Modifying OSM Code

Eduardo Sousa (Canonical)
Fabian Bravo (Whitestack)

Repositories

URL: https://osm.etsi.org/gitweb/

Reviews

URL: https://osm.etsi.org/gerrit/

It always starts the same way

1. Get the code
   - git clone <repository-url>
2. Setup the git repository information
   - git config --local user.name <username>
   - git config --local user.email <email>
   - git config --local pull.rebase true
3. Install gerrit hooks
   - curl -Lo .git/hooks/commit-msg http://osm.etsi.org/gerrit/tools/hooks/commit-msg
   - chmod u+x .git/hooks/commit-msg
4. Setup your .gitignore
   - cp .gitignore-common .gitignore
   - Verify if it includes all the extra files generated by your IDE/development environment
5. (Optional) Create a .dockerignore
   - When creating your own local Docker images, there are extra files that are not needed
   - Check example
6. Local repository is now setup

OSM is moving to tox

Most projects now include a tox file. This enables different virtual environments to perform the following tasks:

- black - uses [black](#) to check code style formatting

- flake8 - uses [flake8](#) to check code style formatting

- cover - uses [nose2](#) and [coverage](#) to run unit tests and generate coverage reports

- pylint - uses [pylint](#) to perform code analysis

- safety - uses [safety](#) to check for security problems in the dependencies used

There are other virtual environments defined in tox not covered here, mainly used to build software or other management tasks.

Why you should run tox?

No need to wait for Jenkins to show you formatting or unit test errors.
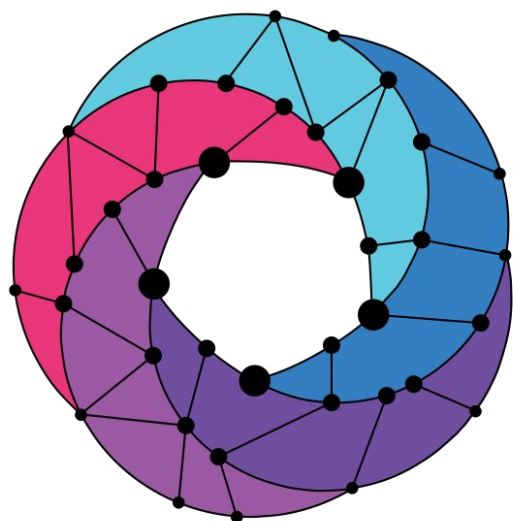
Testing & debugging

The most common ways of testing code are:

1. Build local Docker images

2. Run changed component locally

Submit your changes for code review

1. Stage the files in the local repository
    ○   git add <file(s)>
2. Create the commit with the change
    ○   git commit -s -m "Message of the commit"
3. (Optional) Add a more detailed description
    ○   git commit --amend
4. Rebase the change (if needed)
    ○   git pull origin master
    ○   Change the branch from master to v9.0, v8.0 or others (if needed)
5. Push your changes to the right branch
    ○   git push origin HEAD:refs/for/master
    ○   To push to other branches, replace master with the name of the branch e.g.:
        i.   git push origin HEAD:refs/for/v9.0

Find us at:
osm.etsi.org
osm.etsi.org/wikipub

© ETSI