# Common NFVI Telco Taskforce Antwerp Face-To-Face Sessions

Fu Qiao, China Mobile
Mark Shostak, AT&T
Mike Fix, AT&T
Lincoln Lavoie, UNH-IOL

Test & Validation

**September 2019**

THE LINUX FOUNDATION

GSMA

# Discussion Outline*

**CVC Overview**

- Structure within LFN
- Philosophy & Terms
- OVP Roadmap & Deliverables

**CNTT Relationship with CVC, OPNFV, and OVP**

**Chapter 8 Team**

- North Star & Scope
- Team Progress & Table of Contents

**Methodology & Goals**

**OVP ETE Framework** (Intake & Requirements)

**Entrance & Exit Criteria**

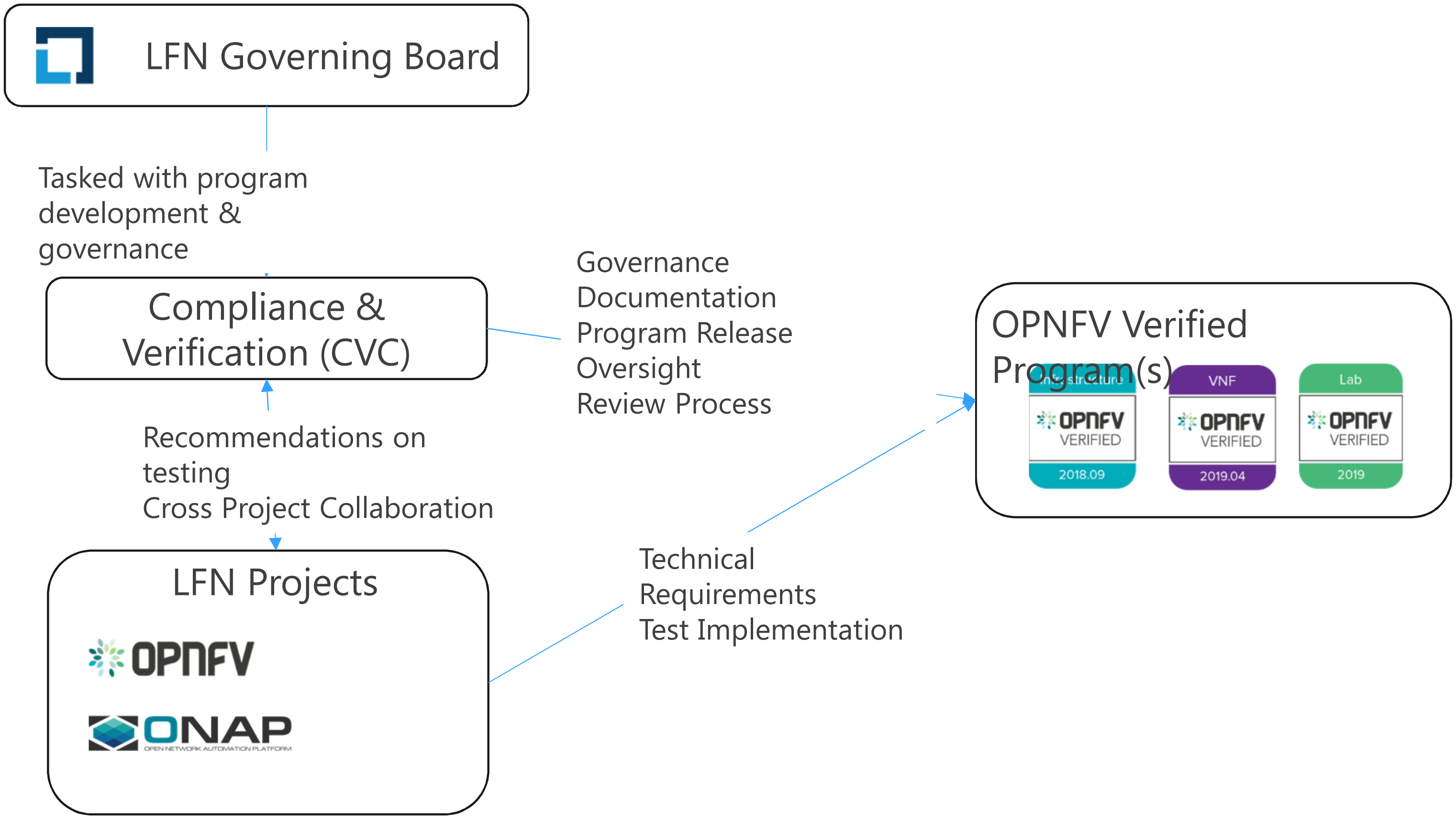
**Test Category/Case Gap Review**

Infrastructure
OPNFV
VERIFIED
2018.09

VNF
OPNFV
VERIFIED
2019.04

***Challenges:*** *Req Gathering, Test Case Traceability, Automated Validation, Normalized Results, …?*

# CVC Structure Within LFN



LFN Governing Board

Tasked with program development & governance

Compliance & Verification (CVC)

Governance
Documentation
Program Release
Oversight
Review Process

Recommendations on testing
Cross Project Collaboration

LFN Projects

Technical
Requirements
Test Implementation

OPNFV Verified Program(s)

