paligo

October 2025

Paligo vs Confluence

From Unstructured Content in Confluence to Structured Content in Paligo

Paligo AB

Executive Summary

Effective content creation and management are foundational to high-quality technical documentation, scalable operations, and getting AI-ready. Yet, many teams rely on unstructured content tools like Confluence. These tools frequently result in inefficiencies, duplication of content, and high operational costs, especially as businesses scale, introduce product variations, or expand into new markets.

Through workflow visualizations and real-world scenarios, we demonstrate how structured content:

- Reduces duplication
- Reduces errors
- Improves formatting consistency
- Simplifies multilingual documentation management
- Streamlines multichannel publishing
- Optimizes content for AI apps, such as chatbots and knowledge bases

This guide will help you compare and contrast structured and unstructured content creation workflows and, therefore, understand the tangible benefits of adopting structured content.

Structured authoring has changed many technical writers' working day for the better. Single-sourcing their content reduces the stressful "cognitive load" of managing large sets of complex documentation. Knowing that everything is in control, consistent, and high-quality gives those who switch to structured authoring confidence and pride in the work that they're delivering.

Index

paligo

The Modern Challenges of Managing Technical Documentation

The Modern Challenges of Managing Technical Documentation

Customer self-service is growing in demand every day, with the majority of customers preferring to resolve issues independently. A recent Salesforce study found that 61% of customers prefer to use self-service to resolve simple issues. As self-service becomes the cornerstone of modern support strategies, the demand for consistent, accurate, and high-quality technical and product documentation has never been greater.

However, as much as customers demand access to accurate documentation, most organizations struggle to deliver it; not because they lack the information, but because they don't have the tools and processes to create, manage, and publish it efficiently and effectively. For those organizations that seek to leverage the power of AI in their knowledgebases and help centers, the challenges of unstructured content and content chaos also hold them back.

The challenges of managing technical documentation stem from several areas. For starters, documentation teams often use a mix of tools, each with its own capabilities and issues. Even when a single tool is used, issues can result (as we'll show you).

Documentation is typically in an unstructured format, such as MS Word, Google Docs, Confluence wikis, or PDFs. It's page- or article-based, with style and formatting mixed with the content. Unstructured content does not support content reuse, which means technical authors must frequently copy and paste the same information within and across documentation, and then manually track where this is happening.

In some cases, instead of copy and pasting content, authors are creating it from scratch, leading to inconsistencies in how information is written and shared.

Collaboration among authors, editors, and subject matter experts is also manual, with a lot of emailing of documents back and forth, and trying to resolve multiple edits from multiple reviewers.

For organizations that provide documentation in multiple languages, the translation workflow is often manual, more costly, and takes longer, resulting in inconsistent information across languages.

And organizations that need to deliver technical documentation to more than one channel (and in more than one format), such as an online Help Center, a Knowledge Base, a customer support portal, or via PDF, find they must create and manage separate versions of their documentation, making it difficult to maintain and keep the versions in sync.

One other important challenge is budget. Although organizations understand the need to provide customers with accurate information, technical documentation teams work with limited budgets for resources and technology.

Is there an answer to these challenges? Yes.

paligo

Structured Content and Component Content Management Systems (CCMS)

Structured Content and Component Content Management Systems (CCMS)

Technical documentation

Technical documentation teams are smart. They know that unstructured content is the biggest challenge they need to overcome if they want to develop scalable, accurate documentation across customer channels. And they know the answer is switching to a structured content model.

Structured content is content that's separated from its presentation and is written by breaking the content into reusable topics (or components). You can create structured content manually but you'll have more success if you use a component content management system (CCMS).

A CCMS, like Paligo, provides a structured authoring environment that enables you to create all your technical documentation in one place (called single sourcing). With a CCMS, you can create your content, collaborate on it with your team and subject matter experts, track versions, send it for translation (and track translations against the source language), and leverage features like content variants, filters, and profiles. Additionally, a CCMS enables the export of content in multiple formats (e.g., HTML, HTML5, XML, PDF) and across multiple channels (e.g., Knowledge Base, website, customer portal).

It sounds like a CCMS is the best tool for managing technical documentation, right? But not every organization uses a CCMS. Instead, some are using unstructured content tools, such as Microsoft Word, Google Docs, Confluence, Adobe RoboHelp, and Zendesk, among others.

We're not saying these tools are inherently bad. There is a time and place where they might be right for your needs. The key is to understand how they work in comparison to a CCMS, so you can determine the right tool for your documentation team. That's what we're going to do.

paligo

Confluence Wiki Flexibility, Documentation Chaos

Confluence
Wiki Flexibility, Documentation Chaos

Atlassian Confluence is team collaboration and knowledge management software designed to help teams create, share, and organize information. It is primarily used by internal teams to share information and collaborate. However, some organizations have chosen to use it as an external knowledge base for customers.

Confluence provides:

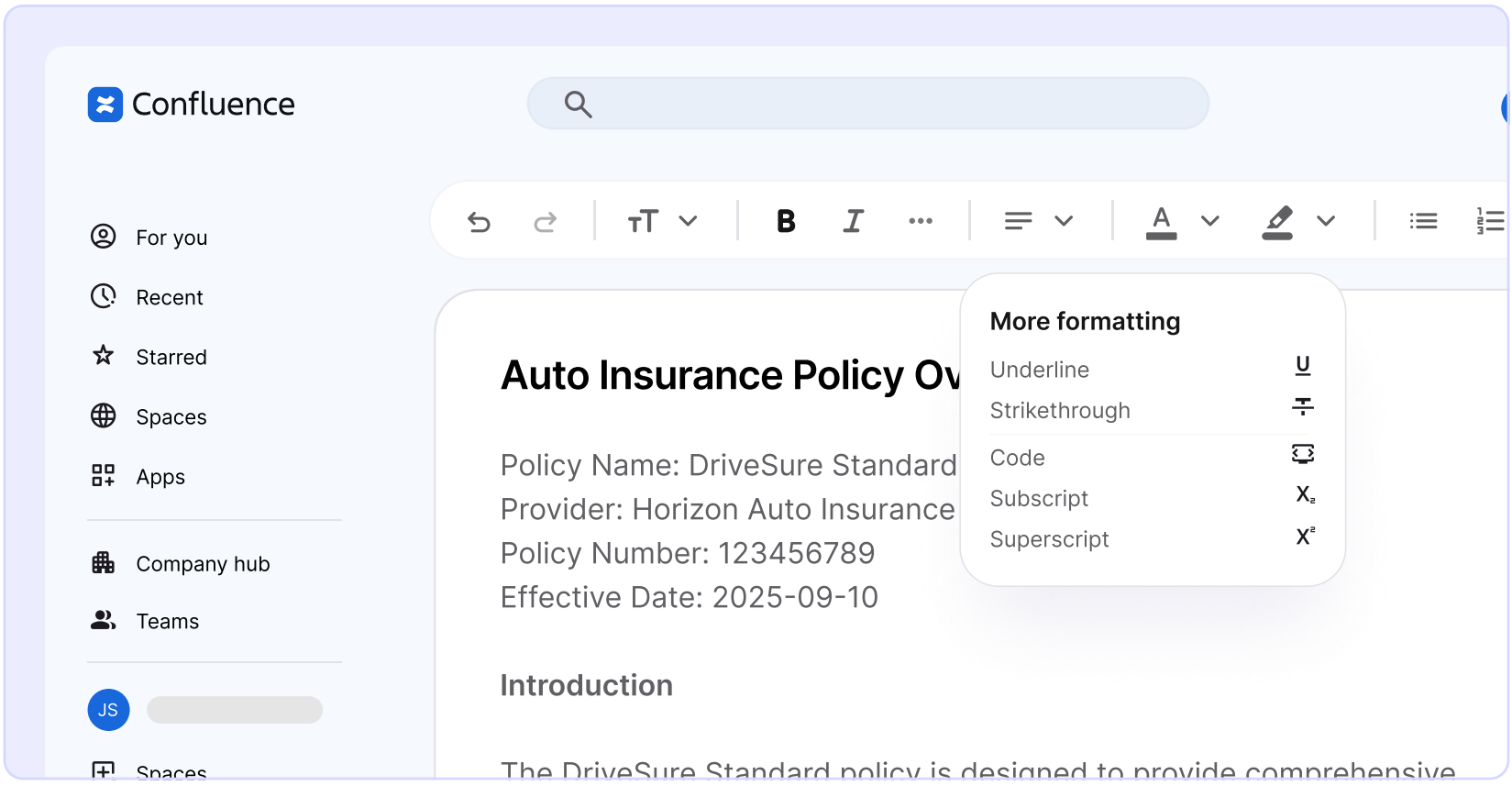
- A Knowledge Base Space that includes article templates, a pre-configured home page (with search and labels)
- A technical documentation space, including a home page with search.

Creating a Publication

Confluence works much like Microsoft Word or Google Docs. It provides a page-centric basic WYSIWYG interface. You can create blogs, live docs (e.g., meeting notes, brainstorming sessions, and other real-time live content), and pages.

