Pull Requests

The process described here has several goals:

- Maintain Product quality
- Fix problems that are important to users
- Engage the community in working toward the best possible product
- Enable a sustainable system for maintainers to review contributions
- Further explanation on PR & commit messages can be found in the [How to Write a Git Commit Message] article by [Chris Beams].

Please follow these steps to have your contribution considered by the approvers:

1. Ensure all commits have a Sign-off for DCO, as described in [DCO.md].
2. Follow all instructions in PULL-REQUEST-TEMPLATE.md.
3. Include appropriate test coverage. Testing is 100% automated. All submissions must be testable in an automated fashion.
4. Follow the Style Guides.
5. After you submit your pull request, verify that all status checks are passing.

What Makes A Good Pull Request?

The following guidelines, based on Hyperledger Fabric's contribution guidelines will help ensure that your pull request gets promptly reviewed.

One Pull Request, One Change

- This limits the surface area of the change, and makes it easier to identify root causes when issues arise.

Link to Issue

- When submitting your PR, include the issue's link in the description and number in the title (i.e. [BESU-99] My Awesome PR), this helps provide more context on your work and auto-update the issue ticket to include a link to your PR.

Minimize LOCs per PR

- PRs get near exponentially longer to review as the number of lines of code increase. Ideally, try and keep your changes to under 300 LOC. If that is not possible, try and break up your PR into smaller ones for reviewers to review sequentially.
- One way to do this if, for some reason, the change has to all go in the codebase at once, is to have a PR open on the Besu repository linking to smaller PRs on your Besu fork.

Write Meaningful Commit Messages

- As mentioned above, your commit title should include the issue number (i.e. [BESU-99]) while the description should link to the issue. Please include a comprehensive description of the changes in your commit description.

Be Responsive

- Don’t let a PR sit idle with unaddressed comments until it gets to a point where you need to rebase the whole thing. If you are pausing your work on an issue, please indicate it in the PR comments.

What if the status checks are failing?

- If a status check is failing, and you believe that the failure is unrelated to your change, please leave a comment on the pull request explaining why you believe the failure is unrelated. A maintainer will re-run the status check for you.
- If we conclude that the failure was a false positive, then we will open an issue to track that problem with our status check suite.

Code Review

While the prerequisites above must be satisfied prior to having your pull request reviewed, the reviewer(s) may ask you to complete additional design work, tests, or other changes before your pull request can be ultimately accepted. Please refer to Code Review.