

## Governance

This document describes the rules and governance of the project. It is meant to be followed by all the developers of the project and the Grafana community. Common terminology used in this governance document are listed below:

- **Team members:** Any members of the private grafana-team Google group.
- **Maintainers:** Maintainers lead an individual project or parts thereof (MAINTAINERS.md).
- **Projects:** A single repository in the Grafana GitHub organization and listed below is referred to as a project:
  - clock-panel
  - devtools
  - gel-app
  - grafana
  - grafana-github-datasource
  - grafana-image-renderer
  - grafana-kiosk
  - grafana-plugin-sdk-go
  - grafana-polystat-panel
  - grafonnet-lib
  - kaisrosdb-datasource
  - piechart-panel
  - simple-angular-panel
  - simple-app-plugin
  - simple-datasource
  - simple-datasource-backend
  - simple-json-backend-datasource
  - simple-json-datasource
  - simple-react-panel
  - strava-datasource
  - tutorials
  - worldmap-panel
- **The Grafana project:** The sum of all activities performed under this governance, concerning one or more repositories or the community.

## Values

The Grafana developers and community are expected to follow the values defined in the Grafana Code of Conduct. Furthermore, the Grafana community strives for kindness, giving feedback effectively, and building a welcoming environment. The Grafana developers generally decide by consensus and only resort to conflict resolution by a majority vote if consensus cannot be reached.

## Projects

Each project must have a `MAINTAINERS.md` file with at least one maintainer. Where a project has a release process, access and documentation should be such that more than one person can perform a release. Releases should be announced on the Grafana Labs blog. Any new projects should be first proposed on the team mailing list following the voting procedures listed below.

## Decision making

### Team members

Team member status may be given to those who have made ongoing contributions to the Grafana project for at least 3 months. This is usually in the form of code improvements and/or notable work on documentation, but organizing events or user support could also be taken into account.

New members may be proposed by any existing member by email to grafana-team. It is highly desirable to reach consensus about acceptance of a new member. However, the proposal is ultimately voted on by a formal supermajority vote.

If the new member proposal is accepted, the proposed team member should be contacted privately via email to confirm or deny their acceptance of team membership. This email will also be CC'd to grafana-team for record-keeping purposes.

If they choose to accept, the onboarding procedure is followed.

Team members may retire at any time by emailing the team.

Team members can be removed by supermajority vote on the team mailing list. For this vote, the member in question is not eligible to vote and does not count towards the quorum. Any removal vote can cover only one single person.

Upon death of a member, they leave the team automatically.

In case a member leaves, the offboarding procedure is applied.

The current team members are:

- Alexander Zobnin (Grafana Labs)
- Alex Khomenko (Grafana Labs)
- Andrej Ocenas (Grafana Labs)
- Arve Knudsen (Grafana Labs)
- Brian Gann (Grafana Labs)
- Carl Bergquist (Grafana Labs)
- Chris Trott (Grafana Labs)
- Daniel Lee (Grafana Labs)
- David Kaltschmidt (Grafana Labs)
- Diana Payton (Grafana Labs)
- Diana Sarlinska (Grafana Labs)

- Dominik Prokop (Grafana Labs)
- Emil Tullstedt (Grafana Labs)
- Erik Sundell (Grafana Labs)
- Fredrik Enestad (Embark Studios)
- Hugo Häggmark (Grafana Labs)
- Ivana Huckova (Grafana Labs)
- Jeroen Op 't Eynde (Grafana Labs)
- Jessica Müller (Grafana Labs)
- Julien Pivotto (Inuits)
- Kay Delaney (Grafana Labs)
- Kyle Brandt (Grafana Labs)
- Leonard Gram (Grafana Labs)
- Lukas Siatka (Grafana Labs)
- Malcolm Holmes (Grafana Labs)
- Marcus Andersson (Grafana Labs)
- Marcus Efraimsson (Grafana Labs)
- Marcus Olsson (Grafana Labs)
- Mitsuhiro Tanda (GREE)
- Patrick O'Carroll (Grafana Labs)
- Peter Holmberg (Grafana Labs)
- Richard Hartmann (Grafana Labs)
- Ryan McKinley (Grafana Labs)
- Sofia Papagiannaki (Grafana Labs)
- Stephanie Closson (Grafana Labs)
- Tobias Skarhed (CERN)
- Torkel Ödegaard (Grafana Labs)
- Utkarsh Bhatnagar (Tinder)
- Will Browne (Grafana Labs)
- Zoltán Bedi (Grafana Labs)

## Maintainers

Maintainers lead one or more project(s) or parts thereof and serve as a point of conflict resolution amongst the contributors to this project. Ideally, maintainers are also team members, but exceptions are possible for suitable maintainers that, for whatever reason, are not yet team members.

