



Open Science: Yes You Can

Cédric H. David

2022-11-09

NASA Earth Surface and Interior Solid Earth Team
Meeting, La Jolla, CA.



Jet Propulsion Laboratory
California Institute of Technology

© 2022 Jet Propulsion Laboratory, California Institute of Technology

This document has been reviewed and determined not to contain
export controlled technical data.

The components of Open Science

“This AGU journal wants me to publish my code and data, I have no clue how”

“My team members keep building the same code to analyze the same data, I wish there was a better way”

“Can you believe how these authors analyzed data from my satellite? They have no clue!”

“Sorry, that was like 10 years ago, I don’t remember this specific detail of my analysis”

“I’m afraid to update my code, it might break things”

“Can’t do open science at NASA, I’ll get fired!”



1. Paper



4. Methods

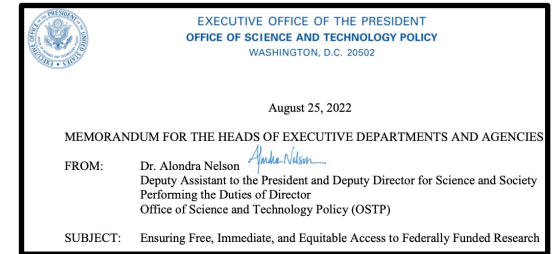
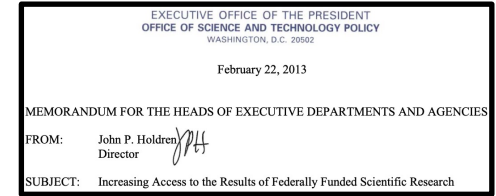


2. Data



3. Software

Open science, you don’t have to do it (or do you?), but you might learn some tricks regardless



5. Example
6. Bonus
7. Challenges/Limits
8. “How to” at JPL?
9. Recommendations
10. Reads

Open Paper

- Publish your paper
 - Good: Drop your submitted manuscript in a NASA repository or an open archive (e.g. Earth and Space Science Open Archive ESSOAr)
 - Better: Choose journals that offer an option for open access fees and pay the fee
 - Best: Choose one of the many journals that only offers open access because NASA/JPL won't pay for optional open access fees

TL, DR

Where?

Science Advances
Scientific Reports
AGU Advances
AGU Earth and
Space Science
EGU journals

License?

Selected by journal

Open Data

- Publish your data
 - Good: pick a license (<https://creativecommons.org/choose/>) and drag/drop your files to Zenodo (or FigShare)
 - Better: include description of the data
 - Best: adopt popular file formats and metadata conventions in your field (e.g. netCDF Climate and Forecast Conventions)

Free of charge, version controlled, and you get DOIs!

TL, DR

Where?

<https://zenodo.org>

License?

CC-BY

Open Software

- Publish your code
 - Good: pick a license (<https://choosealicense.com/>) and drag/drop your files to GitHub (or BitBucket)
 - Better: include some description (can be comments within code) and basic installation instructions
 - Best: Describe software dependencies (Linux: apt-get, Mac: brew, Windows: Chocolatey; Python: pip), use continuous integration (e.g. Travis), release an image (e.g. Docker)

Free of charge, version controlled, DOIs with Zenodo!

TL, DR

Where?

<https://github.com>

License?

BSD 3-Clause

Open Methods

- Publish your methods
 - Good: short file with copy/paste of how your program was run
 - Better: executable scripts
 - Best: Jupyter Notebook

It'll help remember how you actually did this!

TL, DR

Where?

With software

License?

Same as software

Bonus stuff you'll learn to love about open science

Continuous Integration (Travis CI)

Travis CI Dashboard for repository `c-h-david / rapid`. The interface shows a list of recent builds on the left and a detailed view of the current build on the right. The current build is for commit `20210423`, split into two jobs, both of which passed. The build status is `passing`.

Repository	Build #	Status	Duration	Finished
<code>c-h-david/rrr</code>	# 262	✓	27 min 45 sec	10 days ago
<code>c-h-david/rapid</code>	# 186	✓	10 hrs 55 min 24 sec	27 days ago
<code>c-h-david/shbaam</code>	# 122	✓	31 min 8 sec	about a year ago
<code>c-h-david/moshpyt</code>	# 8	✓	4 min 24 sec	2 years ago

Commit	Author	Build #	Status	Duration	Finished
<code>20210423</code>	Cedric H. David	#186	passed	10 hrs 55 min 24 sec	27 days ago
<code>master</code>	Cedric H. David	#185	passed	11 hrs 22 min 50 sec	27 days ago
<code>master</code>	Cedric H. David	#184	errored	11 hrs 26 min 52 sec	28 days ago
<code>master</code>	Cedric H. David	#183	errored	10 hrs 55 min 5 sec	28 days ago
<code>v1.8.0</code>	Cedric H. David	#182	passed	10 hrs 7 min 3 sec	about a year ago
<code>20200424</code>	Cedric H. David	#181	passed	9 hrs 56 min 35 sec	about a year ago

Containerization (Docker)

Docker Hub repository page for `chdavid/rapid`. The page shows the repository name, a description of the routing application, and a list of tags. The latest tag is `20210423`, pushed about a month ago.

