Open comments for the NIH Request for Information (RFI): *Best Practices for Sharing NIH Supported Research Software* (NIH, 2023)

Teresa Gomez-Diaz (University Gustave Eiffel-LIGM-CNRS, France),
Tomas Recio (University Antonio de Nebrija, Spain)
Contact: Teresa.Gomez-Diaz@univ-eiffel.fr, trecio@nebrija.es

January 17th, 2024, V1.2

1 Foreword

The goal of this document is to openly contribute with our comments to the Request for Information (RFI): *Best Practices for Sharing NIH Supported Research Software*[1] (NIH, 2023), request that we have found in [22].

The USA National Institutes of Health (NIH) seeks responses to the following questions regarding research software sharing best practices.

Item1 Comment on the current NIH Best Practices for Sharing Research Software[2]:

- I1.1 Why should I share software and code as “open source” software?
- I1.2 How do I make software source code “open”?
- I1.3 Why should I use a license when distributing code?
- I1.4 How do I choose a license under which to release software developed as part of an NIH award?
- I1.5 How can I make my software citable?
- I1.6 How should I acknowledge NIH as the funder?
- I1.7 Are there any restrictions I should consider in deciding whether to share the research software I develop?
- I1.8 Can research software I have developed be allowed for use in medical practice or clinical settings?
- I1.9 Do I have to check software developed for security vulnerabilities prior to sharing it?
- I1.10 What metadata should be considered when sharing research software?
- I1.11 To what extent should I include documentation for the software?
- I1.12 Does NIH have any requirements or benchmarks for research software quality before releasing it?

I2 Describe how, when, and where you share your research software. What, if any, resources for best practices do you rely upon to make your shared software open and reusable?

I3 What existing standards or criteria do you use to evaluate the openness, FAIRness, quality, and/or security of the software you share or reuse?

[1] https://grants.nih.gov/grants/guide/notice-files/NOT-OD-24-005.html

[2] https://datascience.nih.gov/tools-and-analytics/best-practices-for-sharing-research-software-faq

I4 Describe the collaborative settings in which you develop and share research software. Name communities or organizations, if any, you participate in that are actively promoting or developing software sharing best practices.

I5 What factors influence your decision to share or reuse your research software (or not)? What technical, policy, financial, institutional, and/or social barriers to sharing or reuse of research software have you encountered?

I6 Comment on your ability to reuse open-source research software developed by others. Describe factors used to determine whether to reuse existing research software or develop anew.

I7 How can NIH support research software communities of practice to better aid development of best practices for sharing and reuse of high-quality research software?

I8 Comment on any other topic which may be relevant for NIH to consider in enhancing the sharing of research software.

The detailed statement of these questions is included here for two reasons. First, because they provide a complete and thorough vision of the issues that appear in the Research Software (RS) development, sharing, and dissemination landscape. Secondly, to highlight that the context of this vision is very ample, and to provide answers to all the raised questions is too difficult and out of the scope of a single reaction to the *Request for Information*. Thus, the present contribution selects a partial set of these questions and proposes some answers, based on our expertise and published work, referenced at the end of the document. We hope that they will hep to the NIH to improve RS sharing and dissemination conditions and practices in research communities.

2 Questions and Answers

Note: in this document we use **I** for Item, **Q** for Question, **A** for Answer. In our experience [3, 4, 14] sustained by [8, 11] and many of the other publications referenced here, the general concept of Research Software (RS) is one of the most important issues to deal with, as it is behind most of the queries collected in the Items of the above list. For this reason, our Questions & Answers reflections start (see Questions & Answers 1 to 3) reflecting on this global concept. Then (Questions & Answers 4 to 15) we will address (and include a reference to) specific items from the precedent list.

Question1. What means Research Software (RS)?

Answer1. The research software definition provided in section 2.1 of [11] is the following:

> **Research Software** *is a well identified set of code that has been written by a (again, well identified) research team. It is software that has been built and used to produce a result published or disseminated in some article or scientific contribution. Each research software encloses a set (of files) that contains the source code and the compiled code. It can also include other elements as the documentation, specifications, use cases, a test suite, examples of input data and corresponding output data, and even preparatory material.*

In [11] we provide a thorough discussion and analysis that has driven us to propose this definition. As a conclusion of this RS definition, we have:

- what is done: code, software as a well identified set of files,
- who does it: author(s), but also contributors and/or scientific expert(s),
- to make what: research, science, that is, there are associated scholar publications,
- and the most important point is the quality and correctness of the produced scientific results.