Changes in maintainership have to be announced on the developers mailing list. They are decided by rough consensus and formalized by changing the `MAINTAINERS.md` file of the respective repository.

Maintainers are granted commit rights to all projects covered by this governance.

A maintainer or committer may resign by notifying the team mailing list. A maintainer with no project activity for a year is considered to have resigned. Maintainers that wish to resign are encouraged to propose another team member to take over the project.

A project may have multiple maintainers, as long as the responsibilities are clearly agreed upon between them. This includes coordinating who handles which issues and pull requests.

### **Technical decisions**

Technical decisions that only affect a single project are made informally by the maintainer of this project, and rough consensus is assumed. Technical decisions that span multiple parts of the Grafana project should be discussed and made on the Grafana developer mailing list.

Decisions are usually made by rough consensus. If no consensus can be reached, the matter may be resolved by majority vote.

### **Governance changes**

Changes to this document are made by Grafana Labs.

### **Other matters**

Any matter that needs a decision may be called to a vote by any member if they deem it necessary. For private or personnel matters, discussion and voting takes place on the team mailing list, otherwise on the developer mailing list.

## **Voting**

The Grafana project usually runs by informal consensus, however sometimes a formal decision must be made.

Depending on the subject matter, as laid out above, different methods of voting are used.

For all votes, voting must be open for at least one week. The end date should be clearly stated in the call to vote. A vote may be called and closed early if enough votes have come in one way so that further votes cannot change the final decision.

In all cases, all and only team members are eligible to vote, with the sole exception of the forced removal of a team member, in which said member is not eligible to vote.

Discussion and votes on personnel matters (including but not limited to team membership and maintainership) are held in private on the team mailing list. All other discussion and votes are held in public on the developer mailing list.

For public discussions, anyone interested is encouraged to participate. Formal power to object or vote is limited to team members.

## **Consensus**

The default decision making mechanism for the Grafana project is rough consensus. This means that any decision on technical issues is considered supported by the team as long as nobody objects or the objection has been considered but not necessarily accommodated.

Silence on any consensus decision is implicit agreement and equivalent to explicit agreement. Explicit agreement may be stated at will. Decisions may, but do not need to be called out and put up for decision on the developers mailing list at any time and by anyone.

Consensus decisions can never override or go against the spirit of an earlier explicit vote.

If any team member raises objections, the team members work together towards a solution that all involved can accept. This solution is again subject to rough consensus.

In case no consensus can be found, but a decision one way or the other must be made, any team member may call a formal majority vote.

## **Majority vote**

Majority votes must be called explicitly in a separate thread on the appropriate mailing list. The subject must be prefixed with **[VOTE]**. In the body, the call to vote must state the proposal being voted on. It should reference any discussion leading up to this point.

Votes may take the form of a single proposal, with the option to vote yes or no, or the form of multiple alternatives.

A vote on a single proposal is considered successful if more vote in favor than against.

If there are multiple alternatives, members may vote for one or more alternatives, or vote “no” to object to all alternatives. It is not possible to cast an “abstain” vote. A vote on multiple alternatives is considered decided in favor of one alternative if it has received the most votes in favor, and a vote from more than half of those voting. Should no alternative reach this quorum, another vote on a reduced number of options may be called separately.

## **Supermajority vote**

Supermajority votes must be called explicitly in a separate thread on the appropriate mailing list. The subject must be prefixed with **[VOTE]**. In the body, the call to vote must state the proposal being voted on. It should reference any discussion leading up to this point.

Votes may take the form of a single proposal, with the option to vote yes or no, or the form of multiple alternatives.

A vote on a single proposal is considered successful if at least two thirds of those eligible to vote vote in favor.

If there are multiple alternatives, members may vote for one or more alternatives, or vote “no” to object to all alternatives. A vote on multiple alternatives is considered decided in favor of one alternative if it has received the most votes in favor, and a vote from at least two thirds of those eligible to vote. Should no alternative reach this quorum, another vote on a reduced number of options may be called separately.

## **On- / Offboarding**

### **Onboarding**

The new member is

- added to the list of team members. Ideally by sending a PR of their own, at least approving said PR.
- announced on the developers mailing list by an existing team member. Ideally, the new member replies in this thread, acknowledging team membership.
- added to the projects with commit rights.
- added to the team mailing list.

### **Offboarding**

The ex-member is

- removed from the list of team members. Ideally by sending a PR of their own, at least approving said PR. In case of forced removal, no approval is needed.
- removed from the projects. Optionally, they can retain maintainership of one or more repositories if the team agrees.
- removed from the team mailing list and demoted to a normal member of the other mailing lists.
- not allowed to call themselves an active team member any more, nor allowed to imply this to be the case.
- added to a list of previous members if they so choose.

If needed, we reserve the right to publicly announce removal.