Documentation is typically built using pages because pages are versioned and can be organized into a hierarchy and nested. All content lives in Spaces, which are collections of pages.

Content creation is straightforward. Open the application, create a new page, and start writing content. You can include text, images, headings at multiple levels, code, tables, and Figma files in a blank document or a template, and begin writing content. Confluence automatically saves your work.



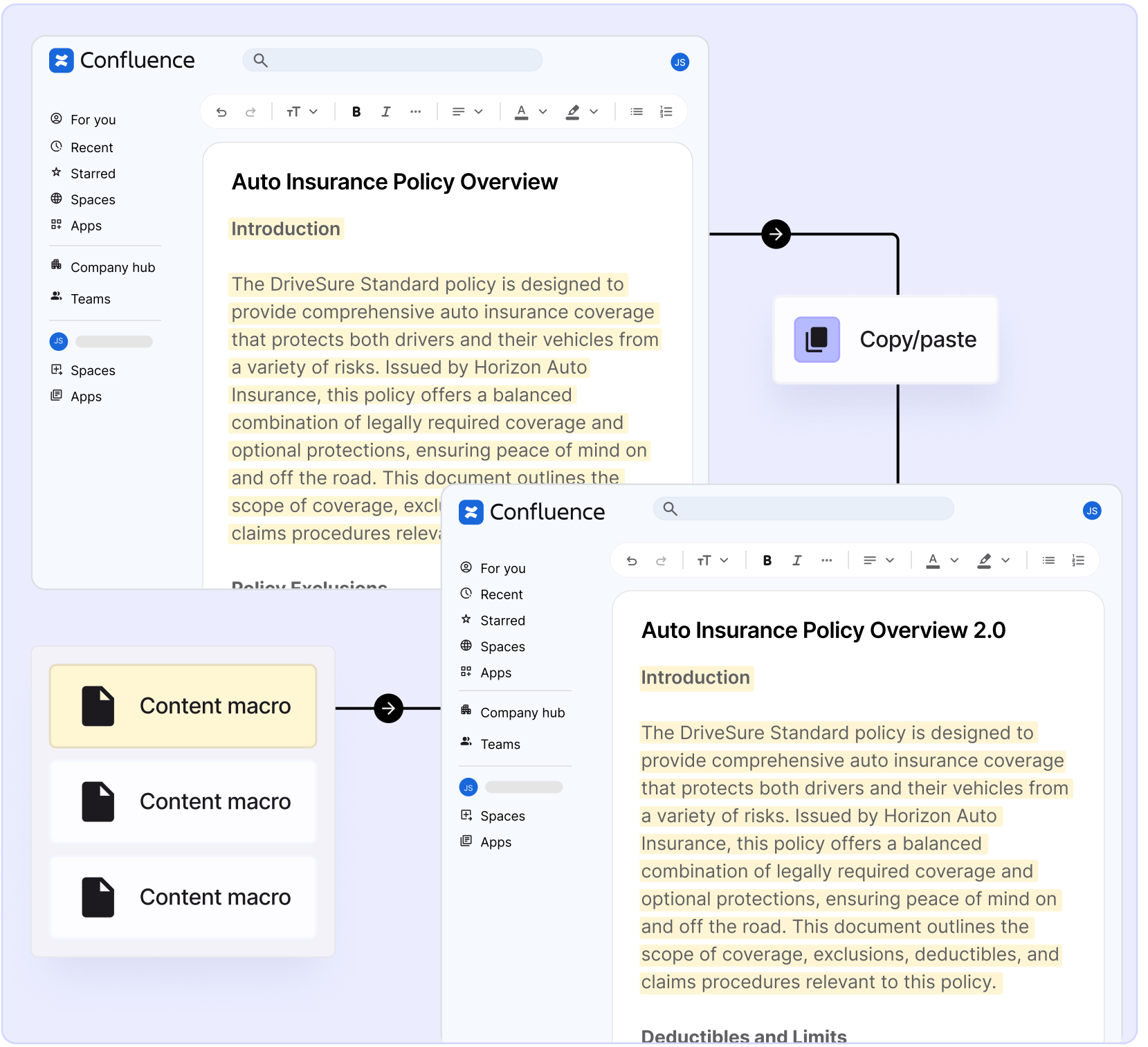
As you write content, you format it. You can also select to create a page from a template, providing a predefined structure and styling. You can also apply labels (keywords) to a page to help categorize it.

All pages exist as drafts until you publish them. Pages are published live or can be scheduled to go live at a specific date and time. Once a page is published, any changes made must be resaved as an update to the page.

Every time you publish or update a page, a new version is saved. For each version, you can write a version note to explain the changes. View who made changes to the page and revert to a previous version if necessary. You can also select two versions to see the changes between them.

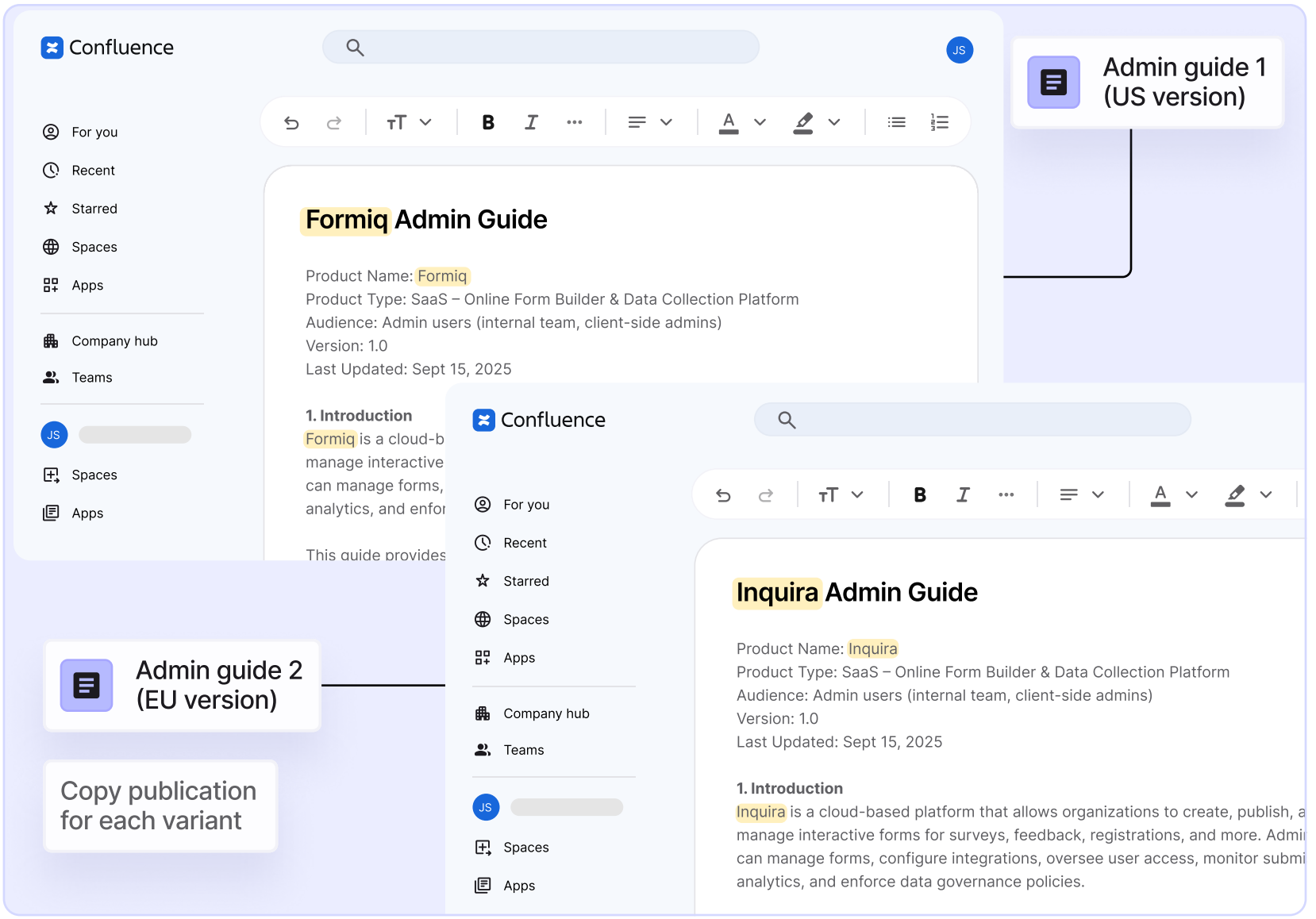
Managing Reuse and Product Variants

It is possible to reuse content in Confluence. Reuse (e.g, a word, sentence, paragraph, image, a product name/version number) is done using three macros: one to create the reusable content, one to insert it into a page, and one to insert an entire page into another page.



