

Creating Release Ready Documentation Versions

August 25, 2026

Preparing source content before you create a release-ready version

Before you create a release-ready version in Atloria, review the source content set you plan to freeze for review. This is the point where you stop treating the content as an active draft and start treating it as a candidate for approval and publication. If you have already been tracking generation activity in [Monitoring Version Generation Jobs and Results](#), use that context to confirm which content set is ready instead of starting over from scratch.

Focus first on the pages reviewers are most likely to inspect: the project home page, section landing pages, release notes, setup guides, and any pages linked from top-level navigation. Open those pages in the project workspace and confirm the final wording is present. If a page is still being rewritten, still missing screenshots, or still sitting in a draft-only workstream, do not include it in the release-ready cut yet.

Next, check the page details that appear in generated output. In Atloria, that usually means the page title, description, navigation label, and any release-specific values that affect how the version appears in menus or page headers. A version can look complete at first glance but still confuse reviewers if the navigation label uses an old product name or the page description still reflects a previous release.

Then review assets and references. Open pages with screenshots, linked guides, reusable content blocks, and cross-links to other sections. Make sure images load, links open the intended destination, and shared content displays correctly in context. Pay extra attention to legal wording, product naming, release identifiers, and any approval-sensitive text. Those details often trigger review feedback even when the rest of the page is correct.

[SCREENSHOT: Project documentation page showing title, description, navigation label, and linked assets before version creation]

Choosing the right moment to cut a documentation version

The best time to create a documentation version in Atloria depends on how stable your release content is. In practice, most teams choose one of three moments: feature freeze, release candidate, or general availability. The right choice comes down to how much content churn is still happening and how much time reviewers need before publication.

If you cut the version at feature freeze, reviewers get more time to read and comment, but the version may miss late product changes, renamed features, or final screenshots. If you wait until general availability, your content may be more accurate, but formal review can become rushed and publication may slip. A release candidate often gives the best balance because the structure is usually stable while there is still time to fix review findings.

Use Atloria's version workspace as a coordination point with product, engineering, and review stakeholders. Before you create the version, confirm everyone is validating the same release scope. If engineering is still changing feature names, or product is still deciding what ships, your review copy can become outdated almost immediately. It is better to delay the cut slightly than to send reviewers a version that no longer matches the planned release.

You should also define what becomes locked once the version is created. For example:

- Allow minor corrections such as typos, punctuation fixes, and small wording clarifications
- Allow urgent factual fixes that prevent incorrect instructions
- Hold back larger rewrites, structural changes, and new sections for the next working draft
- Keep release naming, navigation structure, and review links stable unless there is a clear approval reason to change them

This decision helps everyone treat the version as a review candidate instead of an open editing space. That discipline is what makes a version truly release-ready rather than simply generated.

Creating the version intended for review and publication

Once the source content is stable, create the review version from the version management area in your project workspace. This is the step where Atloria turns your approved content snapshot into a distinct version that reviewers can inspect without being affected by later edits in ongoing work.

1. Open your project in Atloria and go to the documentation version area for that project. Use the version list or version workspace where existing versions and their statuses are shown.
2. Start a new version from the approved source content. If your team works from a current documentation set that is ready for release, choose that exact source. If multiple content snapshots are available, select the one your team agreed to freeze for review.
3. Enter the release version label exactly as it should appear to reviewers. This label should match the release naming used in navigation, page headers, and any version selector. Be consistent with punctuation, spacing, and numbering. A small mismatch here can create confusion later when reviewers compare comments against the version list.
4. Confirm that Atloria is creating a frozen version rather than pointing reviewers to the live working content. The goal is to preserve the review candidate so later edits in the main documentation flow do not silently change what reviewers are seeing.
5. Run the version generation action and wait for Atloria to create the reviewable output. When the process finishes, open the new version from the version list and verify that it appears as a separate item rather than replacing the current published documentation.

[SCREENSHOT: Version management screen with New Version action, version label field, and generated version listed separately]

If your team also uses comparison and readiness checks, pair this step with the review approach described in [Comparing Documentation Versions for Release Decisions](#). That helps you confirm the version you just created is the right candidate before formal approval begins.

Validating the generated output before formal review

After Atloria creates the version, open it immediately and validate the output before sending it for formal review. This step is not a full editorial pass. It is a focused check to make sure the generated version behaves like a stable release candidate and does not contain obvious publishing problems.

1. Start with the main entry points. Open the version home page, major section landing pages, release notes, and a few task pages that matter most for the

release. These pages are the first places reviewers usually visit, so they should look complete and use the correct release wording.

- 2. Check navigation inside the version. Open the sidebar, follow breadcrumbs, use previous and next links, and switch between a few pages to confirm you stay inside the same version. If Atloria shows a version selector, verify it displays the expected label and does not send you to unversioned or latest content by mistake.
- 3. Test links and media. Open pages with screenshots, linked procedures, and section cross-references. Watch for broken anchors, missing images, or links that jump outside the intended version. A single broken reference can make reviewers question the reliability of the whole release candidate.
- 4. Compare a sample of high-priority pages against the source content you froze. Check headings, tables, callouts, and embedded content. You are looking for rendering issues, missing sections, or formatting changes that appeared during generation.

