

# Documentation style guide

This style guide applies to all documentation created for Grafana products.

For information about how to write technical documentation, refer to the following resources:

- [Google Technical Writing courses](#)
- [Divio documentation system](#)
- [Vue writing principles](#)

## Contributing

The *Documentation style guide* is a living document. Add to it whenever a style decision is made or a question is answered regarding style, grammar, or word choice.

## Published guides

For all items that are not covered in this guide, refer to the [Google developer documentation style guide](#) and the [Microsoft style guide](#), in that order.

## Spelling

To catch common misspellings, the [codespell](#) tool is run for every change.

## Inclusive language

Avoid using charged language.

## Allowing and blocking

When referring to *allowing* or *blocking* content or traffic, use a form of *allow* or *block*:

- (noun) *allowlist* or *blocklist*
- (verb) *allow* or *block*

Example: To **allow** outgoing traffic, add the IP to the **allowlist**.

Avoid *whitelist* or *blacklist*.

## Primary and secondary

To describe relationships between nodes or processes, there are several options:

- Use *primary*, *main*, or *parent*, instead of *master*.
- Use *secondary*, *replica*, or *child*, instead of *slave*.

Avoid *master* or *slave*.

## Grafana-specific style

The following guidelines are specific to Grafana documentation. For the most part, these are *guidelines* are not rigid rules. If you have questions, then please ask in the #docs channel of Grafana Slack.

## General

Per the [Voice and tone](#) section of the Google developer documentation style guide:

*In your documents, aim for a voice and tone that's conversational, friendly, and respectful without being overly colloquial or frivolous; a voice that's casual and natural and approachable, not pedantic or pushy. Try to sound like a knowledgeable friend who understands what the developer wants to do.*

- Use active voice:
  - Active: Grafana displays the heatmap visualization.
  - Passive: The heatmap visualization is displayed.
- Write directly to the reader:
  - Use: "After you create a dashboard, you can add a panel to it."
  - Avoid: "After you create a dashboard, it is possible to add a panel to it."
- Write in the imperative second person:
  - "Click the panel."
  - "Close the window."
- Write in present tense:
  - Use: "The panel opens."
  - Avoid: "The panel will open."
- Do not use an ampersand (&) as an abbreviation for *and*.
  - **Exceptions:** If an ampersand is used in the Grafana UI, then match the UI.
- Avoid using internal jargon or slang.
- Do not use two spaces after a period; use one space after a sentence.
- Remove any extra space characters at the end of a paragraph.
- Aim for your sentences to be fewer than 25 words. Instead, use smaller complete phrases or change the format, such as using a list.
- Aim for paragraphs to be three sentences or fewer. Make the text more concise, use more headings, or both.

## File naming conventions

- Files that are displayed in the help system should have names that are all lowercase, no spaces. Use hyphens instead of spaces. Example: glossary.md
- Documentation file names should match the title. **Note:** This only applies to new files at this time. Do not change the names of older files unless directed to do so.
- Internal reference file names should be all uppercase except the file extension. Example: CONTRIBUTING.md
- Image file names should be descriptive and unique. Also, add the software version number that the image applies to or the screenshot was taken in. Example: share-dashboard-link-7-3.png

## Headings

- Write headings in sentence case, not title case.
  - This is sentence case
  - This is Title Case
- Task topic headings start with a verb.
  - Write a query
  - Create a dashboard
- Concept and reference topic headings should be nouns or gerunds. Examples: Contributing to docs, Visualizations, Style guide

## Heading don'ts

- Avoid stacked headings, which is following one heading with another heading.

- Avoid skipping heading levels. For example, an h1 should be followed by an h2 rather than an h3.
- Avoid having just one lower-level heading. For example, h1, h2, h2, h3, h3, h2, h2 is a good order. Do not go h1, h2, h3, h2, h3, h2.
- Avoid using hyphens in headings.
- Do not include parenthetical words like (Important!) in headings.

### Step-by-step headings

In most cases, headings should not be numbered steps.

However, sometimes we need to use headings as numbered steps. This is mostly in cases where each step is complex or a series of other procedures. For example, in [Getting started with Grafana and Prometheus](#).

If that is the case, then use the following format for headings:

#### Step 1. Install the software

#### Step 2. Run the software

### Images

- Preferred format is .png
- File extension should be all lowercase.
- Preferred DPI is 72.
- Assume all graphics will be exclusively viewed on the web.
- Maximum image size is 3840px X 2160px.
- Screenshots should be readable, but not too large.
- *Do not* use image shortcodes. Follow the guidance in [Markdown style guide](#).
- Markdown image links are preferred. Only use the HTML image links if you need to style the image in ways unsupported in Markdown.
- When you name a file, follow the [file naming conventions](#). Example: image-name-7-3.png

### Unordered lists

Here are a few general rules about unordered lists. For more guidance, refer to [Lists](#) in the [Google developer style guide](#).

- List items should begin with a capital letter unless there is a strong reason not to. For example, you are listing case-sensitive parameters.
- List items should end with periods if they are complete sentences. If one item in a list ends with a period, then apply periods to all of them.

### Capitalization

- Grafana, Loki, and Prometheus are always capitalized unless part of a code block.
- API names are always Title Case, followed by "API"—for example, "Dashboard Permissions API"
- Abbreviations are always capitalized (such as API, HTTP, ID, JSON, SQL, or URL) unless they are part of a code block.
- Menu and submenu titles always use sentence case: capitalize the first word, and lowercase the rest.
  - "Dashboards" when referring to the submenu title.
  - "Keyboard shortcuts" when referring to the submenu topic.
- Generic and plural versions are always lowercase.
  - Lowercase "dashboard" when referring to a dashboard generally.
  - Lowercase "dashboards" when referring to multiple dashboards.
- **Exceptions:** If a term is lowercased in the Grafana UI, then match the UI.

