# Components of an Open Science Publication

Replicability and Reproducibility in Geospatial Research: a SPARC Workshop

Vaclav Petras, Helena Mitasova, Ross K. Meentemeyer
Center for Geospatial Analytics, North Carolina State University, Raleigh, NC, USA

October 26, 2018

As there is limited consensus on the definitions and importance of replicability and reproducibility (R&R) (Barba, 2018), we propose an evaluation framework assessing R&R of publications by identifying components of a publication in a broad open science context similarly to, e.g., Peng et al., 2006. The components we describe are: text, data, reusable code, publication-specific code, computational environment, and versioning. Additionally, we also introduce concept of a scientific software platform which is especially important for the reusable code component.

**Geospatial and Computational Science** Here we mainly focus on publications with novel methods and software. Consequently, the components would to a large extent apply to geospatial science as well as computational science. However, there are some important differences. Computational science usually involves writing new code while in the geospatial field many studies involve little or no code development, but rely on applications or extensions of existing software and code.. Recommendations for R&R in computational science are primarily aimed at scientists themselves (e.g., Gent et al., 2014), while geospatial research results need to be often applied outside of the original research field by professionals who don't consider themselves scientists or by scientists who don't develop computational methods. This brings additional challenges to specifying the components of an open geospatial science publication.

**Text Component** Text, as a natural language description of the research, contains sections such as background, methods, results, and discussion and is combined with tables and figures. The text should include description of where to access all the other components. This can be described in an availability section, required or recommended by some journals today, or in the most relevant section, such as methods or results.

**Data Component** Researchers can take advantage of a number of different repositories to make scientific data broadly available. These range from general scientific repositories to repositories dedicated to a specific type of data such as hydrologic or topographic data. In order to reduce number of steps needed for reproducibility, the data published following the presented framework should be included in the computational environment as discussed below. However, for many geospatial applications that is not possible due to large size of datasets. In that case, smaller dataset can be provided as part of the environment or an automatic download from an online storage can be included.

**Reusable Code Component** Reusable code consists of implementation of methods presented in the paper (text component) and it should provide user or programming interfaces to be reusable by other scientists with different data. However, to provide interfaces and other convenience functions, we propose leveraging existing software projects. The new published reusable code can be integrated into the code base and submitted as a code contribution or, if the software provides it, the code can be submitted as a user-developed extension, plugin, tool, library, or package. This integration is further discussed in the context of a scientific software platform below.

**Publication-Specific Code Component** The code which is not reusable but was used to generate figures, tables, and any other computational results is called publication-specific code in this framework. The publication-specific code should make use of the reusable code for any significant and novel data analysis. The code contains specific values like interpolation parameters, number of stochastic runs, and even hardcoded paths if it is distributed in a given computational environment as discussed below.

**Computational Environment Component** To reproduce, specifically recompute, a published result, we typically need publication-specific code, reusable code, and the data. However, just bundling these three components together is not enough. Research usually depends on other software, e.g., specific libraries for statistical analysis. Additionally, all code and software require certain version of dependencies and environment to run in. To address the need to specify, share, and distribute these dependencies, the computational environment component should contain all the necessary information to run the published code.

**Versioning Component** As all of the components are changed over time in the research and development process, preserving the history, explaining the changes, and marking important milestones becomes necessary part of identifying the status of the research and understanding the reasons behind particular decisions. One or more repositories then contain all previous versions of the text, data, or code, authoritative copy of the current version, and possibly also all alternative versions.

**Scientific Software Platform** Although more broad R&R and even reusability is addressed by the reusable code component, R&R of a publication as discussed above is mainly focusing on obtaining again the published computational results. The challenge to long-term R&R and reusability is that if the reusable code is standalone, it is less likely to be preserved, e.g., when the original researcher moves on to a different topic. We envision that a scientific software platform or platforms should ensure, among other things, active code and longevity. Platforms can be new or existing software packages which fulfill certain criteria such as integration and interoperability with other software, accessibility in terms of distribution and research code dissemination, different language choices because of applicability of different languages to different problems (Baxter et al., 2006), user interfaces to accommodate different workflows, and finally also handling of citations (Löwe et al., 2017).

**Extending Curriculum** Clearly, getting a working knowledge of the R&R methods and tools as well as understanding the needs is a significant endeavor for students learning about their field which, in case of geospatial research, is already likely interdisciplinary. However, we also view these skills as something which is simply needed alongside with writing, communication, and visualization skills. At the Center for Geospatial Analytics at North Carolina State University, we created a new course called Tools for Open Geospatial Science which is aimed at providing the minimal required skill set.

# References


Barba, L. A. (2018). "Terminologies for Reproducible Research". In: *arXiv:1802.03311 [cs]*. arXiv: 1802.03311.

Baxter, S. M., S. W. Day, J. S. Fetrow, and S. J. Reisinger (2006). "Scientific Software Development Is Not an Oxymoron". In: *PLOS Computational Biology* 2.9, e87. ISSN: 1553-7358. DOI: 10.1371/journal.pcbi.0020087.

Gent, I. P. and L. Kotthoff (2014). "Recomputation.org: Experiences of Its First Year and Lessons Learned". In: *Proceedings of the 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing*. UCC '14. Washington, DC, USA: IEEE Computer Society, pp. 968–973. ISBN: 978-1-4799-7881-6. DOI: 10.1109/UCC.2014.158.

Löwe, P., M. Neteler, J. Goebel, and M. Tullney (2017). "Towards OSGeo Best Practices for Scientific Software Citation: Integration Options for Persistent Identifiers in OSGeo Project Repositories". In: *Free and Open Source Software for Geospatial (FOSS4G) Conference Proceedings*. Vol. 17, p. 29.

Peng, R. D., F. Dominici, and S. L. Zeger (2006). "Reproducible Epidemiologic Research". In: *American Journal of Epidemiology* 163.9, pp. 783–789. ISSN: 0002-9262. DOI: 10.1093/aje/kwj093.