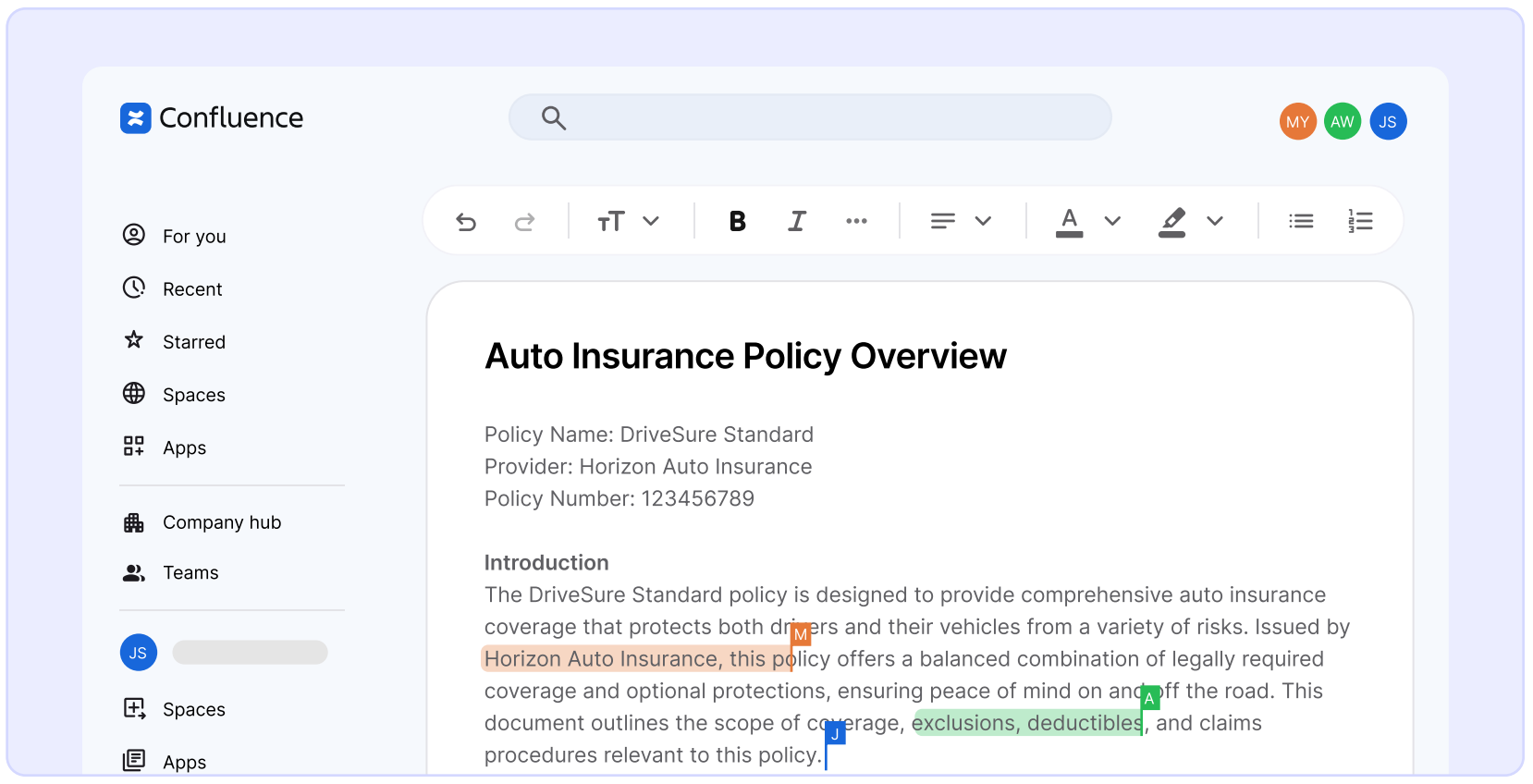
There are also two additional macros to create a table of contents and to create “warnings, tips, and info” type content.

Confluence does not support variants or filtering, or profiling. For product variants, users often create "additional spaces" in Confluence, leading to fragmented content.



Collaboration

In Confluence, up to 11 people can collaborate on a page simultaneously. Changes are saved and synced automatically, ensuring that all authors, editors, and reviewers view the same content. Invite people to work on a page by sending them the page link or sending a notification in Confluence.



Collaborators can make changes to the page directly, provide in-line comments, or make overall page comments. Comments can be assigned to a specific person by using the @ name syntax.

Confluence also enables you to invite external collaborators to work on content in a specific space. These external contributors have limited permissions.

Translating Documentation

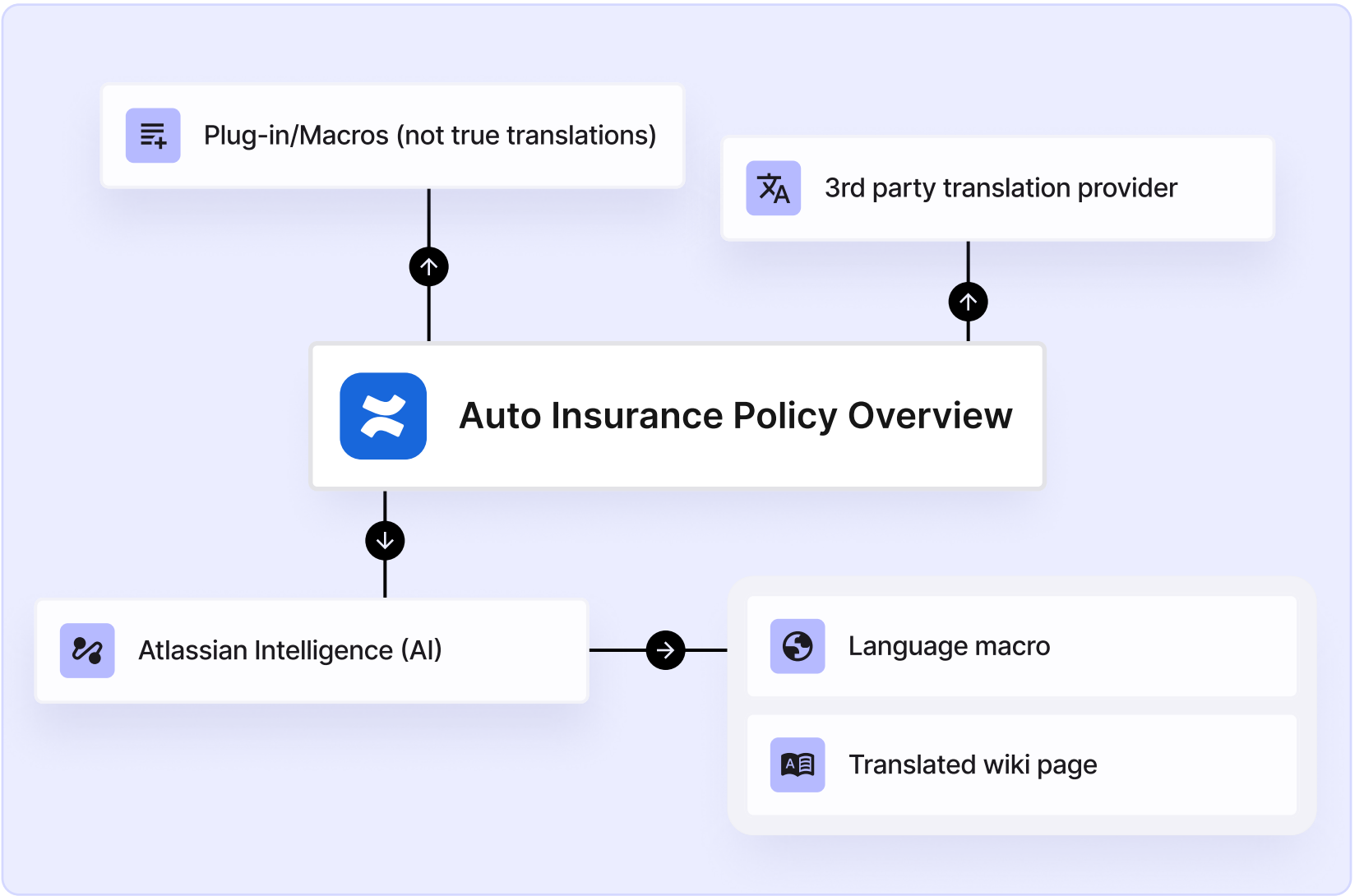
Confluence does not offer out-of-the-box support for automated language translations of page content. Several partners offer translation capabilities through add-ons, including:

- Bitvoodoo AG: Translations for Confluence allows you to display page content in different languages on the same page.
- Seibert/Media: Use Language Manager to create different versions of content in different languages (including ensuring links also follow the same language).
- EPS Software Engineering AG: The Easy Translator for Confluence integrates with Deepl and Google to provide translations for a page.

There are also some additional add-ons that provide browser-based translation capabilities, leveraging services such as Google Translate. Note: Some of these add-ons are only compatible with the Data Center version of Confluence, not the Server or Cloud version. Check the marketplace to find which add-ons are compatible with the Cloud version.

There are workarounds using language macros or plug-ins, such as the Visibility and Composition plug-ins; however, these are not specifically designed for translation solutions and can complicate the management of language versions.

