

# Common NFVI Telco Taskforce

Technical F2F Work Shop – January 13-16, 2020

## RC Workstream: Key Updates

**Facilitators:** Mike Fix, Cedric Olivier, Rajesh Rajamani,  
Kanagaraj Manickam

 THE **LINUX** FOUNDATION



# Content & MVP Targets



# Level Set on MVPs

## Initial Badging Framework for NFVI | VNFs

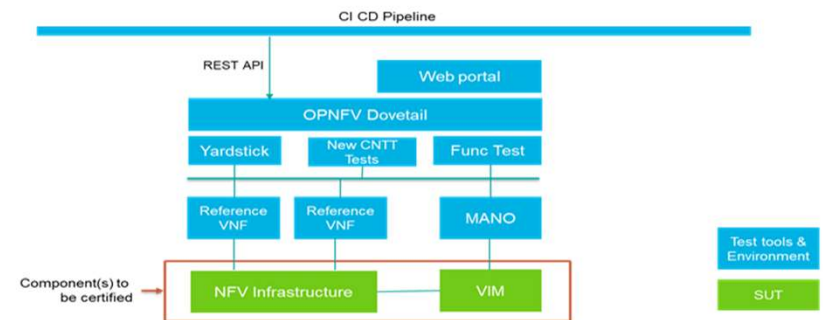
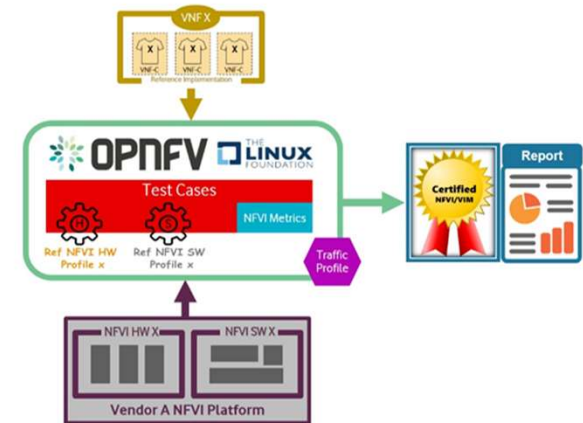
- **Lab** setup
- **Compliant** requirements
- **Execution** empirical, verification & validation
- **Evidence** meeting qualifications
- **Governance (Badging)** reviews & badging

## Certification Process Drafted

- **Certified Lab** utilized
- **Test Case Traceability** to req's
- **Execution** complete & passing
- **Results Collation** normalized & centralized
- **Evidence** meeting qualifications
- **Governance** reviews of Entry/Exit criteria and certification

## Automation Tool Chain Framework



- **Refactor** existing OVP toolchain
- **Versatile** test harnesses using standard interfaces & services
- **Supplier Integration enabling** VNF testing using Supplier Apparatus
- **Adaptable** & Portable Tool Chaining across releases





# Progress: Initial Content Creation

## Initial Content





### NFVI

- ✓ Ch01: Introduction
- ✓ Ch02: NFVI E2E C&V Framework Requirements
- alpha  Ch03: NFVI Test Case Requirements
- alpha  Ch04: NFVI TC Traceability to RA Requirements

### VFN

- ✓ Ch05: VNF E2E C&V Framework Requirements
- alpha  Ch06: VNF Test Case Requirements
- alpha  Ch07: VNF TC Traceability to RM Requirements

### DEV

- alpha  Ch08: E2E Framework Integration
- alpha  Ch09: NFVI Tests Traceability to TC Requirements
- alpha  Ch10: VNF Tests Traceability to TC Requirements
- alpha  Ch11: Gap analysis & Development

## “RI-Alpha & RC-Pre Alpha Phase”

### Delivered - Snezka MVP

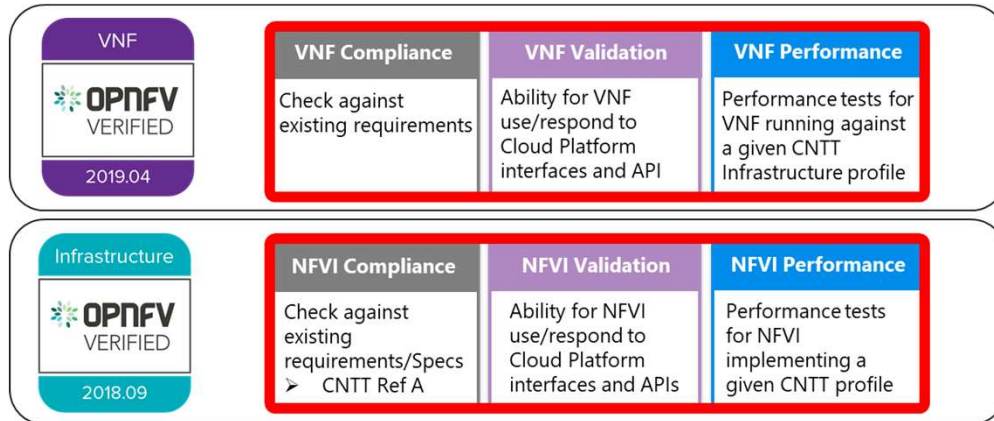
- Defined Certification
- Provide NFVI and VNF Certification Methodology
- Outline E2E Frameworks for Tools, Badges, and Process
- Included Gaps Identified During Installation

### Next Release

- Include Normalized Results Template & Repo
- Finalize TC Requirements and Traceability
- Reach consensus on Entry & Exit Criteria

# Progress: Badging Requirements

## Badging Requirements



## Badging Defined

Granting of certification by the OVP to Suppliers of NFVI+VNF upon demonstrating testing confirms:

- NFVI adheres to CNTT RA/RM requirements.
- VNFs pass interoperability tests on target NFVI with acceptable levels of stability and performance.