Advanced Image Management: View all your images and tags in this repository, clean up unused content, recover untagged images. Available for Pro and Team accounts.

TAG	DIGEST	OS/ARCH	LAST PULL	COMPRESSED SIZE
<code>20210423</code>	<code>c734c858063b</code>	linux/amd64	25 days ago	773.55 MB

Peace of mind: every update (even minor is fully tested), you can know if you broke anything

Less email traffic: “I can’t install your code!”, “what’s wrong with my file?”

Faster research: next team member (yourself included) can grab the previous study and build on it

Community Challenges and the “Limits of Sharing”

- Technical training → we need to know/teach how to do this
- Self-perceived inadequacy → “not good enough”
- Documentation → takes time and effort
- Acknowledgment, evaluation, recognition of digital scholarship → “carrot” (citations, metrics, annual reviews, awards)
- Sustainable sharing → open science is not free support
- Keep up → rapidly-evolving tools for open science require time/effort

Suggested reads

commentary

Of carrots and sticks

Jens Katgtte, Sandra Diaz and Christian Wirth

Journals and funders increasingly require public archiving of the data that support publications. We argue that this mandate is necessary, but not sufficient: more incentives for data sharing are needed.

With the digital revolution, data exchange has become easy in a technical sense. As a result, data that were originally collected for one specific purpose can now be used in different contexts, to answer new scientific questions. Data sharing offers prospects for progress, and not only for data-intensive sciences like remote-sensing. Scientific domains that are typically dominated by numerous small data sets—such as ecology, biodiversity or medicine—stand to benefit, too.

Historically, data sharing was limited by the absence of centralized easily accessible archives for scientific data. Data were usually stored at the research institutions where they were produced, and formats ranged from a centralized institutional digital repository to hand-written field notebooks. Upon publication, the underlying data sets were typically shared via bilateral communication between researchers, and mostly used solely to repeat a given study and verify its results. In such informal structures, metadata are often lacking and data are prone to rapid loss of information content, which makes reuse difficult.

As the focus of much research is shifting towards larger-scale questions—such as global biodiversity, workforce or organizational specialisation patterns and

continental pandemics—and data collection is becoming ever more efficient, the approach to storing and disseminating data needs to change.

We argue that data sharing is already rewarded with recognition, increased citations and collaborations, but stronger incentives in terms of citations are overdue. Only if the full scientific value of generating and disseminating data is acknowledged, will data sharing become the integral part of the scientific system that it needs to be.

Three stumbling blocks

Constraints to data sharing are not more social than technical. Scientists tend to embrace the opportunity of sharing data in bilateral contexts, but they are often reluctant to release their data to the broader scientific community. The reasons for this are manifold, but we feel that three are particularly prominent.

First, high-quality data are hard to obtain. Researchers should expect their fieldwork to yield one or several primary publications. The originality of these publications might be jeopardised if the data are widely available before they are published, or if individual data sets are amalgamated in large collective scientific publications.

The challenge towards making hard-earned data publicly available

is understandable—at least as long as the measurements have not been sufficiently explored by those who obtained them.

Second, data are context dependent. Without appropriate contextual information, methods and shortcomings, they can easily be misinterpreted and misused. Opening data sets to other researchers means that the circumstances of collection—known by those who performed the measurements—need to be carefully recorded and communicated.

The third factor is related to the previous one: significant effort is often necessary to prepare the data for reuse before they can be made available to the scientific community. In addition to contextualising data, formats may have to be adjusted and a point of contact may be necessary in case there are questions.

In order to overcome these obstacles to voluntary data sharing, public archiving has been made mandatory by many publishers and funding agencies. However, this approach—using a proverbial stick to encourage data sharing—has not (yet) led to a broad cultural change in researcher actions. We therefore advocate complementary incentives—a carrot that will supplement the stick.

Big data, many references

There are already incentives for sharing high-quality data accompanied by all the relevant contextual information. Benefits to those who readily make their data available include the facilitation of new collaborations and the development of their professional network; researchers can enhance their own original data by allowing others to access it and contribute; data sharing can also result in joint publications with other groups who use the data, and it may yield data publications and hence citations beyond those of the original papers.