If your site has Atlassian Intelligence enabled (AI offering), you can translate sections or an entire page, then paste the AI output into language macros or new pages. This approach also requires manual work to track and manage translations.



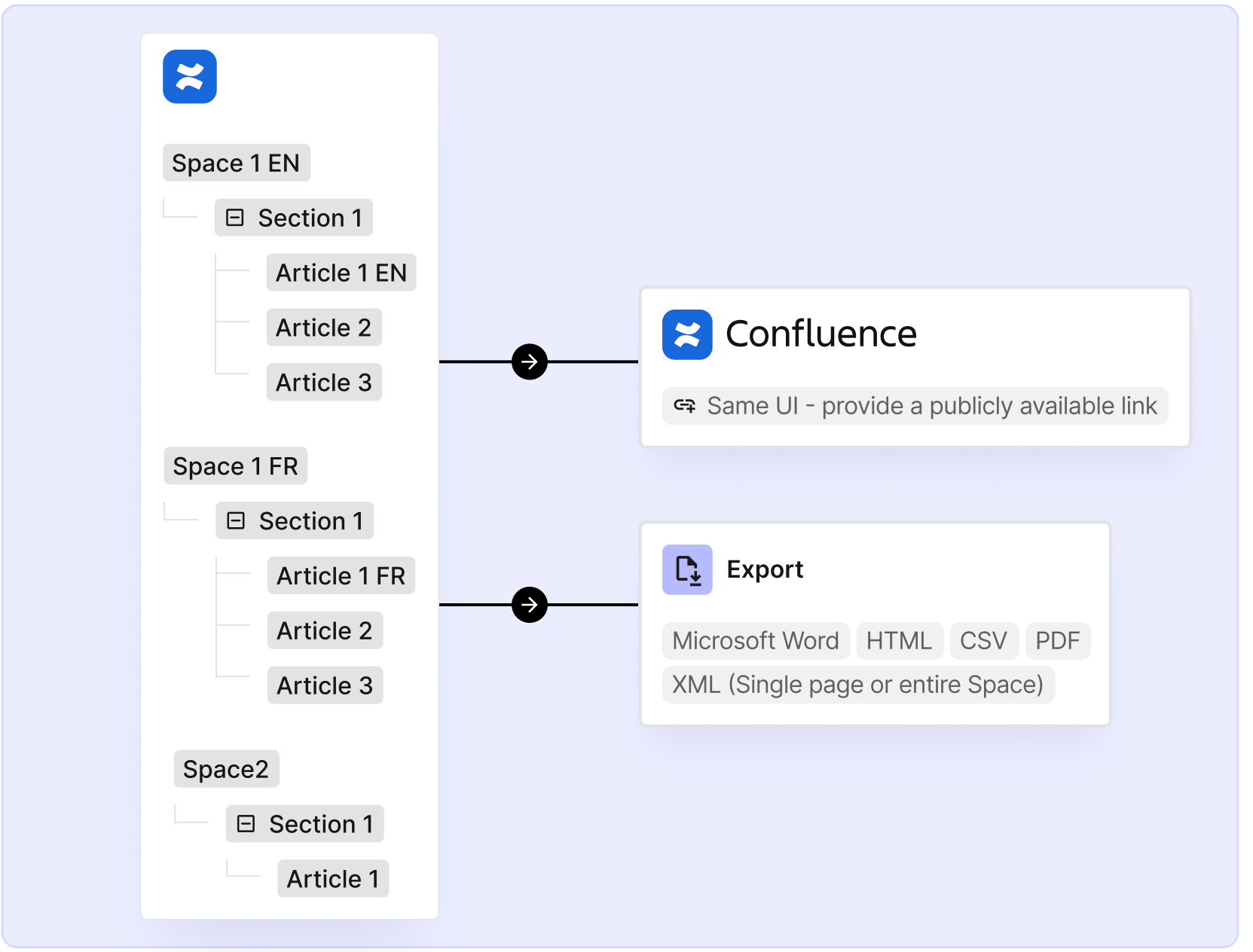
Some Confluence customers have implemented translations using a simpler approach. For each language, they create a separate space, essentially creating a translated version of the source. Links are an issue using this approach because someone will need to manually change all links in the content to point to the right translation.

When you use Spaces for each language, there is no connection between versions, so when the source is updated, a manual process is required to track and make the necessary updates to the translated copies. If you translate into many languages, this approach does not scale at all.

Another option is to use add-ons from K15T, an Atlassian partner. Third-party add-ons like K15T's "Scroll" apps extend Confluence's documentation capabilities (sections, variables, reuse). However, these are limited, and the more sophisticated you become, the more difficult it is to maintain, as it becomes too complicated. These add-ons also introduce non-standard HTML code, which complicates migration to structured tools.

Multi-Channel Publishing

Confluence is both a content creation and content delivery application. So, the primary publishing channel is Confluence itself. To make content available to customers, you generate a read-only public link. Public links provide customers with access to published pages without granting them access to Confluence. It's important to note that a public link is available to anyone on the internet but is not indexed by search engines. You must have a paid Confluence plan to create public links.



Confluence pages can also be exported in Microsoft Word, HTML, CSV, PDF, and XML formats. You can export a single page or a set of pages in a Space, or an entire Space. Note that exported Word documents are not compatible with Open Office, Libre Office, or Google Docs.

PDFs are formatted using a combination of HTML and CSS and PDF templates are unique to a Space.

Although a page may look good in Confluence, it is not guaranteed to look good in PDF, particularly when tables are used in the content. Note: Atlassian does not provide support for issues related to PDF customizations.

Important Considerations

Cloud vs Data Center

Confluence discontinued support for the server-based versions of its software in February 2024, so there is no longer an on-premises version available. Instead, Atlassian is encouraging existing customers to move to Atlassian Cloud or Atlassian Data Center.
These are two different offerings. Cloud is a SaaS subscription and Data Center is a self-hosted version of the platform. Depending on the offering you choose, double-check the features and functionality you will have access to.

Working with Add-Ons

Many of the functionalities you may need in Confluence are available through add-ons, such as translation provider integrations. While this isn’t necessarily a bad thing, it’s essential to recognize and work with trusted add-on partners.

Why use Confluence

If you are using Jira Service Management, it may make sense to build your knowledge base and technical documentation content within it, as it features a built-in Help Center, and Confluence powers the Jira Knowledge Base.

If you are working with a small documentation set or only a small number of knowledge base articles with little to no reuse, then Confluence will be suitable for you, depending on your specific requirements and budget.

Customer Story

From Confluence Bottlenecks to Rapid Release

KIX Service Software is an open-source service management solution. Its documentation team was relying on Confluence (with multiple add-ons) to produce all its user and administration guides, self-service portal documentation, and mobile app manuals. While Confluence made writing and basic HTML publishing straightforward, creating polished PDF manuals was arduous—tables and info-boxes were split across pag es, there was endless format cleanup, and it took two-plus days to publish a 2,000-page guide. Reviews required exporting PDFs, emailing them to SMEs, then manually merging comments back into Confluence.

KIX knew they needed a true cloud-based CCMS that could handle integrated translation workflows, structured reviews, and streamlined multi-channel publishing. After a proof-of-concept, they chose Paligo for its browser-based SaaS model (with no on-premise or Windows requirements), topic-based authoring with content reuse, and batch publishing that delivers HTML and PDF in under half a day instead of days.

Today, KIX leverages Paligo’s versioning, filters, and variables, in-source commenting, and review workflows, and plans to integrate a translation management system, dramatically reducing manual effort and accelerating its publishing and review cycles.

“It took two people two and a half days to publish a 2000-page PDF in Confluence. Even small changes would throw off the formatting. With Paligo, we were able to publish the entire set of documentation, PDF and HTML, in a half day or less. And it looked good!”

Heidi Meißner, Kix

paligo

Managing Technical Documentation with Paligo CCMS

Managing Technical Documentation with Paligo CCMS

True Structured Authoring and Management

We've walked through the key capabilities of five unstructured authoring tools. Now, we want to describe how Paligo CCMS differs from these tools to offer true structured authoring and single-sourcing capabilities.

Paligo is a cloud-native component content management system that gives you granular control over your content. Technical documentation teams use Paligo CCMS to create, manage, and reuse content through structured, topic-based authoring. It provides full version control, built-in review workflows, automated translation management, and multi-channel publishing.

