenpeter, M., Katz, D.S., Lamprecht, A-L., Honeyman, T., Garijo, D., Struck, A., Niehues, A., Martinez, P.A., Castro, L.J., Rabemanantsoa, T., Plomp, E., Chue Hong, N., Martinez-Ortiz, C., Sesink, L., Liffers, M., Fouilloux, A.C., Erdmann, C., Peroni, S., Lavanchy, P.M., & Todorov, I. (2021). Defining Research Software: a controversial discussion: Summary Report of FAIR4RS Subgroup 3 activity and discussion. [Manuscript in preparation.](#)

Hata, H., Guo, J. L., Kula, R. G., & Treude, C. (2021). Science-Software Linkage: The Challenges of Traceability between Scientific Knowledge and Software Artifacts. [arXiv preprint arXiv:2104.05891.](#)

Katz, D. S., Gruenpeter, M., & Honeyman, T. (2021). Taking a fresh look at FAIR for research software. *Patterns*, 2(3), 100222. <https://doi.org/10.1016/j.patter.2021.100222>

**Commented [339]:** Thank you for this suggestion - in the next version of this document it has been added, and a reference added in the text.

Katz, Daniel S., Chue Hong, Neil P., Barker, Michelle, & Gruenpeter, Morane. (2021). FAIR4RS WG subgroup community consultation March 2021. Zenodo.

<http://doi.org/10.5281/zenodo.4635410>

Lamprecht, A.-L., Garcia, L., Kuzak, M., Martinez, C., Arcila, R., Martin Del Pico, E., Dominguez Del Angel, V., van de Sandt, S., Ison, J., Martinez, P. A., McQuilton, P., Valencia, A., Harrow, J., Psomopoulos, F., Gelpi, J. Ll., Chue Hong, N., Goble, C., & Capella-Gutierrez, S. (2020).

Towards FAIR principles for research software [JB]. *Data Science*, 3(1), 37–59.

<https://doi.org/10.3233/DS-190026>

Research Data Alliance/FORCE11 Software Source Code Identification WG, Allen, A., Bandrowski, A., Chan, P., Di Cosmo, R., Fenner, M., Garcia, L., Gruenpeter, M., Jones, C. M., Katz, D. S., Kunze, J., Schubotz, M. & Todorov, I. T. (2020). Use cases and identifier schemes for persistent software source code identification (V1.1). Research Data Alliance.

<https://doi.org/10.15497/RDA00053>

Wilkinson, M. D., Dumontier, M., Aalbersberg, Ij. J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J.-W., da Silva Santos, L. B., Bourne, P. E., Bouwman, J., Brookes, A. J., Clark, T., Crosas, M., Dillo, I., Dumon, O., Edmunds, S., Evelo, C. T., Finkers, R., Gonzalez-Beltran, A., Gray, A.J.G., Groth, P., Goble, C., Grethe, J.S., Heringa, J., 't Hoen, P.A.C., Hooft, R., Kuhn, T., Kok, R., Kok, J., Lusher, S.J., Martone, M.E., Mons, A., Packer, A.L., Persson, B., Rocca-Serra, P., Roos, M., van Schaik, R., Sansone, S-A., Schultes, E., Sengstag, T., Slater, T., Strawn, G., Swertz, M.A., Thompson, M., van der Lei, J., van Mulligen, E., Velterop, J., Waagmeester, A., Wittenburg, P., Wolstencroft, K., Zhao, J. & Mons, B. (2016). The FAIR Guiding Principles for scientific data management and stewardship. *Scientific Data*, 3(1). <https://doi.org/10.1038/sdata.2016.18>



