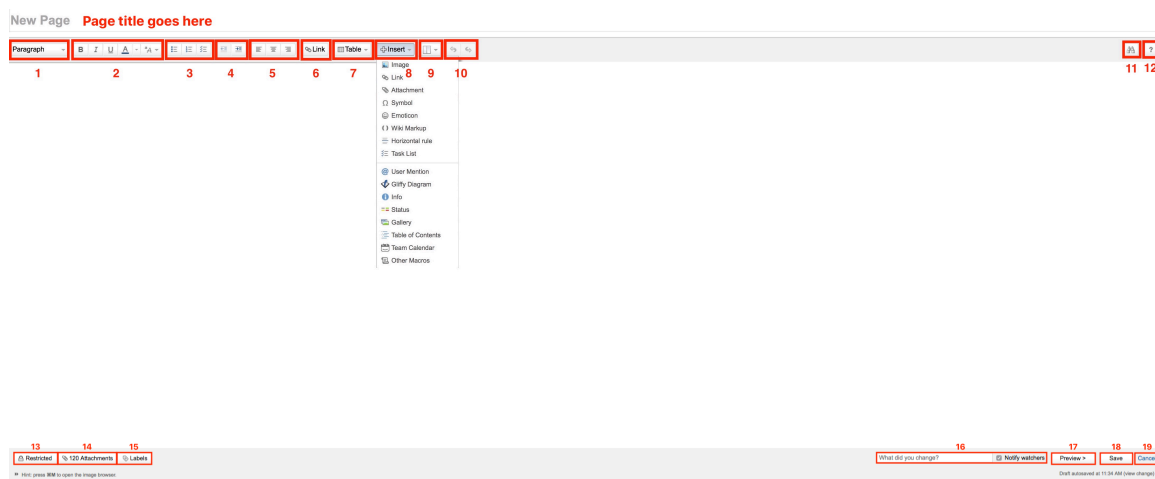


	Arrows	
	Confluence help	https://confluence.atlassian.com/display/CONF43/Confluence+Documentation+Home
HOW	Filing BPNs	BPNs must be submitted before 8 AM PST on Fridays. How to request: Basemap Training and Documentation Processes

2. Introduction to Confluence and the Wiki

For the introduction to the wiki for editors, see [How to Use the Wiki](#).

Confluence is the web-based software we use to host and edit our wiki; almost all of our work is within Confluence. Below are the main editing tools in Confluence.



#	Action
1	Change text style. We only use the header styles and paragraph style.
2	Font tools
3	Add a bulleted, numbered or task list
4	Outdent and indent text
5	Text alignment tools
6	Add link to page or anchor
7	Insert a new table
8	Insert element. You'll use the following most often: Image, link, attachment, symbol, Gliffy diagram, table of contents, and other macros.

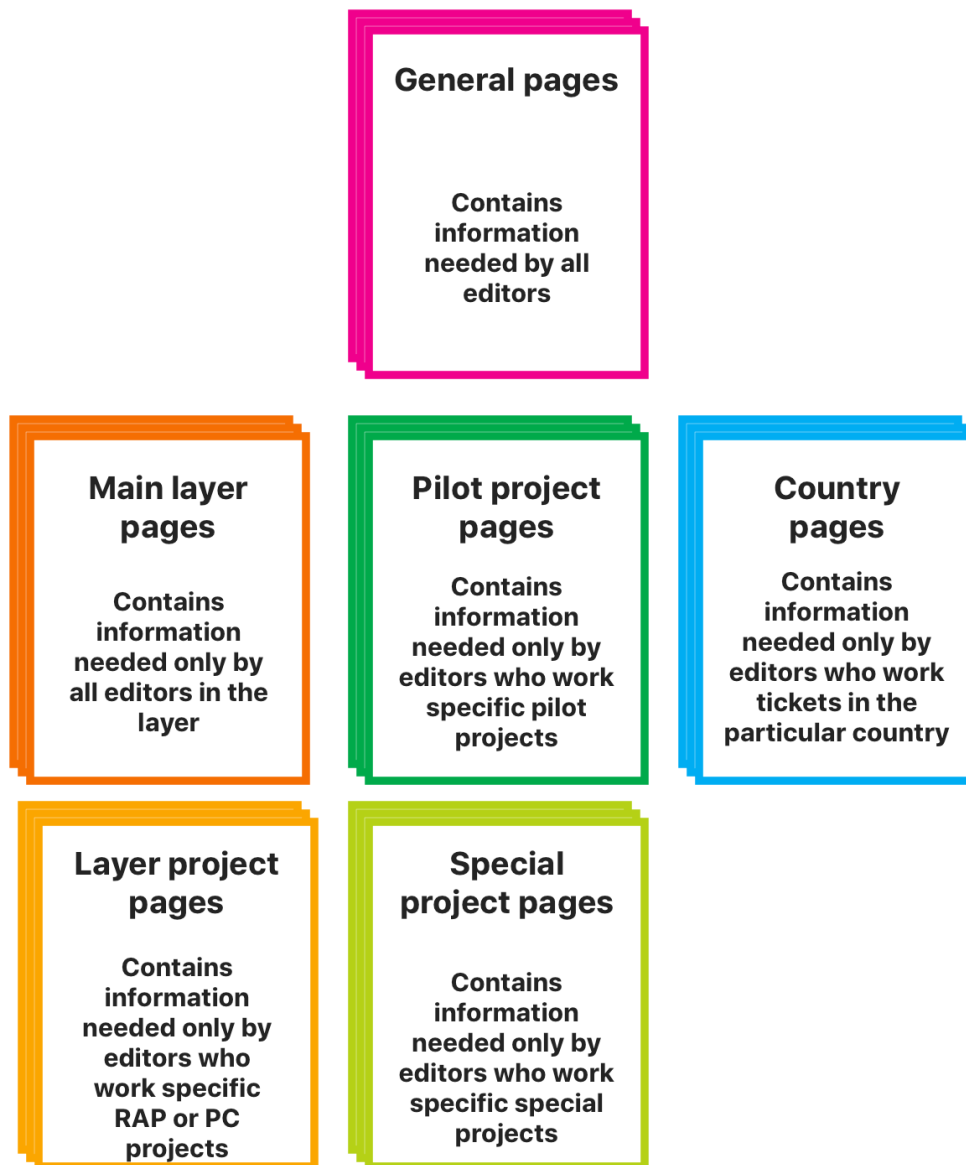
9	Choose page layout. This determines how many columns and sidebars there are.
10	Undo and redo
11	Find and replace
12	Help with keyboard shortcuts
13	Change viewing and editing restrictions
14	Add an attachment or view existing attachments
15	Add labels
16	Track changes and notify anyone watching the page <ul style="list-style-type: none"> • In the changes field, you should either provide a short description of the change or paste the ID & title link for the Radar requesting the change. • Uncheck the "Notify Watchers" box when making small changes to the page.
17	Preview changes without saving
18	Save changes. These changes will be visible to all readers.
19	Cancel without saving any changes