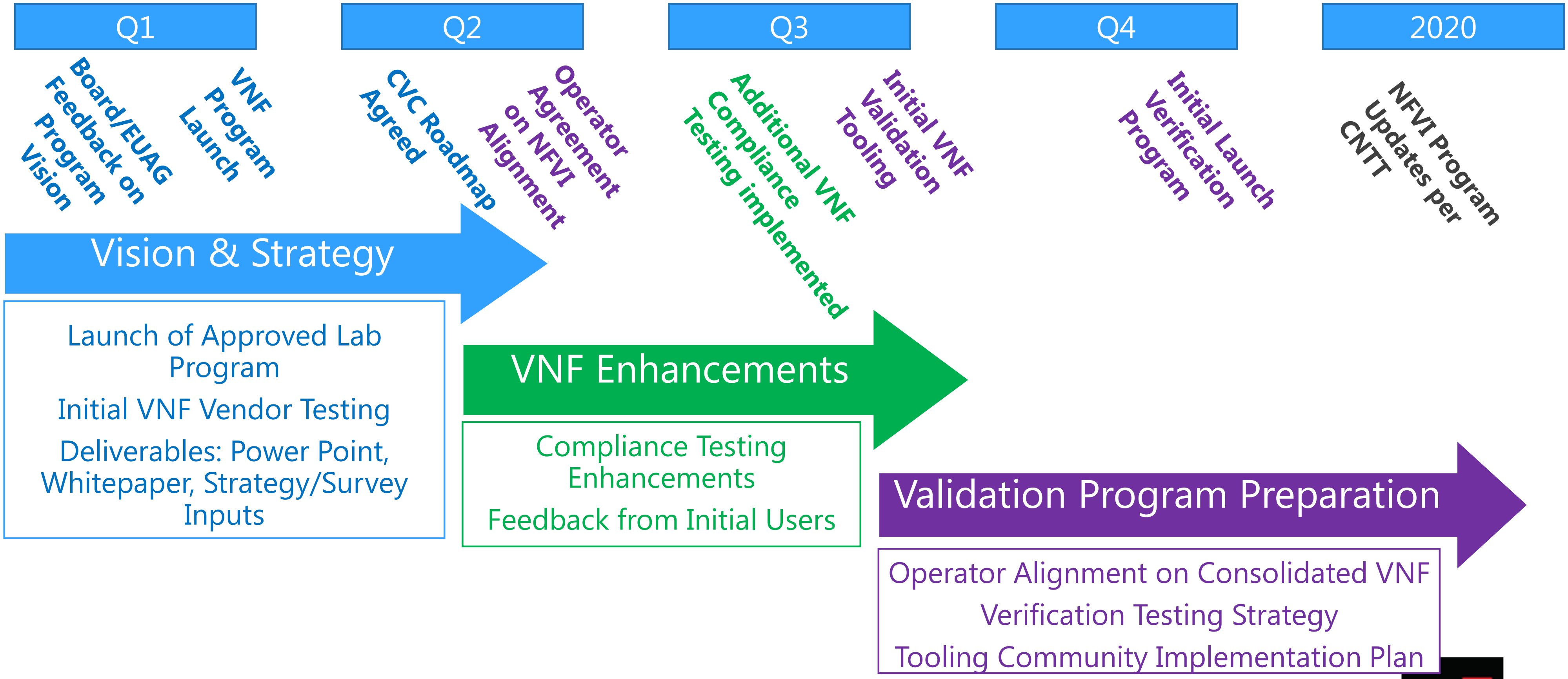
# Program Philosophy

- Open program, well aligned with open source best practices
  - Support self-testing and 3rd party lab testing
- Reliance on community review processes
  - Requirements, Test Implementation, and Results
- Uniform approach across projects
  - Single release schedule for badges / updates
  - Consistent definition of test types and badges

# Some Terms

- Compliance – Testing to ensure product meets requirements defined by the project
  - API testing for correct implementation, format, responses, etc.
  - VNF Template testing per requirements defined by ONAP
- Validation – Testing to ensure product operation meets requirements
  - Testing of API control over the larger system, i.e. use API to create network with expected access controls, etc.
- Performance – Testing to measure the capability of the product
  - Testing of traffic throughput on a VNF to meet a minimum requirement
  - Testing the minimum number of sessions supported by a VNF
- Should we consider formalizing additional terms?
