

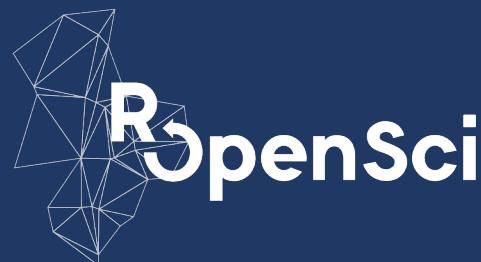
Governance Strategies for Open Source Research Software Organizations

Tuesday, December 18, 2018

Dan Sholler

rOpenSci Project at UC Berkeley

dsholler@berkeley.edu - @DanSholler



Overview

- Background about an ethnographic study
- What is open science?
- What is open research software's (ORS) role in open science?
- What is governance?
- What are the governance challenges ORS projects face?
- What strategies do ORS projects use to overcome governance challenges?

Qualitative, ethnographic study

- Four open source research software projects across two (broadly-defined) disciplines
- Varying lengths of existence
- Interviews with leaders, maintainers, contributors, and users
- Observations/documentation of:
 - Hackweeks, workshops, and unconferences (9)
 - Online discussion forums (e.g., Slack groups), blog posts, website announcements, community calls, and policy papers
 - Social media threads

Interviews

Role	Interviews
Project Leaders and Founders	10
Core maintainers	21
Contributors and users	28
Total	59

Analysis

- Qualitative data analysis
 - Developing themes from interview responses and observation notes
 - Archival analysis of documents
 - Member-checking
- Putting themes in conversation with one another as traditionally done in Science and Technology Studies
- Outcomes: Inform organization science and information systems theories; document a period of change in the sciences; inform governance strategies in open science

“Infrastructures”

“Open Science”

“Governance”

Open Science

Open Data

Open Source
Software (OSS)

Open Access

Open Science

Open Data

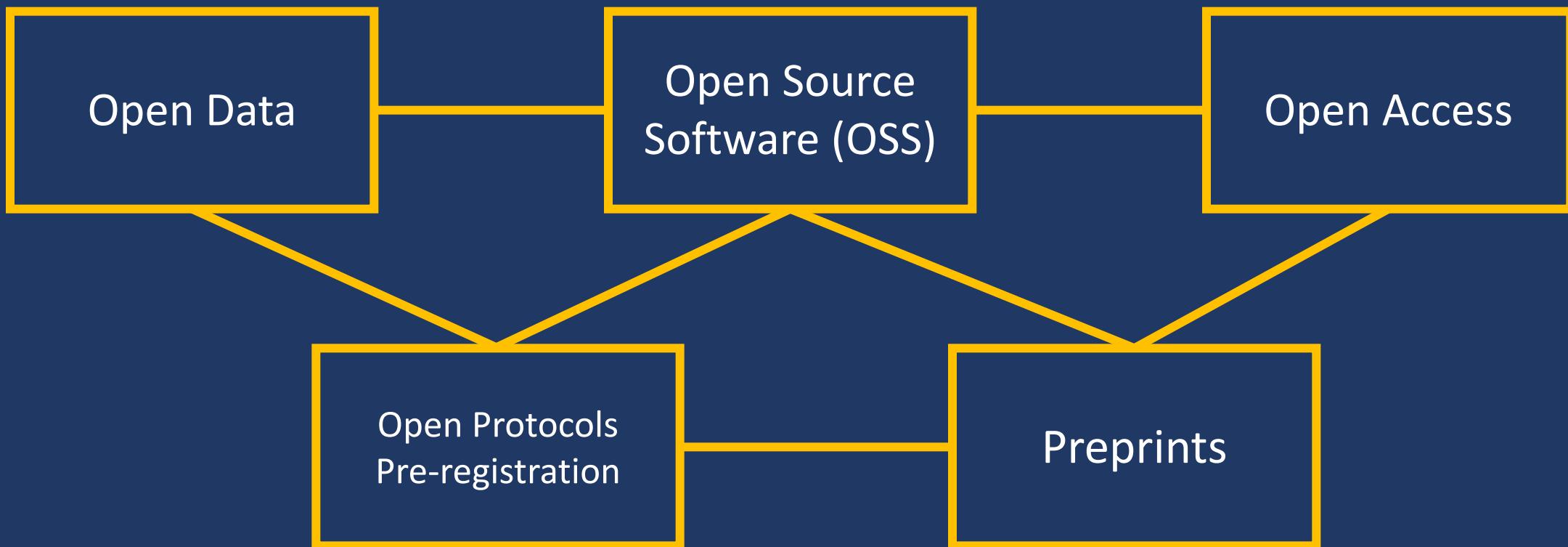
Open Source
Software (OSS)

Open Access

Open Protocols
Pre-registration

Preprints

Open Science



Benefits of Open Science

- Transparency
- Reproducibility
- Replicability
- Access to data, information, and knowledge

Reproducibility and Replicability

Reproducibility: A study is reproducible if you can take the original data and the computer code used to analyze the data and reproduce all of the numerical findings from the study. This may initially sound like a trivial task but experience has shown that it's not always easy to achieve this seemingly minimal standard.