## Suppliers seeking NFVI & VNF certification will furnish the following:

Category	OVP/CVC Expectation	Supporting Artifact(s)
Lab	Delivered test lab conforms to RI-x lab requirements for SUT	Bare-metal H/W Validations
Compliance	Installed software conforms to RM/RA requirements for components & options	Manifest S/W Validations
Validation	FR Validation of Component and API functional behavior meets requirements	API & Platform Test Results
Performance	NFR Validation of Component, Interface, and API, results are within baseline tolerance	Performance Test Results
Results Reporting	Test Results published into centralized and common repository & portal	Normalized Results per Standards
Release Notes	Supplier provides concluding remarks, links to artifacts, having met exit criteria for testing	Release Notes

# Progress: Badging Requirements.. Test Results

## Categorization

Test suites – Functional/Platform or Performance based

## Results

Test results communicated as boolean (pass/fail), or Measurements Only

- **Functional Pass/Fail** - assertions in a test script verify the FR met its stated objective delivered by the developer
- **Performance-based Pass/Fail** - compares measured results with NFR KPIs &/or Reference VNF KPIs
- **Measurement Results** - baseline measurements when no benchmarks available to compare

## Collation | Portal

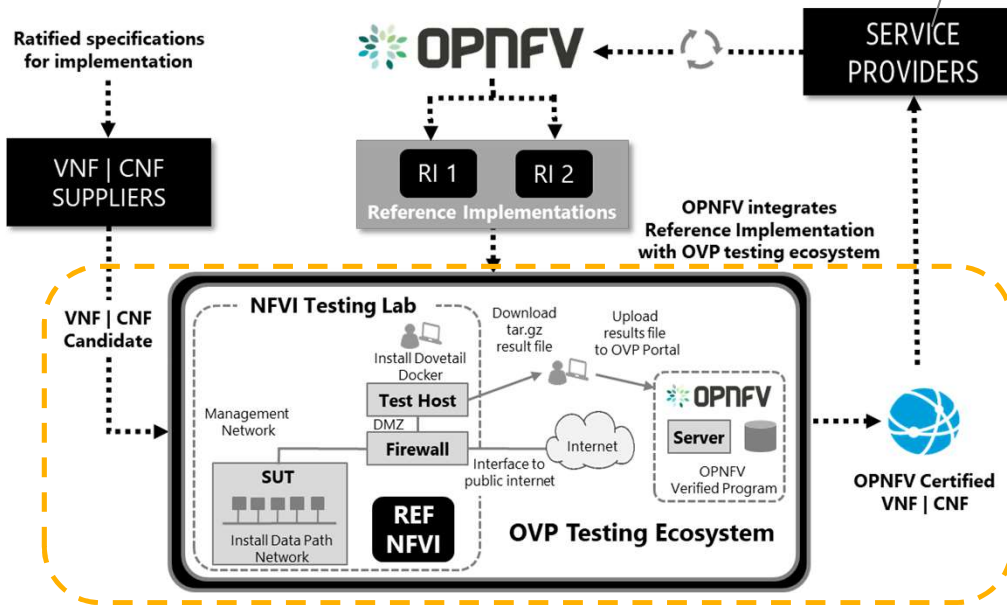
Criteria applied to collation and presentation of test-result data:

- RA number and name (e.g. RA-1 OpenStack)
- Version of software tested (e.g. OpenStack Ocata)
- Normalized results will be collated across all test runs (i.e. centralized database)
- Clear time stamps of test runs will be provided.
- Identification of test engineer / executor.
- Traceability to requirements.
- Summarized conclusion if conditions warrant test certification (see Badging Section).
- Portal contains links to certification badge(s) received.





# Reference Certification



## Objective

Deliver community certified NFVI | VNFs | CNFs to the Service Provider Marketplace

## Goals

- Provide uniform approach for NFVI | VNF | CNF certification process, lifecycle, & badging
- Certify VNF | CNF on infrastructure, instantiation, tear-down, performance, & resiliency
- Provide VNFs | CNFs with effective & efficient intake & onboarding for Lab Management
- Ensure test framework can be reused for Manifest, Empirical, and Interoperability validations for new distributions

## Target Delivery

March | April 2020 (v 1.0 - Alpha)

Aligns with Reference Architecture # 1 (OpenStack)

