

Pulsar Pull Request Naming Convention Guide



Summary

This guide explains why you need good PR titles and how you do that with various self-explanatory examples.

ToC

Why do PR titles matter?	2
How to write good PR titles?	4
Quick examples	5
type	7
scope	9
Pulsar	10
Client	16
Summary	18
Full examples	19
Submit breaking changes	19
Submit PIP changes	19
Cherry pick changes	20
Revert changes	20
Add features	20
Fix bugs	20

Submit improvements	20
Update tests	20
Update docs	21
Update website	21
Update instructions/ guidelines	21

Why do PR titles matter?

Engineers and writers submit or review PRs almost every day.

A PR title is a summary of your changes.

- Vague, boring, and unclear PR titles decrease team efficiency and productivity.
- PR titles should be engaging, easy to understand and readable.

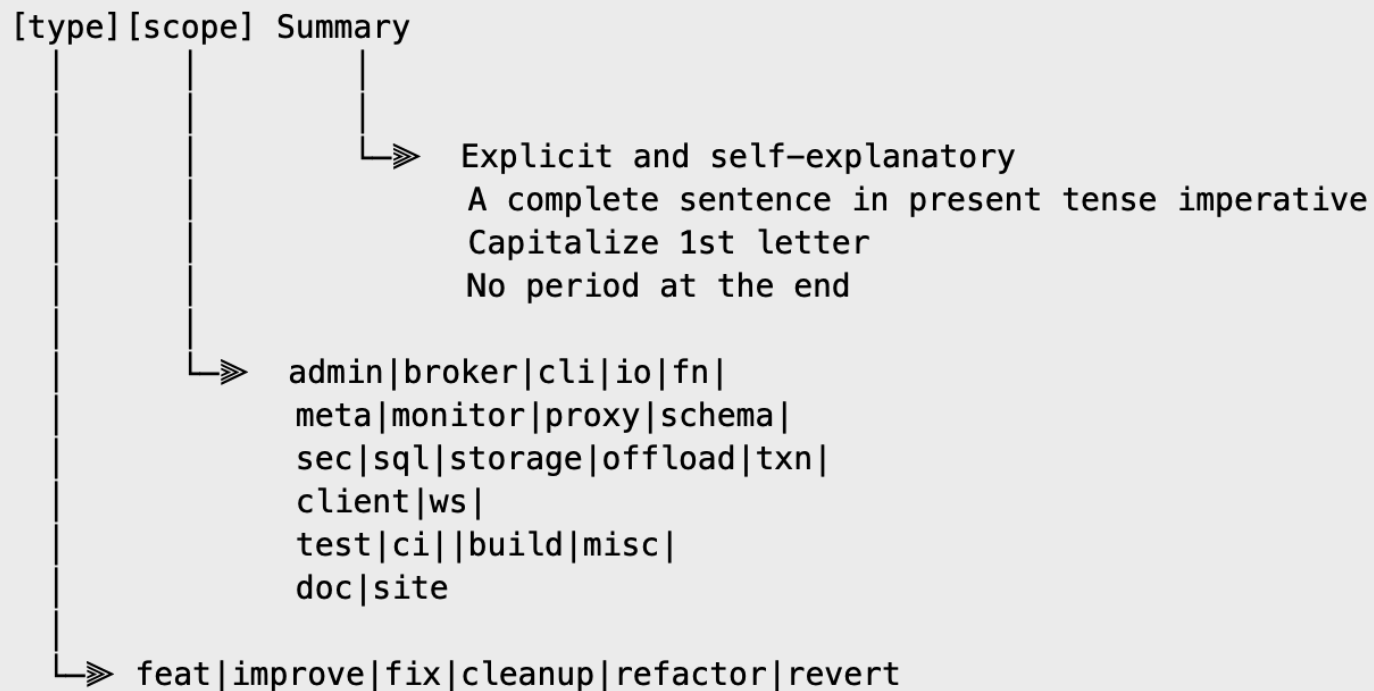
Good titles often bring many benefits, including but not limited to the following:

- Speed up the review process.
You can tell from the title what changes the PR introduces.
- Facilitate understanding of PR changes.

- PR titles are shown on Pulsar release notes as items.
Concise PR titles make your changes easier to understand.
- Especially when you read commit logs in command-line tools, clear commit messages show PR changes quickly.
- Increase search efficiency.
You can skim through hundreds of commits and locate desired information quickly.
- Remind you to think about your PR.
If you can not write a PR title in a simple way (eg. [\[type\]](#) [\[scope\]](#) [summary](#)), or you need to use several types / scopes, consider whether your PR contains too many changes across various scopes. If so, consider splitting this big PR into several small PRs. In this way, you might get your PRs reviewed faster.