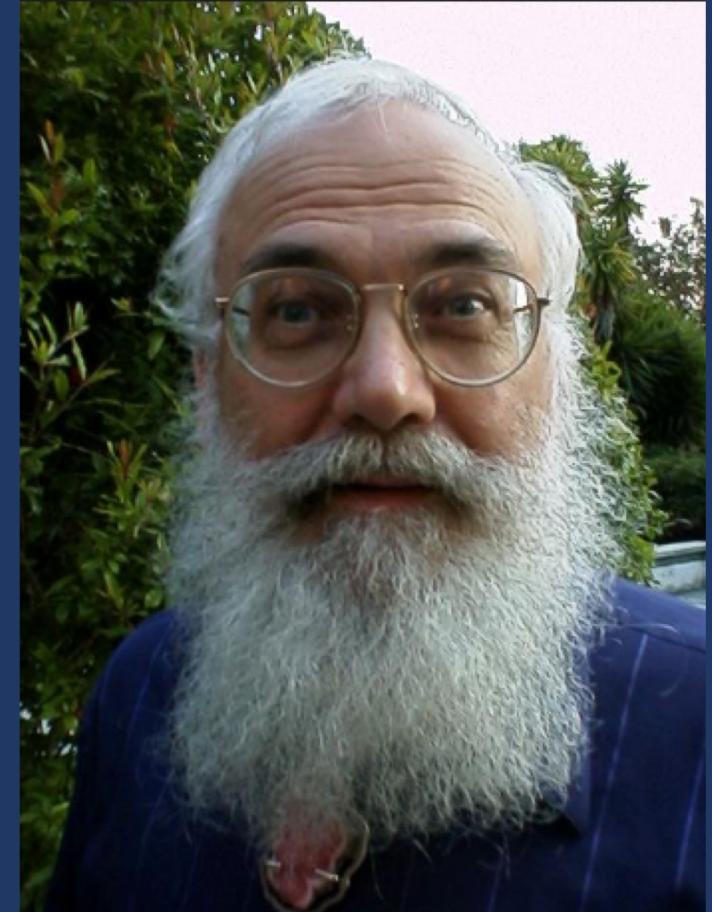
Replicability: This is the act of repeating an entire study, independently of the original investigator without the use of original data (but generally using the same methods)."

American Statistical Association, 2017

Jon Claerbout's vision

“In this marriage, an author attaches to every figure caption a push button to recalculate the figures from all its data, parameters, and programs ... This provides a concrete definition of reproducibility in computationally oriented research.”

Claerbout and Karrenbach (1992)
via Karthik Ram <http://inundata.org/talks/ernz18/#/0/1>



Open Research Software (ORS) Projects

- Open (Source) Research Software projects aim to make software used in scientific research freely/permissibly available, customizable, shareable, versionable, and citeable
- Often built in programming environments such as R or Python
- Alternatives to analysis software such as SAS and SPSS