## Appendices

### Appendix A - Additional Figures

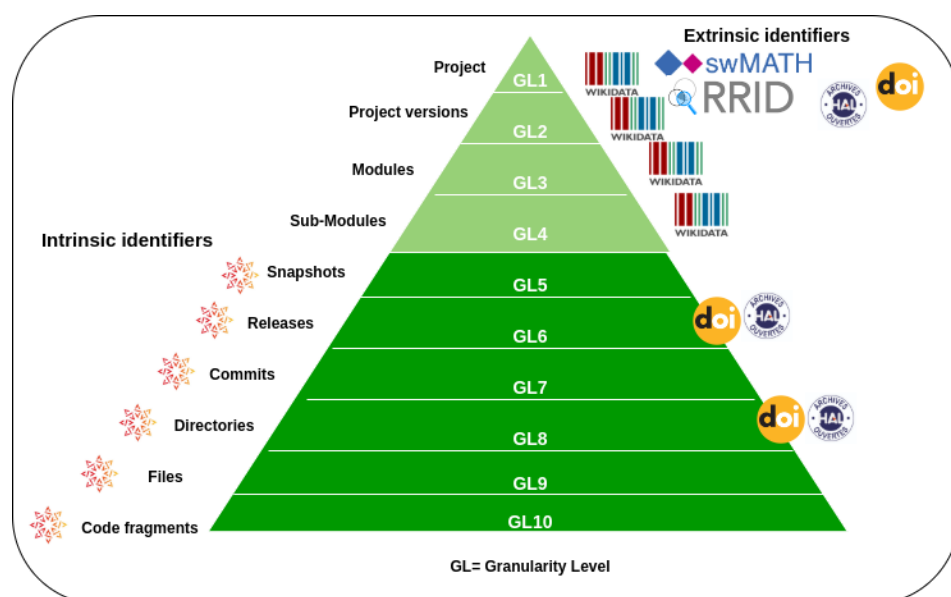


Figure 1: Granularity levels and identifiers currently in use for software [based on granularity levels definition from (RDA/FORCE11 SSCID WG et al., 2020)]



Figure 2: Summarizing software as increasingly FAIR research objects (Katz, Gruenpeter & Honeyman, 2021)

**Commented [340]:** I don't think this figure was referenced in the text

**Commented [341]:** No, it's not currently - we need to make a decision about how we choose to use this figure (or a version of it) in the version for formal review.



## Appendix B - Comparison of FAIR principles

As background information, this section details how the development of the FAIR4RS principles has evolved, by comparison of The FAIR Guiding Principles for scientific data management and stewardship (Wilkinson et al., 2016, [with foundational principle text taken from GO FAIR, 2018](#)) with the Towards FAIR Principles for research software (Lamprecht et al., 2020) and Taking a fresh look at FAIR for research software report (Katz, Gruenpeter & Honeyman, 2021), and the FAIR4RS principles described in this document.

**Commented [342]:** This is absolutely fabulous! Great overview and it really provides an overview of the past years!

**Commented [343]:** 👍

FAIR Guiding Principles (2016)	Towards FAIR Principles for research software (2020)	Taking a fresh look at FAIR for research software (2021)	Draft FAIR4RS Principles (2021)
<b>F. Findable</b>			
The first step in (re)using data is to find them. Metadata and data should be easy to find for both humans and computers. Machine-readable metadata are essential for automatic discovery of datasets and services, so this is an essential component of the FAIRification process.	The main concern of findability for research software is to ensure software can be identified unambiguously when looking for it using common search strategies.	The first step in (re)using software is to find it. Metadata and software should be easy to find for both humans and computers. Machine-readable metadata are essential for automatic discovery of software, so this is an essential component of the FAIRification process.	The software, and its associated metadata, should be easy to find for both humans and computers.
F1. (Meta)data are assigned a globally unique and persistent identifier	F1. Software and its associated metadata have a global, unique and persistent identifier for each released version.	F1. Software is assigned a globally unique and persistent identifier	F1. Software is assigned a globally unique and persistent identifier.
F2. Data are described with rich metadata (defined by R1 below)	F2. Software is described with rich metadata.	F2. Software is described with rich metadata (defined first by R1 below, and then by the original FAIR principles for metadata)	F2. Software is described with rich metadata.
F3. Metadata clearly and	F3. Metadata clearly and	F3. Metadata clearly and	F3. Metadata clearly and

explicitly include the identifier of the data they describe	explicitly include identifiers for all the versions of the software it describes.	explicitly include the identifier of the software they describe	explicitly include the identifier of the software they describe.
F4. (Meta)data are registered or indexed in a searchable resource	F4. Software and its associated metadata are included in a searchable software registry.	F4. Software is registered or indexed in a searchable resource	F4. Software is registered or indexed in a searchable resource.
<b>A. Accessible</b>			
Once the user finds the required data, she/he needs to know how can they be accessed, possibly including authentication and authorisation.	Accessibility translates into retrievability [...] however, we found mere retrievability not enough. In order for anyone to use any research software, a working version of the software needs to be available.	Once the user finds the required software, they need to know how it can be accessed, possibly including authentication and authorization.	The software, and its metadata, must be retrievable via standardized protocols.
A1. (Meta)data are retrievable by their identifier using a standardized communications protocol	A1. Software and its associated metadata are accessible by their identifier using a standardized communications protocol.	A1. Software is retrievable by its identifier using a standardized communications protocol	A1. Software is retrievable by its identifier using a standardized communications protocol.
A1.1. The protocol is open, free, and universally implementable	A1.1. The protocol is open, free, and universally implementable.	A1.1. The protocol is open, free, and universally implementable	A1.1. The protocol is open, free, and universally implementable.
A1.2. The protocol allows for an authentication and authorization procedure, where necessary	A1.2. The protocol allows for an authentication and authorization procedure, where necessary.	A1.2. The protocol allows for an authentication and authorization procedure, where necessary	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	A2. Software metadata are accessible, even when the software is no longer available.	A2. Metadata are accessible, even when the software is no longer available	A2. Metadata are accessible, even when the software is no longer available.
<b>I. Interoperable</b>			