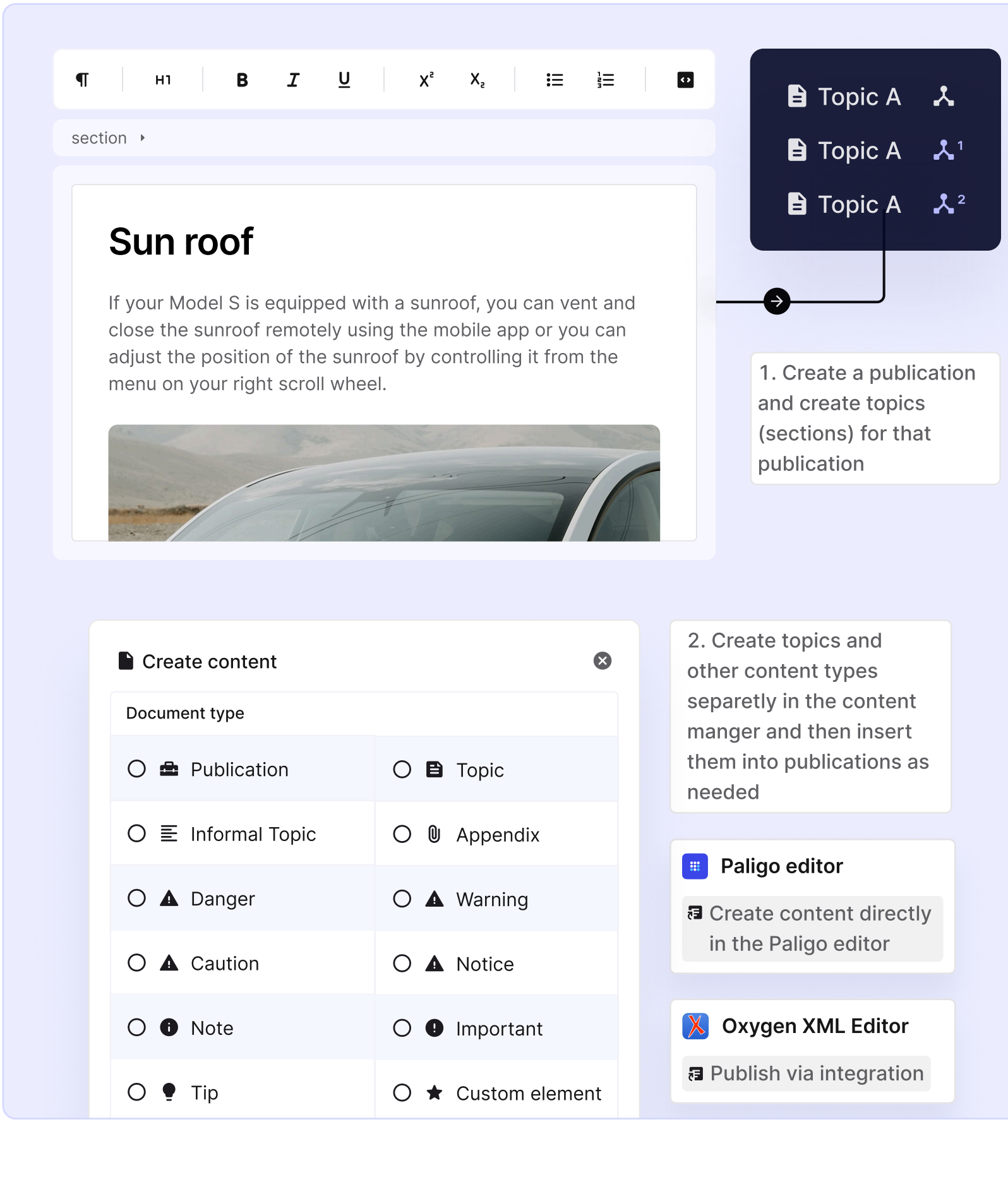
Let's examine these capabilities in more detail to understand how a structured authoring solution offers numerous improvements over an unstructured authoring tool.

Creating a Publication

Paligo has a structured authoring environment. Content is organized by publications and topics. Topics can be as small as a tip, note, or warning, or as large as a paragraph or section. The key with a topic is that it can stand alone and be reused across publications.

Each topic follows a structured XML-based layout, which includes at a minimum a title and a paragraph. Authors can add additional elements as needed using the Paligo Editor, which resembles a word processor, hiding XML tags to keep the interface uncluttered. The Editor also provides some basic formatting features.

Paligo tracks all changes to a component, maintaining a complete history. You can roll back to previous revisions, and compare different revisions of a component. Every time you save a component, Paligo stores a record called a revision. You can also create branches of your content at the publication or component level.

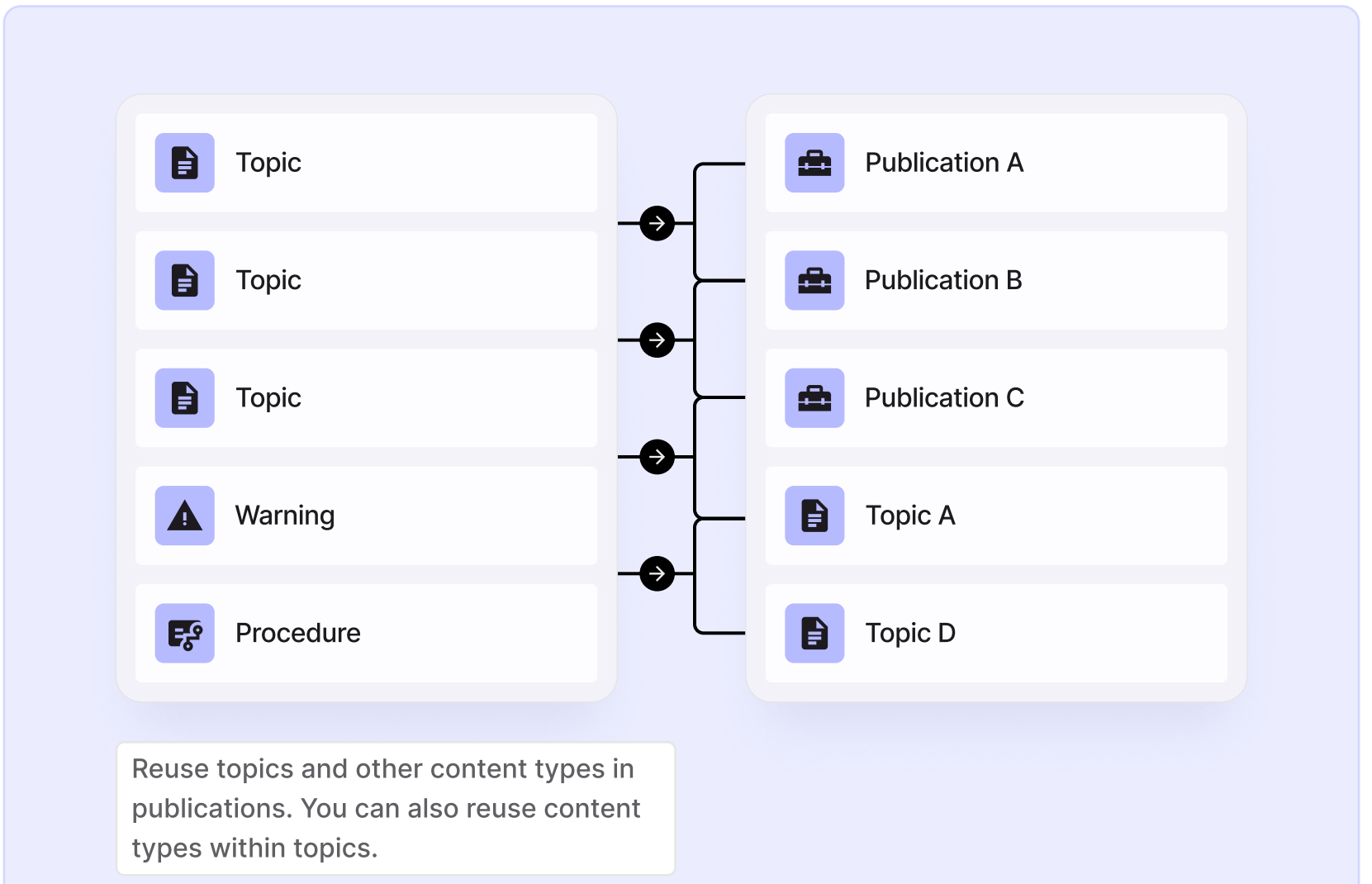


Branching allows you to create concurrent or parallel versions of content, often used to work on major updates to documentation that support a new product update before the product is released. Once the product is officially available, you can merge the branches back together and have the new content available.