  - Stability, on-boarding, interoperability, etc.

# OVP Roadmap (2019)



Q1

Board/EUAG Feedback on Program Vision

VNF Program Launch

Q2

CVC Roadmap Agreed

Operator Agreement on NFVI Alignment

Q3

Additional VNF Compliance Testing implemented

Initial VNF Validation Tooling

Q4

Initial Launch Verification Program

2020

NFVI Program Updates per CNTT

Vision & Strategy

Launch of Approved Lab Program

Initial VNF Vendor Testing

Deliverables: Power Point, Whitepaper, Strategy/Survey Inputs

VNF Enhancements

Compliance Testing Enhancements

Feedback from Initial Users

Validation Program Preparation

Operator Alignment on Consolidated VNF Verification Testing Strategy

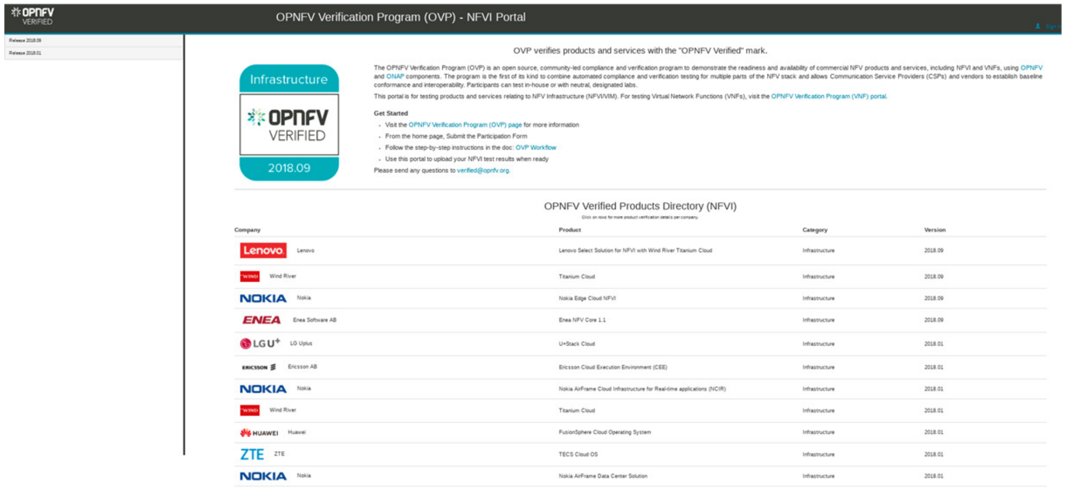
Tooling Community Implementation Plan

THE LINUX FOUNDATION

GSMA

# Program Deliverables & Components

- Test Requirements
- Test Definitions & Implementation
- End User Documentation
- Updates to OVP Portal
- Beta Testing



OPNFV Verification Program (OVP) - NFVI Portal

OVP verifies products and services with the "OPNFV Verified" mark.

