

Managing Project Version Timelines and Status Decisions

August 25, 2026

Reading version timeline details and progress signals

In Atloria, start from your project workspace and open the version you want to review. If you need help getting back to the right version workspace, use [Managing Project Version Workspaces](#). On the version record, focus first on the timeline details shown around the version status and progress information. These details help you decide whether the version is moving normally or slipping behind the project schedule.

Look for date fields that show the planned timing of the version. In most cases, the key dates to compare are the **planned start date**, the **target release date**, and any visible completion or release marker that shows when work actually finished. When the target date is still ahead, the version may still be on track. When that date has already passed and the version is not complete, treat it as a delay that needs a status check before anyone moves it forward.

The progress area is the next place to review. Atloria may show a completion percentage, a status badge, or other progress signals that tell you whether work is still active, already finished, delayed, or no longer continuing. Read these items together rather than one at a time. A version can show high completion but still be delayed if the target release date has passed. It can also show an active status even though the planned timeline suggests it should already be under review.

Use the version status area to separate:

- versions still being worked on
- versions delayed against the target date
- versions completed and ready for the next decision
- versions canceled or no longer moving forward

[SCREENSHOT: Version record showing status badge, progress indicator, planned start date, target release date, and completion marker]

Before making any release decision, compare the version's timeline with the broader project schedule shown in the surrounding project workspace. That quick comparison

helps you see whether the version is aligned with the release plan or whether its dates and status need to be corrected first.

Checking whether a version is ready for review or release

When you open a version in Atloria, do not rely on a single badge to decide that it is ready. Review readiness is usually a combination of progress, current status, and whether any blocking items are still visible on the version. Start by checking whether the version appears complete enough to leave active work and enter review. If the version still shows unresolved work, incomplete progress, or a status that clearly indicates ongoing changes, it is usually too early to move it toward release.

Compare the current state of the version with its planned scope. If the version was intended to include a specific set of documentation updates, make sure those updates are actually reflected in the version workspace before changing any status. A version can look nearly complete from a timeline point of view while still missing important content or review work. That is why readiness should be checked from both the schedule side and the work-completion side.

Use the review-related status area on the version to confirm whether required checks have already happened. In Atloria, the version may show review status badges or decision indicators that tell you whether the version is still in progress, under review, approved, deferred, or blocked by a decision that requires more work. If a blocking decision is still visible, do not treat the version as ready for release even if the progress indicator looks complete.

Keep these three states separate when you review a version:

Version state	What it means
Functionally complete	The work appears finished enough to stop editing and begin review
Approved	A review decision has been recorded allowing the version to move forward
Released	The version has actually been moved into its released state

These are not the same milestone. A version can be complete but not approved, or approved but not yet released. [SCREENSHOT: Version detail view with readiness indicators, review status badge, and release-related status area]

If you need a deeper comparison before deciding, pair this check with [Working with Version Comparison Views](#) or [Comparing Version Output and Release Readiness](#).

Recording review outcomes and updating version decisions

Once a version has reached review, open its review or decision controls and record the outcome based on the current state of the work. In Atloria, this is where you move beyond general progress tracking and make a formal decision that other team members can see. Choose the outcome that matches the review result, such as **Approved**, **Needs Changes**, **Deferred**, or **Rejected**, if those options are available on the version.

1. Open the version record from the project workspace.
2. Find the review decision area or status controls on the version.
3. Select the outcome that matches the review result.
4. Add any notes, comments, or internal explanation that should stay with the version.
5. Save the change and confirm the updated status appears on the version.

Be specific when you enter notes. Instead of leaving a short comment that only says the version is delayed, explain what caused the hold, what still needs attention, or why the version was approved despite a schedule shift. These notes are especially useful when another documentation manager, approver, or project lead returns to the version later and needs to understand the decision without reopening every related document.

After saving the review outcome, make sure the lifecycle status also reflects that decision. For example, if the review result is positive, the version should no longer look like active draft work. If the review result is **Needs Changes** or **Deferred**, the visible status should make that clear so nobody mistakes it for a release-ready version.

Then check the version history or activity area. Atloria should show that the review decision was saved, making the change visible for later reference. This is important when teams need to confirm who made the decision and when it happened.

[SCREENSHOT: Review decision area with outcome selection, notes field, and saved status shown in the version activity history]

For broader approval workflows, see [Managing Version Review Requests and Decisions](#).

Managing release-related status changes across the version lifecycle

A version's status in Atloria should follow the actual stage of work, not just the date on the calendar. As the version moves from planning into active work, then review, approval, release, and closure, use the version status selector carefully so the visible lifecycle matches reality. This matters because teams often use the status badge as the quickest signal for whether a version can still change, needs attention, or is ready to ship.

1. Start with the version in its early planning or draft state while dates and scope are still being set.
2. Move it into an active work state when documentation updates are underway.
3. Change it to a review-related state when editing is complete enough for formal review.
4. Mark it approved only after the review decision supports moving forward.
5. Change it to released only when the version is actually being released.
6. Use a closed or completed state only after release work is finished and no further action is expected.

Avoid skipping steps. A version should not be marked **Released** just because the target release date has arrived. If review is incomplete or approval has not been recorded, keep the version in the correct pre-release state until those actions are done. This keeps the project workspace trustworthy for everyone who depends on it.

Atloria also needs to reflect exceptions clearly. If a release is delayed, update the status so it no longer looks on schedule. If an approved version must be reopened, change the status back to an active or review state and adjust the dates if needed. If the version will not ship at all, use the canceled or postponed state available on the version rather than leaving it in a misleading in-progress or approved state.