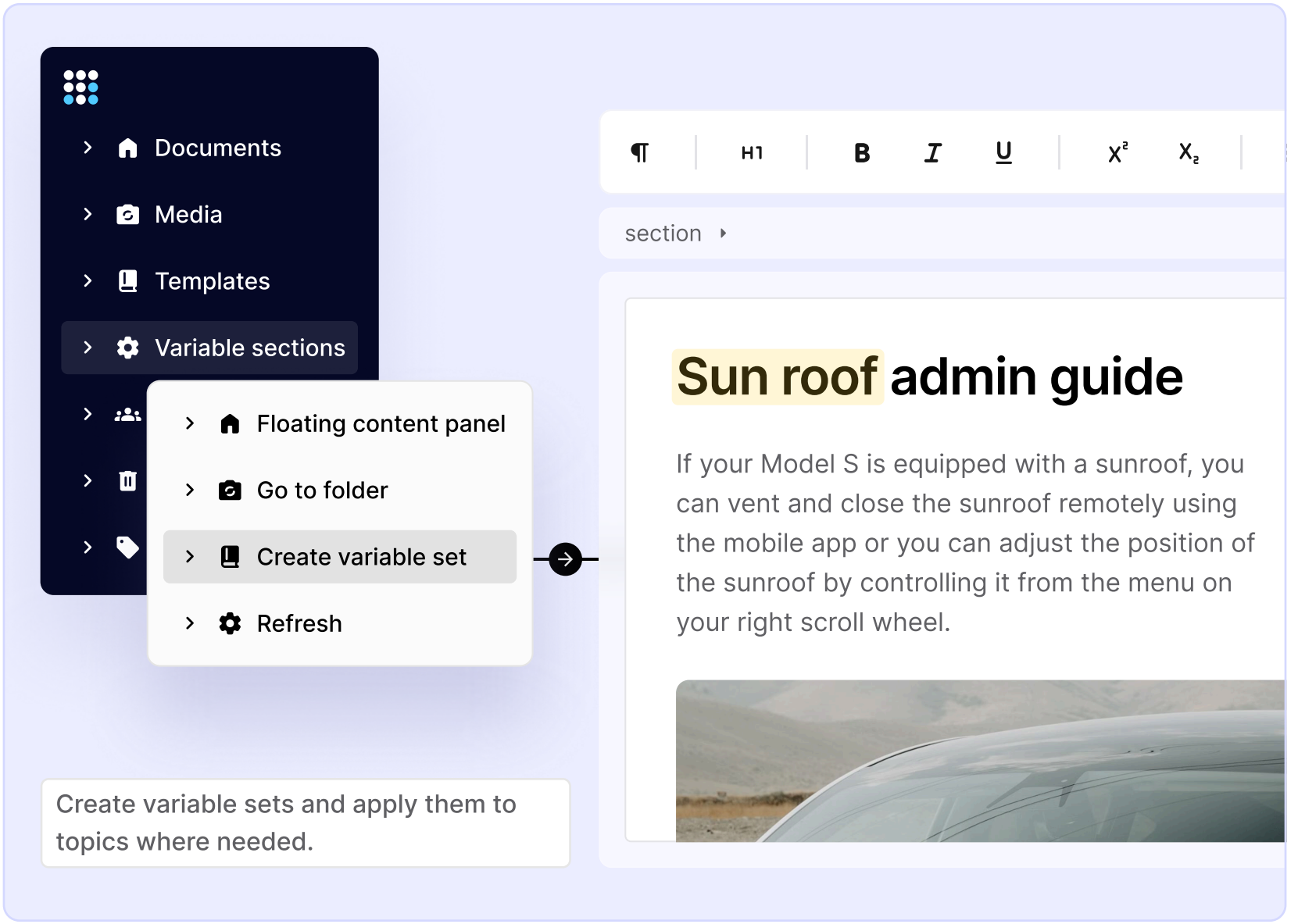
Managing Reuse and Product Variants

Paligo CCMS supports content reuse. Because content is structured, you can include topics inside other topics.

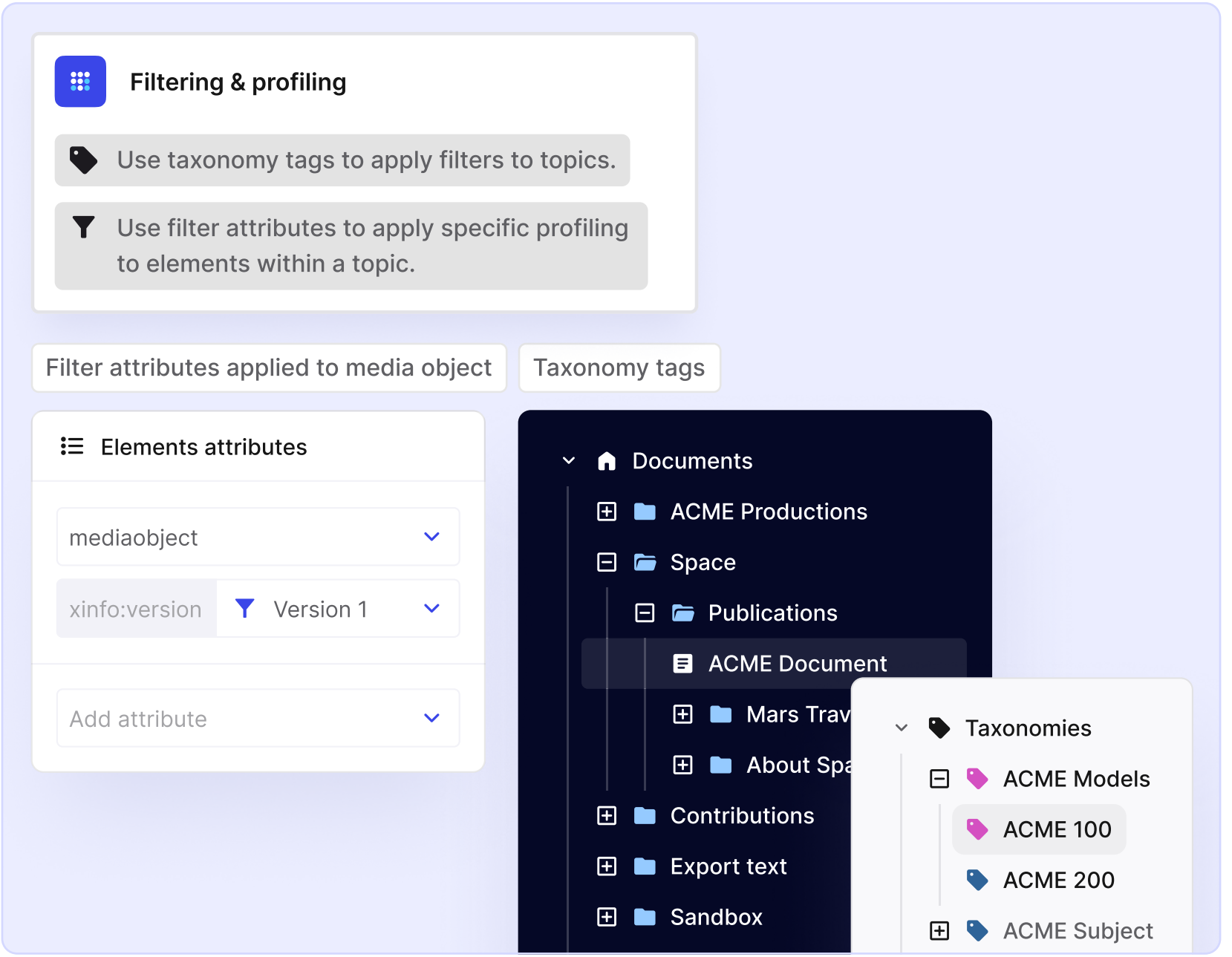
For example, if you have a series of Tips or Procedures created as reusable topics, you can insert one or more inside another topic. When that Tip or Procedure is updated, it’s automatically updated everywhere it’s reused.



You can also create product variants and conditional content in Paligo. Let’s say you have a product that you sell globally, but the product has a different name in several countries. The rest of your documentation remains the same; it’s only the name that differs. Instead of creating multiple copies of your documentation for each country that needs a different product name, you can use a variable. You create a variable set in Paligo that contains the product names you need, and you insert the variable everywhere the product name is mentioned. When you export your publication for publishing, you select which variable value to use.



In addition to variables, you can also perform filtering or profiling to enable a topic to be reusable in multiple contexts. For example, you are creating an administration guide that supports multiple subscription levels. However, some of the content is only applicable to a specific subscription. In this example, you apply a filter to the content that you only want to publish with the higher-level subscription documentation. Filtering is also known as conditional text and can be used to change data, images, or hide information for certain products, models, or markets.



Collaboration

Paligo CCMS includes built-in collaboration capabilities to support multiple authors, editors, and reviewers. First, you can create assignments, including Review, Contribution, Translation, and Translation Review. An email notification is sent to the assignee containing a link to the assignment. This link also appears in the Assignments panel of their Dashboard.

