

# Documentation Toolchain

**OBS: This page needs to be updated to point to the Documentation section of the wiki**

**Remove details and add general help and links to relevant content.**

OPNFV documentation is automated and integrated into our git & gerrit toolchains.

We use RST document templates in our repositories and automatically render to HTML and PDF versions of the documents in our artifact store, our Wiki is also able to integrate these rendered documents directly allowing projects to use the revision controlled documentation process for project information, content and deliverables.

## Enabling the documentation toolchain in your project

There are some basic steps you will need to perform in order to get the toolchain working in your repository, this include setting up Jenkins jobs and establishing your project documentation structure. See the [documents toolchain instructions](#) for more details on getting started in your project. The linked page above is written completely in generated artifacts, check the source to see how to do it yourself!

## Licencing your documentation

All contributions to the OPNFV project are done in accordance with the OPNFV licensing requirements. Documentation in OPNFV is contributed in accordance with the [Creative Commons 4.0](#) licence.

All documentation files need to be licensed using the creative commons licence. The following example may be applied in the first lines of all contributed RST files:

```
.. This work is licensed under a Creative Commons Attribution 4.0 International License.
.. http://creativecommons.org/licenses/by/4.0
.. (c) <optionally add copywriters name>
```

These lines will not be rendered in the html and pdf files.

## How and where to store the document content files in your repository

All documentation for your project should be structured and stored in the `<repo>/docs/` directory. The documentation toolchain will look in these directories and be triggered on events in these directories when generating documents.

### Project documentation

While many of your documents may be specific to your work there is a general structure proposed for storing and handling documents that are common across many projects. You are encouraged to establish the following basic structure for your project as needed:

- Requirement Documentation: `<repo>/docs/requirements/`
  - Should be the folder for your requirement documentation
- Design Documentation: `<repo>/docs/design/`
  - Should be the folder for your upstream design documents (blueprints, development proposals, etc..)
  - For details on requirements projects' structures see the [Requirements Projects](#) page

### Release documentation

Release documentation is the set of documents that are published for each OPNFV release. These documents are created and developed following the OPNFV release process and timelines and should reflect the content of the OPNFV release.

For release documents specific to a project, the documentation should be created in the projects' release directory.

- Release Documentation: `<repo>/docs/release-notes/`
  - Should be the directory where any release specific documentation (release notes) is produced

Release documents that represent the OPNFV release as a whole have a specific structure and content collation method. These documents have a master `index.rst` file in the `<opnfvdocs>` repository and extract content from other repositories. To provide content into these documents place your `<content>.rst` files in a directory in your repository that matches the master document and add a reference to that file in the correct place in the master `index.rst` file in `<opnfvdocs>`.

- Platform Overview: `<repo>/docs/platformoverview`
  - Should be the folder for the overview of a release deliverable
  - Note this document will be compiled into OPNFVDOCS 'platformoverview'
- [Installation and Configuration Guide](#): `<repo>/docs/configguide`
  - Should be the folder for manuals to install/configure project specific components and features
  - Guides to validate project specific features can be included

- Note this document will be compiled into OPNFVDOCS 'configguide'
- NOTE: see [Configuration guide](#).
- **User Guide:** <repo>/docs/userguide
  - Should be the folder for manuals to use specific features
  - Can be the directory where API reference for project specific features are stored
  - Can be the place for sotring typical scenario as an example
  - Note this document will be compiled into OPNFVDOCS 'userguide'

## Documentation projects

[OPNFV Documentation Project](#)

### Recent space activity



#### David McBride

[2019 Election Nominees](#) updated Jul 09, 2019 • [view change](#)

[Hunter Award Winners](#) updated Jul 02, 2019 • [view change](#)

[Community-awards](#) updated Jul 02, 2019 • [view change](#)



#### Min Yu

[Intern Project: -- Large Scale Resource Pool High Availability testing--](#) updated Jun 26, 2019 • [view change](#)

[Intern Project: Containerizing VSPERF - Automated Deployment, Control and Results-Management](#) updated Jun 26, 2019 • [view change](#)

### Space contributors

- [Mark Beierl](#) (1199 days ago)
- [Christopher Price](#) (1199 days ago)