The OPNFV Verification Program (OVP) is an open source, community-led compliance and verification program to demonstrate the readiness and availability of commercial NFV products and services, including NFVI and VNFs, using OPNFV and ONAP components. The program is the first of its kind to combine automated compliance and verification testing for multiple parts of the NFV stack and allows Communication Service Providers (CSPs) and vendors to establish baseline conformance and interoperability. Participants can test in-house or with neutral, designated labs.

This portal is for testing products and services relating to NFV Infrastructure (NFVI/VIM). For testing Virtual Network Functions (VNFs), visit the OPNFV Verification Program (VNF) portal.

**Get Started**

- Visit the OPNFV Verification Program (OVP) page for more information
- From the home page, Submit the Participation Form
- Follow the step-by-step instructions in the doc: OVP Workflow
- Use this portal to upload your NFVI test results when ready

Please send any questions to verified@opnfv.org.

OPNFV Verified Products Directory (NFVI)

Click on rows for more product verification details per company.

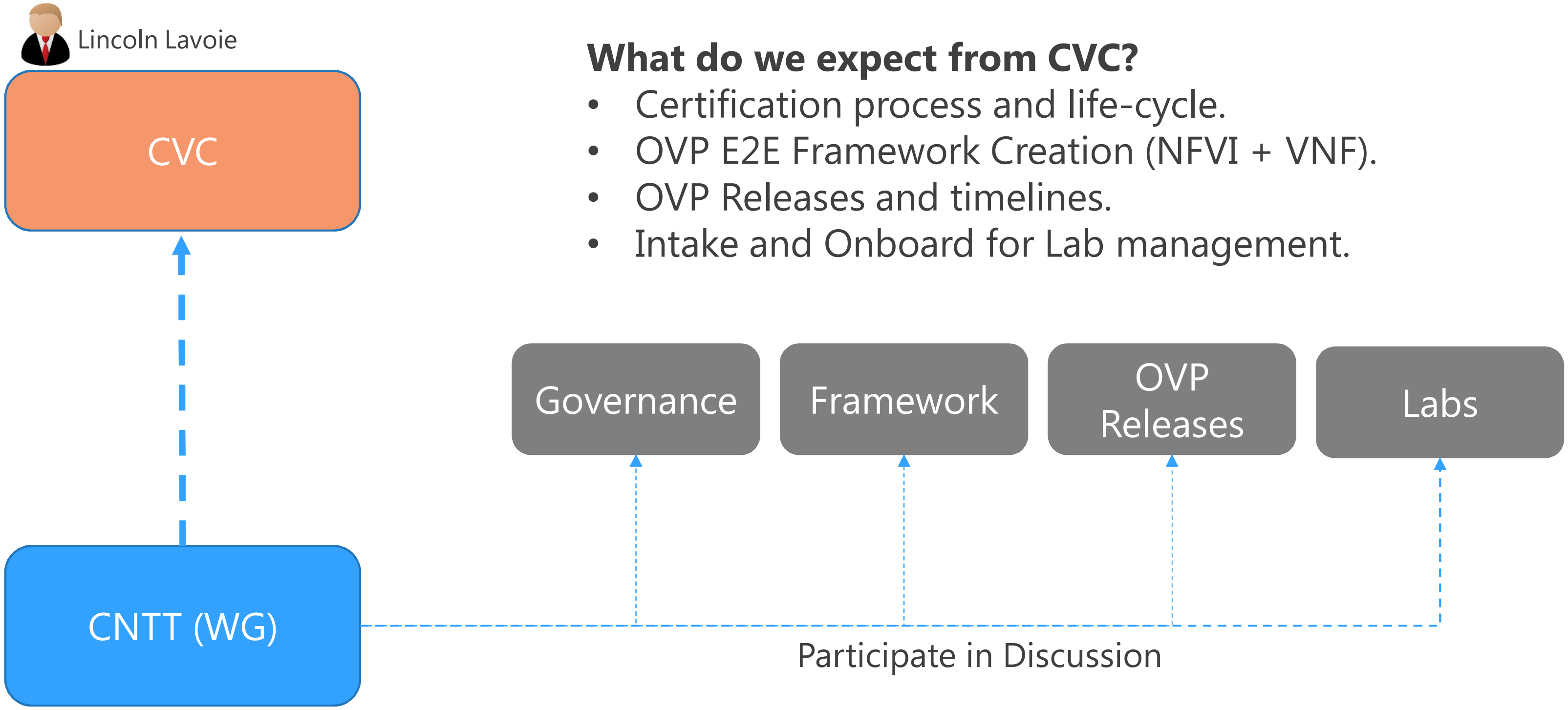
| Company | Product | Category | Version |
|---|---|---|---|
| Lenovo | Lenovo Select Solution for NFVI with Wind River Titanium Cloud | Infrastructure | 2018.09 |
| Wind River | Titanium Cloud | Infrastructure | 2018.09 |
| Nokia | Nokia Edge Cloud NFVI | Infrastructure | 2018.09 |
| Enea Software AB | Enea NFV Core 1.1 | Infrastructure | 2018.09 |
| LG Uplus | U+Stack Cloud | Infrastructure | 2018.01 |
| Ericsson AB | Ericsson Cloud Execution Environment (CEE) | Infrastructure | 2018.01 |
| Nokia | Nokia AirFrame Cloud Infrastructure for Real-time applications (NCIR) | Infrastructure | 2018.01 |
| Wind River | Titanium Cloud | Infrastructure | 2018.01 |
| Huawei | FusionSphere Cloud Operating System | Infrastructure | 2018.01 |
| ZTE | TECS Cloud OS | Infrastructure | 2018.01 |
| Nokia | Nokia AirFrame Data Center Solution | Infrastructure | 2018.01 |