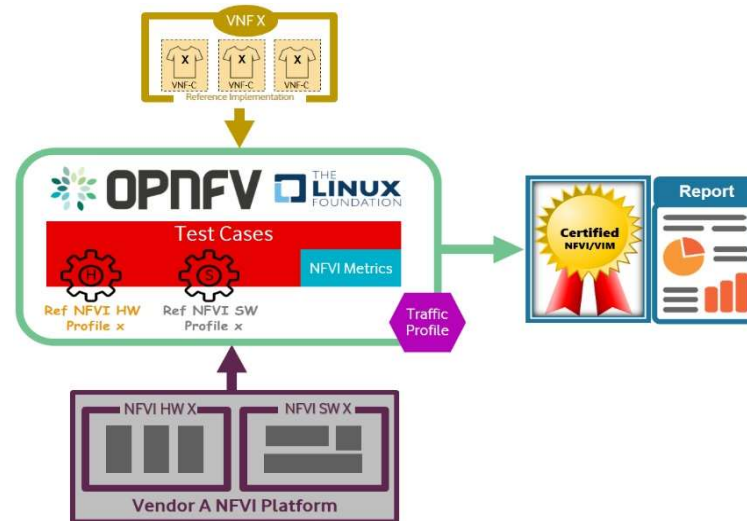


# Progress: Certification Process Framework

## Certification Process Framework

### Core Principles

- **Certification fulfilled** by the OPNFV Verified Program (**OVP**), under the Linux Foundation Networking (LFN) umbrella
- Program **overseen** by the Compliance Verification Committee (**CVC**) providing tracking and governance
- **NFVI and VNFs supplied** by vendors **must adhere** to Reference Model (RM) and Reference Architecture (RA)



## By Definition

- **Verification** conformance that NFVI is delivered per implementation specifications
- **Validation** testing performed confirms the actual output of a product meets the expected or desired outcome, or behavior
- **Certification** issuance of NFVI/VNF badges in recognition of the successful completion of verification and validation testing

## Certification and Issuance of NFVI+VNF Badges



- ✓ Utilization of target RM/RA-x certified RI lab
- ✓ Traceable test cases to requirements
- ✓ Adoption & Execution of XTesting for RC pre-alpha validations

- ✓ Collation of Normalized Results in Centralized Repository
- ✓ Entry and exit criteria satisfied
- ✓ Required artifacts supplied to the OVP



# Reference Certification Achievements | Targets for Alpha

# NFVI Compliance



## Scope & Test Strategy

---

- **Manifest Verifications** verify NFVI matches hardware and software profile specifications for RM/RA
- **Empirical Validations** baseline NFVI and Ref/Golden VNFs behaviors for future comparison
- **Interoperability Validation** performed leveraging VVP/CVC test suites to ensure VNF can be spun up, modified, or removed, on the target NFVI

## Not In Scope

---

- VNF functional testing
- MANO for VNFs
- Validating VNF's ability to be upgraded
- Georedundant and Load Testing

## RI-Alpha & RC-Pre Alpha Release

---

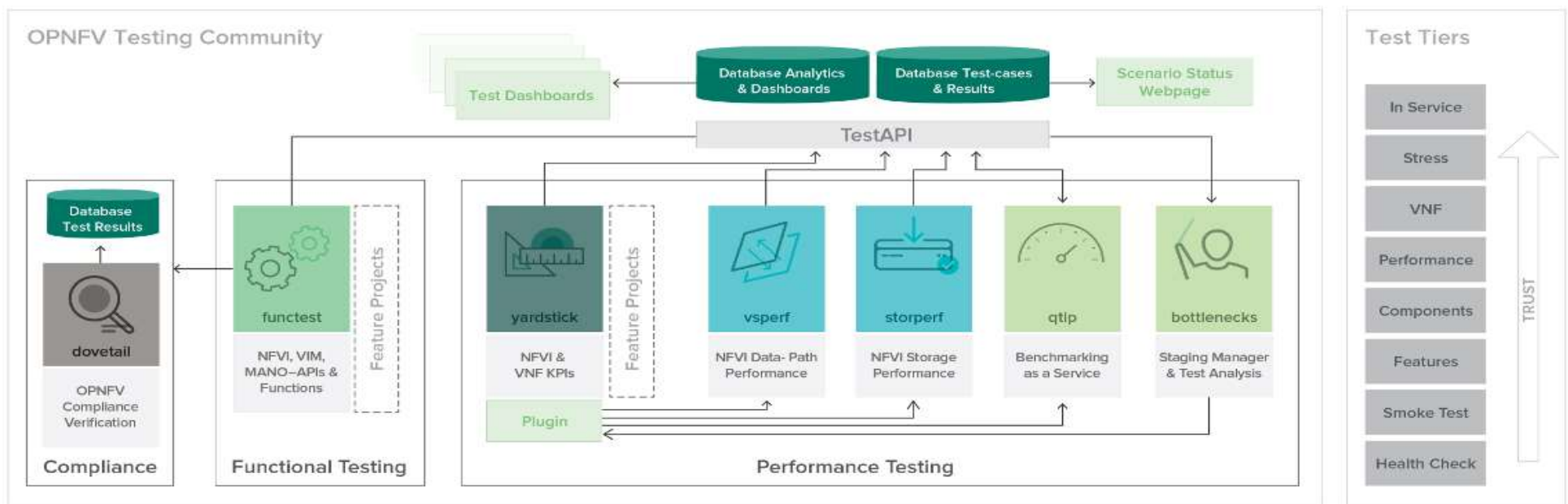
**Xtesting and Xtesting CI meet Requirements** for verification, compliance and certification:

- Assembly of multiple heterogeneous test cases
- OPNFV Release Engineering Jenkins jobs to verify RI
- Test case results & logs for third-party certification review
- Deploy local CI/CD toolchains to verify RI compliance

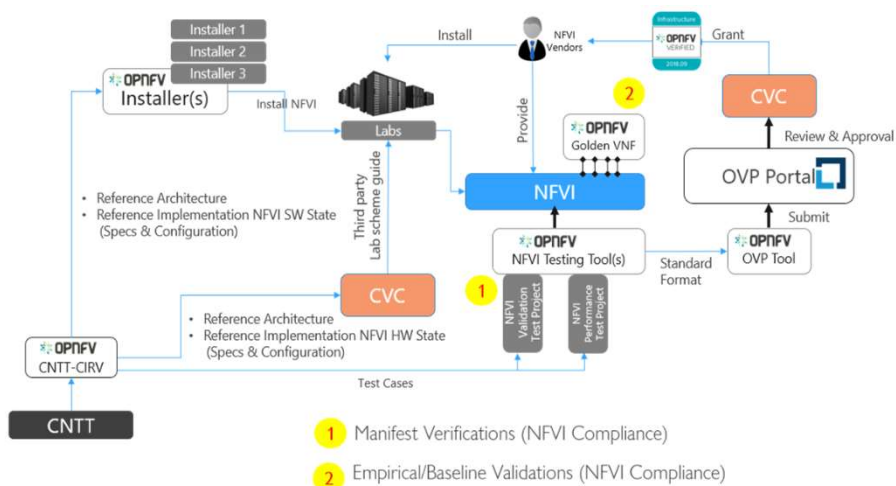
# Framework



- ✓ CI/CD approach utilizing common components and frameworks:
  - ✓ RI verification: centralized LF CI toolchain
  - ✓ RC compliance and cert: local LF-compatible CI toolchains
- ✓ Defining first framework requirements to simplify CI toolchain:
  - ✓ Common test case execution
  - ✓ United way to manage interactions with CI/CD components and with third-parties
- ✓ Xtesting was pragmatically selected:
  - ✓ Meets CNTT requirements
  - ✓ Allows easy adding any new test case
- ✓ The first test case requirements have been published:
  - ✓ Test cases delivered as Docker containers
  - ✓ Xtesting package and the related test case execution description file included



# Certification Process | Gap Analysis



## Process

### ➤ Certification based on successful delivery of:

- 1) **Manifest Verifications** – confirming NFVI delivered per RI-x requirements.
- 2) **Baseline Validations** – measuring FR/NFR behaviors using reference VNFs
- 3) **Interoperability Validations** – confirming capabilities, stability and perf

### ➤ For NFVI Certification:

- Vendor NFVI/VIM images under Test will be installed and configured
- For Phase 1, OPNFV **FuncTest** tests used for Compliance and Validation

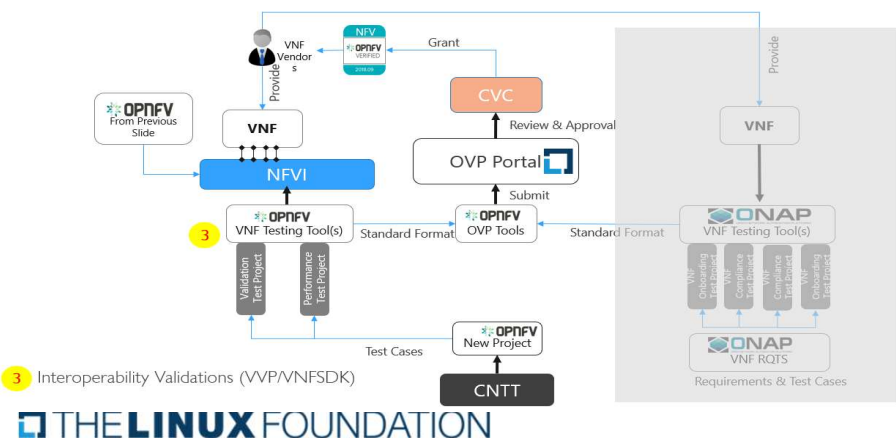
### ➤ For VNF Certification.:

- Vendor VNF images are ingested by the CICD pipeline, implemented in lab
- VNF on-boarding and lifecycle operations validation is performed using upstream projects such as VNFSDK and VPP