Q2. What means a Research Team (RT)?

A2. We have also provided a definition of what means Research Team in [21]:

> **Research Team** *is a well identified set of persons that are involved in whatever ways to produce a result published or disseminated in some article or scientific contribution in the academic context.*

In this case, the academic contribution is a RS.

Q3. What means sharing a RS? This question is related to item **I1.1**.

A3. Sharing RS means that the producer RT gives the RS to another person/team external to the project or that the team makes the RS available in some web page or a repository, cloud, etc.

Q4. Why should I share software and code as "open source" software? This question is related to item **I1.1**.

A4. At the very moment the RT gives the RS away, that is, the *dissemination step*, the legal conditions to use, study, copy, modify, and redistribute the software should be insured, that is, it is necessary to establish the RS *sharing conditions.*

Please note that the authors of the present document are not legal experts, in spite of the fact that they have acquired some basic legal notions in copyright and licensing issues [5, 8, 9, 19], mainly in the French, Spanish and European legal context. It is not our intention, nor the intention of the present document, to provide answers to this kind of legal questions. If you have some, please refer to the legal experts in your Head Institutions.

Q5. How do I make software source code "open"? Why should I use a license when distributing a RS? This question is related to items **I1.2**, **I1.3**.

A5. In order to ensure the legal conditions for use, study, copy, modify, and redistribute the RS, our work has been centered into accompany the RS with a free/open source license [5, 6, 7, 8, 14, 19], that is, to disseminate the RS as free/open source software. Other legal means do exists, like establishing collaboration contracts, which should be discussed with the legal experts of the RT Head Institutions.

Please note that sharing only the executable part of the RS provides with black boxes that can not ensure the study nor the reproducibility of the scientific results obtained with the RS, as no access to the source code hinders, and most of the times severely obstructs, the study of the functioning of the RS.

The license corresponds to the legal means to protect potential users, collaborators and authors of the RS. Software licenses and licensing information can be found at the Free Software Foundation (FSF)[3], the Open Source Initiative (OSI)[4], and the Software Package Data Exchange (SPDX)[5].

The license can contain *as is* clauses to deal with warranties (or no warranties) about quality or features of the RS, as well as *reciprocity clauses* that should be respected.

Q6. How do I choose a license under which to release a RS? This question is related to item **I1.4**.

A6. There are several kinds of arguments to choose a license, for example related to a specific legal context (USA law, French law, European law...). For example, in France, the list of licenses that can be used in your own RS is given in a Décret[6]. The use of other licenses is possible, but this needs to follow a legal procedure.

Other reasons can be related to the reciprocity clauses that appear in the licenses of software (or other RS) components that have been included in your own RS. These clauses should be respected, see for example [2, 1, 5, 19] for more information.

[3] `https://www.fsf.org/licensing/`

[4] `https://opensource.org/licenses/`

[5] `https://spdx.org/licenses/`

[6] `https://www.data.gouv.fr/fr/licences`

Q7. How can I make my software citable? This question is related to item **I1.5**.

A7. Among many other works on software citation issues, we have proposed in section 2.5 of [11] three possible ways to cite your RS that take into account practices that already well established in some research communities:

- *the reference to a research software paper or other kind of scientific publication that includes, and relies on, a software peer review procedure, or*
- *the reference to a standard research article that includes a description of the RS and the implemented algorithms, explaining motivations, goals and results, or*
- *a typical label, associated to the RS itself, and that identifies it as a research output, specifying its title, authors, version, date, and the place the software can be recovered from.*

See also section 5 of [21] for further discussion on citation and referencing issues.

Q8. Can research software I have developed be allowed for use in medical practice or clinical settings? This question is related to item **I1.8**.

A8. The conditions in which your RS is to be used in medical practice or clinical settings should be carefully considered by the RT and clearly stated in its user documentation, the website of the project... It should also be carefully studied by the user team, and in particular, users should be well aware of clauses *as is* specified in the RS license.

