

Using Project Activity Signals to Improve Release Planning

August 25, 2026

Identifying the activity signals that matter before a release

Before you start a release review in Atloria, focus on the signals that tell you which pages may need attention. The most useful signals are the ones you can see directly in the project workspace: recently updated pages, pages with open review comments, topics that have not been updated in a long time, pages with broken or outdated links, and pages that show low recent views even though they belong to an important release area. These signals help you spot risk before the version moves into review or publishing.

Documentation Managers and Technical Writers usually read these signals together rather than one at a time. A page with a recent edit is not automatically a problem, but a page with a recent edit plus unresolved comments is a stronger sign that the content may not be ready. In the same way, a stale topic is not always wrong, but if it sits in an area that changed for the upcoming release, it deserves a closer look. Atloria's analytics views, page status labels, and activity feed give you this context in one place.

Use these signals at three release checkpoints:

- **Internal version review:** check whether pages are still changing, waiting for review, or carrying unresolved comments
- **Publishing approval:** confirm that pages expected to ship are in the right status and do not show obvious warning signs
- **Audience-facing release:** verify that the most visible pages are accurate and aligned with the version being released

Treat activity signals as prioritization tools, not as a replacement for page-by-page review. Analytics can tell you where to look first, but they do not decide whether the wording, screenshots, structure, or version details are correct. If you need a refresher on reading project metrics before this step, use [Reading Project Analytics for Documentation Decisions](#).

[SCREENSHOT: project analytics area showing page status labels, recent activity, and content signals before release]

Reviewing project dashboards and activity feeds

In Atloria, start from the project workspace and open the dashboard or analytics area for the release you are preparing. Look for panels that show recently updated pages, pages with open review comments, and any content tied to an upcoming release window. These areas help you narrow your attention to the topics most likely to affect release readiness.

1. Open the project workspace for the release you are planning.
2. Go to the dashboard or analytics view that shows project activity.
3. Review the panels for recently updated pages and note any topics changed close to the release date.
4. Open the activity feed and check who made the change, when it happened, and whether it happened after the last review cycle.
5. Apply available filters such as version, collection, or content owner so you are only looking at pages included in the upcoming release.
6. Compare page status labels across the filtered list to find pages still marked **Draft**, **In Review**, **Approved**, or **Published**.

The activity feed is especially useful when a page looks ready at first glance but may have changed after approval. If a topic is marked **Approved** but the feed shows a late edit, that page should go back into your release discussion. If a page is still marked **Draft** while related pages are already **Approved**, that mismatch can delay the release even if the content itself looks complete.

When you filter by version or collection, keep your release scope tight. This prevents older pages or unrelated work from crowding the list. Filtering by content owner is also helpful when you need to see which writer or reviewer still has open work.

[SCREENSHOT: project dashboard filtered by version with recent updates, open comments, and page statuses visible]

Using analytics to decide which content needs attention

Once you have the right release scope on screen, use Atloria's analytics to decide which pages deserve immediate review. Start with page-view trends. High-traffic pages usually need extra validation because any mistake will affect more readers after

release. Low-traffic pages still matter, especially if they cover setup, permissions, or advanced tasks, but they may not need the same level of urgent review unless another signal points to risk.

Search behavior can reveal gaps that page views alone do not show. If Atloria shows failed search terms or weak engagement on a page, that may mean readers are not finding the answer they expect. For release planning, this is a strong reason to inspect the page title, headings, version details, and links to related content. A page that readers search for often but do not engage with may need a clearer update before publishing.

Also check for outdated content signals:

- Old **Last updated** timing on a page in an area that changed for this release
- Missing or inconsistent version labeling
- Pages that have not changed even though the surrounding feature area has changed
- Topics with recent edits but no matching review progress

The most important pages are usually the ones where signals overlap. For example:

Signal combination	What it usually means for release planning
High views + unresolved comments	Review before publish
Recent edit + already approved status	Recheck approval and content changes
Low engagement + failed searches	Improve findability or fill a content gap
Stale page + changed feature area	Validate accuracy before release

Do not rank pages by traffic alone. In Atloria, the best release decisions come from combining audience behavior, activity history, and page status. That gives you a practical list of pages that are both visible and potentially risky.

Turning signals into a release-ready content priority list

After reviewing the dashboard, activity feed, and analytics, turn what you found into a working priority list inside your release workflow. The goal is not to create a perfect report. The goal is to produce a short, clear list of pages that the team can act on before version review and publishing approval.

1. Filter the project view to show pages with open comments, stale review status, recent edits, or strong audience traffic.

2. Add the pages that matter to your release working list.
3. For each page, assign a release decision based on what you found.
4. Route each page to the correct owner or reviewer.
5. Record why the page was flagged so the team can understand the decision quickly during review.

A simple release decision set works well in Atloria:

Release decision	When to use it
Update now	The page has clear issues that must be fixed before release
Review before publish	The page may be correct, but recent activity or comments require a final check
Defer to next version	The issue is real but outside the current release scope
Monitor after release	The page is acceptable to ship, but usage should be watched closely

Use ownership fields, assignee columns, or team labels to send urgent pages to the right Technical Writer or reviewer. This is especially helpful when one release includes multiple collections or content areas. A page should not sit in a general list without an owner if you expect action before publishing.