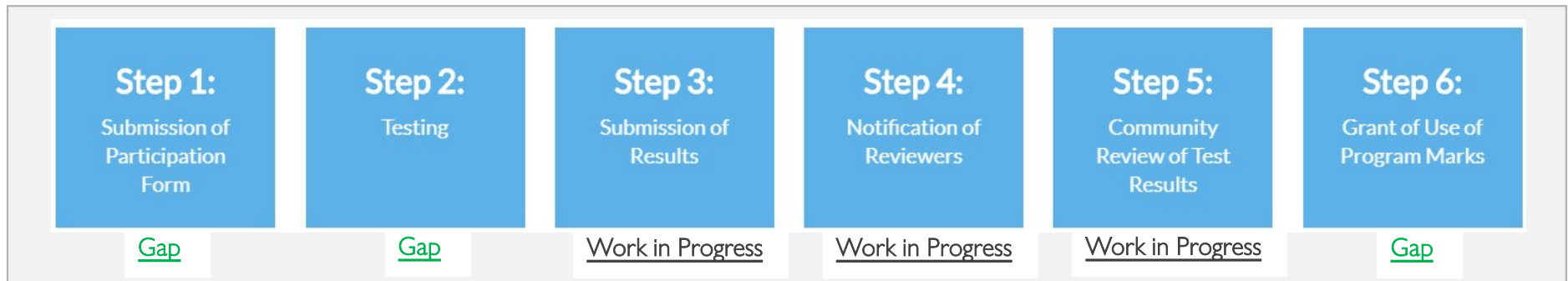
➤ Test results submitted to OVP for review by committee. If results are validated, the vendor's application for certification is approved and the badge(s) awarded.

## Gaps

- No automated means for Manifest (s/w) or Lab (h/w) Validations
- Need support vehicle for Installer changes (to meet RI-x specs)
- Normalization and Centralization of Results Alignment needed
- Storage and Performance scenarios (or tools) missing
- Need Installers which are OpenStack release agnostic
- Need priority on lab support with proactive monitoring



# Exceptions: Certifications From the OVP Process



OPNFV Verification Program (OVP) is a five step process resulting in the issuance of three badges: NFVI, VNF, and Lab



**CNTT seeks to align with the OVP process, noting three Gaps above in process:**

- **Step 1** for client (NFVI/VNF) participation in badging and certification
- **Step 2** for Test(ing) & tools utilized for testing
- **Step 6** for Badging Guidelines

**Mitigating Process Gaps:**

- **Step 1:** Friendly and Controlled Introductions in 2020, using key learnings to create formal participation mechanism
- **Step 2:** Adopt XTesting as OVP certified Platform Alternative, Expand Test Coverage onboarding/augmenting Test Projects
- **Step 6:** OVP accepts cookbook results as fulfillment for CNTT NFVI conformance/certification (all actors will run cookbook)

# Results: Compliance

## With RI-Alpha

- Running deployment verification and CNTT compliance
- Commands to mostly configure the daily jobs within minutes
- OVP test coverage increased to include 100% CNTT Compliance validations, forbidding skipping of mandatory tests
- Tests run for Compliance Verification include:
  - API testing benchmarking
  - API and data plane benchmarking
  - VNF deployment and testing



**Conclusion:** Successfully audited the RI System Under Tests (SUT), identifying installer changes to enable RI deployments, & confirming RA1 Chapter 5 (API) feature capability and exposure per OSH Ocata.

## Observations

- OpenStack Helm (OSH) doesn't support live migration and resize server for Ocata
- Compliance checks for Live Migration and Resize Servers blacklisted
- Metrics needed for API / data-plane benchmarking

## Recommendations (for RC-Alpha)

- Implement Stein, OSH supported
- Update RM/RA-1 OpenStack documentation for Stein
- Link RI gates to verification tests
- Leverage XTesting to wrap the RI deployment calls, simplifying the RI “cookbook”



# Next Steps

## Current Status

### Completed:

- RM | RA | RI Requirements
- Lab Requirements
- Initial Lab Secured
- S/W Deployed | Config
- Smoke Test | Sanity
- Continuous Deployment (with errors)

### In-Progress

- Manifest > PDF & IDF
- Complete Lab PoC & Deliver Lab
- Create Cookbook & RI Topology artifact
- Implement PoC Key Learnings

### Completed:

- Define & Vet Verification Methodology
- Stage Jenkins Hosts
- Prep Health/Smoke Suites
- Completed API test harness setup
- Initial API Compliance Validation