### Git, GitHub

Git is always capitalized, unless part of a code block. GitHub is the correct spelling and capitalization.

## Integrations

In general, "integration" is not capitalized. Only capitalize it if it is capitalized in the UI or part of a proper noun, like the name of a specific integration.

The first letter of the name of an integration is always capitalized, even if the original named source is lowercase.

### Examples:

- MySQL Integration
- CockroachDB Integration
- Etcd Integration
- I installed an integration on my local Grafana.

## Kubernetes objects

Capitalize Kubernetes objects such as Job, Pod, and StatefulSet when it is clear you are specifically talking about them and not generic jobs, pods, or whatever.

Introduce the object as "Kubernetes XX" on the first usage, then just the object in subsequent uses.

### Example:

Create the Kubernetes Job and check the logs to retrieve the generated token:

The Job requires the token be submitted as ...

## Links and references

When referencing another document, use "Refer to" rather than alternatives such as "See" or "Check out."

Always give the reader some idea of what to expect in the reference. Avoid blind references, such as, "Refer to [this file](#)."

When possible, use the exact title of the page or section you are linking to as the link text.

**Example** Refer to the [Documentation style guide](#) for information about word usage and capitalization guidelines.

## Notes, tips, cautions, and warnings

Grafana documentation uses notes, tips, cautions, and warnings. Notes are the most common. The format for all of them is indented, bold, sentence case:

```
> **Note:**
```

### Notes

Notes provide additional information that the user should be extra aware of. For example:

**Note:** This page describes a feature for Grafana 7.0 beta.

### Tips

Tips describe alternate or more efficient ways of doing things. Rarely used.

### Cautions

Cautions warn the user that they should proceed with caution. Use cautions to emphasize the potential downside of a course of action.

**Caution:** *If you turn off authentication requirements, then anyone can access your Grafana instance. This poses a considerable security risk.*

## Warnings

Warnings tell the user not to do something. For example:

**Warning:** *Grafana does not back up your dashboards. If you delete a dashboard, then you might not be able to recover it.*

## Command line examples

- Do not assume everyone is using Linux. Make sure instructions include enough information for Windows and Mac users to successfully complete procedures.
- Do not add `$` before commands. Make it easy for users to copy and paste commands.
  - **Right:** `sudo yum install grafana`
  - **Wrong:** `$ sudo yum install grafana`
- Include `sudo` before commands that require `sudo` to work.

For terminal examples and Grafana configuration, use a `bash` code block:

```
sudo yum install grafana
```

For HTTP request/response, use an `http` code block:

```
GET /api/dashboards/id/1/permissions HTTP/1.1
Accept: application/json
Content-Type: application/json
Authorization: Bearer eyJrIjoiT0tTcG1pU1Y2RnVKZTFVaDFsNFZXdE9ZWmNrMkZYbk
```

## Word usage

Grafana products has some words, abbreviations, and terms particular to the Grafana discourse community.

### changelog

One word, not two.

#### Example

- Read the full changelog.

#### Exception:

- When referring to the file containing the official changelog, use the filename: `CHANGELOG.md`.

### checkout, check out

Two words if used as a verb, one word if used as a noun.

## Examples

- Check out these new features!
- Proceed to checkout.

## data source

Two words, not one.

## Exceptions:

- "datasource" used as an identifier
- "datasource" in a URL
- Use "data source" instead of "datasource" unless used as an identifier, in code, or as part of a URL.
- Spell out "repository" and avoid the shorter "repo."
- Use "Unix" as the preferred spelling (as opposed to "UNIX", or "unix") when referring to the family of operating systems.

## display (verb)

*Display* is a transitive verb, which means it always needs a direct object.

- Correct, active voice: Grafana displays your list of active alarms.
- Correct, but passive voice: Your list of active alarms is displayed.
- Incorrect: The list of active alarms displays.

## drawer

Do not use. This is developer jargon that refers to a UI panel. Refer to the panel or feature by its proper name.

## intro, introduction

"Introduction" is the preferred word. Use "intro" if there are space constraints (like on the side menu) or you are specifically trying for a less formal, more conversational tone.

## metadata

One word, not two.

## mixin

One word, not two. Also, not hyphenated.

## open source, open-source

Do not hyphenate when used as an adjective unless the lack of hyphen would cause confusion. For example: *Open source software design is the most open open-source system I can imagine.*

Do not hyphenate when it is used as a noun. For example: *Open source is the best way to develop software.*

## plugin, plug in

Two words if used as a verb, one word if used as a noun. Do not use *plug-in*.

## Examples

- Plug in the appliance.
- Download the plugin.

## setup, set up

Two words if used as a verb, one word if used as a noun.

### Examples

- Set up the workspace.
- Initial setup might take five minutes.

### **node\_exporter, windows\_exporter**

When referencing the Prometheus data source exporters, always use "node\_exporter" and "windows\_exporter" when referring to those tools.

**Correct:** node\_exporter, windows\_exporter **Incorrect:** Node Exporter, node exporter, Windows Exporter, Windows exporter, windows exporter.

### **web server**

Two words, not one.

**Correct:** web server **Incorrect:** webserver

### **MS SQL Server**

Always use "MS SQL" when referring to MS SQL Server application.

Incorrect UI spellings will be corrected in a later version of Grafana.