Q9. What metadata should be considered when sharing research software? This question is related to item **I1.10**.

A9. As presented in section 2.5 of [11] (references have been revisited and do refer to the ones included at the end of the present document):

> *A more complex way for RS identification than a citation form is the use of metadata sets. The Software Citation Implementation Working Group has worked over several possibilities for software metadata sets*[7]*. The PRESOFT (Preservation for REsearch SOFTware) metadata set proposed for RS in [10] is built over the skeleton of the RS reference cards that where published between 2008 and 2013 by the PLUME project [12]. This metadata set benefits from the PLUME experience, which validates the proposed model, and sets a reasonable level of standards to manage RS identification.*

The metadata set proposed in section 1 of [10] is the following:

1. Software name. If you need to choose a name, avoid the name of a brand and other software names.
2. Short software description. A short sentence describing your software.
3. Software web page or website.
4. Link to source code or package.
5. Contact. Email address.
6. Research unit in charge of the software.
7. Main developers and their affiliations.
8. Software version.
9. Date of the software version.

[7] `https://force11.org/groups/software-citation-implementation-working-group/`

10. License.
11. Scientific discipline. Classifications may follow ACM Computing Classification System[8] or more general classifications[9].
12. Main functionalities. Give for example some Keywords.
13. Main technical characteristics. Give for example some Keywords.
14. Other keywords.

Q10. To what extent should I include documentation for the software? This question is related to item **I1.11**.

A10. Our vision on this important issue of documentation is that it can have several levels. The more basic level is to explain summarily the main functionalities of the RS, and provide basic examples of use, with input and output files. The use examples give hints about how the software is to be launched, how to write an input file, and which is the kind of output that users are to expect. The idea is to give basic information about what the RS is able to do, and the examples can be easily modified by potential users to understand how to use the software, and how to write their own use examples.

Large development RTs may have the necessary resources to build and give sound documentations for users and for the potential future development teams.

But the users, as well as the potential future development teams, should be well aware that the main goal of a RT providing a RS is the research work, and maybe not to provide sound maintenance or documentation during months or years after the RS has been released, as these tasks usually do take time and resources, and the RT may have not the needed support or resources (human, financial,...) to deal correctly with these tasks at short or long term.

The RT should carefully consider the level of documentation and/or maintenance that is to be provided and clearly stated the sharing and dissemination conditions of the RS.

Q11. Which best practices do you rely upon to share and disseminate your software? This question is related to item **I2**.

A11. A sharing and dissemination procedure has been proposed in [6, 7, 18]. The last and revised version has been presented in [18], but the most detailed study is in the initial reference [6] (in French). We include here a short and slightly modified version of the RS dissemination procedure that can be found in [18]. Steps marked with (*) are to be revisited regularly for each version release.

- Choose a name or title to identify the RS, avoid trademarks and other proprietary names, you can associate date, version number, target platform... Consider best practices in file names.
- (*) Establish the list of authors and affiliations (this is the so called *research team step*). An associated percentage of participation, completed with minor contributors can be useful. If the list is too long, keep updated information in a web page or another document like a Software Management Plan (SMP) [10], for example, where you can mention the different contributor roles. This question about authors and contributors has been revisited in [21]. This is the step in which the intellectual property producer's rights are to be established. Producers include the RS authors and rightholders. This is then the step in which RS legal issues related to copyright information are dealt with.
- (*) Establish the list of included software and data components, indicate their licenses (or other documents like the component's documentation...) giving the rights to access, copying, modification and redistribution for each component. Take into consideration best citation practices.