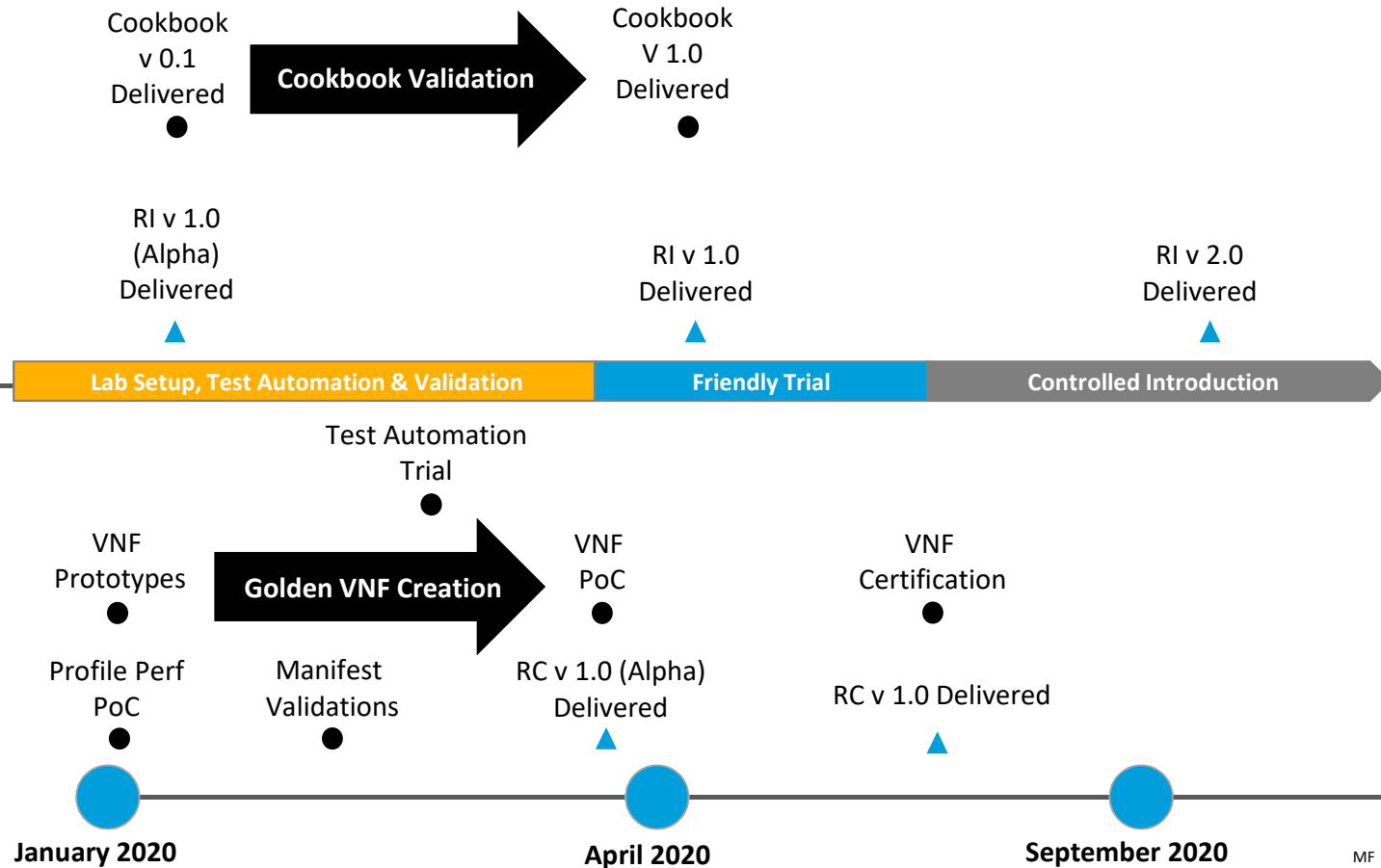
### In-Progress

- Design & deliver VNF Prototypes
- Integrate upstream community performance & storage test cases

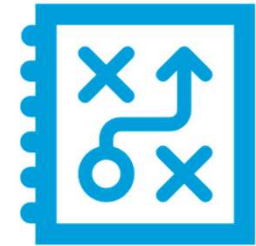
*Note: All events & dates are pending community alignment*

Reference Implementation

Reference Certification



# Reference Certification Challenges



- Availability of OVP | CNTT resources & active engagement
- **OVP's ability to scale to demand, & alignment with CNTT objectives, including:**
  - Fully automated ecosystem & badging process
  - Intuitive and efficient VNF on-boarding processes
  - Lab optimization & rationalization
  - Lab-use management & control strategy
    - Augmentation of OVP processes to include third party certification platforms & process
    - Clear and efficient third party certification requirements
  - Audits of lab certifications confirming availability, state (current), & accessibility
- Normalized test results with a centralized repository
- Manifest validations require an automated check of requirements

***CNTT will maintain ownership of the Reference Certification until a satisfactory level of support, stability, & maturity is attained***

# Appendix

 THE **LINUX** FOUNDATION



# Reference Certification Approach | Outcomes

## Actions Underway

## Outcomes



Define requirements & criteria for profile based implementations & certifications

Clearly defined requirements to assess vendor software, hardware, and VNF solutions



Define badging requirements to achieve certification

Deliver compliant & stable VNFs | CNFs to CSPs



Define, align, and select NFVI | VNF | CNF test case requirements, covering:

- Infrastructure
- Instantiation
- Tear-Down
- Performance
- Resiliency
- Security

Ensure certification criteria and badging requirements are met through an robust & optimized test suite



Establish guidelines and/or processes for:

- Entry | Exit Criteria
- Test Categories
- Quality Assurance
- Test Case In-Take

Ensure proper level of structure & discipline exists within test ecosystem to effectively manage & scale to demand




Define & align across communities on the test framework & tooling

Optimized test ecosystem, designed to scale, manage, & perform VNF | CNF certifications

