

OPNFV Releases and Scenario Editions

Frank Brockners

March 22, 2016 – updated from March/14

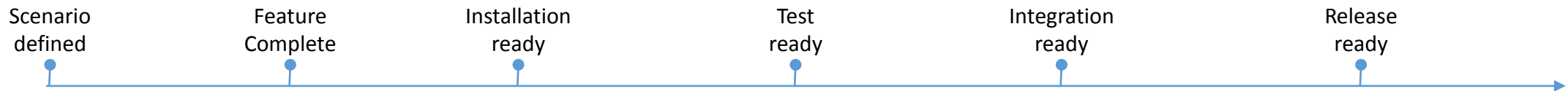
Background and Objectives

- OPNFV test components and features at system level
 - This is a unique value OPNFV provides to component/feature developers, who can typically only run unit tests
- OPNFV scenarios provide rapid feedback to developers
- The faster the results of a scenario become available, the higher the value to a component developer
 - Providing system-level test-results quickly to (upstream) developers, allows for more rapid innovation and increases the relevancy of OPNFV to developers of components
- Accommodate for the fact that not all scenarios are created equal (some are complex, some are simple)

Approach: Scenario editions

- Per scenario schedules
 - Not all scenarios are of the same complexity: Allow for per scenario schedules; allow scenarios to define their own schedule
- Scenario editions:
 - Frequently (once per month) make completed scenarios available (i.e. publish results of deployment and testing)
 - Scenario schedules work in parallel to macro OPNFV release schedule
 - Completed scenarios fold into the next OPNFV release

Per Scenario milestones



- **Feature complete:** All features for a scenario are code complete. Unit tests are successful
- **Installation ready:** Features and components for a scenario can be deployed using one of the installers. Integration into at least one of the installers is complete.
- **Test ready:** Test infra and Test cases for system level testing of features available and code complete.
- **Integration ready:** Combination of components in scenarios work “as designed”
- **Release ready:** Test runs complete, documentation ready etc.