LL

# Relationship with CVC

**What do we expect from CVC?**

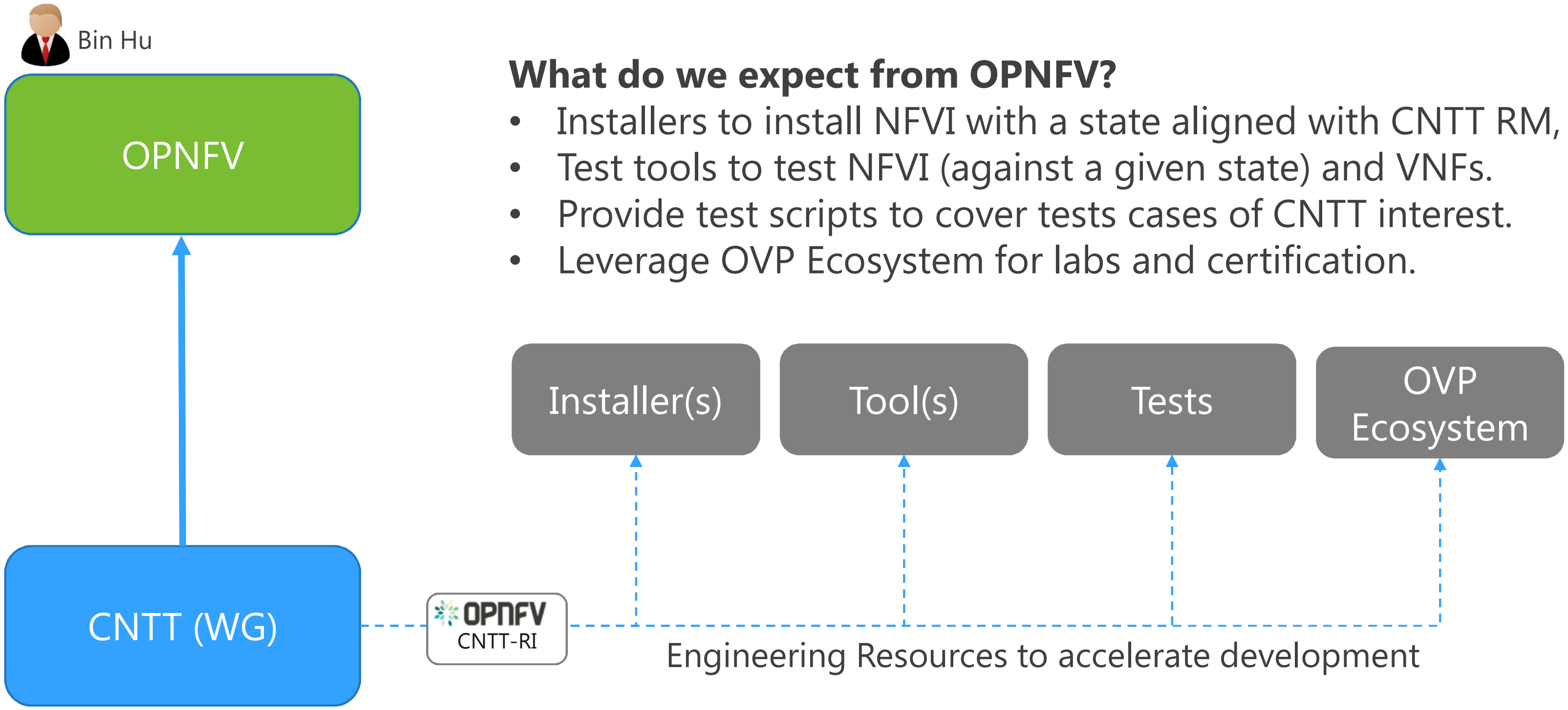
- Certification process and life-cycle.
- OVP E2E Framework Creation (NFVI + VNF).
- OVP Releases and timelines.
- Intake and Onboard for Lab management.



