

Patch submission and review

- [Overview](#)
- [Patch Review Tool](#)
- [Contributor and Reviewer Workflow](#)
- [Committer Workflow](#)
- [How to get your patches reviewed](#)

This page is meant to document the various steps to working with git to contribute or review Kafka code. There are probably a lot of bugs in these steps or possible better recipes, so help make this page better. If you want to push your commits without passwd, please see [apache git wiki](#).

Overview

The Kafka project development ecosystem involves git (for version control), JIRA (for issue tracking) and either GitHub pull requests or Review Board (for reviewing code changes made by contributors). See below for more details.

Patch Review Tool

We have moved the contribution workflow to Github pull requests. The patch review tool has been phased out. If you are submitting a new patch, please use a Github pull request. If you are still interested in learning more about it, [you can find the instructions here](#).

Contributor and Reviewer Workflow

The process for contributing or reviewing a patch is documented in the [Contributing Code Changes](#) page.

Committer Workflow

If you are merging a patch attached to a JIRA (and not a GitHub PR), [here is a suggested workflow](#).

Instructions on merging GitHub pull requests [are here](#).

How to get your patches reviewed

Please ping the dev mailing list if you have a patch that needs a review and it will be added to the queue. The following (JIRA link) are issues that currently have patches available and have an assigned reviewer:

type	key	summary	assignee	reporter	priority	status	resolution	created	updated	due
------	-----	---------	----------	----------	----------	--------	------------	---------	---------	-----



JQL and issue key arguments for this macro require at least one Jira application link to be configured