A quick validation table can help you stay focused:

What to check	Where to look	What to confirm
Entry pages	Home page and section landing pages	Correct release wording and complete content
Navigation	Sidebar, breadcrumbs, previous/next links, version selector	Reviewers remain in the intended version
References	Internal links, anchors, screenshots, shared assets	No broken links or missing media
Page rendering	High-priority task pages and release notes	Headings, tables, and callouts display correctly

[SCREENSHOT: Generated version open in review mode with sidebar, breadcrumbs, and version selector visible]

Handling review updates without compromising the release candidate

Once reviewers begin commenting, protect the release candidate from turning into a moving target. In Atloria, the safest approach is to treat the generated version as the approved review baseline and make only deliberate, documented updates. That keeps review feedback tied to a stable version instead of a constantly shifting draft.

Start by separating feedback into two groups: changes that must be made before publication and changes that belong in the next documentation cycle. Publication-critical fixes usually include incorrect steps, wrong release names, missing screenshots, legal wording issues, or broken links on high-traffic pages. Broader rewrites, new examples, or content expansion requests are often better left in the ongoing draft unless they block release approval.

When you do need to update the release-ready version, track those edits clearly against the frozen version. Keep a simple record of what changed between review rounds, especially if you regenerate the version. Reviewers should be able to tell whether they are looking at the same release candidate with a few approved fixes or a substantially different copy. This is especially important when multiple teams are reviewing at once.

To avoid confusion:

- Keep the version label the same unless the release name itself changed
- Reuse the same review destination instead of creating parallel copies with similar names
- Limit updates to approved publication fixes
- Tell reviewers what changed before asking for another pass
- Avoid mixing next-release edits into the current review candidate

If the update is small, verify only the affected pages and linked paths. If you regenerate more broadly, repeat the validation checks from the previous section before sending the version back for review. For approval-specific workflows, align your update process with [Preparing a Version for Final Release Review](#) so the release candidate stays controlled all the way to sign-off.

Fixing common mistakes when creating release-ready versions

Most release-version problems in Atloria come from a small set of avoidable mistakes. When something looks wrong after generation, start by checking the version source, the version label, and whether the content was actually frozen at the time you created it.

If the version shows incomplete content, the most likely cause is that the wrong source content set was selected. Open the version details and compare the generated pages with the content you intended to freeze. If key pages are missing or older

wording appears, create the version again from the correct source snapshot rather than trying to patch around the problem page by page.

If reviewers report broken links or missing media, inspect pages that use screenshots, shared assets, and cross-links to other sections. These problems often appear when a page points to content outside the versioned set or when an asset was not available in the source you used. Open the affected page in the generated version and follow each visible link so you can identify exactly where the path breaks.

If the wrong release name appears in page headers, navigation, or the version selector, check both the version label and the page-level release wording used in the source content. Correct the naming mismatch and regenerate the version so the review copy stays consistent everywhere reviewers see it.

If changes keep appearing unexpectedly after version creation, confirm that your version was created from a frozen snapshot and not from live working content. A release candidate should not keep shifting as teammates edit the main documentation set.

Use this table to diagnose quickly:

Problem	Likely cause	What to do
Incomplete pages	Wrong source content selected	Recreate the version from the intended source snapshot
Broken links or missing images	References outside the versioned set or missing assets	Recheck linked pages, shared assets, and media paths
Wrong release name	Version label or page wording not updated	Correct naming and regenerate the version
Unexpected content changes	Version not frozen from a stable snapshot	Recreate the version from a fixed review source

[SCREENSHOT: Version details view showing version label, source selection, and generated output status]

Overview

Creating a release-ready documentation version in Atloria means more than clicking a generate action. You are preparing a stable review candidate that matches the release scope, uses the correct naming, and can move through formal review without constant rework. The version should be distinct from your ongoing draft content so reviewers can evaluate a fixed snapshot instead of chasing live edits.

This workflow usually starts in the project's version management area, where you create a new version from the content set your team has agreed to freeze. From there, you validate the generated output, confirm navigation and links behave correctly, and decide how tightly to control changes during review. The goal is to give stakeholders a version they can approve with confidence.

A release-ready version in Atloria should meet a few practical standards:

- The source content is complete enough for formal review
- The version label matches the release name reviewers expect to see
- The generated output appears as its own reviewable version
- Key pages, links, screenshots, and navigation elements work correctly
- Any updates during review are intentional, limited, and easy to explain

This guide focuses on that release-preparation stage. It does not repeat the broader job tracking process covered in [Managing Version Generation Jobs and Results](#) or [Monitoring Version Generation Jobs and Results](#). Instead, it shows how to turn a generated version into a reliable publication candidate that is ready for formal review and eventual release.

Prerequisites

Before you start, make sure you have the right project access and that the content you plan to version is actually ready to freeze. In Atloria, release-ready versioning works best when the project team has already aligned on scope, naming, and review timing.

You should have the following in place:

- Access to the project workspace where documentation versions are managed
- Permission to open the version list and create a new version
- A stable source content set that is ready for formal review
- Agreed release naming for the version label
- Reviewer awareness of which release scope is being documented
- Enough time to validate the generated output before sending review links

It also helps if you have already completed these related tasks:

- Generated at least one documentation version before, using the workflow in [Generating New Documentation Versions](#)
- Reviewed how version comparisons support release decisions in [Comparing Version Output and Release Readiness](#)

- Checked current generation activity in [Monitoring Version Generation Jobs and Results](#)

Before cutting the version, confirm that high-visibility pages are ready. That includes the documentation home page, section landing pages, release notes, setup instructions, and any pages likely to be reviewed by product, support, or legal stakeholders. If those pages still contain placeholder text, incomplete screenshots, or draft-only wording, finish that work first.

After your release-ready version is created and validated, continue with [Monitoring Version Generation Progress and Results](#) to follow the version through the next stage.