How to write good PR titles?

A PR title should be structured as follows:



Rule

A good title = clear format (**type** and **scope**) + self-explanatory **summary**

Quick examples

Here are some examples of unclear and good PR titles for your quick reference. Good PR titles are concise and self-explanatory since they tell you the changes in a clear and direct way.

For more examples with correct formats, see [Example](#).

PR title examples	
Vague ❌	Clear ✅
Producer getting producer busy is removing existing producer from list	[fix][broker] Active producers with the same name are no longer removed from the topic map
Forbid to read other topic's data in managedLedger layer	[improve][broker] Consumers are not allowed to read data on topics to which they are not subscribed

PR title examples	
Vague ❌	Clear ✅
Fix kinesis sink backoff class not found	[improve][connector] xx connectors can now use the Kinesis Backoff class
K8s Function Name Length Check Allows Invalid StatefulSet	[improve][function] Function name length cannot exceed 52 characters when using Kubernetes runtime



Steps

How to write a good PR title?

- 1] Select a [type](#).
- 2] Select a [scope](#).
- 3] Write a [summary](#).

type

type is “what actions do you take”.

It must be one of the following.

type	Pulsar PR label	What actions do you take?
cleanup	type/cleanup	Remove unused code or doc.
improve	type/improvement	Submit enhancements that are neither new features nor bug fixes.
feat	type/feature	Submit new features.
fix	type/fix	Submit bug fixes.
refactor	type/refactor	Restructure existing code while preserving its external behavior.
revert	To be created	Revert changes



Note

Choose the correct labels for your PR so that your PR will automatically go to the correct chapter in release notes. If you do not specify a type label, the PR might go to the wrong place or not be included in the release notes at all. For more information about release note automation for Pulsar and clients, see [PIP 112: Generate Release Notes Automatically](#).

scope

scope is “where do you make changes”.

Pulsar and clients have separate release notes, so they have different scopes.



Note

If your PR affects several scopes, do not choose several scope labels at the same time since different scopes go to different chapters in release notes. Instead, choose the most affected label (scope), or else your PR goes to several chapters in release notes, which causes redundancies. Choose only one label as much as possible.

Pulsar

scope and PR labels must be one of the following.

scope	Pulsar PR label	Where do you make changes?
admin	<ul style="list-style-type: none">• scope/admin• scope/topic-policy	<ul style="list-style-type: none">• pulsar-admin• REST API• Java admin API
broker	<ul style="list-style-type: none">• scope/broker	<p>It's difficult to maintain an exhaustive list since many changes belong to brokers.</p> <p>Here just lists some frequently updated areas, it includes but not limited to:</p> <ul style="list-style-type: none">• key_shared• replication• metadata• compaction

scope	Pulsar PR label	Where do you make changes?
cli	<ul style="list-style-type: none">● scope/tool	Pulsar CLI tools . It includes: <ul style="list-style-type: none">● pulsar● pulsar-client● pulsar-daemon● pulsar-perf● bookkeeper● broker-tool
io (connector)	<ul style="list-style-type: none">● scope/connector● scope/connect● scope/kafka	Connector
fn (function)	<ul style="list-style-type: none">● scope/function	Function
dependency	<ul style="list-style-type: none">● scope/dependency	Dependencies that used to run Pulsar.

scope	Pulsar PR label	Where do you make changes?
		It includes but not limited to: <ul style="list-style-type: none">• Netty• Log4j
meta (metadata)	<ul style="list-style-type: none">• scope/zookeeper	Metadata
monitor	<ul style="list-style-type: none">• scope/metrics• scope/stats	Monitoring
proxy	<ul style="list-style-type: none">• <u>scope/proxy</u>	Proxy
schema	<ul style="list-style-type: none">• scope/schema• scope/schemaregistry	Schema
sec (security)	<ul style="list-style-type: none">• scope/security• scope/authentication• scope/authorization	Security

scope	Pulsar PR label	Where do you make changes?
sql	<ul style="list-style-type: none">scope/sql	Pulsar SQL
storage	<ul style="list-style-type: none">scope/bookkeeper storage	Managed ledger
offload (tiered storage)	<ul style="list-style-type: none">scope/tieredstorage	Tiered storage
txn	<ul style="list-style-type: none">scope/transactionscope/transaction-coordinator	Transaction
test	<ul style="list-style-type: none">scope/test	Code tests.
ci	<ul style="list-style-type: none">scope/ci	CI workflow changes or debugging
build	<ul style="list-style-type: none">scope/build	<ul style="list-style-type: none">Dependency (Maven)DockerBuild or release script
misc	<ul style="list-style-type: none">scope/misc	Changes that do not belong to any

scope	Pulsar PR label	Where do you make changes?
		scopes above.
doc	<ul style="list-style-type: none">• doc	Documentation
blog	<ul style="list-style-type: none">• blog	Blogs
site (website)	<ul style="list-style-type: none">• website	Website

Below are **deprecated** scopes.

