ATLORIA

Using Analytics to Prioritize Documentation Improvements

August 25, 2026

Defining the documentation signals you will track

Before you start ranking documentation work, decide exactly which signals you will review in Atloria and keep that list stable from one review cycle to the next. If you already worked through Analyzing Documentation Performance Across Projects, use the same project and reporting views here so your comparisons stay consistent.

In Atloria, focus on a small set of signals that clearly support content decisions:

Signal	What to look for	What it usually helps you decide
Pageviews	Pages visited often	Which pages affect the most readers
Unique visitors	How many different readers reached a page	Whether demand is broad or concentrated
Average time on page	Very short or unusually long reading time	Whether readers are skimming, getting stuck, or reading deeply
Exit rate	Readers leaving after a page	Whether a page may be ending the journey too early
Search queries	Repeated words and phrases	Which topics readers expect to find
Zero-result searches	Searches with no matching page	Which topics or terms are missing

Do not mix every page type into one list. Separate onboarding guides, troubleshooting articles, release notes, and reference content before you compare them. A short visit on a release note may be normal, while the same pattern on a setup guide may point to a problem.

When you open your analytics view, confirm three things before you record any findings:

- The date range, such as the last 7 days or last 30 days
- Any traffic filters you are using
- The content group or project area you are reviewing

[SCREENSHOT: Analytics view showing date range, filters, and grouped documentation content]

Keeping these settings the same each week or month makes your priority list more trustworthy. Otherwise, a page can appear to improve or decline simply because you changed the reporting window or compared different content groups.

Finding where readers struggle in the docs journey

Once your signals are defined, use them to find where readers lose momentum. In Atloria, start with the pages that bring readers in. Top landing pages often reveal the first place someone tries to solve a problem, whether they arrived from a search engine, a product link, or a support response.

Look closely at landing pages that combine high traffic with weak engagement. A page that attracts many visits but shows short reading time or quick exits may not be answering the question the title promised. This is especially important for task-based pages, where readers usually expect a direct set of steps and a clear result.

Site search behavior gives you another strong clue. Review repeated searches, alternate spellings, and searches that return no results. These patterns often point to one of three issues:

- The topic does not exist yet
- The page exists, but the title uses different wording
- Navigation labels do not match the language readers use

For example, if readers repeatedly search for the same phrase after landing on a popular overview page, that overview page may need clearer links to the next task or troubleshooting article.

Support activity helps confirm whether the documentation gap is real. Compare common support topics with the pages readers visit before contacting support. If a topic receives both heavy documentation traffic and frequent support requests, that page deserves attention even if its pageviews alone do not make it your top item.

[SCREENSHOT: Report showing top landing pages, engagement metrics, and search terms side by side]

As you review, group findings by journey stage rather than by page alone. A reader may land on an overview, search for a setup term, then open a troubleshooting page. When several weak signals appear across that path, you have found a documentation journey that likely needs improvement.

Turning analytics patterns into a content priority list

After you identify weak spots, turn them into a ranked list that your team can act on. In Atloria, avoid sorting pages by pageviews alone. A page with moderate traffic can still deserve urgent work if it affects a critical workflow, drives support volume, or has not been updated in a long time.

A simple priority matrix works well when you combine four factors:

Factor	What to consider
Traffic volume	How many readers the page affects
Support impact	Whether the topic still generates support requests
Content freshness	How old the page is and whether the interface has changed
Business importance	Whether the page supports onboarding, setup, publishing, or another key workflow

Use these patterns to decide what kind of work each page needs:

- **Rewrite** pages when you see high visits, high exits, and continued support demand on the same topic.
- **Expand** pages when readers stay on the page but continue searching for related terms, which usually means the content is helpful but incomplete.
- **Consolidate** pages when several low-traffic articles compete for the same search phrase or cover the same workflow in fragments.

This approach helps you avoid spending time on pages that are merely popular. Instead, you focus on pages where better content can improve reader outcomes.

[SCREENSHOT: Documentation priority list with impact level, traffic, support volume, and recommended action]

When you build the list, add one clear reason beside each page. For example: “High-traffic onboarding page with short engagement and repeated follow-up searches” is much more useful than “Needs update.” That wording makes it easier for writers, support leads, and project owners to agree on why the page belongs near the top of the backlog.

Improving pages based on the signals you found

Once you have a priority list, update each page based on the specific signal that triggered it. In Atloria, the most effective improvements usually start with language. If

internal search reports show that readers use a different phrase than your page title or heading, rename the page and section headings to match the wording readers already use.

For pages with quick exits after readers arrive from product links or support responses, review the page structure first. Make sure the page includes:

- A title that matches the task
- Clear step-by-step instructions
- Any required setup or conditions before the task starts
- The expected result at the end of the task

If readers spend time on a page and then search again, the page may need expansion rather than a rewrite. Add the next logical step, a troubleshooting section, or links to related reference pages. This is especially useful for overview pages that introduce a feature but do not yet guide readers to the exact screen or action they need next.

Cross-linking is another high-value fix. Add links between overview pages, troubleshooting articles, and reference content that readers commonly open in sequence. This reduces the need for repeated searching and helps readers move through the documentation more naturally.

Also review screenshots and interface wording on high-traffic pages with weak engagement. If button labels, menu names, or screen titles have changed in Atloria, readers may leave because the instructions no longer match what they see.