[8] `https://dl.acm.org/ccs`

[9] `https://en.wikipedia.org/wiki/Outline_of_academic_disciplines`

- Choose a software license, with the agreement of all the rightholders and authors, and establish a signed agreement if possible. The licenses of the software components that have been included and/or modified to produce the RS can have impact in your license decision. Software licenses and licensing information can be found at the Free Software Foundation (FSF)[10], the Open Source Initiative (OSI)[11], and the Software Package Data Exchange (SPDX)[12]. Consider using FLOSS licenses to give the rights of use, copy, modification, and/or redistribution. This is then the step in which legal issues related to the RS sharing conditions are to be taken into consideration. Indicate the license in the RS files, its documentation, the project web pages... Give licenses like GNU FDL[13], Creative Commons (CC)[14], LAL[15]... to documentation and to web sites related to the RS.
- Choose a web site, forge, or deposit to distribute your product; licensing and/or conditions of use, copy, modification, and/or redistribution should be clearly stated, as well as the best way to cite your work. Good metadata and respect of open standards are always important when giving away new components to a large community: it helps others to reuse your work and increases its longevity. Use Persistent Identifiers (PIDs)[16] if possible.
- (*) This step deals with the utility of the RS and how it has been used for your research (this is the *research work step*). Establish the list of main functionalities, and archive a tar.gz or similar for the main RS versions in safe place. Keep a list of the associated research work, including published articles. Update your documentation, SMP, web site... with the new information in each main RS version.
- Inform your laboratories and head institutions about this RS dissemination (if this has not be done in the license step).
- Create and indicate clearly an address of contact.
- Release the RS.
- Inform the community (mailing lists...), consider the publication of a software paper, see for example the list of Journals where you can publish articles focusing on software[17].

This proposed procedure is flexible and can be adapted to many different situations, and can also be used for Research Data [18].

Q12. What existing standards or criteria do you use to evaluate the RS? This question is related to item **I3**.

A12. The CDUR evaluation protocol has been initially formulated for RS in [11] and has been extended to Research Data (RD) in [18]. It has been designed to help evaluators, members of evaluation committees, and evaluated researchers, members of the RT responsible of the RS or RD. We include here a short presentation as given in [20] (with minor modifications), but, please, do refer to [11] for a thorough presentation and discussion of this protocol.

There are four steps in the proposed evaluation protocol, which is flexible enough to be applied in different evaluation contexts:

(C) Citation. This step measures if the RS or RD are well identified as a research output, i.e. if there is a good citation form, or good metadata. We look here to best citation practices: te citation form that is proposed for your RS and/or RD, and how you cite RS and RD done by other RTs.

This is a legal related point where we ask for authors (if any) are well identified, which are their affiliations, and for example the % of their participation in software writing.

[10] `https://www.fsf.org/licensing/`

[11] `https://opensource.org/licenses`

[12] `https://spdx.org/licenses/`

[13] `http://www.gnu.org/copyleft/fdl.html`

[14] `https://creativecommons.org/choose/`

[15] `http://artlibre.org/licence/lal/en/`

[16] `http://en.wikipedia.org/wiki/Persistent_identifier`

[17] This list is mantained by Neil Chue Hong in the Software Sustainability Institute web page `https://www.software.ac.uk/which-journals-should-i-publish-my-software`

(D) Dissemination. In this point we look to best dissemination practices, in agreement with the scientific policy of the evaluation context. The dissemination of RS and RD needs a license to set the sharing conditions. For RD there are maybe further legal issues to look at (personal data, *sui generis* database rights...).

This is a policy point in which we look at Open Science requirements [13].

(U) Use. This point examines "software" or "data" aspects, in particular the correct results that have been obtained, and we can also look if their reuse has been facilitated, the output quality, best software/data practices such as documentation, testing, installation or reuse protocols, up to read the code, launch the RS, use examples...

This is the reproducibility point that looks at the validation of the scientific results obtained with the RS and/or the RD.

(R) Research. This point examines the research aspects associated to the RS and/or RD production: the quality of the scientific work, the proposed and coded algorithms and data structures, which are the related publications, the collaborations, the funded projects...

This point measures the impact of the RD and/or RS related research work.

Q13. What technical, policy, financial, institutional, and/or social barriers to sharing or reuse of RS have you encountered? This question is related to item **I5**.

A13. Among all the possible barriers that RTs may usually encounter when sharing RS, we would like to mention the following ones.

To share RS does take time and ressources, and many times it involves tasks (technical, administrative...) that are far away of the research interests of a RT. It also needs to take into account legal issues (intellectual property rigths, licensing...), and usually RTs are not very much aware of legal matters, nor do find easily the help needed to deal with these questions. Many times, the research work is done in the framework of international collaborations, which does not help to deal with these issues easily.