Scope	PR label	Where do you make changes ?	Why deprecate it?
deploy	<ul style="list-style-type: none">● scope/deploy● scope/helm● scope/k8s	Deployment	Changes to deployment can be represented by other [scope]s.
deps (dependency)	<ul style="list-style-type: none">● scope/dependency	Dependencies	Changes to dependency (Maven) belong to [build]
package	<ul style="list-style-type: none">● scope/package	Package management	It belongs to [admin].
workflow	<ul style="list-style-type: none">● workflow	Process / standard / guide Examples: <ul style="list-style-type: none">● Update issue templates: [feat][workflow] change doc issue template from markdown to yml;● Update PR templates: [improve][workflow] Add PR guideline info to PR template;● Add standards: [feature][workflow] Add Pulsar PR Naming Convention● Add guides: [feat][workflow] add instructions for previewing website changes	It belongs to [misc].

Client

The following changes are shown on client release notes.

scope and PR label must be one of the following.

scope	Pulsar PR label	Where do you make changes?
Java client (Java client)	<ul style="list-style-type: none">scope/client-java	Java client
cpp (C++ client)	<ul style="list-style-type: none">scope/client-c++	C++ client
py (Python client)	<ul style="list-style-type: none">scope/client-python	Python client
ws (WebSocket)	<ul style="list-style-type: none">scope/client-websocket	WebSocket API

Below are **deprecated** scopes.

Scope	PR label	Where do you make changes ?	Why deprecate it?
rest	● scope/admin	REST	rest belongs to [admin] scope
go (Go client)	✗	Go client	Note: Go client release notes are automatically synced from apache/pulsar-client-go .
nodejs (Node.js client)	✗	Node.js client	Note: Node.js client release notes are automatically synced from apache/pulsar-client-node .
C# (C# client)	✗	C# client	Note: C# client release notes are automatically synced from apache / pulsar-dotpulsar .

Summary

Summary is a single line that best sums up the changes made in the commit.

Follow the best practice below.

- Keep the summary concise and descriptive.
- Use the second person and present tense.
- Write [complete sentences](#) rather than fragments.
- Capitalize the first letter.
- No period at the end. ❌
- Do not include back quotes (` `).
- Limit the length to 50 characters.
- If you cherry-pick changes to branches, name your PR title the same as the original PR title and label your PR with cherry-pick related labels.
- Do not use [GitHub keywords](#) followed by a #<issue-number>. This information should be provided in PR descriptions or commit messages rather than in PR titles. ❌

Full examples

As explained in the [How to write good PR titles](#) chapter:

Rule

A good title = clear format (**type** and **scope**) + self-explanatory **summary**

Here are some format examples. For self-explanatory summary examples, see [here](#).

Changes	Unclear format ❌	Clear format ✅
Submit breaking changes	[Breaking change] xxx	[feat][broker]! Support xx
Submit PIP changes	[PIP-198] Support xx	[feat][broker] PIP-198: Support xx

Changes	Unclear format ❌	Clear format ✅
Cherry pick changes	[Branch-2.9] Fix xxx issue.	[fix][broker][branch-2.9] Fix xxx issue
Revert changes	Revert xxx	[revert][broker] Revert changes about xxx
Add features	Adding xx feature Support delete schema forcefully	[feat][java client] Add xx feature [feat][schema] Support xx
Fix bugs	[Issue 14633][pulsar-broker] Fixed xxx	[fix][broker] Fix xxx
Submit improvements	Enhances xx Bump netty version to 4.1.75	[improve][sql] Improve xx performance [improve][build] Bump Netty version to 4.1.75
Update tests	reduce xx test flakiness	[improve][test] Reduce xxx flaky tests

Changes	Unclear format ❌	Clear format ✅
Update docs	[Doc] add explanations for xxx 2.8.3 Release Notes Fix typos in xx	[feat][doc] Add explanations for xxx [feat][doc] Add 2.8.3 release note [fix][doc] Fix typos in xx
Update website	[Website] adjust xxx	[improve][site] Adjust xxx
Update instructions/ guidelines	Update xxx guideline Add Release Plan to Pulsar Website doc	[improve][doc] Update xx guidelines [feat][website] Add release plan