After every status change, recheck:

- the current status label
- the target release date
- any actual release or completion marker
- whether the progress display still matches the lifecycle stage

[SCREENSHOT: Version status selector showing lifecycle stages from draft through released, with release date fields nearby]

Using version history to explain schedule and status decisions

The version history or activity stream in Atloria is where you confirm how a version reached its current state. When a target release date changes, a review decision is reversed, or a version suddenly appears delayed, the history gives you the clearest explanation without relying on memory or side conversations. Open the version and look for the activity area that lists recent changes tied to status, dates, and review updates.

Use this history to answer practical questions:

- Who changed the version status?
- When was the target release date updated?
- Was the version moved because of a review decision or a release action?
- Did the version move from approved back into active work?
- When was it marked ready, deferred, or released?

Read the entries in order so you can follow the timeline of decisions. For example, if the version now shows a later release date, check whether the history also shows a deferred review outcome or a reopened status change around the same time. If the visible status says **Approved** but the latest activity shows a newer blocking decision, that mismatch tells you the version needs correction.

This history is especially useful when you need to explain schedule changes to project leads or other administrators. Instead of saying the version was delayed at some point, you can point to the exact activity sequence: the original target date, the review result that paused progress, and the later update that changed the release plan. That makes the version record easier to trust and easier to hand off between team members.

[SCREENSHOT: Version activity stream showing date changes, status updates, review outcomes, and user names]

When you use history this way, always compare it with the current version header. The latest activity and the visible status should tell the same story. If they do not, update the version before anyone uses it for release planning.

Fixing inconsistent timelines, readiness states, and release statuses

Sometimes a version in Atloria shows mixed signals. You might see a version that looks 100% complete but is still marked as active work, or a version with a release date filled in even though it is not marked released. These mismatches can confuse

reviewers and lead to the wrong release decision, so fix them directly on the version record before moving forward.

Start by comparing the most visible fields on the version:

- progress or completion indicator
- current status badge
- target release date
- actual completion or release marker
- review outcome or approval state

If the progress looks complete but the version is still blocked from review, check whether the review decision is missing or whether the status was never advanced from draft or in progress. If the version appears ready but still carries a blocking review result such as **Needs Changes**, **Deferred**, or **Rejected**, update the review outcome only after the underlying issue has actually been resolved.

For release-date mismatches, make sure the dates and status agree with each other. A version should not show an actual release marker while still carrying an unreleased status. In the same way, a version marked **Released** should not still look like it is awaiting review. If a previously approved version was reopened, update both the lifecycle status and any related timeline fields so the record no longer looks final.

1. Open the version and review the status, dates, and decision area together.
2. Check the version history to see which value was changed last.
3. Correct the status or review decision that no longer matches the current state.
4. Update any date fields that should reflect the latest plan.
5. Save the version and confirm the header, readiness signals, and history now align.

[SCREENSHOT: Version with mismatched progress, review status, and release date fields highlighted for correction]

If inconsistencies keep appearing, compare the version against your project's normal release flow and the guidance in [Understanding Version Lifecycle and Release Readiness](#).

Overview

This guide focuses on the decision-making side of version management in Atloria. You use these steps after a version workspace already exists and work is underway. The goal is to read the version's timeline, understand whether it is on schedule, decide if it

is ready for review, and keep the status aligned with what is actually happening in the release cycle.

The most important screens are the project version record, the version status area, the review decision controls, and the version history or activity stream. Together, these parts of Atloria show whether a version is still being worked on, delayed against its target release date, approved for release, or already released. They also help you explain why a decision was made when someone asks later.

Use this guide when you need to:

- judge whether a version is on track
- decide if a version should move into review
- record an approval, deferral, rejection, or change request
- update release-related statuses without creating confusion
- investigate why a version's dates and status do not match

This guide does not repeat workspace navigation or general version browsing. For that foundation, refer to [Managing Project Version Workspaces](#). It also does not go deep into side-by-side comparisons or detailed approval workflows, which are covered in related version documents.

A good working habit in Atloria is to read the version header, timeline fields, readiness indicators, and activity history together before changing anything. That quick review helps you avoid marking a version ready too early or leaving it in an outdated state after a review decision.

[SCREENSHOT: Project version page showing the main areas used for timeline review, status decisions, and history checks]

Prerequisites

Before you manage timelines and status decisions in Atloria, make sure you already have access to a project that includes at least one documentation version. You should also be comfortable opening the version workspace and identifying the version record you want to review. If you are not there yet, start with [Managing Project Version Workspaces](#).

You will get the most value from this guide if the following are already true:

- You can open a project and locate its versions.

- You understand the basic difference between draft work, review, approval, and release states.
- You have permission to update version details or record review decisions.
- The version already contains enough work to evaluate progress and readiness.
- Your team is using version statuses consistently enough that the visible badges and dates matter for release planning.

It also helps to have related information available before you begin reviewing a version:

- the planned release timing for the project
- the version's current target release date
- any review feedback already recorded on the version
- the latest activity history for recent status or date changes

If your team is still deciding how statuses should be interpreted, read [Managing Version Lists Statuses and Comparisons](#) and [Understanding Version Lifecycle and Release Readiness](#) before making release decisions.

After you are comfortable reading timelines and recording status changes, continue with [Coordinating Version Work Before Release](#) to organize the final handoff from active version work into release preparation.