



Getting to M1 Release Planning Training

Gildas Lanilis – ONAP Release Manager

June 15 , 2017

Where do we stand?

- Congrats, your project has been Approved!!! (or is in the pipe for next week)
- You attended the ONAP “[Project Roles and Responsibilities \[1\]](#)” training.

(Thanks Chris)

- Now, you know what it entails to be a PTL. Run for election 😊
- What else is needed to reach M1 Release Planning (scheduled for June 29)?
 - That is what this presentation is about

Agenda

- Release Planning Template
- Release Planning Checklist
- Resources & Repositories
- Intellectual Property Rights, FOSS
- Wiki Project Space
- Development Best Practices

Release Planning Template

- Release Planning is all about the commitments to the community.
- Template is available in [wiki](#) [2]
- It covers:
 - Scope
 - Architecture
 - API
 - Dependencies
 - Resources
 - Risks
 - Gaps
- Actions:
 - Create a copy of the template in your [project space](#) [3] and fill it out.
 - Let the TSC and Release Manager know when you are done.

Release Planning Checklist

- They are so many things to do. Right? Checklist helps.
- Checklist is available in [wiki](#) [4]
- It covers multiple products area:
 - Product Management
 - Release Management
 - Integration and Testing
 - Development
- Actions:
 - Create a copy of the template in your project space and fill it out.
 - Let the TSC and Release Manager know when you are done.

Resources & Repositories

- Idea: **ONE central place** [5] to define
 1. Project Artifacts (PTL, Jira Project Name, ...)
 2. Repositories (name, maven ID, description)
 3. Resources: list of all the committers

- Actions:
 - Existing projects repositories (former open ecomp) have already been defined. But please checkout:
 - Repos: name, maven ID, description
 - Resources: list of committers (including LFID, email address, Location)
 - Let the TSC and Release Manager know when you are done.

Intellectual Property Rights

Members agree that **all** new inbound **code** contributions to ONAP shall be made under the Apache License, Version 2.0

- Anything **re-compile** MUST be **Apache 2**
- Linux Foundation runs **FOSSOLOGY** [16]
- Awareness of any executable embedded



<https://wiki.onap.org/display/DW/Open+Source>

- For Legal purpose, there is a need to record all Third Party code and libraries embedded within ONAP
- Info in FOSS Table (for both Open Source Code and Libraries):
 - Open Source Code Name
 - Version
 - License Type
 - Description
 - Link toward License Info
 - Notes (including reason not using Apache Version 2)
- Action:
 - Foss table for each project have been created in [wiki](#) [6]
 - Fill out FOSS table for each projects (important for M2, but not show stopper for M1)

Project Page

- Most of us like to find things in the right place and the right format
- LF has created a project structure, where each Team can:
 - Upload their Release deliverables (proposal, planning, checklist)
 - Record meeting minutes
 - Post and discuss any other item relevant to the Team
- Project Pages are available under “[ONAP Project](#)” page [7]

Development Best Practices

- Draft Development Best Practices in [wiki](#) [15]
 - Continuous Integration
 - Commit Process
 - Code Review
 - Code Coverage and Static Code Analysis
 - Functional Test Case Testing
 - Java Code Style
 - Scrum and Jira

Continuous Integration

- [CI Practice](#) [8]
- The core goal of Continuous Integration is to get **fast feedback**
- There is **ONE** Branch: Master
- If there is 1 word to remember "Commit, commit, commit" multiple times a day
- Don't check In on a broken build: If the build breaks, the **developers** are **responsible** for fixing it
- Always run all commit tests locally before committing
- Wait for commit tests to pass before moving on. If the build fails, **fix it immediately** or reverse your changes
- Never go home on a broken build ...

Commit Process

- [Commit Practice](#) [9]
- 3 reasons to perform a good commit:
 - ONAP is public, so everything you do is widely visible. As you are proud of your work, you want to make good impressions
 - It speeds up the reviewing process
 - It helps to write good Release Notes
- **JIRA ID mandatory** in message (not in the header)
- **ONLY 1** JIRA issue per commit
- Self-commits are **not allowed**
- **NEVER** embed binaries (jar, war, tar, gz, gzip, zip) in Gerrit

Code Review

- [Practice](#) [11]
- You may not find bugs while doing Code Review, however you will find mistakes overlooked in the initial development phase
- Code Review is mandatory
- What to check?
 - Comply to commit Practices
 - Check for FOSS compliance
 - Check if only **ONE** JIRA issue per commit
 - Is the code easily readable?
 - All Code is in English. Comment is Code
 - ...
- Shall we recommend a due time to complete the code Review? 24-36 hours?