These benefits have perhaps not been cultivated as they could be, but improvements are under way. Examples include the development of domain-specific data repositories, which explicitly support networking opportunities (see Box 1). That collaborations and networks tend to form

WORLDVIEW

A personal take on events

Publish your computer code: it is good enough

Freely provided working code—whatever its quality—improves programming and enables others to engage with your research, says Nick Barnes.

I am a professional software engineer and I want to share a trade secret with scientists: most professional computer software isn't very good. The code inside your laptop, television, phone or car is often badly documented, inconsistent and poorly tested.

Why does this matter to science? Because to turn raw data into published research papers often requires a little programming, which means that most scientists write software. And you scientists generally think the code you write is poor: it doesn't contain good comments, has enable/disable variable names or proper indentation. It breaks if you introduce badly formatted data, and you need to edit the output by hand to get the columns to line up. It includes a routine written by a graduate student which you never completely understood, and so on.

That the code is a little raw is one of the main reasons scientists give for not sharing it with others. Yet, software in all trades is written to be good enough for the job intended. If your code is good enough to do the job, then it is good enough to release—and releasing it will help your research and your field. At the Climate Code Foundation, we encourage scientists to publish their software. Our experience shows why this is important, and how researchers in all fields can benefit.

Programs written by scientists may be small scripts to draw charts and analyse data, trends and significance, larger routines to process and filter data in more complex ways, or telemetry software to not be backed up from lab or field equipment. Often they are an awkward mix of these different parts, glued together with piecemeal scripts. What they have in common is that, after a paper's publication, they often languish in an obscure folder or are simply deleted.

Although the paper may include a brief mathematical description of the processing algorithm, it is rare for science software to be published or even readily pre-empted for reuse.

Last year's global fuss over the release of climate-science e-mails from the University of East Anglia (UEA) in Norwich, UK, highlighted the issue, and led to calls for scientists to publish code. My efforts pre-date the UEA incident and grew from work in 2008 based on software used by NASA to report global temperatures. Released on its website in 2007, the NASA code was messy and provided difficult for critics to run on their own computers. Most did not seem to try very hard, and nonsense was written about fraud and conspiracy. With other volunteers, I rewrote the software to make it easier for non-experts to understand and run. All software has bugs, and we found a number of minor problems, which had no bearing on the results. NASA fixed

them and now intends to replace its original software with ours. So, openness improved both the code used by the scientists and the ability of the public to engage with their work. This is to be expected. Other scientific methods improve through peer review. The openness movement has led to rapid improvements within the software industry. But science source code, not exposed to scrutiny, cannot benefit in this way.

NEEDS

If scientists stand to gain, why do you not publish your code? I have already discussed misplaced concerns about quality. Here are my responses to some other common excuses.

It is not open source. As explained above, this is more an issue of climate science and should do so across all fields. Some disciplines, such as bioinformatics, are already changing.

People will pick holes and demand support and bug fixes. Publishing code may see you accused of sloppiness. Not publishing can draw allegations of fraud. Which is worse? Nobody is entitled to demand technical support for freely provided code: if the feedback is unhelpful, ignore it.

The code is unscientific. Really, that little MATLAB routine to calculate a two-part fit with your own data is not backed by skilled experts' abandonment. Institutions should support publishing those who refuse are blocking progress.

It is too much work to polish the code. For scientists, the word publication is totemic, and signifies perfectionism. But your papers need not include meticulous pages of Fortran, the original code can be published as supplementary information, available from an institutional or journal website.

I accept that the necessary and inevitable change of culture cannot be made by scientists alone. Governments, agencies and funding bodies have all called for transparency. To make it happen, they have to be able to make the necessary policy changes, and to pay for training, workshops and initiatives. But the most important change must come in the attitude of scientists. If you are still resistant about releasing your code, then ask yourself this question: does it perform the algorithm you describe in your paper? If it does, your audience will accept it, and maybe feel happier with its own efforts to write programs. If not, well, you should fix that anyway. **#OPENRESEARCH**

Nick Barnes is director of the Climate Code Foundation, Sheffield S17 4DZ, UK. E-mail: nb@climatedcode.org

DISCUSS THIS ARTICLE ONLINE AT www.nature.com/naturecommentary

AGU PUBLICATIONS

Earth and Space Science

RESEARCH ARTICLE
10.1002/2015EA000142

Special Section:
Geoscience Papers of the Future

Key Points:
• A reflection on the open source development of geoscience code is presented.
• Sharing can be broken down into three phases: opening, exposing, and consolidating.
• Free online services facilitate sharing and allow for further academic credit.

