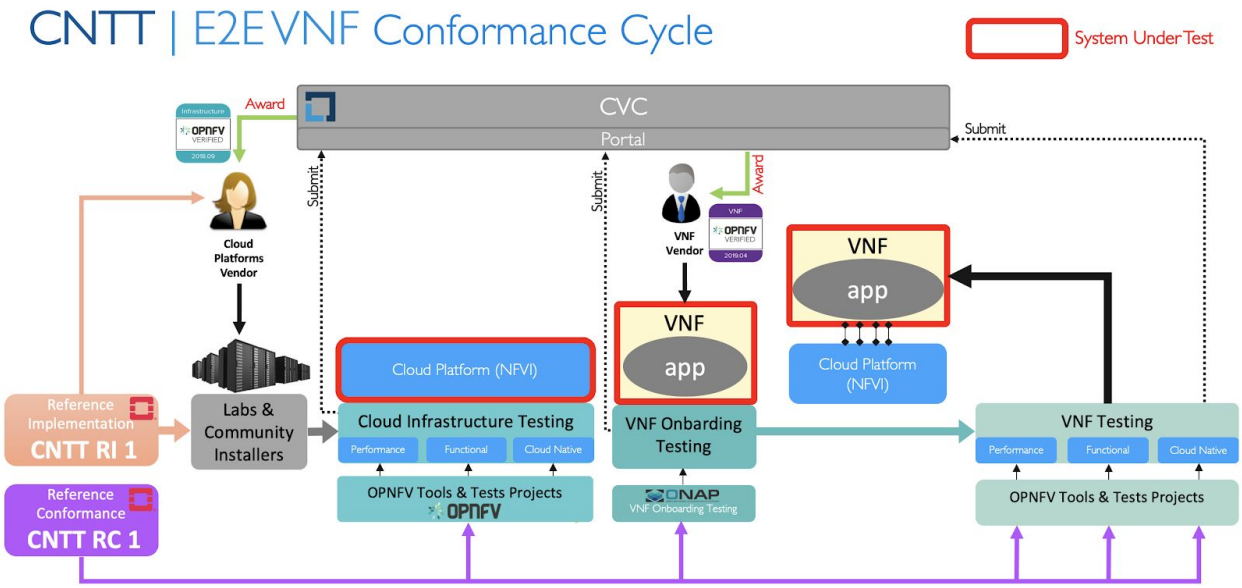
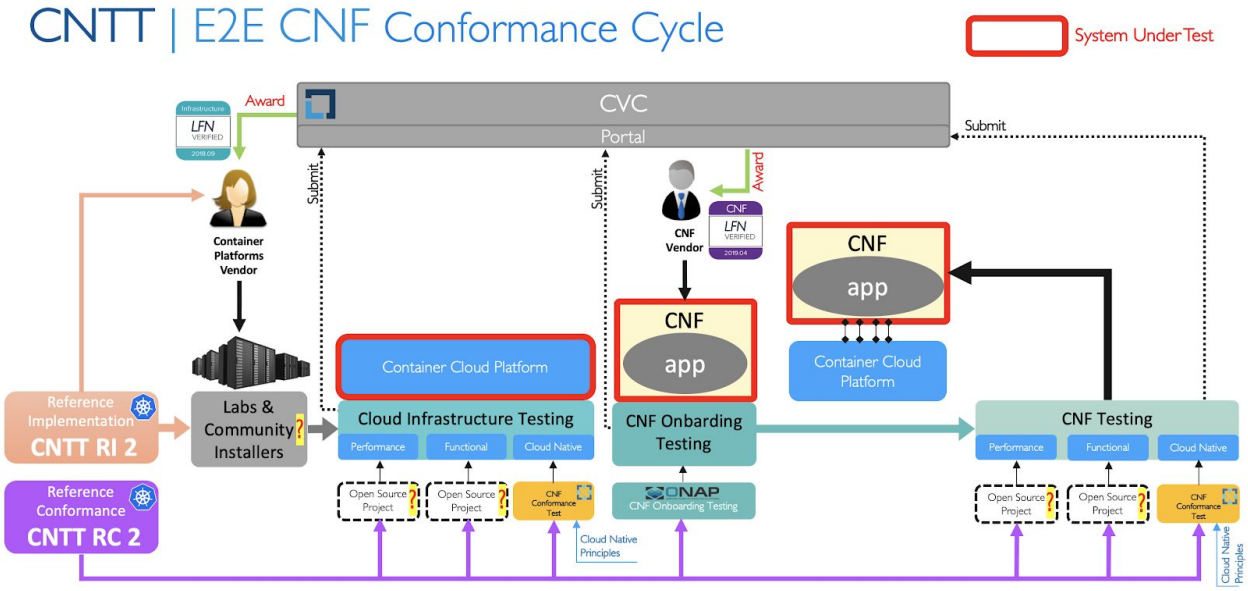
Organizing the work for OVP phase 2