On the other hand, and even if this is changing a lot nowadays, the scientific work is evaluated mainly regarding the publications, and does not take into account the production of RS or other outputs. Until now, to do the effort of RS dissemination has not much scientific value, and it is usually done in order to facilitate the reproducibility of the scientific results, to increase citations or to look for collaborations with other RTs that may have common scientific interests.

Finally, the Research Institutions are currently adopting new Open Science policies [13, 23], and it will take time until these policies are well installed, adopted and followed.

Q14. Comment on your ability to reuse RS developed by others. This question is related to item **I6**.

A14. Our intention with the proposition of the dissemination procedures and the CDUR protocols is to help the scientific community at large to promote and adopt RS (and RD) best practices on sharing, dissemination and evaluation of these research outputs. As a consequence, RTs will be more aware of procedures that should be followed in order to share and disseminate their outputs, and their work will be better recognized. It will also be taken in to account in evaluation contexts such as recruitment or career evolution. The RTs will also facilitate the reuse conditions for their outputs.

If these proposed procedures and protocols are largely adopted, they will also help to potential users to better and quickly recognize research outputs shared and disseminated in good conditions, that have taken into account, in particular, the facilitation of their reuse, as the Use step of the CDUR protocols deals with this issue in a precise and transparent way.

As studied in [13, 15, 16], there is a dissemination/evaluation loop: if you improve the evaluation context, the RTs will adapt to this new context and improve the dissemination conditions of their research outputs, facilitating thus their reusability.

Q15. How can NIH support research software communities of practice to better aid development of best practices for sharing and reuse of high-quality research software? This question is related to item **I7**.

A15. We would like to propose three points that could be considered by NIH:

1. Establish clear Open Science policies that include the sharing and dissemination conditions in which NIH Research Software is to be rendered visible, accessible and reusable [13, 23].
2. Establish clear dissemination procedures to widespread best dissemination practices, like the ones proposed in [6, 7, 18].
3. Establish clear RS evaluation protocols like the CDUR ones proposed in [11, 18].

Acknowledgments. We would like to thank the NIH for such an inspiring work.

References

[1] Aimé T, (2010). A Practical Guide to Using Free Software in the Public Sector, DGI - Ministry for the Budget, Public Accounts and the Civil Service, https://zenodo.org/records/7096100

[2] Fogel K, (2005-2022). Open Source Software. How to Run a Successful Free Software Project https://producingoss.com/

[3] Gomez-Diaz T (2007-2013). Le thème PLUME Patrimoine logiciel d'un laboratoire, Zenodo Community, https://zenodo.org/communities/plume-patrimoine-logiciel-laboratoire/

[4] Gomez-Diaz T, (2007-2024). Page web with presentations, articles, posters and other productions since 2007, Laboratoire d'informatique Gaspard-Monge (LIGM), http://igm.univ-mlv.fr/~teresa/logicielsLIGM/

[5] Archimbaud JL, Gomez-Diaz T, (2009). FAQ : licence & copyright pour les développements de logiciels libres de laboratoires de recherche, Publication for the French PLUME project (2007-2013), https://zenodo.org/records/7063146

[6] Gomez-Diaz T, (2010). Diffuser un logiciel de laboratoire : recommandations juridiques et administratives, Publication for the French PLUME project (2007-2013), https://zenodo.org/record/7096216

[7] Gomez-Diaz T, (2014). Free software, Open source software, licenses. A short presentation including a procedure for research software and data dissemination, Zenodo Preprint, Septembre 2014, http://zenodo.org/record/11709/. Presented at the Workshop on open licenses: Data licencing and policies, EGI Conference 2015, Lisbon, May 2015, https://indico.egi.eu/event/2452/sessions/1522/#20150520. Spanish version: Software libre, software de código abierto, licencias. Donde se propone un procedimiento de distribución de software y datos de investigación, Zenodo Preprint, Septembre 2015, https://zenodo.org/record/31547/

[8] Gomez-Diaz T, (2015). Article vs. Logiciel : questions juridiques et de politique scientifique dans la production de logiciels. 1024 - Bulletin de la société informatique de France, N. 5, 2015, http://www.societe-informatique-de-france.fr/wp-content/uploads/2015/04/1024-5-gomez-diaz.pdf, also available at https://hal.science/hal-01158010. First version published for the French PLUME project (2007-2013) in 2011, https://zenodo.org/record/7063154