Always capture the reason for each priority call. For example, note whether the page was flagged because of high traffic, unresolved comments, a stale update date, or a late edit after approval. That short note gives version reviewers the context they need without reopening every page from scratch.

[SCREENSHOT: release planning list with page title, status, owner, and priority decision]

Coordinating version review and publishing decisions with the team

A prioritized list is most useful when the whole team works from the same view in Atloria. During version review meetings, Documentation Managers can use that list to keep the discussion focused on pages with the highest release risk instead of reviewing every topic in the same depth. This helps the team spend time where it matters most: pages with late edits, unresolved comments, status mismatches, or strong audience impact.

1. Open the release-scoped page list during the version review meeting.
2. Start with pages marked **Update now** and **Review before publish**.

3. Ask the assigned writer or reviewer to confirm what action is needed on each page.
4. Update the page status, comments, or approval state as decisions are made.
5. Recheck the release view before moving the version toward publishing approval.

Technical Writers should confirm the exact type of work required for each flagged page. In practice, that usually falls into one of these categories:

- Content edits for accuracy or completeness
- Metadata updates such as version details or page organization
- Link fixes for broken or outdated navigation paths
- Approval changes when a page was edited after review

Keep the project workspace current while the meeting is happening. If a page moves from **Draft** to **In Review**, or from **In Review** to **Approved**, update that status right away. If a reviewer leaves a decision in comments, make sure the comment thread reflects whether the issue is still open. This keeps the release view trustworthy for everyone involved.

For handoff, use one lightweight flow: flagged page review, content update if needed, approval confirmation, then publishing approval. Because the same activity signals stay visible throughout the process, the team can track whether a page still carries risk without switching to a separate planning method.

Fixing common issues when activity signals are unclear or misleading

Sometimes the signals in Atloria do not tell a clear story on the first pass. When that happens, the fix is usually to compare more than one view instead of trusting a single number or label.

If analytics show low traffic for a page you know is important, first check your filters. The page may be outside the selected version, tied to a different audience segment, or affected by a recent page URL change. A page that moved or was renamed can appear quieter than expected if you are only looking at the latest path or a narrow version view.

If a page looks release-ready but still feels risky, inspect the details behind the status label. Open the page and look for unresolved inline comments, pending approvals, or edits that happened after the last review timestamp. A page marked **Approved** is not truly ready if someone changed key content after that approval.

When too many pages appear stale, reduce the scope so the team can act on the current release instead of the entire project history. Narrow the view by:

- Release label
- Content owner
- Modified date
- Collection or section included in the release

If recent activity is visible but priorities are still unclear, combine the signals. A late edit on a low-impact page may not matter. A late edit on a high-traffic page with open comments probably does. In Atloria, the clearest priority decisions come from reading activity history together with page status and audience metrics.

If you still have trouble deciding what belongs in the release conversation, return to the release scope and ask one practical question for each page: does this page create a meaningful risk for internal review, publishing approval, or the audience-facing release? That question usually separates background noise from real release work.

Overview

- This guide focuses on using visible project signals in Atloria to improve release planning for documentation work.
- The main signals covered here are:
 - Recently updated pages
 - Open review comments
 - Stale topics
 - Broken or outdated links
 - Low recent views
 - Page status labels such as **Draft**, **In Review**, **Approved**, and **Published**
- You use these signals during three key release checkpoints:
 - Internal version review
 - Publishing approval
 - Audience-facing release
- The workflow in this guide builds on the analytics-reading approach from [Reading Project Analytics for Documentation Decisions](#).
- In Atloria, the recommended process is:

- Review the project dashboard and activity feed
- Filter the release scope by version, collection, or owner
- Compare analytics with page status and review history
- Build a short priority list for the team
- Update statuses, comments, and approvals as release decisions are made
- This guide does not replace editorial review. It helps you decide where to look first so the team can spend review time on the pages most likely to affect release quality.

[SCREENSHOT: release planning workflow in Atloria from analytics review to publishing approval]

Prerequisites

- You should already know how to read project-level analytics in Atloria. If needed, review [Reading Project Analytics for Documentation Decisions](#).
- You need access to the project workspace that contains the upcoming release.
- You should be able to open the project dashboard or analytics area and view the activity feed.
- You need visibility into the pages included in the release, ideally with filters for:
 - Version
 - Collection
 - Content owner
- The release workflow is easier to manage if your team is already using page status labels such as **Draft**, **In Review**, **Approved**, and **Published**.
- You should also have access to the page comments or review history used by your team during version review.
- This guide is most useful for:
 - Documentation Managers preparing version review
 - Technical Writers checking pages before publishing
 - Reviewers helping confirm release readiness across a project

For the next step in this workflow, continue with [Using Project Analytics and Audit Signals for Ongoing Operations](#).