Utilizing the successful process flow example from OVP, let's understand the workstreams required for phase 2 of OVP.



In the below diagram the "?" indicate processes that need definition and refinement for the differing execution environment of OVP Phase 2.



Test Categories

- Cloud Infrastructure Testing
    - Functional
        - Cloud Native ready post-install
    - Manifest validation
        - Security patches
    - Performance
- CNF Compliance Testing
    - Cloud Native:
    - Helm v3.0 testing:
    - TOSCA (?)
- CNF Validation Testing:
    - Functional Testing
        - API/Interfaces Testing
        - Major subsytem connectivity
    - On-Boarding and Lifecycle Mgmt:
    - Performance Testing
    - Interoperability (?)

Potential Workstreams

R2 Definition Requirements Activities (RA2/RI2/RC2) [Primary Owner: CNTT]

Development/Lab environment [Primary Owner OPNFV+CNTT] - What are the lab resources for hosting configurations for developing the NVFI and running on-going CI/CD verification tests? How can a “Lab as a Service” (LaaS) be instantiated for CNF/NFVI testing, development, and validation efforts? Does the CNCF TestBed meet the needs?

Test Tooling/Test Suite Development Based on Above Categories [Primary Owner: OPNFV]: Understand dependencies and what can be parallel processed. Also, what is the overall program test framework (e.g., Dovetail or something similar) that can plug in tests from projects and communities….

- **On-boarding/Lifecycle Management testing** - What are the preferred tools/methodologies for repeatably creating an NFVI environment supporting containers? Can CNFs be deployed and respond to basic lifecycle events?

- **Platform test** - What is the source for the tests and toolchains for developing and verifying the NFVI platform for container based CNFs? Are there existing CNCF infrastructure tests/standards that should be co-opted? What level of micro-services

should be integrated into the standard platform? How do we validate each of the micro-services against a specific version of the platform?

- **CNF test** - What is the source for the tests and toolchains for developing and verifying the CNFs as the SUT atop a verified NFVI? How does the CNCF test bed integrate into the OVPp2 workflow and test process? With many micro-services as “helper functions”, how do we validate each of the micro-services against a specific version of the CNF?

- **Performance testing** - Both the infrastructure and CNF performance characterization would be useful - this is very difficult to cooperatively define/execute **(DEFER)**

CVC portal [Primary Owner: CVC] - Define the UI for consumers of the CVC and on-ramp for producers of NFVIs and CNFs to publish their successful validation process results? How to the CNF compliance tests (CNCF) figure into the telco focused OVPp2 process?

Governance/Structure/Mktg Framework [Primary Owner: CVC] - Includes 3rd party labs, white papers, slide decks. With input from MAC

Roadmap

- Define MVP - by April 1
- Validate approach with partners (CNTT, CNCF, LFN projects) - during ONES technicall event (April 20)
- Implement “HelloWorld” sampleCNF test - Hackfest at June Developer and Testing Forum
- Implement MVP - by Q1/Q2 2021

Possible Items in MVP proposal

- Cloud Infrastructure Installation.
    - Installation tools and labs.
- Cloud Infrastructure testing:
    - Functional Testing:
        - API/Interfaces Testing
- CNF Onboarding Testing:
    - ONAP Testing
- CNF Testing

- Cloud Native Testing:
  - Against Cloud Native Principles. (CNF Conformance Test)
  - + Against CNTT Requirements.

Immediate task list for bootstrap:

1. Identify cloud infrastructure installation tools.
2. Identify Labs for testing.
3. Identify projects to develop test scripts and tools to perform Cloud infrastructure functional testing and agree on scope.
4. Agree on scope of CNF Onboarding testing.
5. Identify projects to develop test scripts and tools to perform CNF testing and agree on scope.
6. Agree on CVC portal.