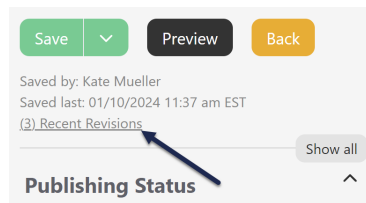


Version guidelines

Last Modified on 10/10/2024 1:24 pm EDT

When you make any type of change to an article and save it, that save is tracked as a KnowledgeOwl [revision](#). The 10 most [recent revisions](#) for any article are stored and available in the upper right, just under the Publishing Status section. You can [compare revisions](#) and [recover a previous revision](#).



However, since only the ten most recent are stored, we strongly encourage you to use versions to track important historical changes.

[Versions](#) have several advantages:

- They are stored as long as you'd like them to be.
- You can control when a new version is [activated](#), so you don't risk half-complete information being displayed to a reader while edits are in-progress.
- Since you have control over version activation, you can [bulk activate](#) new versions of existing content to coincide with a software release, yearly policy change, and so on.
- You can add [version notes](#) to summarize changes that were made or track content reviews.
- You can choose to make versions [visible to specific reader groups](#) so that review teams can preview/review content in the live knowledge base before activation.

To get familiar with versions, check out [Versions](#) or get started [creating a version](#).

Set version guidelines

Once you have some understanding of how versions work, we suggest you establish some version guidelines.

These guidelines should help any of your authors determine:

- When a version should be created
- How to select/set the version number
- When and how to use version notes
- What the version review process is
- What the version activation process is

Having clear guidelines will:

- Reduce stress or confusion for your authors
- Keep version creation consistent across your knowledge base

- Provide appropriately-detailed historical tracking of article versions

You can set these guidelines up in any way you'd like, but we'd recommend hitting on these key points:

Version creation guidelines

This should answer the question: When should I create a new version?

Without clear guidance on this, we've seen some authors create a new version every time they correct a typo or grammar error, and other authors never create versions. Give your authors a sense of which edits should warrant the creation of a new version and which shouldn't.

Here in our support knowledge base, small wording or content updates that might be more stylistic don't get new versions. But we do create new versions for:

- Any update that changes a screenshot and related content
- Major updates to how a feature or function works or is configured
- Substantive style/readability/layout edits--if it moves a heading section around, for example

Version numbering guidelines

These guidelines should answer the question: Which version number should I use?

KnowledgeOwl provides three options for creating a new version:

- Minor version (+0.01 numbering change)
- Major version (+1.0 numbering change)
- Custom version (any format of your choice)

Especially if you're using custom versions, providing guidance here is essential (should those versions correspond to a software release, or the date the version was created?). But it's still helpful if you're using the built-in major and minor versions, so your authors make them consistently.

Here are our three version causes from above, with the version numbering system we use:

- Screenshot/related content update: minor version
- Major feature/function update: custom version using year.mm.dd format
- Substantive style/readability/layout update: major version

Version note guidelines

These guidelines should answer the question: when should I add [version notes](#), and what kind of notes do I include?

These guidelines can vary sharply based on whether you're only using the notes to document the changes made, or also using the version notes field as part of an internal review process.

For our support knowledge base, we do not use the version notes field for internal reviews, so we have a very simple guideline:

Any time you create a new version, add a version note that indicates the reason for version creation. (For example: updated Snippet Details screenshot to include the new References section.)

Version review guidelines

These guidelines should answer the question: once I create a new version, who reviews it and how?

These guidelines can also vary a lot. If you're a solo writer, you might not need a review process at all.

If you only have a couple other authors, the policy might be: set version as ready for review and then DM or email the other person to review.

If you have many authors or subject matter experts, you might make use of the [In-app version review process](#) or [Version review process for readers](#).

If you're using any type of review process, be clear:

- When a version should be marked as "Ready for review"
- Who should be doing the reviewing
- How they'll be notified
- How any feedback from reviewer to version writer should be communicated

Version activation guidelines

These guidelines are often combined with the version review guidelines. They should answer the question: when is a new version activated?

Here, again, this will depend a lot on your review process and how you're using versions.

If you're creating versions in advance of a major release or update, you might be doing bulk activation on a specific date.

If you're creating versions as more one-off updates, it's more likely that your reviewer might just activate the version when they're done with their review.

Sample complete guidelines

If you're creating a style guide, author's guide, or something similar, we recommend including the version guidelines as part of that document.

Here's a sample of our own version guidelines, broken down above:

Small wording or content updates that might be more stylistic don't get new versions.

We create new versions for:

- Any update that changes a screenshot and related content: minor version
- Major updates to how a feature or function works or is configured: custom version using year.mm.dd format

- **Substantive style/readability/layout edits--if it moves a heading section around, for example: major version**

Any time you create a new version, add a version note that indicates the reason for version creation. (For example: updated Snippet Details screenshot to include the new References section.) In the rare case where we're creating versions based on the QA environment because the change is not yet live, indicate this in the version note by adding "Not released yet".

When you're done making edits:

1. Save the version with the Ready to review checkbox marked.
2. Create an Asana task in the Docs or it Didn't Happen project referencing this article/version, and assign to the other docs owl.

When reviewing a version:

1. Review the version note for a summary of changes.
2. Review the entire article for style, content, and layout adherence.
 - a. Fix any typos or wording issues.
 - b. If major issues with content, uncheck the ready for review box in KnowledgeOwl. Assign the Asana task back to the original writer owl with some notes in the task about what seemed wrong.
3. Once content looks good:
 - a. If this is for changes already released, activate version.
 - b. If for changes not already released, close the Asana task to indicate completeness. Version will be bulk-activated using Manage interface after release.