

# Integration V1 Release Plan

## V1 Release Plan Priorities

The following categories are shown in order of priority.

### Autorelease

- Define community-wide artifact versioning and release strategy
  - i.e synchronized artifact versions vs. staggered/independent release cycles
- Create scripts to detect/resolve cross-project build dependency issues
- Create jobs to build the final release artifacts

### Test

- Define Test Coverage Expectations
  - e.g. require 50% UT coverage across all projects
- CSIT
  - Implement CSIT test framework and guidelines
  - Reuse CSIT framework from OPEN-O if possible
- Individual project teams will write their own respective CSIT test suites
  - Define framework and guidelines for building custom docker images for mocked services used in CSIT flows

### Integration Lab / Open Lab

- CI / CD lab(s) for developer
  1. Setup testing env
  2. ??
- E2E
  1. [Define gating use case](#), VNF selection
  2. Implement full integration between E2E test execution with results reporting back to LF Jenkins master
  3. Help individual project teams add test flows to the ETE test suites
  4. Define service template / DGs, etc.

### CI-Management

- Assist project teams with creation of Jenkins Jobs
  - Help with use of the Jenkins-Job-Builder tool
- Review/revise Jenkins job definitions for consistency and coverage
- Implement CI/CD strategy for OpenLabs
  - e.g. integrate central LF Jenkins master with lab-based Jenkins instances
  - Implementation of Jenkins jobs to be run inside OpenLabs

### O-Parent

- Parent POM to be inherited by all ONAP projects
- Centrally define:
  - Nexus repository locations
  - Code style checks (checkstyle)
  - License checks
  - Artifact versioning (depending on autorelease strategy)
  - Swagger API documentation generation

### S3P

- Focus on Longevity testing for V1
- Provide framework, infrastructure, and guidelines on writing Longevity tests
  - Individual project teams will write their own respective longevity tests

### Bootstrap

- Develop framework, tools, guidelines, and tutorials to help onboard new developers to ONAP
- Assist developers in setting up a development environment for code/deploy/test cycle
  - Leverage ONAP Operations Manager project for deployment in a dev environment if possible