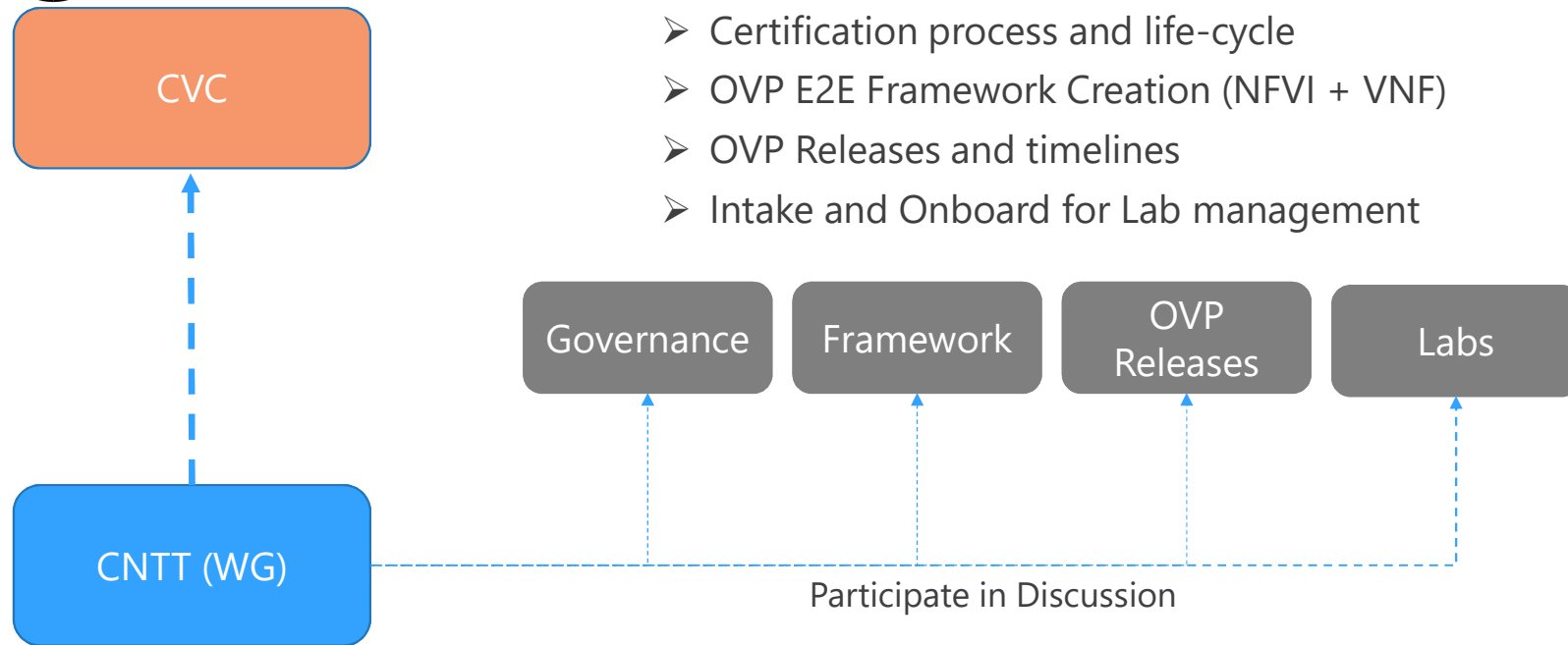
# Relationship with CVC



 Lincoln Lavoie

## What do we expect from CVC?

- Certification process and life-cycle
- OVP E2E Framework Creation (NFVI + VNF)
- OVP Releases and timelines
- Intake and Onboard for Lab management



- CNTT will work directly with CVC to align with governance
- Output of CNTT will be input to release scope, labs needs, and augment governance where needed

# Relationship with OPNFV and OVP



Bin Hu



## What do we expect from OPNFV?

- Installers to install NFVI with a state aligned with CNTT RM, RA.
- Test tools to test NFVI (against a given state) and VNFs.
- Provide test scripts to cover tests cases of CNTT interest.
- Leverage OVP Ecosystem for labs and certification.



Engineering Resources to accelerate development

- CNTT will work directly with OPNFV via the RI Project
- Output of CNTT-RI will be RI requirements and test cases



# Chapter 8 Team: North Star



## Mission

Ensure Implementation of CNTT Reference Model and Reference Architecture meets industry driven quality assurance standards for compliance, verification and validation.

---

## Objectives

- **Data Driven** RA Verification and Validations
- **OPNFV, CVC, and OVP Processes used** to onboard and check for NFVI compliance
- **Entry** and **Exit** Quality **Standards** are satisfied
- Ensure **test harnesses** can be **ported** and utilized **across multiple distributions**

---

## Guiding Tenets

- **Verification** and **Validations** determine NFVI+VNF compliance
- **Verification** signals conformance to design requirement specifications
- **Validations** signals compliance that output of a product meets the expected, or desired outcome
- **Certifications**, are out of scope as this measures adherence to development, however, no code is being delivered by testing
- OVP and CVC track and govern RM/RA verification



# Test Category / Case Gap Summary



## Projects Identified

- |                                     |                                     |
|-------------------------------------|-------------------------------------|
| <a href="#">* Airship Installer</a> | <a href="#">* High Availability</a> |
| <a href="#">Barometer</a>           | <a href="#">NFVBench</a>            |
| <a href="#">Bottlenecks</a>         | <a href="#">* Pharos</a>            |
| <a href="#">Doctor</a>              | <a href="#">SampleVNF</a>           |
| <a href="#">* Dovetail</a>          | <a href="#">VSPerf</a>              |
| <a href="#">* Fuel</a>              | <a href="#">Yardstick</a>           |
| <a href="#">FuncTest</a>            |                                     |