The data usually need <sup>s</sup> to be integrated with other data. In addition, the data need to interoperate with applications or workflows for analysis, storage, and processing.	Interoperability for research software can be understood in two dimensions: as part of workflows (horizontal dimension) and as stack of digital objects that need to work together at compilation and execution times (vertical dimension)	The software usually needs to communicate with other software via exchanged data (or possibly its metadata). Software tools can interoperate via common support for the data they exchange.	The software interoperates with other software through exchanging data and/or metadata, and/or through interaction via application programming interfaces (APIs).
I1. (Meta)data use a formal, accessible, shared, and broadly applicable language for knowledge representation.	I1. Software and its associated metadata use a formal, accessible, shared and broadly applicable language to facilitate machine readability and data exchange.	I1. Software should read, write or exchange data in a way that meets domain-relevant community standards	I1. Software reads, writes and exchanges data in a way that meets domain-relevant community standards.
I2. (Meta)data use vocabularies that follow FAIR principles	I2.1. Software and its associated metadata are formally described using controlled vocabularies that follow the FAIR principles. I2.2. Software use and produce data in types and formats that are formally described using controlled vocabularies that follow the FAIR principles.		
I3. (Meta)data include qualified references to other (meta)data		I2. Software includes qualified references to other objects.	I2. Software includes qualified references to other objects.
	I4S. Software dependencies are documented and mechanisms to access them exist.		
<b>R. Reusable</b>			

**Commented [344]:** This is in the original text from the GO FAIR initiative, where "data" here is the plural form. I've added some text in the intro to this appendix to make it clearer.

The ultimate goal of FAIR is to optimize the reuse of data. To achieve this, metadata and data should be well-described so that they can be replicated and/or combined in different settings.	Reusability in the context of software has many dimensions. At its core, reusability aims for someone to be able to reuse software reproducibly.	The ultimate goal of FAIR is to enable and encourage the use and reuse of software. To achieve this, software should be well-described (by metadata) and appropriately structured so that it can be replicated, combined, reinterpreted, reimplemented, and/or used in different settings.	The software is both usable (it can be executed) and reusable (it can be understood, modified, built upon, or incorporated into other software).
R1. (Meta)data are richly described with a plurality of accurate and relevant attributes	R1. Software and its associated metadata are richly described with a plurality of accurate and relevant attributes.	R1. Software is richly described with a plurality of accurate and relevant attributes	R1. Software is richly described with a plurality of accurate and relevant attributes.
R1.1. (Meta)data are released with a clear and accessible data usage license	R1.1. Software and its associated metadata have independent, clear and accessible usage licenses compatible with the software dependencies.	R1.1. Software is made available with a clear and accessible software usage license	R1.1. Software is made available with a clear and accessible license.
R1.2. (Meta)data are associated with detailed provenance	R1.2. Software metadata include detailed provenance, detail level should be community agreed.	R1.2. Software is associated with detailed provenance	R1.2. Software is associated with detailed provenance.
R1.3. (Meta)data meet domain-relevant community standards	R1.3. Software metadata and documentation meet domain-relevant community standards.	R1.3. Software meets domain-relevant community standards	R3. Software meets domain-relevant community standards
		R2. Software includes qualified references to other software	R2. Software includes qualified references to other software.

## Appendix C - Contributor List

The following table lists all people who have been recorded as having made a significant contribution towards the development of the FAIR4RS principles, listed in alphabetical order by first name. If your contribution has not been properly recognized, please contact N.ChueHong@software.ac.uk.