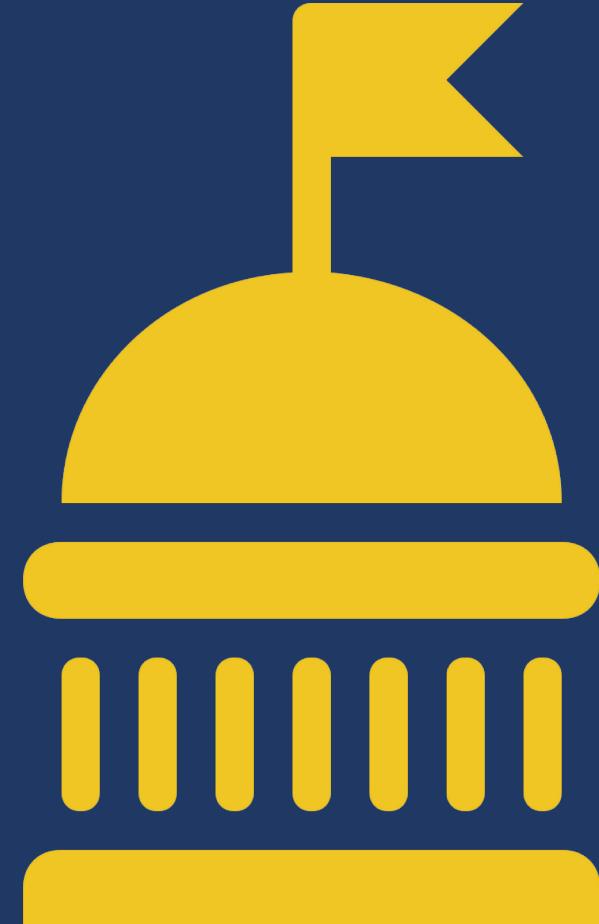
Governing bodies have the “responsibility to see that the organization is acting in the best interests of the public, and more specifically the ‘stakeholders’ who are served by the organization’s mission.”

Bader (2008), p. 2



Components of governance

- Mission and strategic direction
- Finances and investments
- Quality
- Community benefit
- Compliance with laws and regulations



Components of ORS project governance

- Mission and strategic direction – What do we want to build? (technically and socially)

Components of ORS project governance

- Mission and strategic direction – What do we want to build? (technically and socially)
- Finances and investments – How will we sustain our efforts?

Components of ORS project governance

- Mission and strategic direction – What do we want to build? (technically and socially)
- Finances and investments – How will we sustain our efforts?
- **Quality** – What constitutes impactful software? How do we formalize and evaluate “quality”?

Components of ORS project governance

- Mission and strategic direction – What do we want to build? (technically and socially)
- Finances and investments – How will we sustain our efforts?
- Quality – What constitutes impactful software? How do we formalize and evaluate “quality”?
- **Community benefit** – How does the project promote better science?

Components of ORS project governance

- Mission and strategic direction – What do we want to build? (technically and socially)
- Finances and investments – How will we sustain our efforts?
- Quality – What constitutes impactful software? How do we formalize and evaluate “quality”?
- Community benefit – How does the project promote better science?
- **Compliance with laws and regulations** – How do we navigate institutional, journal, governmental, and other policy landscapes?

Why is governance important for ORS projects?

Why is governance important for ORS projects?

They rely on volunteer contributions

- Package development and maintenance
- Heterogeneity in volunteer characteristics



Why is governance important for ORS projects?

They operate with non-hierarchical authority structures



Why is governance important for ORS projects?

They work within an ecosystem of organizations

- Rely on one another in the stack, but often little coordination
- Fluidity of membership; members may contribute to multiple projects



Governance challenges in OSR Development

- Incentivizing contributions and growing the user base or use cases
 - What's in it for the contributors and maintainers?
 - Why should researchers change existing, long-held workflows?
- Balancing tight control with flexibility
- Sustaining the project

What types of governance strategies
exist in ORS projects?

Where do they come from, and
what purposes do they serve?

Early Stages – Founders identify problems

Computational problem with no existing solution: “*There were no tutorials on the web, StackOverflow did not exist, and I did not have a peer community of people to talk to, and there was no training program like Software or Data Carpentry.*”

Connecting to others with the same problem: “*This was at the time Twitter was becoming ‘science Twitter.’ ... This is how I ended up getting in a conversation with [the cofounders] ... who were both graduate students at the time.*”

Developing solutions: “*...magically, we were making faster and faster progress. And that sort of provided like a quick hit of endorphins ... I purchased a domain name, created a website...*”

Early Stages – Advertising the project

Establishing the contribution: “*For every package that we wrote, we just had a simple table, linking to the package then describing what it did.*”

“*I traveled a ton talking about the project.*”

Demonstrating the value: “*We accepted just about everything.*” / “*I gave tutorials.*” / “*People wanted to write us into their grants.*”

Early Stages – Shoring up the project

Building Advisory Boards: “*I just started reaching out to people I knew.*” / “*The project we built this off of had a handful of people who wanted to stay involved, and one way was through the steering board.*”

Securing Funding: “*At the time I didn’t know if it was the right move. I knew it would take me down a different career path, with the time it would take to stay on soft money.*”

Early Stage Governance – Summary

- 1. Identify the common problem to draft mission**
- 2. Develop some solutions (technical)**
- 3. Build advisory boards to set strategic direction**
- 4. Secure funding**