- # Total OPNFV Projects = 31
- # CNTT-NFVI = 13 (potential value)
- # 5/13 NA for Review – \*Already Covered by Yardstick and Functest

## Results

- # 6 Projects can be adopted (as is)
- # 4 We can add/augment TCs for gaps
- # 3 Create projects for new testing

 THE **LINUX** FOUNDATION

MF

## Assessment Strategy

- Select Project by activity, use, and maturity state
- Compare Against Test Categories
- Identify Gaps
- Form Professional Opinion – e.g. augment, adopt
- Solicit Strategic Partner Contributions

## Test Categories

- **(Hardware Validations)** BareMetal – HW & O/S validations
- **(Component Validation and VNF Validation Config Only)** VNF Interoperability – validations
- **(Platform Stability)** Compute Component – validations
- **(Platform Resiliency)** Control Plane Component – validations

## Next Steps

- **Review Projects** Identified during Antwerp not considered
- **Integrate TCs** from these projects into the delivery stream
- **Discuss augmenting existing test projects**, or create new
- **Onboard Spirent Test Case contributions** – **Where?**
- **Finalize** Test-/Use- Case Needs



# Test Category / Case Gap Summary.. Continued



Project	Purpose	Recommendation
<b>FuncTest</b>	<ul style="list-style-type: none"> <li>Functional interoperability validations</li> </ul>	<ul style="list-style-type: none"> <li>✓ <b>ADOPT</b>, as an RI suite. <a href="#">Covers 2k+ Openstack Interoperability Validations</a></li> <li>✓ <b>Augment</b> to include Baremetal testing for Manifest Validations</li> </ul>
<b>Yardstick</b>	<ul style="list-style-type: none"> <li>VNF/Payload performance validations</li> </ul>	<ul style="list-style-type: none"> <li>✓ <b>ADOPT</b>, 62 TCs, leverages Shaker and YAML for test-case development</li> <li>✓ <b>Augment</b> to perform POD restarts and HA for Maria/Ceph restarts</li> </ul>
<b>VSPerf</b>	<ul style="list-style-type: none"> <li>vSwitch perf testing</li> </ul>	<ul style="list-style-type: none"> <li>✓ <b>ADOPT</b>, for OVS-DPDK validations with 32 perf and functional TCs</li> <li>✓ <b>Setup</b> external packet generator to avoid latency caused by the tool.</li> </ul>
<b>DoveTail</b>	<ul style="list-style-type: none"> <li>Automation framework</li> </ul>	<ul style="list-style-type: none"> <li>✓ <b>ADOPT</b>, with large number of test cases for conformance evaluation</li> </ul>
<b>Barometer</b>	<ul style="list-style-type: none"> <li>Platform availability and NW usage validations</li> </ul>	<ul style="list-style-type: none"> <li>✓ <b>ADOPT</b>, for use of NFVI+VNF validations capturing Telemetry data</li> <li>✓ <b>Augment</b> to include device specific resiliency testing and monitoring.</li> </ul>
"NEW"	<ul style="list-style-type: none"> <li><b>Baremetal Validations</b></li> </ul>	<ul style="list-style-type: none"> <li>✓ <b>CREATE New</b> Baremetal Validations to verify engineering packages</li> </ul>
"Augment"	<ul style="list-style-type: none"> <li><b>Spirent Validations</b></li> </ul>	<ul style="list-style-type: none"> <li>✓ <b>Augment</b> projects with 240 TC adds for load, scaling, cloud migration.</li> </ul>
"NEW"	<ul style="list-style-type: none"> <li><b>Chaos Toolkit</b></li> </ul>	<ul style="list-style-type: none"> <li>✓ <b>CREATE New</b>, project to test POD resiliency by injecting chaos (failover)</li> </ul>
<b>Bottlenecks</b>	<ul style="list-style-type: none"> <li>Stress Testing</li> </ul>	<ul style="list-style-type: none"> <li>✗ <b>Not recommended</b> with limited test sets and results categorization</li> </ul>
<b>Doctor</b>	<ul style="list-style-type: none"> <li>Computer NFVI Fault Mgmt validations</li> </ul>	<ul style="list-style-type: none"> <li>✗ <b>Not recommended</b> with limited/no coverage for SDN, KVM, or containers</li> </ul>
<b>XTesting</b>	<ul style="list-style-type: none"> <li>CICD tool chaining in CNTT validations.</li> </ul>	<ul style="list-style-type: none"> <li>? <b>REQUIRES POC</b> if CNTT NFVI requires ADOPTION and USE of tool chaining.</li> <li>? <b>There are no specific TCs</b>, as XTesting is for chaining together CICD test projects, and not for NFVI validation.</li> </ul>
<b>NFVBench</b> MF	<ul style="list-style-type: none"> <li>NFVI Perf Measurements (at physical hardware/host level)</li> </ul>	<ul style="list-style-type: none"> <li>✓ <b>ADOPT</b>, as a complement to vsperf and yardstick</li> <li>✓ <b>Augment</b> to expand SRIOV and/or OVS-DPDK test cases.</li> </ul>