- 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

MF

# Relationship with OPNFV and OVP



**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.

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

MF

# 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

# Scope

## 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

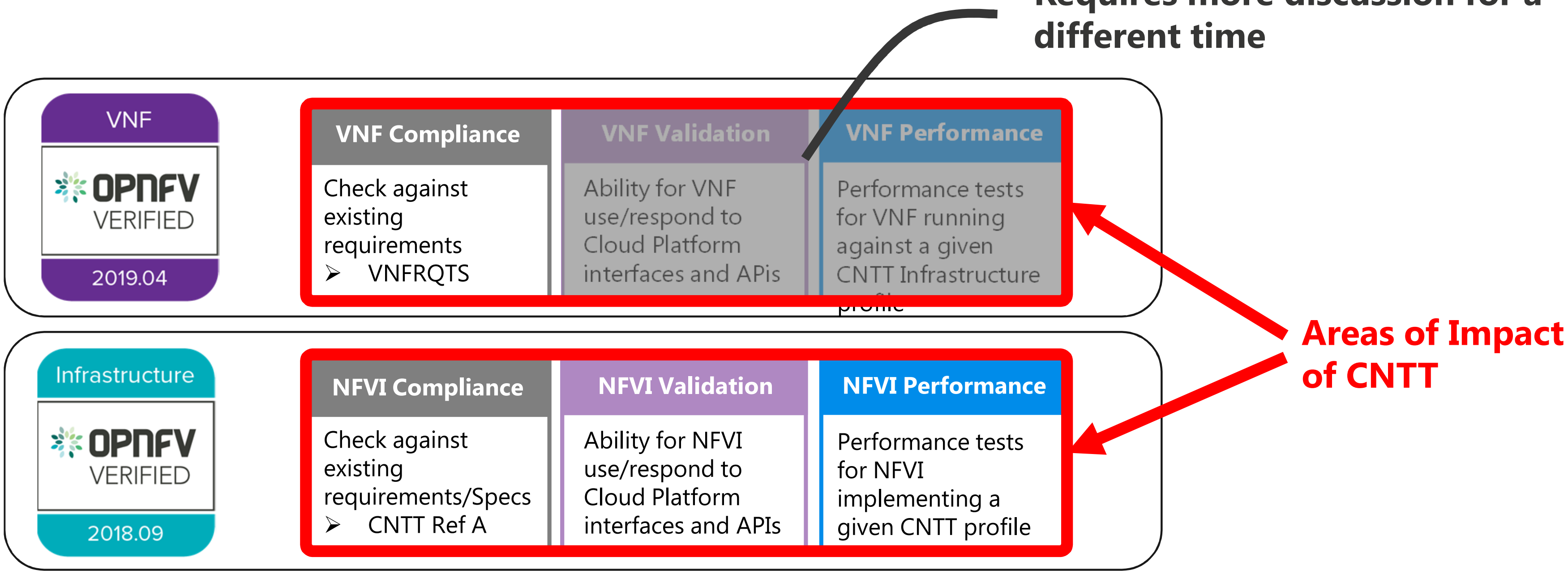
## Not In Scope

- VNF functional testing
- ONAP as a MANO for VNFs
- Validating VNF’s ability to be upgraded
- Georedundant and Load Testing

## Different Distributions

**Repeat** the strategy of Manifest Verification, Empirical Golden VNF Validations, and Interoperability Testing for any new Distributions.

# Verification & Validation Scope



Requires more discussion for a different time

| VNF OPNFV VERIFIED 2019.04 | VNF Compliance | VNF Validation | VNF Performance |
|---|---|---|---|
| | Check against existing requirements<br>➢ VNFRQTS | Ability for VNF use/respond to Cloud Platform interfaces and APIs | Performance tests for VNF running against a given CNTT Infrastructure profile |

| Infrastructure OPNFV VERIFIED 2018.09 | NFVI Compliance | NFVI Validation | NFVI Performance |
|---|---|---|---|
| | Check against existing requirements/Specs<br>➢ CNTT Ref A | Ability for NFVI use/respond to Cloud Platform interfaces and APIs | Performance tests for NFVI implementing a given CNTT profile |

Areas of Impact of CNTT

**Example Tests**

- VNF Interoperability Testing
- Security Testing
- Scalability Testing
- Fault Recovery Testing
- VNF Coexistence
- HA Testing

THE LINUX FOUNDATION

GSMA

MF

# Team Progress

9/19/2019