Correspondence to:
C. H. David,
cedric.david@nasa.gov

Citation:
David, C. H., J. S. Famiglietti, Z. Yang, F. Habets, and D. M. Maitland. 2016. A decade of RPFD—Reflections on the development of an open source geoscience code. *Earth and Space Science*, 3, 226–244. doi:10.1002/2015EA000142

Received 19 OCT 2015
Accepted 22 MAR 2016
Accepted article online 7 APR 2016
Published online 19 MAY 2016

© 2016 The Authors.
This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution in any medium, provided the original work is properly cited. See <http://www.nature.com/open> for non-commercial and no modifications or adaptations are made.

A decade of RAPID—Reflections on the development of an open source geoscience code

Cédric H. David^{1,2}, James S. Famiglietti^{1,2,3}, Zong-Liang Yang⁴, Florence Habets⁵, and David M. Maitland⁶

¹Jet Propulsion Laboratory, California Institute of Technology, Pasadena, California, USA, ²Center for Hydrologic Modeling, University of California, Irvine, California, USA, ³Department of Earth System Science, University of California, Irvine, California, USA, ⁴Department of Geological Sciences, Jackson School of Geosciences, University of Texas at Austin, Austin, Texas, USA, ⁵UMR 7619 METIS, CNRS, UPMC, Paris, France, ⁶Center for Research in Water Resources, University of Texas at Austin, Austin, Texas, USA

Abstract Earth science increasingly relies on computer-based methods and many government agencies now require further sharing of the digital products they helped fund. Earth scientists, while often supportive of more transparency in the methods they develop, are concerned by this recent requirement and puzzled by its multiple implications. This paper therefore presents a reflection on the numerous aspects of sharing code and data in the general field of computer modeling of dynamic Earth processes. Our reflection is based on 10 years of development of an open source model called the Routing Algorithm for Parallel Computation of Discharge (RAPID) that simulates the propagation of water flow waves in river networks. Three consecutive but distinct phases of the sharing process are highlighted here: opening, exposing, and consolidating. Each one of these phases is presented as an independent and tractable increment aligned with the various stages of code development and justified based on the side of the users' community. Several aspects of digital scholarship are presented here including licenses, documentation, websites, stable code and data repositories, and testing. While many existing services facilitate the sharing of digital research products, digital scholarship also raises community challenges related to technical training, self-perceived inadequacy, community contribution, acknowledgment and performance assessment, and sustainable sharing.

1. Introduction

Driven by the need to understand Earth's dynamic climate, geoscientists have dedicated much effort to creating numerical models of the major components of the climate system and to analyzing their outputs. Early modeling studies date back to the 1950s and include simulations of the Earth's atmosphere (Phillips, 1956), oceans (Byron and Cox, 1967), land (Monsieur, 1969), and rivers (Miller et al., 1994). Decades later, computer modeling and data-intensive analysis have become key elements upon which modern climate science has been built (e.g., *Intergovernmental Panel on Climate Change*, 2013) and numerous geoscientists therefore dedicate considerable research energy to such endeavors. Computer-assisted research, especially ubiquitous in the broad scientific community, such that some have argued that computer modeling and data-intensive science is considered legitimate pillars of science, hence joining experimental science and theoretical science (Bell, 1987; Bell et al., 2009; Hey et al., 2009; Hey, 2010; Hey and Payne, 2015), although such a view is not without its critics (Ward, 2010a, 2010b). Nevertheless, computer modeling and analysis are now integral parts of many geoscientific investigations.

The recent mandate (Maitland, 2013) requesting that the direct results of federally funded scientific research in the U.S. be made further accessible—including availability of digital data—has spurred much discussion in the scientific community on ways to improve data sharing in research. Associated hurdles, however, exist, and proper means of acknowledgment (i.e., citations) are needed so that scientists can benefit from the added burden. This argument was further supported by the survey of Kratz and Stasser (2015). Others have also suggested that the computer codes used to generate or to analyze data are equally important and should hence be made similarly accessible (Hartze, 2014; Hartze Geoscience, 2014). Prior to the recent mandate, Barnes (2010) had already advocated for sharing computer code so that—like any other scientific method—code development could benefit from the peer review process. Additionally, the description of computations—using only natural language or equations has inherent ambiguities that have unpredictable effects on results; hence, access to the source code is essential to reproducing the central findings of studies.

<https://doi.org/10.1038/nge025180>

<https://doi.org/10.1038/467753a>

<https://doi.org/10.1038/221215EA00142>



Jet Propulsion Laboratory
California Institute of Technology

jpl.nasa.gov