[9] Gomez-Diaz T, (2017). Software legal protection, European IPR Helpdesk Bulletin Issue (26), http://igm.univ-mlv.fr/~teresa/logicielsLIGM/documents/Internacional/European_IPR_Helpdesk_Bulletin_Issue_26.pdf

[10] Gomez-Diaz T, Romier G, (2018). Research Software management Plan Template V3.2. Projet PRESOFT, Bilingual document (FR/EN), Zenodo Preprint, https://zenodo.org/records/1405614

[11] Gomez-Diaz T and Recio T, (2019). On the evaluation of research software: the CDUR procedure [version 2; peer review: 2 approved], F1000Research 2019, 8:1353, https://doi.org/10.12688/f1000research.19994.2. See also https://pubmed.ncbi.nlm.nih.gov/31814965/

[12] Gomez-Diaz T, (2019). Le Projet PLUME et le paysage actuel des logiciels de la recherche dans la science ouverte. Zenodo Preprint, 2019. https://zenodo.org/record/2591474

[13] Gomez-Diaz T and Recio T, (2020-21). Towards an Open Science definition as a political and legal framework: on the sharing and dissemination of research outputs, Version 2 published in POLIS N. 19, december 2020, https://doi.org/10.58944/yuro5734. Version 3 available in Zenodo https://zenodo.org/record/4577066

[14] Gomez-Diaz T (2021). Free/Open source Research Software production at the Gaspard-Monge Computer Science laboratory (LIGM). Lessons learnt, FOSDEM21, Open Research Tools and Technologies devroom, https://fosdem.org/2021/schedule/event/open_research_gaspard_monge/

[15] Gomez-Diaz T and Recio T, (2022). On the dissemination/evaluation loop for Research Software, FOSDEM 2022, Open Research Tools and Technologies devroom, https://archive.fosdem.org/2022/schedule/event/open_research_cdur/

[16] Gomez-Diaz T and Recio T, (2022). Research Software and Research Data: dissemination, evaluation and reusability in the Open Science context, Poster, 17th International Digital Curation Conference IDCC22, https://zenodo.org/records/6778872

[17] Gomez-Diaz T and Recio T, (2022). Research Software vs. Research Data I: Towards a Research Data definition in the Open Science context [version 2; peer review: 3 approved], F1000Research 2022, 11:118, https://f1000research.com/articles/11-118/v2. See also https://pubmed.ncbi.nlm.nih.gov/36415208/

[18] Gomez-Diaz T and Recio T, (2022). Research Software vs. Research Data II: Protocols for Research Data dissemination and evaluation in the Open Science context [version 2; peer review: 2 approved], F1000Research 2022, 11:117, https://f1000research.com/articles/11-117/v2. See also https://pubmed.ncbi.nlm.nih.gov/36483317/

[19] Gomez-Diaz T, (2023). Les logiciels de la recherche et leurs licences : trois visions sur un objet, Version 2, Training support, University Gustave Eiffel, Marne-la-Vallée, France, 2023, 58 pp., https://hal.science/hal-02434287v2

[20] Gomez-Diaz T and Recio T, (2023). A Code for Thought podcast: Research software and Research Data in Open Science, July 14th 2023, Zenodo Preprint, https://zenodo.org/record/8159906

[21] Gomez-Diaz T and Recio T, (2023). Articles, software, data: An Open Science ethological study, Maple Transactions, 3, 4, Article 17132 (December 2023), https://doi.org/10.5206/mt.v3i4.17132

[22] Nature Computational Science Editorial (2023). Code sharing in the spotlight, Nature Computational Science volume 3, page 907 (2023), https://doi.org/10.1038/s43588-023-00566-4

[23] Office of Science and Technology Policy (OSTP) (2022). Memorandum for the heads of executive departments and agencies for Ensuring Free, Immediate, and Equitable Access to Federally Funded Research, August 25, 2022, https://www.whitehouse.gov/wp-content/uploads/2022/08/08-2022-OSTP-Public-access-Memo.pdf