Anuket Release Process Overview

THE **LINUX** FOUNDATION

LF NETWORKING

Introduction

› The Anuket release process has been in development since the end of CY '20.

› A major challenge has been to integrate two previously separate processes into one.

› This process will be put into practice for the first time with the "Lakelse" release, which kicked off on June 1 and is scheduled for sign off on Dec 7.

› Thanks to the members of the former CNTT and OPNFV organizations for their participation in the weekly release planning process meetings and for their patience and resolve in getting us to this point.

Release Milestone Overview

Milestone	Description	Delta from M0
M0	Start of release	
M1	Release Definition	M0+4w
M2	Scope Freeze	M0+14w
M3	Code: RC Validation Testing Spec: Content Freeze	M0 + 18w
M4	Code: RI Validation Testing Spec: Proofreading	M0+22w
M5	Release Readiness	M0+25w
Sign Off	Release	M0+26w

M0 ⇒ M1 - Release Definition

Purpose: propose the scope of the release

› Workstream leads recommend RM and RA versions to the TSC for use in the release

› Workstream leads recommend RA features to TSC for implementation in the release

› RC validation plan created or updated

› RI validation plan created or updated

› Release string created in Jira

› Release string created in GitHub

› Release milestone dates added to GitHub

› PTLs submit project-level release plans

› Workstream leads complete specification planning template for release

› RC workstream leads propose plan for RC development (based on an updated RA)

› Release manager prepares release schedule and presents to the TSC for approval

M1 ⇒ M2 - Scope Freeze

Purpose: freeze the scope of the release

› RC workstream lead(s) prepare RC for RI implementation, then confirm to the TSC, including documentation of RA coverage

› Project PTLs create Jira issues for features and bugs and assign them to the release, using the "fix version" field in Jira

› Developers review new specification development and create github issues, as necessary

› Specification team creates high level issues in GitHub and lists them in dashboard

M2 ⇒ M3 - RC Val Testing (code) / Content Freeze (spec)

Purpose: RC software team completes validation testing. Specification team freezes content.

› RC(s) tested and validated. Results are reported to the TSC.

› RI(s) developed and ready for validation

› Software projects resolve high priority Jira issues assigned to the release

› Prepare preliminary documentation (directory structure and placeholder documents, at a minimum)

› Workstream leads share closed PRs with the TSC. The PRs define the scope of the release. PRs are marked complete in the release dashboard.

M3 ⇒ M4 - RI Val Test (code) / Proofreading (spec)

Purpose: RI team completes validation testing. Specification team resolves issues on dashboard.

› RI(s) tested and validated

› All high priority Jira issues resolved

› All specification issues identified on the dashboard are resolved or moved to backlog

M4 ⇒ M5 - Release Readiness

Purpose: ensure that all aspects of the specifications and software are ready for release

› Complete final documentation, including RI cookbook(s)

› Resolve all Jira issues assigned to the release (fix version field), or move them to the next release

› Prepare and deliver software artifacts

› Finalize specifications, including version numbering

› Create traceability matrix

› All PTLs and workstream leads complete marketing highlights page

M5 ⇒ Sign Off

Purpose: assess status and complete the release

›   Release manager assesses status and makes recommendation to the TSC

›   The TSC reviews the release manager's recommendation and votes on the release

›   Repositories are branched

›   Website and download page updated

›   Unified documentation compiled and deployed to Read The Docs

Sign Off ⇒ Post-Release

Purpose: provides an opportunity for projects to deliver release artifacts late

> › Process and tasks are TBD