[SCREENSHOT: Documentation page showing updated title, clearer headings, related links, and refreshed screenshot]

Treat every edit as a response to a measured problem. That keeps your updates focused and makes it easier to check later whether the change actually improved the page.

Sharing analytics-backed priorities with writers and support teams

A priority list only becomes useful when other teams can review it quickly and understand why each item matters. In Atloria, create a shared reporting view that brings the most important signals together in one place. Keep it simple enough that a writer, support lead, or admin can scan it without opening several separate reports.

A practical shared view should include:

Column	Why it matters
Page or topic	Identifies the content to update
Owner	Shows who is responsible
Last updated	Highlights stale content
Pageviews	Shows reader demand
Search demand	Confirms what readers are trying to find
Related support volume	Shows customer impact
Priority level	Helps teams agree on order

Use a common label set such as **High-impact**, **Medium-impact**, and **Maintenance**. These labels are easier to discuss than raw numbers alone, especially when different teams care about different outcomes. A support lead may focus on ticket volume, while a writer may focus on content age and clarity.

For each proposed change, include the metric you expect to improve. Examples include:

- Reduce zero-result searches for a missing topic
- Lower exits on a setup page
- Increase engagement on a troubleshooting article
- Reduce support requests tied to a known issue

[SCREENSHOT: Shared documentation priority report with owner, impact level, and expected outcome]

Review this list on a regular schedule with the people who manage documentation, support, and project priorities. A fixed weekly or monthly review works better than ad hoc updates because urgent customer pain points can be added quickly, while lower-impact maintenance work stays visible without taking over the queue.

Checking whether your documentation changes improved outcomes

After you publish an update, return to the same Atloria analytics view you used during planning and compare the results against your earlier baseline. Use the same date range style, filters, and content grouping so the before-and-after comparison is fair.

For each updated page, check whether the signals moved in the direction you expected:

- Did exits decrease?
- Did time on page become healthier for that page type?
- Did repeated searches or refinements drop?
- Did related support requests decline?
- Did readers continue to the next page more often?

Be careful with misleading changes. Traffic spikes caused by a release announcement, a support campaign, or a seasonal event can make a page look better or worse than it really is. URL changes can also break comparisons if the old page and new page are not being reviewed together.

If engagement improves but support demand does not change, the page may still be missing the real issue readers contact support about. In that case, compare the updated page with recent support topics and check whether the article covers the exact workflow support is handling.

If traffic drops sharply after an update, do not assume the page became less useful. First verify that the page can still be found through search, internal links, and any product or help links that previously sent readers there.

[SCREENSHOT: Before-and-after analytics comparison for an updated documentation page]

A good review cycle ends with a decision, not just a measurement. Mark each page as successful, needs another revision, or needs a broader content change. The next document, Managing Enterprise Analytics for Documentation Programs, shows how to scale this process across larger documentation teams and multiple projects.

Overview

This guide shows how to use Atloria analytics to decide which documentation pages should be updated first. The goal is not to collect every available number. Instead, you use a focused set of signals to find pages that attract readers, fail to answer questions, or continue to generate support demand even after people visit the docs.

The workflow in this guide follows a practical sequence:

1. Define the signals you will track consistently.
2. Review landing pages, search behavior, and support patterns to find friction.

3. Turn those findings into a ranked content priority list.
4. Update pages based on the specific issue the data revealed.
5. Share the priorities with writers and support teams.
6. Measure whether the changes improved outcomes.

This document builds on Analyzing Documentation Performance Across Projects. That earlier guide focuses on reading performance across projects. Here, the focus is narrower: deciding what to rewrite, expand, consolidate, or relabel inside your documentation backlog.

Use this guide when you already have documentation traffic and want to make better editorial decisions with it. It is especially useful when your team needs to choose between several possible updates, such as improving onboarding guides, expanding troubleshooting coverage, or cleaning up duplicate content around the same workflow.

In Atloria, this work is most effective when you compare analytics with real reader behavior, including internal search terms and support activity. That combination helps you move beyond “most visited pages” and identify which content changes are most likely to improve customer outcomes.

Prerequisites

Before you use this workflow in Atloria, make sure you have access to the reporting areas and content views needed to compare documentation performance. You do not need every admin feature, but you do need enough visibility to review page activity and connect it to real documentation work.

Prepare the following before you begin:

- Access to the relevant analytics view for the project or documentation area you want to review
- A clear date range you plan to use consistently, such as weekly or monthly reporting
- A content grouping approach, such as onboarding guides, troubleshooting pages, release notes, or reference content
- Access to internal search reporting, including repeated searches and zero-result searches if available in your reporting setup
- A way to review related support trends, such as topic tags, case categories, or another shared support summary

- A current list of documentation pages, topics, or owners so you can assign follow-up work

It also helps to know which pages are tied to important workflows in Atloria, such as project onboarding, publishing, approvals, analytics, or support agent setup. Those pages may deserve higher priority even if they do not have the highest traffic.

If you are still getting familiar with Atloria navigation or reporting areas, review Using the Admin Workspace and Monitoring Administrative Analytics and Activity. If your focus is project-level performance rather than cross-project reporting, Analyzing Project Performance and Activity can help you frame the right comparisons before you build your priority list.