

# Contribute docs

**PUBLIC PREVIEW DOCUMENT** This document is currently in public preview and may change significantly as feedback is captured from readers like you. Click the [Request an update](#) button at the top of this document or [join the Arbitrum Discord](#) to share your feedback. The [docs.arbitrum.io](https://docs.arbitrum.io) docs portal is the single source of truth for documentation that supports Offchain Labs' product portfolio. Contributions are welcome from the entire Ethereum community.

This document shows you how to craft and publish Arbitrum documentation. Familiarity with [Markdown](#) syntax, Github, and [Docusaurus](#) is expected.

## Add a new core document

If a document isn't in a [Third-party content sidebar node](#), it's a core document. To contribute a new core doc:

1. Begin by creating a branch (internal) or fork (external) of the [Arbitrum docs repo](#)
2. .
3. Issue a Draft
4. pull request into master
5. . Pull requests into master
6. generate a preview of your changes via a PR-specific Docusaurus deployment; this preview will update as you push commits to your remote.
7. Include answers to the following questions in your PR description:
  1. Audience: Who am I writing for?
  - 2.
  - 3.
  - 4.
  - 5.
8. Problem: What specific problem are they trying to solve?
9. 2.
10. Discovery: How are they looking for a solution to this problem? What search terms are they using?
11. 3.
12. Document type: Which document type is most suitable?
13. 4.
14. 5.
15. Policy acknowledgment (Third-party docs only): Do you agree to the third-party content policy outlined within "Contribute docs"?
16. As you craft your contribution, refer to the [document types](#)
17. [Style guidance](#)
18. , and other conventions below.
19. Mark your PR as Open
20. when it's ready for review.
- 21.

## Add a new third-party document

Third-party docs are documents that help readers of Arbitrum docs use other products, services, and protocols (like the ones listed in the [Arbitrum portal](#)) with Arbitrum products.

See [Contribute third-party docs](#) for detailed instructions.

## Request an update

If you'd like to request an update or share a suggestion related to an existing document without submitting a pull request to implement the improvement yourself, click the [Request an update](#) button located at the top of each published document. This button will lead you to a prefilled Github issue that you can use to elaborate on your request or suggestion.

## Add a new translation page

If you would like to participate in translating the Arbitrum docs, you can:

1. Check whether `/website/i18n`
2. has a corresponding language (currently there are `ja`
3. and `zh`
4. ). If not, you can use the following command to add it (we take adding French as an example):

```
cd ./website && npm run write-translations -- --locale fr
```

It will help generate folder `website/i18n/fr`.

1. Create the folders `current`
2. and `translated`
3. under the newly generated folder `website/i18n/fr/docusaurus-plugin-content-docs`
4. :

```
mkdir i18n/{Your_language}/docusaurus-plugin-content-docs/current && mkdir i18n/{Your_language}/docusaurus-plugin-
```

content-docs/translated 1. Translate one of more docs files located in/arbitrum-docs 2. . 3. Place the translated document into the folderi18n/{Your\_language}/docusaurus-plugin-content-docs/translated 4. according to its relative path inarbitrum-docs 5. . For example, if you translated/arbitrum-docs/arbos/arbos.mdx 6. , then its path in i18n should bei18n/{Your\_language}/docusaurus-plugin-content-docs/translated/arbos/arbos.mdx 7. .

Test run:

1. Check that the i18n settings inwebsite/docusaurus.config.js
2. have included your new language:

i18n: { defaultLocale: 'en', // locales: ['en', 'ja', 'zh'], locales: ['en'], // You can add your new language to this array }, 1. Check whether thelocale Dropdown 2. component exists in navbar, if not, add it:

navbar: { title: 'Arbitrum Docs', logo: { alt: 'My Site Logo', src: 'img/logo.svg', href: '/welcome/arbitrum-gentle-introduction', }, items: [ // note: we can uncomment this when we want to display the locale dropdown in the top navbar // if we enable this now, the dropdown will appear above every document; if ja is selected for a document that isn't yet translated, it will 404 // there may be a way to show the dropdown only on pages that have been translated, but that's out of scope for the initial version { type: 'localeDropdown', position: 'right', } ], }, 1. Build translation and docs:

yarn build\_translation && yarn build 1. Start docs:

npm run serve

## Document type conventions

Every document should be a specifictype of document. Each type of document has its own purpose:

Document type Purpose Example(s) to refer to Gentle introduction Onboard a specific reader audience with tailored questions and answers [A gentle introduction to Orbit](#) Quickstart Onboard a specific reader audience with step-by-step "learn by doing" instructions [Quickstart: Build dApps](#) How-to Provide task-oriented procedural guidance [How to run a local dev node](#) Concept Explain what things are and how they work [Token bridging](#)

[Nodes and networks](#) FAQ Address frequently asked questions [FAQ: Run a node](#) Troubleshooting List common troubleshooting scenarios and solutions [Troubleshooting: Run a node](#) Reference Lists and tables of things, such as API endpoints and developer resources [RPC endpoints and providers](#) This isn't an exhaustive list, but it includes most of the document types that we use.

## Style conventions

The following style guidelines provide a number of loose recommendations that help us delivera consistent content experience across our docs:

1. Casing
2.
  - Sentence-case "content labels": document titles, sidebar titles, menu items, section headers, etc.
3. Linking
4.
  - Avoid anchoring links to words like "here" or "this". Descriptive anchor text can help set expectations for readers who may hesitate to click on ambiguous links. When linking to docs, try to link to the document's title verbatim.
5. Titling
6.
  - Titles should balance brevity with precision -Node running overview
7.
  - is preferred toOverview
8.
  - . This helps with SEO and reader UX.
9. Separate procedural from conceptual (most of the time)
10.
  - Within procedural docs like how-tos and quickstarts, avoid including too much conceptual content. Provide only the conceptual information that the target readerneeds
11.
  - in order to complete the task at hand. Otherwise, organize conceptual information within conceptual docs, and link to them "just in case" from other docs.
12. Voice
13.
  - Address the reader as "you".
14.
  - Write like you'd speak to a really smart friend who's in a rush.
15.
  - Opt for short, clear sentences that use translation-friendly, plain language.

