Committer Best Practices

Role Definitions

Contributor

Anyone who wants to participate in the project.

Examples:

- providing input/responses on the email list
- contributing a bug fix
- contributing code for a new feature
- writing test case or documentation
- Reviewing commits
- Integration/Deployment testing of merged commits
- Triaging failed builds/deployments and runtime use cases
- Jira/Confluence authoring
- Contributing to weekly meetings and getting involved in assigned tasks

Contributors always have a voice and are welcome to provide thoughts and insights and in any technical discussion within the project as well as assist in direct use/testing of the project artifacts.

Committer

A Committer is a contributor that has the authority, and responsibility to submit changes to the ONAP software repository.

Typical characteristics of a Committer are:

1. Deep expertise in the code base over which they are committers
2. Time dedicated to reviewing code contributions made by other contributors
3. Knowledge and understanding of the overall development activities occurring within the project - this is important so that the review of new code is taken in the context of the overall development for the project.
4. Knowledge and understanding of other, interdependent projects within ONAP and how contributions to this project affect work being done elsewhere by others.

The Committers on a project will review each code contribution made by the Contributors, and other Committers on the project. Often, a Committer will need to enter into a dialog with a Contributor to have them make changes to the contribution to better fit the functional, structural makeup or style of the existing codebase. It is preferable to have at least 2 Committers show approval (with a +1) for a contribution before it is accepted into the repository. It is also ideally best practice to never have a Committer review and/or approve their own contribution into the repository.

Project Technical Lead (PTL)

The PTL is a committer who is the one point of contact responsible for representing the project to the rest of the ONAP community.

For projects that become part of any given release, the PTL is responsible for reporting milestone status and release readiness to the rest of the community. The PTL for each project is elected by the other committers within the project. Please note that it is very common for individuals to be a committer on one project, and an contributor on another. However, there is nothing stopping an individual from being a committer on multiple projects. Also, it is rare, but not unheard of, that an individual can be a PTL on more than one project.
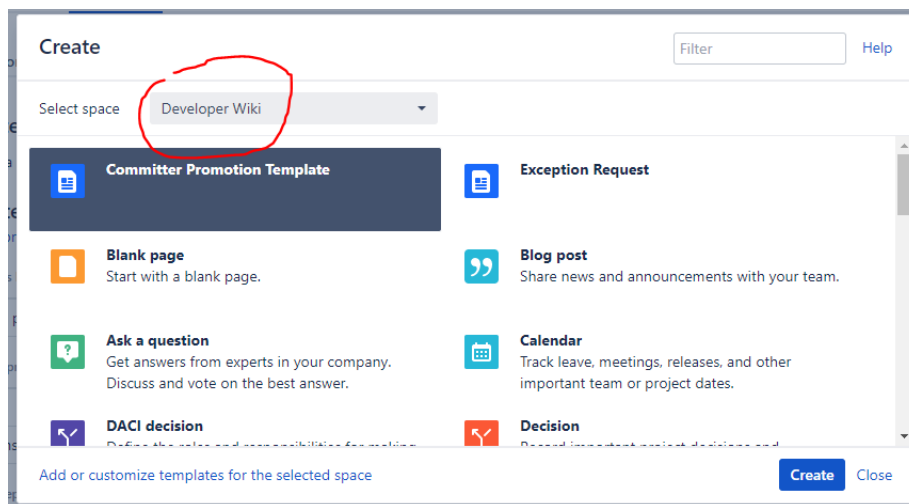
Change

A change is a portion of code submitted to https://gerrit.onap.org. Sometimes referred to as a 'gerrit' or patch, it is essentially a git commit pushed to gerrit code review system.

Committer Promotion Prerequisites

ONAP contributors seeking promotion to committer status should demonstrate the following abilities and behaviors before being considered for promotion. Ideally the individual has been participating in the community and several projects including the one they are seeking promotion to. Contributors can perform 95% of the duties of a formal committer including the following which would be expected from both contributors and committers.

- Gerrit involvment - putting up reviews of their own, reviewing other reviews of contributors/committers - you can add yourself as a reviewer as well.
- Ability to build, deploy and run your own project in the context of a larger ONAP deploy
- Ability to build and debug your own project in an IDE like Eclipse or IntelliJ
- Proactive - do not wait to be part of a epic/story/task or wait to be part of a review, or wait to bring forward build/design issues, or wait to bring new capabilities to the project and any other members of ONAP pushing or pulling from your component
- JIRA involvement - including one or more of raising, reviewing, commenting, testing, linking, attaching to, answering questions on, closing JIRAs
- Design review of all the release Epics and wiki documentation for the current and pending release.
- Confluence involvement - writing/editing/reviewing/answering-on wiki pages around their projects, building in general
- Git involvement - being able to clone, build, patch, commit and run the code in their project and ideally any of the north or southbound dependend projects
- Knowledge of the architecture and API of the project - ideally across ONAP
- Participation in onap-discuss work including helping with issues, raising issues
- Participation in the weekly project meeting
- Participation in pre-release blitzes
- Participation as secondary PTL in the TSC when required
- Integration ability - the developer should be able to bring up the entire ONAP deployment either in OOM or HEAT to be able to E2E test their particular component and any flows involving it

To start the promotion process, create a new page by clicking the 3 dots next to the create button and select the **Committer Promotion Template** as shown below. After that follow these instructions **Committer Management Automation via INFO.yaml**:

Best Practices

- Committers are prohibited from a "self merge" of their own changes
  refer to

  https://lists.onap.org/pipermail/onap-discuss/2018-February/008068.html

  https://lists.onap.org/pipermail/onap-discuss/2018-February/008181.html

  https://lists.onap.org/pipermail/onap-tsc/2017-May/002341.html

  TSC 2018-03-08

  - Committers should add other committers as reviewers and have at least 2 Committers show approval (with a +1). The first reviewing committer will +1, the second reviewing committer will +2.
  - One exception to the rule is self merging critical changes required to unblock broken builds - *maybe*.
- Committers should only merge proposed changes from contributors after at least 2 (+1) reviews.
- Committers should remove themselves from a review under the following circumstances:
  - Committer has been added as a reviewer to a change they do not feel comfortable reviewing
  - Committer anticipates a lack of time to review within 1 week
- Committers who remove themselves should leave a short comment in the change explaining to the owner the reason for removal
  - This allows owners opportunity to find others willing to review (or to make changes to significantly simplify or further document change)
- Committers should aim to review all pending changes which have passed verify, have no merge conflicts or (-1) or (-2) reviews in a timely manner.

- Committers are welcome to ignore pending changes which do not pass verify, have merge conflicts or have been reviewed (-1) or (-2).
- Committers should occasionally review the list of all old changes (To be defined). Owners of old changes should be contacted to ascertain if the change can be abandoned. If change owners do not respond or respond that the change has been abandoned, it should be abandoned.

Sources: TSC Discussion - https://lists.onap.org/pipermail/onap-tsc/2017-May/000476.html, Open Source Practice in OpenDaylight https://wiki.opendaylight.org/view/Genius:Main#Information_for_committers and Open Source Practice in OpenStack via TSC Discussion - https://lists.onap.org/pipermail/onap-tsc/2017-May/000496.html