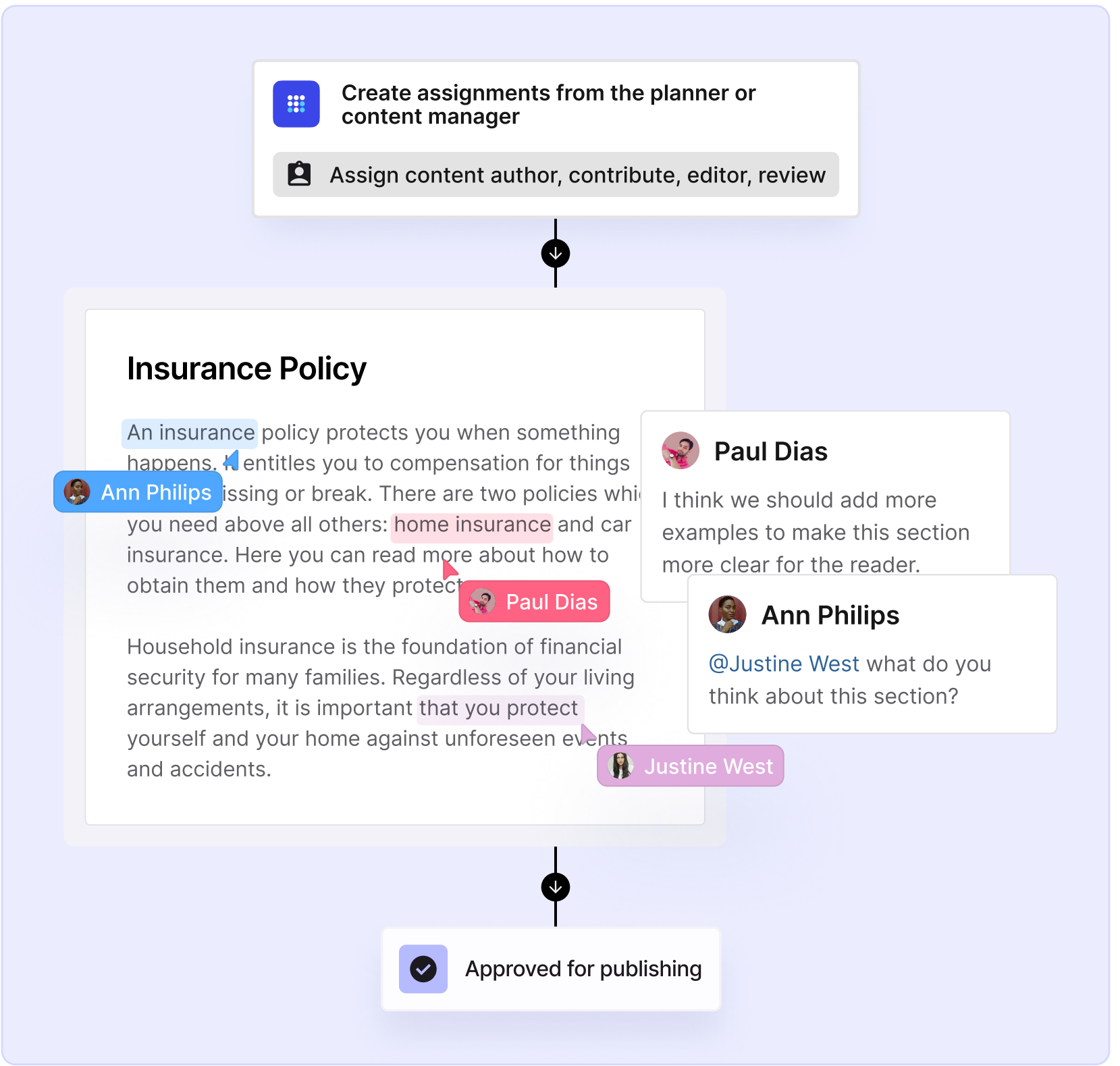
You track the status of the assignments using the Dashboard or the Planner. The Dashboard is the first thing you see when you log into Paligo. It shows the work you are doing, your assignments, and more. The Planner is a built-in project planning tool, similar to a Gantt chart for planning your technical documentation assignments. Use the Planner to create and monitor assignments.

When an administrator or an author assigns content for review, they can select one of two options: Edit and Suggest, or Suggest Only.

The reviewer receives an email notification, clicks the link, and opens the content within Paligo in a Contributor view (a streamlined editor designed for contributors), then performs their review.

When the Reviewer is done, they click the Finish Assignment button, selecting either Needs Work or Approved and ready for publishing. The Author gets a notification the review is complete and moves forward. If the reviewer has the Author user type, they can create a snapshot of the content to create an archived version for comparing changes later on.

There is also the ability to share content without using the Planner or Assignments feature.



The key to collaborating on content with Paligo CCMS is that everything happens within the same environment, on the source topics, enabling multiple contributors to work on the same publication. There is no need to create copies and share them, or try to synchronize edits and comments. All edits and comments are saved and tracked with the content giving you a complete picture of everything that has happened with the content.

Translating Documentation

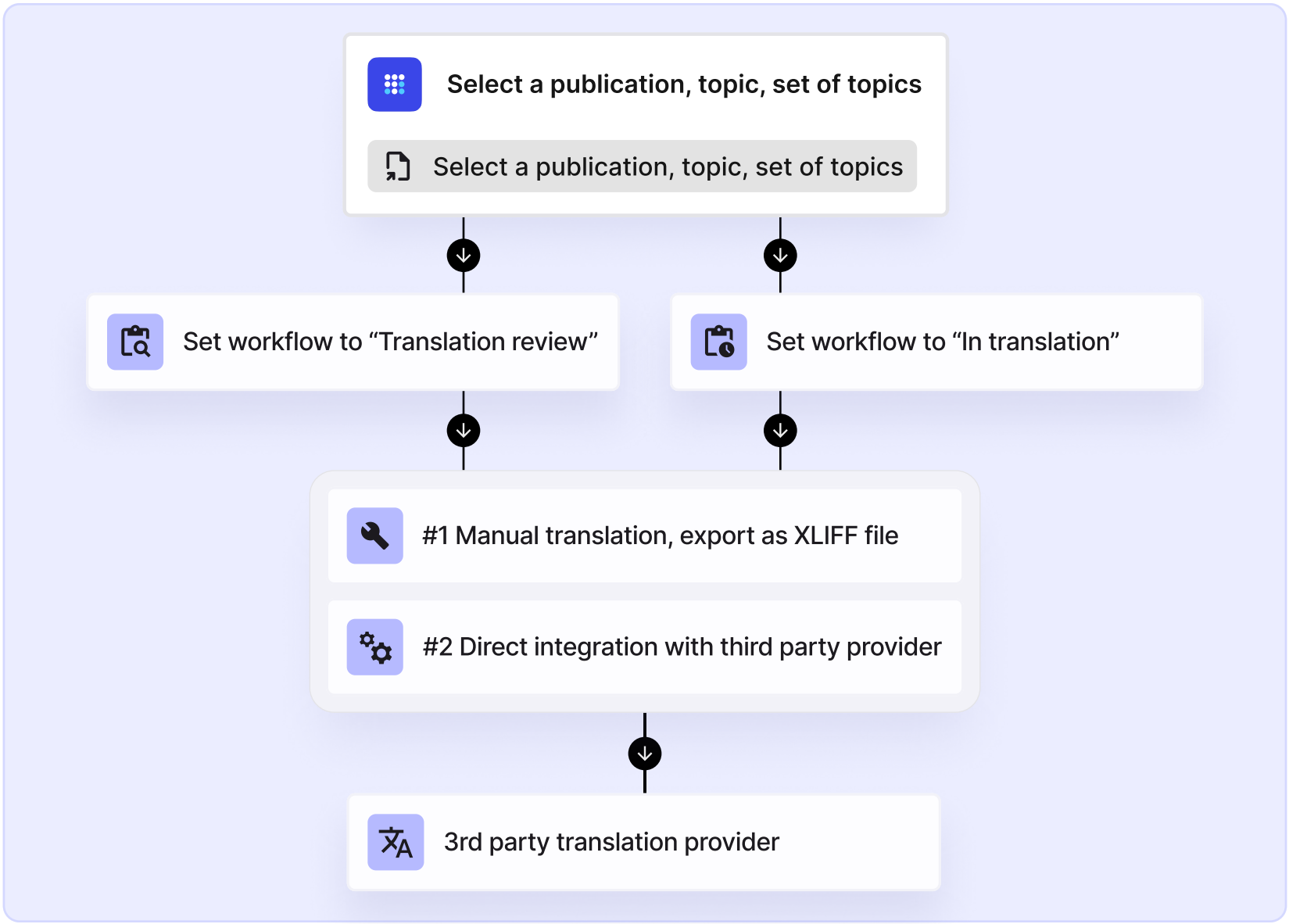
Paligo has built-in features for supporting the translation of your content. These include integrations with translation services, built-in translation workflow statuses, Translation View and assignments.

Managing translations within Paligo CCMS differs significantly from the unstructured tools discussed above. To manage translations, you have to enable translations in your Paligo instance and then add languages to your publications and topics. This means that the translation is connected to the topic, not to the publication as a whole.

To translate content, you select the topics to send for translation. If a topic is reused, you only need to translate it once. When you view a topic, you can select to view the translated versions.