Code Coverage and Static Code Analysis

- Practice [10]
- What % is realistic for ONAP first Release? 30-50%?
- Sonar is available: <https://sonar.onap.org/>

Functional Test Case Testing

- This is about functional testing of your components, not the whole ONAP
- Beside Unit Tests, the team needs to have the confidence they have not broken their surroundings
- Functional Test Case must be automated
- Functional Test Case are developed by Team, executed locally and then within ONAP Jenkins
- What are the steps
 - All functional Test Case defined by M2 Functionality Freeze
 - 50% of functional test case implemented by M3 API Freeze
 - 100% of functional test case implemented by M4 Code Freeze

Java Code Style

- [Practice](#) [12] Based on Google Java Style
- Could we agree on this?
 - **4.2 Block indentation: +4 spaces**
 - **4.4 Column limit: 120**
- Java Code Style
 - How do we want to enforce? By Automated Tool or by “Human” Code Review

Scrum and Jira

- Practice [Scrum](#) [13], [Jira How To?](#) [14]
- We want to be Agile. What does it mean?
 - >Focused and Transparent to the community
- Usage of 2 weeks iteration Scrum Board
- Simplified Jira workflow
 - Simplified [Workflow](#) (4 statuses: Todo, In Progress, Implemented, Done)
 - Simplified [Resolution code](#) (Unresolved, Done, Cannot Reproduce, Duplicate, Won't do)

Key Onboarding Behaviors

- Get connected: join the [mailing lists](#) [17], [IRC](#) [18], [LFID](#) [19]
- Be Public: private message don't scale
- Grab the info: listen first
- Start to participate in [calls](#) [20]
- Ask questions
- Dive in
- Fix defects
- Find your place
- Get familiar with the whole system

Key Principles

- One principle: KISS Keep It Simple S-----d!
- Be responsive
- Be proud, take ownership
- Failure is OK. Do it fast...and learn from it
- Have Fun...you are human
- Celebrate every success

Full URLs to references (1/2)

- [1] Project Roles and Responsibilities

<https://wiki.onap.org/display/DW/Presentations?preview=%2F3246513%2F6592765%2FONAP-Project+Roles+v1.pdf>

- [2] Release Planning Template <https://wiki.onap.org/display/DW/Release+Planning+Template>

- [3] Project Space <https://wiki.onap.org/display/DW/ONAP+Projects>

- [4] Deliverable for Planning Milestone Checklist Template

<https://wiki.onap.org/display/DW/Deliverables+for+Planning+Milestone+Checklist+Template>

- [5] Resources and Repositories <https://wiki.onap.org/display/DW/Resources+and+Repositories>

- [6] FOSS Table <https://wiki.onap.org/display/DW/Project+FOSS>

- [7] ONAP Projects <https://wiki.onap.org/display/DW/ONAP+Projects>

Full URLs to references (2/2)

- [8] CI Practice <https://wiki.onap.org/display/DW/Continuous+Integration>
- [9] Commit Practice <https://wiki.onap.org/display/DW/Commit+Messages>
- [10] Code Coverage <https://wiki.onap.org/display/DW/Code+Coverage+and+Static+Code+Analysis>
- [11] Code Review <https://wiki.onap.org/display/DW/Code+Review>
- [12] Java Code Style <https://wiki.onap.org/display/DW/Java+code+style>
- [13] Scrum <https://wiki.onap.org/display/DW/Scrum>
- [14] Jira How to? <https://wiki.onap.org/pages/viewpage.action?pageId=6592023>
- [15] Development Best Practices <https://wiki.onap.org/display/DW/Developer+Best+Practices>
- [16] Fossology <https://www.fossology.org/>
- [17] Mailing List <https://wiki.onap.org/display/DW/Mailing+Lists>
- [18] IRC <http://webchat.freenode.net/?channels=onap-meeting>
- [19] LFID <https://wiki.onap.org/display/DW/Joining+the+Community>
- [20] Join Meetings <https://wiki.onap.org/display/DW/TSC+Meeting+Resources>



ONAP

OPEN NETWORK AUTOMATION PLATFORM

谢谢