16.
  - Use contractions wherever it feels natural - this can help convey a friendly and conversational tone.
17. Formality
18.
  - Don't worry too much about formality. The most valuable writing is writing that provides value to readers, and readers generally want to "flow" through guidance.
19.
  - Aim at "informal professionalism" that prioritizes audience-tailored problem-solving
20.
  - and consistent style and structure
21.
  - .
22. Targeting
23.
  - Don't try to write for everyone; write for a specific reader persona
24.
  - (also referred to as "audience" in this document) who has a specific need
25.
  - .
26.
  - Make assumptions about prior knowledge (or lack thereof) and make these assumptions explicit in the beginning of your document.
27. Flow
28.
  - Set expectations
29.
  - : Begin documents by setting expectations. Who is the document for? What value will it provide to your target audience? What assumptions are you making about their prior knowledge? Are there any prerequisites?
30.
  - Value up front
31.
  - : Lead with what matters most to the reader persona you're targeting. Then, progressively build a bridge that carries them towards task completion as efficiently as possible.
32. Cross-linking
33.
  - We want to maintain both high discoverability
34.
  - and high relevance
35.
  - . As a general rule of thumb, links to other docs should be "very likely to be useful for most readers". Every link is a subtle call to action; we want to avoid CTA overload.
36. Things to avoid
37.
  - Symbols where words will do
38.
  - : Minimize usage of &
39.
  - and/
40.
  - - spell out words like "and"
41.
  - " and "or
42.
  - ".
43.
  - Jargon
44.
  - : Using precise technical terminology is ok, as long as your target audience is likely to understand the terminology. When in doubt, opt for clear, unambiguous, accessible
45.
  - language.

Don't stress too much about checking off all of these boxes; we periodically review and edit our most heavily-trafficked docs, bringing them up to spec with the latest style guidelines.

Some important disclaimers:

- This isn't an exhaustive list

- . These are just the min-bar guidelines that will be applied to all new content moving forward.
- Many of our docs don't yet follow this guidance
- . Our small-but-mighty team is working on it! If you notice an obvious content bug, feel free to submit an [issue](#)
- or [PR](#)
- .

## Banner conventions

You can use banners (Docusaurus refers to them as "[admonitions](#)" ) to set expectations for your readers and to emphasize important callouts. Use these conservatively, as they interrupt the flow of the document.

### Public preview content banner

Example:

**PUBLIC PREVIEW DOCUMENT** This document is currently in public preview and may change significantly as feedback is captured from readers like you. Click the Request an update button at the top of this document or [join the Arbitrum Discord](#) to share your feedback. Usage:

```
import PublicPreviewBannerPartial from '../partials/_public-preview-banner-partial.md';
```

### Under construction banner

Example:

**UNDER CONSTRUCTION** The following steps are under construction and will be updated with more detailed guidance soon. Stay tuned, and don't hesitate to click the Request an update at the top of this document if you have any feedback along the way. Usage:

```
:::caution UNDER CONSTRUCTION
```

The following steps are under construction and will be updated with more detailed guidance soon. Stay tuned, and don't hesitate to click the Request an update at the top of this document if you have any feedback along the way.

```
:::
```

### Community member contribution banner

Example:

**Community member contribution** The following document was contributed by @todo-twitter-handle. Give them a shoutout if you find it useful! Usage:

```
:::info Community member contribution
```

The following document was contributed by @todo-twitter-handle. Give them a shoutout if you find it useful!

```
:::
```

## Frequently asked questions

**Can I point to my product from core docs? For example - if my product hosts a public RPC endpoint, can I add it to your [RPC endpoints and providers](#)**

page?

These types of contributions are generally not merged unless they're submitted by employees of Offchain Labs.

Instead of opening a PR for this type of contribution, click the Request an update button at the top of the published document to create an issue. Generally, third-party services are included in core docs only if we can confidently assert that the services are "trustworthy, highly relevant to the core document at hand, and battle-tested by Arbitrum developers " under a reasonable amount of scrutiny.

### Can I use AI-generated content?

"No". By issuing PRs into our docs repo, you're acknowledging that your content has been produced organically. Content produced under the influence of AI/ML tooling, such as ChatGPT, is "strictly not allowed".

**Should I be wary of contributing to FAQs and Glossaries? It looks like there are some automations configured**

**against these document types.**

You're right! We draft FAQs and Glossaries on Notion to make it easier for nontechnical internal contributors to contribute. This content is then published to our docs repo using a script that reads from Notion and writes to Markdown.

You can still submit changes to the Glossary and FAQ markdown files; we manually synchronize these types of changes with Notion content whenever we need to.

### **How long does it take for my third-party content contribution to be reviewed?**

Our small-but-mighty team is continuously balancing competing priorities, so we can't guarantee a specific turnaround time for third-party docs PRs. They're processed in the order in which they're received, generally within a week or two.

### **Is there any way to expedite third-party content contribution reviews?**

The most effective way to expedite processing is to ensure that your PR incorporates the conventions outlined in this document. Please don't ask for status updates - if you've submitted a PR, it's on our radar! [Edit this page](#) Last updated on Mar 19, 2024 [Previous Arbitrum glossary](#) [Next Contribute third-party docs](#)