The wiki is organized primarily by audience. There are general pages, which all editors are required to understand and use. These include:

- Tool pages, including [How to Use Fusion](#), [How to Use Basemap Tickets](#), and [How to Use the Wiki](#)
- [Basemap Editing Workflow](#)
- [Basemap Sourcing Terms and Tools](#)
- [Support for Editors](#)

In addition to general pages, there are country-specific pages, layer-specific pages, and project-specific pages. If projects are completed by the same team, then the projects are grouped in the same page. If projects are completed by different teams, then the projects get their own pages.

Pages are nested from general to more specific. Specific pages do not include general information, but instead link to it in the more general pages.



Our team has made the mistake of making too many pages (organized by layer, task, or policy) and too few (organized by layer, instead of audience). These systems failed for a few reasons:

Many Pages, Organized by Task/Policy	Few Pages, Organized by Layer
Editor efficiency was low. Editors had to jump from page to page to complete a single ticket.	Editor efficiency was still fairly low, since they struggled to navigate the single page for their layer. Page tables of contents were so long, it was hard to find what you were looking for.
Editors struggled to locate information because Confluence search is poor.	Because of the length of and amount of content in the pages, +F was less effective.
Landing pages were overwhelming and difficult to navigate, since they had so many pages linked on them.	Editors struggled to determine what information was relevant to them and what information was relevant to other teams in their layer.

Using nested pages (general information to more specific information) and organizing by audience improves editor efficiency without leading to human error:

1. While it would be most efficient for an editor to have ALL the information they need (general and specific) in a single page, repeating general information would lead to more errors. Every time a general policy needs to be edited, our team would need to remember all the places that policy is stated and updated each instance. There's a good chance we'd miss some instances and outdated information would persist on the wiki.
2. By reducing the number of pages an editor needs to reference, however, editor efficiency is greatly improved. If, for example, we separated pages by project instead of audience, then an editor on a team that works on multiple projects would need to navigate from project page to project page during a workday. Keeping that team's multiple projects in a single page prevents the editor from having to do this.
3. By linking general information frequently in the more specific pages, as well as listing general pages in a "Related Pages" section, we decrease the amount of time an editor needs to search for general information outside of their audience-specific page.
4. Editors can find pages with relative ease once they know what team they're on. They do not need to know the different edit types, policies, or Basemap structure to find the pages they need. If you are a member of the PC AOI team, then you look for the PC AOI page; if you're working in a pilot project team, you find the pilot project page.
5. Permissions are streamlined. Permissions are by necessity organized by audience, so they can align fairly well with our page audiences.

3. The Basemap Team and Our Audience

The Basemap Editing team creates and edits data on the Apple Maps Basemap. Different teams work in different Basemap "sandboxes," which are either published to Maps all at once or on a cadence. There are many teams in the Basemap Editing team and editors switch teams pretty frequently. We want Basemap Editing documentation to be consistent in organization, tone, and style, so that these editors do not have to learn how to navigate documentation anew when they switch teams.

The Basemap Editing team is divided into two main categories: **Operations (Ops)** and **Planning (sometimes called Proto)**. We are members of the Ops team.

[Click here for a full organization chart.](#)

Team	Planning	Ops
Manager	Kristen Cayce	Nicole Schowalter
Description	The Planning team develops new projects and edit types. They run their own pilot projects and develop the Prototype Work Package (PWP), which is handed off to Ops.	The Ops team runs pilots projects before they go to scale and includes production.
Relationship	We do not create documentation for the Planning team, but we do use the PWPs they develop to create pilot project documentation. Part of the Planning team includes our trusted partners, who create documentation (mostly country pages) that we edit and publish.	We create almost all the documentation for Ops, who are the main audience for the wiki.