Name	Institution	ORCID	Editor	Member of drafting group	Member of WG steering committee	Contributor to WG meetings	Contributor to subgroup reports	Contributor to first consultation	Contributor to second consultation
Alan Williams	The University of Manchester	0000-0003-3156-2105					X		
Allen Lee	Arizona State University	0000-0002-6523-6079							X
Alexander Struck	Cluster of Excellence Matters of Activity at Humboldt-Universitaet zu Berlin	0000-0002-1173-9228					X		X
Anna Niehues	Radboud university medical center	Please add					X		
Anna-Lena Lamprecht	Utrecht University	0000-0003-1953-5606		X		X	X	X	X
Anne Claire Fouilloux	University of Oslo, Department of Geosciences	0000-0002-1784-2920					X		
Axel Loewe	Karlsruhe Institute of Technology (KIT)	0000-0002-2487-4744				✗		X	X
Bala Desinghu	Rutgers University	0000-0003-2854-9583					X		

**Commented [345]:** In some of the discussions around subgroup 1 and workflows, I see contributions from Stian Soiland-Reyes and Alan Williams, who don't appear to be listed in the table yet

**Commented [346]:** Added, but no details, could please add what you know of?

**Commented [347]:** @soiland-reyes@manchester.ac.uk - can you add your info here if you would like to be credited for your contribution?

**Commented [348]:** @alan.r.williams@manchester.ac.uk - can you add your info here if you would like to be credited for your contribution?

**Commented [349]:** Collaborators please double check your affiliation and ORCID

**Formatted Table**

**Commented [350]:** I'm removing this, as there may have been some confusion over this column - this is to recognise contributions to the monthly WG meetings, rather than any FAIR4RS workshops or plenary sessions.

Ben van Werkhoven	Netherlands eScience Center	0000-0002-7508-3272						X	
Carlos Martinez	Netherlands eScience Center	0000-0001-5565-7577		X	X	X	X	X	X
Carole Goble	The University of Manchester	0000-0003-1219-2137				X	X		
Catherine Jones	Science and Technology Facilities Council (STFC)	0000-0002-5112-835X					X	X	
Céline Richard	CIRAD Centre de coopération internationale en recherche agronomique pour le développement	Please add					X		
Charles Gray	Newcastle University	0000-0002-9978-011X					X		
Chris Erdmann	American Geophysical Union	0000-0003-2554-180X					X		
Daniel Garijo	Information Sciences Institute, University of Southern California (USC)	0000-0003-0454-7145				X	X	X	
Daniel S. Katz	University of Illinois Urbana-Champaign	0000-0001-5934-7525	X	X	X	X	X	X	X
Daniel Nüst	Institute for Geoinformatics , Opening	0000-0002-0024-5046					X		

**Commented [351]:** please contact me or register to be part of the group via <https://www.rd-alliance.org/groups/fair-research-software-fair4rs-wg>

**Commented [352]:** please contact me or register to be part of the group via <https://www.rd-alliance.org/groups/fair-research-software-fair4rs-wg>

	Reproducible Research,, University of Münster; de-RSE								
Daniele Tartarini	University of Sheffield	0000-0002-8913-0156						X	
Elena Rangelova	Netherlands eScience Center	0000-0002-9834-1756						X	
Esther Plomp	Delft University of Technology, Faculty of Applied Sciences	0000-0003-3625-1357					X		X
Fotis E. Psomopoulos	Institute of Applied Biosciences (INAB CERTH)	0000-0002-0222-4273		X	X	X	X	X	X
Francoise Genova	Centre de Donnees astronomiques de Strasbourg (CDS)	0000-0002-6318-5028				X			
Hartwig Anzt	University of Tennessee / Karlsruhe Institute of Technology	0000-0003-2177-952X					X		
Hugh Shanahan	Royal Holloway, University of London	0000-0003-1374-6015						X	
Ilian Todorov	UKRI, Science and Technology Facilities Council (STFC)	0000-0001-7275-1784					X	X	

**Commented [353]:** Daniele please contact me or register to be part of the group via <https://www.rd-alliance.org/groups/fair-research-software-fair4rs-wg>

James McNally	ICPSR University of Michigan	0000-0002- 6807-4538						X	
Javier Moldon	IAA-CSIC	0000-0002- 8079-7608					X		
Jen Harrow	ELIXIR	0000-0003- 0338-3070			X	X	X	X	X
Jessica Burnett	US geological survey	0000-0002- 0896-5099					X		
Joanna Leng	University of Leeds	0000-0001- 9790-162X						X	
Julián Garrido- Sánchez	Instituto de Astrofísica de Andalucía IAA	0000-0002- 6696-4772					X		
Khalid Belhajjame	PSL, Paris- Dauphine University	0000-0001- 6938-0820					X		
Laurents Sesink	Leiden University Libraries	0000-0001- 7880-5413					X		
Leyla Jael Garcia Castro	ZB MED - Information Center Life Sciences	0000-0003- 3986-0510		X	X	X	X	X	X
Lorraine Hwang	UC Davis, CIG	0000-0002- 1021-3101					X		
Maggie Hellström	Lund University, Sweden and ICOS Carbon Portal	0000-0002- 4154-2610					X		
Malin Sandström	International Neuroinformati cs Coordinating Facility (INCF)	0000-0002- 8464-2494				X		X	<del>X</del>