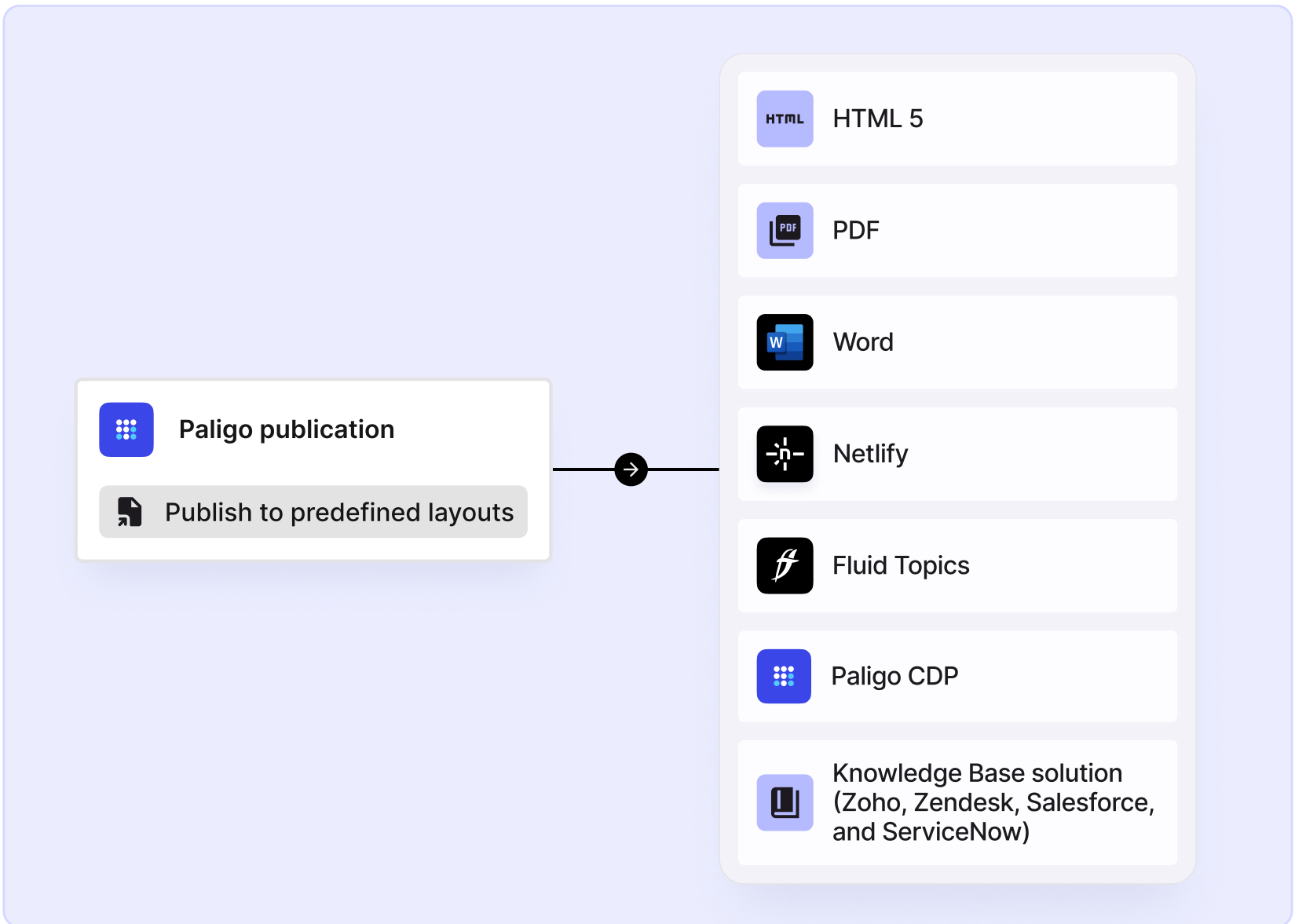
Translation can happen through a third-party integration or using Paligo's built-in translation editor. In addition to translating text-based content, you can also manage translations for images, variables, and filters. You can also exclude content from being translated. Image, variable, and filter translations are not included as part of a translation package for third-party translation, so you will need to export them and send them for translation separately. However, the translations are still connected to the source content the way that translated text is connected to its source text.

Paligo includes built-in integrations with several Translation Management Providers, including Phrase, CrowdIn, and LanguageWire. Additionally, you can export XLIFF packages and manually send them for translation, then reimport the translation package when it's complete. There is also a translation workflow that enables authors and reviewers to review the translations before they are accepted and marked for publication.



Multi-Channel Publishing

Paligo does not provide publishing channels by default. Instead, it provides the ability to publish content in multiple formats, including HTML, HTML5, PDF, Word, online help, and SCORM (for Learning Management Solutions). It also integrates with several content delivery solutions, including Fluid Topics, Zendesk, Salesforce Knowledge, ServiceNow, Netlify, GitHub, and others.



You define layouts in Paligo to support each export format you want. When you select to publish a publication, you configure the Layout settings and choose the publishing settings. Paligo takes care of the actual "transformation" into HTML5, PDF, and other formats. When you are publishing via a direct integration, Paligo also takes care of transforming the content for the destination channel based on how the integration is set up.

paligo

Confluence vs Paligo CCMS At a Glance

Confluence vs Paligo CCMS
At a Glance

Feature	Paligo	Confluence
Creating a Publication	Structured topic-based	Flat wiki pages
Managing Reuse & Product Variants	Topics, variables & filters	Snippet macros, no profiling
Collaboration	Built-in assignments & reviews	Live edits & comments
Translation Management	Integrated translation workflows + XLIFF export/import	3rd-party add-ons required
Multi-Channel Publishing	Multiformat & integrations	Export to Word/HTML/PDF
Best Use Case	Multiple Product & technical documentation, content reuse, high translation requirements	Internal documentation or teams using the Confluence Help Center knowledge base.

paligo

Choosing the Right Model for Your Organization

Choosing the Right Model for Your Organization

The goal of this ebook is to help you understand how various unstructured authoring tools help you create and publish technical documentation compared to Paligo CCMS and structured authoring. While there is no one-size-fits-all answer to “Should you use a CCMS?” There are indicators that suggest which approach will work best in certain citations.

When Unstructured Works

If you have very simple, straightforward documentation requirements, using a tool like Microsoft Word or Google Docs, or one of the other tools mentioned here, may be sufficient. For example, if one or more of these is true, then you may not need a structured authoring model or a CCMS:

- Have one or two authors/editors.
- Manage fewer than three documents with little to no reuse across them.
- Support one or two languages.
- Have a small team with few reviewers (or no reviewers)
- Have no requirements for variants, filters, or profiling.
- Don’t need to maintain version history.
- You are only publishing to a single channel (like PDF).

However, even if these are true, a CCMS can still provide many benefits, especially around maintaining a single, easy-to-use environment for all your documentation.

When Structured is Essential

There are situations where structured content is critical to ensure your ability to create, manage, and publish technical documentation quickly:

- You have a large documentation set supporting multiple product lines that is regularly updated.
- There is significant reuse across documentation.
- Some products have different names in some countries
- Documentation is published in multiple languages, including a lot of translated images.
- You need to track changes and maintain version history for all content by all authors, editors, and contributors/reviewers.
- You publish content across multiple channels in multiple formats and need to ensure consistency and accuracy across channels.

paligo

Checklist: Is Your Organization Ready for Structured Content?

Checklist: Is Your Organization Ready for Structured Content?

If you’re unsure whether you need to adopt a structured content model, and specifically a CCMS, use this brief checklist as an early indicator.

Analyze Current Content Landscape

- [ ] Conduct a preliminary content audit (content types, channels, formats, versions, languages).
- [ ] Map out your content team, including all authors, editors, reviewers, and collaborators, noting their locations and work schedules.
- [ ] Document any current documentation challenges (inconsistency, slow publishing, high translation costs).
- [ ] Document any bottlenecks or hand-offs that consistently slow delivery.
- [ ] Gather team feedback (including pain points and frustrations) from reviewers on the current content development process.
- [ ] Gather customer feedback on your existing content and delivery channels.

Identify Content Reuse Opportunities

- [ ] What output formats do you require today (PDF, HTML, mobile, embedded help, etc.)?
- [ ] Do you need product- or customer-specific variants (e.g., region, language, customer tier)?
- [ ] Do multiple products, versions, or channels share identical, or near-identical, text, graphics or procedures?

- [ ] Are you manually adjusting layouts or continually saving Word documents as PDFs or HTML and dealing with formatting issues?
- [ ] Could content reuse in structured topics reduce translation volume (and cost)?
- [ ] Do you have any future or near-future content needs (new markets, languages, products, channels) that you need to consider?

CCMS Considerations

- [ ] Identify measurable goals (reduce costs, speed up review, improve scalability).
- [ ] Preliminary evaluation to determine CCMS need.
- [ ] If needed, map requirements and research options (features, integrations, costs, support).

paligo

The Path to Smarter, Scalable Content Management

The Path to Smarter, Scalable Content Management

The decision to move away from unstructured authoring tools to structured authoring and component content management makes sense for many companies struggling to manage effective technical and product documentation. The growth of customer self-service, including the demand for clear and accurate documentation, as well as the speed at which it needs to reach customers, employees, and teams, means traditional unstructured authoring approaches to managing it are no longer effective.

Structured authoring introduces a systematic framework emphasizing modularization, consistency, and reusability. It involves using predefined structures and formats to ensure that information is presented in a consistent and standardized manner, making it highly adaptable and scalable to different channels and formats. It also fosters a more agile and collaborative environment for writers, editors, and subject matter experts. The same can’t be said for unstructured authoring.

If you are struggling with your current authoring tool, take the time to think through your challenges and consider how a CCMS, such as Paligo CCMS, can help alleviate them. Ask a lot of questions, request demos, and most importantly, make the decision that is right for your team and situation.