

What Is Governance?

All open source projects operate in and with governance structures. The term "governance" carries multiple meanings in an organizational context. It can refer to regulatory matters or risk management issues, for example. More generally, though, it can also refer to a system of rules, roles, and procedures that determine how power in a project gets distributed.

Because open source projects are organizations, all projects feature governance structures. Some of these structures are more *explicit* than others. Some are more *formal* than others. But every project has them.

Unfortunately, too many discussions of open source project governance focus on activities or resources, like "speaking for the project" or "ownership of the web domain." While documenting these functions is useful, these are not, strictly speaking, governance issues.

At its heart, open source project and community governance is about *people*—their rights and responsibilities as part of a project, and the expectations others have for them.

Why governance?

In some open source communities, "governance" gets a bad rap. This is true in cases where project contributors tend to construe governance as a purely negative force—a set of rules or procedures aimed solely at telling people what they *can't* do, how they *shouldn't* act, or how they should *limit* themselves to acting only within certain boundaries.

But a well-crafted governance model can in fact be a largely positive force in open source communities. A project's governance model outlines that project's "terms of engagement"—the specific, tried-and-tested structures for working together and making decisions that project contributors have found works best for the community. A clear governance model can *encourage* new contributors to become involved in your project.

A well-designed system of governance is much less likely to turn away or de-motivate project participants than a vague or non-existent one is. Consider your project from the perspective of new contributors. Are new contributors *more* or *less* likely to jump into a project without any sense of the role they're supposed to play and the rules they're supposed to follow when they want others to seriously consider your contributions? A clear governance model helps people understand precisely how they can make an immediate contribution to a project, how they can pitch in without "rocking the boat," how they can escalate questions or issues if they have them, and what sorts of leadership positions they can aspire to if they stick around long enough. So a community's goal in architecting a governance model should be *making structures of participation obvious*. When your project's rules are clear, contributors can engage with confidence. Taking this approach to governance can positively impact a project's long-term viability and growth.

Governance is People

Simply put, "governance" refers **to the rules or customs that determine who gets to do what (or**

is supposed to do what), how they're supposed to do it, and when.

All of project governance falls into two categories: **roles** and **policies and procedures**. For the purpose of explanation, we'll discuss each of these separately. In practice, however, they're inseparable—two sides of the same coin.

Roles

A great deal of activity hinges on roles-related governance in open source projects. Think of a **role** as a function someone in the project performs. Project contributors can, and often do, have multiple roles, and project roles are often fulfilled by multiple people.

As a part of governance, all projects have roles, even if they are not explicitly documented. Documenting them will help you recruit people to those roles. A minimal set of project roles will often be something like: contributor, reviewer, maintainer.

When documenting your project's roles, ask the following questions:

- What roles do (or can) project contributors play?
- What qualifies a person to play a particular role in the project?
- What duties, privileges, and forms of authority are associated with each role?
- What project resources are the province or responsibility of people who perform certain roles?

See the **section on Roles** for more detail and examples of how to develop and document them for your project.

Policies and procedures

While role definitions explain how specific contributors participate in the project, there are many activities in a project that involve groups of people, including people external to the project. Examples of this include things like a release procedure or a contribution process. These **Policies and Procedures** (P&P) are what is often thought of as “governance paperwork” for projects.

Some useful questions to ask when developing P&P for your project include:

- How do code and documentation get accepted into the project?
- How does code get released?
- What rules govern communication in the project?
- When do contributors get promoted, and how?
- How do decisions about technical strategy or project resources get made?
- If your project has elections, how are they conducted?

There is a short list of P&P that every CNCF project is expected to have, and a slightly longer list that every project will want to have. See the section on **Policies & Procedures** for more detail.

Accurately Documenting Your Governance

Like technical documentation, governance documentation should explain how things actually work. If there are aspirational goals, those go in their own section under Roadmap or TODO. Remember, if you are a Sandbox or Incubating project, you aren't expected to have all of these things sorted out yet.

It can be tempting to define your project as you would like it to be -- or how you would like to present it to the CNCF TOC -- rather than how it actually is. Particularly, project leaders frequently make the mistake of attempting to make the project appear more organized and mature than it actually is, in documentation. This falls apart when users or contributors expect your project to live up to its governance documentation, and it doesn't. People who would have been fine with being told a project was single-company at the outset become very upset if they ask for their committer status and are refused later. For that matter, the CNCF TOC requires projects to be accurate about the things they are still working on, and may penalize projects for not being completely truthful.