

# ONAP Documentation

LFN Developer & Testing Forum 2020  
Trevor Bramwell

# Problems with Submodules

- › Documentation repository contains a monolithic build of all project documentation
  - › Job triggers on change to *any* project's documentation
  - › Consumes lots of resources
- › Falls over on ReadTheDocs due to resource constraints
  - › Needs to clone every ONAP project
  - › Needs memory / time to load in build all documentation

# Benefits to Transition

- › Local Builds
- › Reduced build time
- › Decoupled build failures
- › Not a en-masse change - Can be done per-repo over time
- › (Drawback) Requires some more overhead for release integration

# Dependencies to Make the Change

- › Create and publish ONAP Sphinx theme as a python module
  - › Contains shared assets (logo, styling) so they don't need to be duplicated to each repository
- › Add base configuration file to the lf-docs projects
  - › De-duplicates the configuration for Sphinx
  - › References the published theme
- › Requires community buy-in to review and approve changes to repos

# Steps to Transition a Repository

- › Create a RTD project for the repo (RTDv3 Job now handles this)
- › Add a docs/conf.py, docs/conf.yaml, .readthedocs.yaml
- › Add tox.ini configuration
  - › Local docs can be built with `tox -e docs`
- › Add the RTD job from Global-JJB
- › Trigger a build on RTD through the merge job
- › Update the documentation repository to intersphinx link the project
- › Remove the submodule from the ONAP docs repository

# Q & A

 THE **LINUX** FOUNDATION

 **LF** NETWORKING