**Commented [354]:** James please contact me or register to be part of the group via <https://www.rd-alliance.org/groups/fair-research-software-fair4rs-wg>

**Commented [355]:** Joanna please contact me or register to be part of the group via <https://www.rd-alliance.org/groups/fair-research-software-fair4rs-wg>

**Commented [356]:** Julian, please contact me or register to be part of the group via <https://www.rd-alliance.org/groups/fair-research-software-fair4rs-wg>



Manodeep Sinha	Swinburne University of Technology	0000-0002-4845-1228					✖	X	
Marcos Roberto Tovani-Palone	University of São Paulo, Brazil; Modestum Ltd, United Kingdom	0000-0003-1149-2437						X	
Mark D. Wilkinson	Polytechnic University of Madrid	0000-0001-6960-357X					X		
Mateusz Kuzak	Netherlands eScience Center	0000-0003-0087-6021			X	X	X		
Mathieu Servillat	Observatoire de Paris	0000-0001-5443-4128				✖		X	X
Matthias Liffers	Australian Research Data Commons	0000-0002-3639-2080					X		
Merc Fox	University of Arizona	0000-0002-0726-7301						X	
Michelle Barker	Research Software Alliance	0000-0002-3623-172X	X		X	X	X	X	X
Morane Gruenpeter	Software Heritage, INRIA	0000-0002-9777-5560		X	X	X	X	X	X
Nadica Mijiković	University of Belgrade	0000-0002-3933-6076							X
Neil P. Chue Hong	Software Sustainability Institute / EPCC, University of Edinburgh	0000-0002-8876-7606	X	X	X	X	X	X	X

**Commented [357]:** subgroup 1 I believe, @dskatz@illinois.edu can you check please

**Commented [358]:** I don't see a record of this, though he was in the group. We probably should just ask him.

**Commented [359]:** I'm removing this, as there may have been some confusion over this column - this is to recognise contributions to the monthly WG meetings, rather than any FAIR4RS workshops or plenary sessions.

**Commented [360]:** please contact me or register to be part of the group via <https://www.rd-alliance.org/groups/fair-research-software-fair4rs-wg>

Nick Lynch	Curlew Research	0000-0002-8997-5298					X		<del>X</del>
Patricia Herterich	Digital Curation Centre	0000-0002-4542-9906						X	<del>X</del>
Paula Andrea Martinez	Research Software Alliance	0000-0002-8990-1985		X	X	X	X	X	X
Paula Martinez Lavanchy	TU Delft Library	0000-0003-1448-0917					X		
<a href="#">Sandra Gesing</a>	<a href="#">University of Notre Dame</a>	<a href="#">0000-0002-6051-0673</a>				<del>X</del>			
Sarah Stevens	University of Wisconsin-Madison	0000-0002-7040-548X					X		
Sergio Martinez Cuesta	University of Cambridge and AstraZeneca	0000-0001-9806-2805					X		
Sharif Islam	Naturalis Biodiversity Center, Distributed System for Scientific Collections (DiSSCo)	0000-0001-8050-0299						X	
Silvio Peroni	University of Bologna, OpenCitations	0000-0003-0530-4305					X		
Stian Soiland-Reyes	The University of Manchester						X		
Susanna-Assunta Sansone	University of Oxford	0000-0001-5306-5690							X
Tom Honeyman	Australian Research Data Commons	0000-0001-9448-4023		X		X	X	X	X

Tom Pollard	PhysioNet	0000-0002-5676-7898						X	
Tovo Rabemanantsoa	French National Institute for Agricultural Research (INRAE), Food and Environment	0000-0002-8362-9474					X		
Udayanto Dwi Atmojo	Aalto University	0000-0002-6865-0806						X	
Vanessa Sochat	Stanford University	0000-0002-4387-3819					X		
Yo Yehudi	<a href="#">Wellcome Trust</a> Please add	0000-0003-2705-1724					X		