Middle Stages – Growth

Growing the user base: “*We started doing lots of tutorials at conferences in [our discipline] and at open science gatherings.” / “If we can show you, yes, this is going to allow you to publish more, and faster, because you’re not reinventing the wheel each time, people are receptive to that.*”

Rebuilding advisory boards: “*At first we just wanted people who were known and who were working on software. But we started to be more purposeful in who we had at our steering meetings, people recommended others, and we started to fill the gaps.*”

Middle Stages – Building Community

Connecting existing communities: “*I’m here [at a hackweek] because I met [a project maintainer] at a conference. I didn’t know so many other people were working on software so much of the time.*”

Diversifying expertise: “*The [project] community gives me a one-stop-shop for any questions I have, even if its really specific to wetland conservation data or some esoteric thing.*”

Fostering psychological safety: Codes of conduct

Middle stages – Founders and leaders transition to administrative tasks

- Writing grants
- Managing personnel
- Navigating the broader ecosystem of ORS organizations
- Deciding on “open” and “closed” decision-making processes
- Anticipating future changes

“I never envisioned that these kinds of things [administrative tasks] would take up so much of my time. And I don’t know if it’s moving me forward. It moves [the project] forward, but I’m not sure where it’s taking me and my career.” (project lead)

How do OSS organization leaders learn leadership and managerial skills?

“I read books.”

“I learned to just triage, and how to signal what’s important.”

Middle Stage Governance – Summary

1. Grow the user base
2. Revise advisory board based on rescoped project
3. Prepare leadership for new tasks

Late stages – Balancing growth and standardization

Codifying the scope: “*All of these things are focused around government data, and we have a science focus, so we should not deal with them ... We started tightening the ship a little bit more.*”

Structuring community: “*We started having some issues with the makeup of the contributors. Either voices were too dominant, or using the package without contributing anything back. So we had a few meetings just looking at, 'If I were completely new to this, didn't know anything about it, what would be a good motivator to fix bugs, contribute an affiliated package?'"*

Late stages – Codifying scope by formalizing contribution guidelines and processes

Package onboarding decision:

- *Does the proposed submission align with the mission and strategic direction?*
- *Is the package in good enough shape to see a way forward?*
- *Would the community find this useful?*

Late stages – Codifying scope by formalizing contribution guidelines and processes

Package maintenance—three (or four) options:

- *Author-maintained*
- *Community-maintained*
- *Staff-maintained*
- *Hybrid*

Late stages – Codifying scope by formalizing contribution guidelines and processes

Contribution review—three (or four) options:

- *Peer-reviewed by external experts*
- *Peer-reviewed by project staff*
- *Community-driven review*
- *Hybrid*

Late stages – Formalizing contribution guidelines and processes

“It’s [package review and onboarding] an extension of what people know already. And they know they’re going to get a citable object out of it.”

Late stages – Offering trajectories

From shy contributor to central community member: “*My first contribution was writing documentation. Just seeing it there was a boost to my confidence, but it also gave me an excuse to reach out to a couple of the major developers, and from there I built relationships, shared more of my shitty code, and I started asking more questions in public places, and then answering people’s questions, and then I got addicted to seeing my ranking [on GitHub Insights] go up and up.*”

Late stages – Offering trajectories

Avoiding volunteer burnout: “*It’s a lot of unpaid effort, and time away from my research.*”

Navigating escalating commitments: “*The age-old problem, right, the more you do, the more people ask of you.*”

Late stages – Partnering with other organizations

Journals: Partnerships that facilitate software review and onboarding alongside manuscript review, revision, and publication

Other projects and organizations: Formalized contribution guidelines can aid in the development of skills for teaching (e.g., with The Carpentries); overlapping membership (e.g., rOpenSci and the R Consortium) can lead to career opportunities for community members

Late Stage Governance – Summary

1. Strike a balance between growth and standardization
2. Formalize contribution guidelines and processes
3. Develop and offer trajectories for members of the community, within and outside the community
4. Partner with other organizations

Components of governance - Summary

- Mission and strategic direction
- Finances and investments
- Quality
- Community benefit
- Compliance with laws and regulations

How can we support the development
of administrative, managerial skills and
roles to promote good governance in
OSS organizations?

Acknowledgments

Work funded by the Leona M. and Harry B. Helmsley Charitable Trust
and the Berkeley Center for Technology, Society & Policy

Special thanks to the interview participants

Research conducted under UC Berkeley Office for the Protection of
Human Subjects Protocol ID 2017-08-10194