



Proposal for Beijing Calendar & Development Best Practices

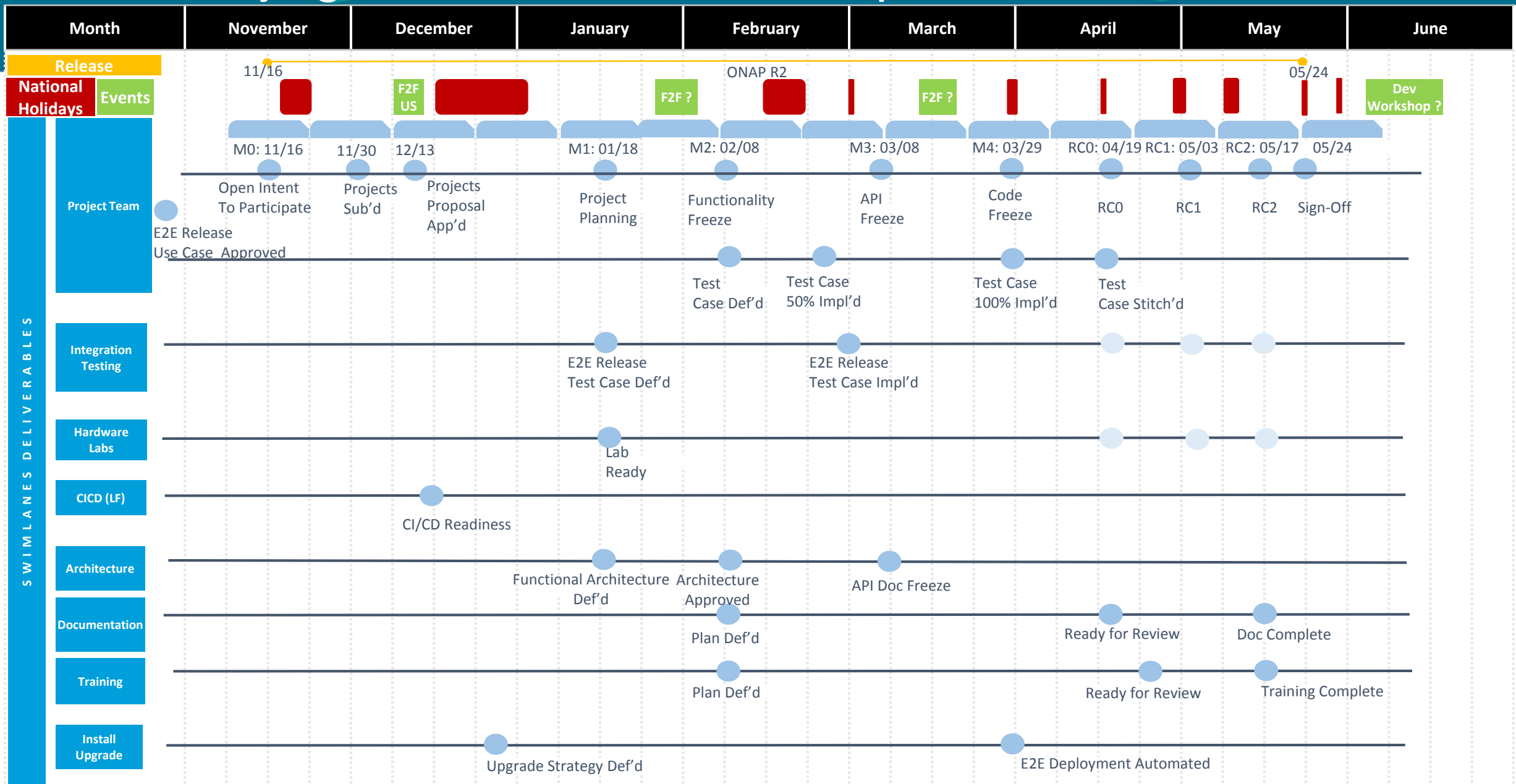
Gildas Lanilis – ONAP Release Manager

F2F Santa Clara Dec 11-13, 2017

Agenda

- Beijing Release Calendar
- Changes on Milestones
- Branch Cut
- Code Coverage and CSIT goals
- Commit Process and Code Review
- Jira

ONAP Beijing Release Calendar Proposal



Changes: Milestones

- M1 Project Planning
 - Project Planning available for review by Jan 8. We want to give time to review the plans.
 - Security goals: Specify what CII badging level the project expects to meet?
 - Platform Maturity: Specify the target to meet?
- M1 Labs Readiness
 - Labs should be ready at same time as release planning
 - Integration Team will permanently test the whole solution

Changes: Milestones

- M2 Functionality Freeze
 - Architecture is approved

Changes: Milestones

- M4-RC0: Code Freeze-Release Candidate 0
 - 3 weeks instead of 2
 - **Week 1:**
 - Project Teams gather to perform their pairwise testing and fix bug.
 - Teams are self-organized and may take any opportunity at any time during development cycle to pairwise.
 - **Week 2 and 3:**
 - Project Teams “Show & Tell” to Integration the outcome of their testing.
 - Project Teams may need to re-iterate as bugs are found.
 - This is orchestrated by Integration.

Branch Cut

- RC0-Sign-Off: Branch Cut
 - When shall we cut the Branch?
 - Or when shall we decide to cut the branch?
 - LF creates Branch and update Jenkins Jobs (consistent across all projects)
- At Sign-Off:
 - LF tags release artifacts (consistent across all projects)
 - PTL bumps the version Digits

Code Coverage and CSIT

- In Amsterdam, 30% code coverage for all repo. Goal for Beijing: 50%
 - For discussion

- In Amsterdam: Team were empowered to define their CSIT suite.
 - What approach for Beijing?
 - How much more?

Note: LF infra to support Python

Commit Process and Code Review

- Commit Process
 - No Self Commit allowed for committers
 - 36 Business Hours. To clarify:
 - To provide feedback. Once feedback is provided the timer is set to 0
 - To respect the effort of the contributor
 - Deferring review: PTL sets a note in Gerrit on the reasons the defer merge.
 - Commit body must document the rationale on the changes: that will help the reviewer and your change to get code through quickly
- Code Review
 - Anyone can review code and provide -1,0,+1
 - Only Committers can provide -2,+2
 - PTL can delegate Code Review to anyone they trust

- Make the following fields mandatory:
 - Affects Version/s: Project version(s) for which the issue is (or was) manifesting
 - Fix Version/s: Project version(s) for the issue to be fixed
- PTLs are in the driving seat:
 - PTL can change bugs priority
 - PTL decides what go into the Sprint
 - PTL decides on Fix Version
 - PTL can remove items from a Sprint
- As a reporter, do not add a bug in the Current Sprint.
 - If a bug is critical, place it on the top of the Product Backlog (you can also kindly email the PTL)



ONAP

OPEN NETWORK AUTOMATION PLATFORM

谢谢