## TC Gap Assessment

- ✓ Deep-dive of OPNFV, CVC, and OVP Process
- ✓ Initial pass assessing Key Active OPNFV Projects for CNTT alignment
- ✓ Normalize TC Review results format
- ◐ PTLs provide Test Suite data
- ✓ Initial Test Category/Case Review

## Partnerships & Communications

- ✓ Stakeholder Verizon as co-author of Verification Process
- ✓ Launch PR Campaign with the OPNFV TSC and CVC
- ◐ Continued CNTT>OPNFV>CVC alignment discussions
- ✓ OPNFV Proposal Review 9/9-9/16
- ✓ OPNFV TSC Project Vote 9/17

## Setup Lab

- ◐ Finalize RM/RA/RI requirements
- ✓ Identify initial lab hardware needs
- (!) Configure SUT, & Execute Sanity
- (!) Identify and Close Gaps in TCs

## Documentation

- ✓ Outline & Initial Merge Complete
- ◕ Working GitHub Issues
- ◐ Define & vet Verification Methodology
- ◐ Create Annex, Developing Content
- (!) Finalize Ch 8 Content & Publish
- (!) Finalize RI Content

THE LINUX FOUNDATION

MF

# Chapter 8 - Table of Contents

## Synopsis

- ✓ **Introduction** - Overview, problem statement, scope
- ✓ **Principles and Guidelines** – Details on objectives, verification methodologies, and governance
- ✓ **Terms and Resources** – Common terms and external documentation

## Process & Management

- ◔ **Lifecycle and Process Flow** – Project management guidelines, onboarding, SLAs and Issue Resolution
- ✓ **Current OVP/CVC Process** – Existing process, test frameworks, tools, test cases/scenarios and test certification guidelines

## Verification & Validation Strategy

- ◕ **CNTT/NFVI Validation Approach** – Augmented OPNFV and OVP certification process using NFVI Verification, Empirical Validation, and VNF Interoperability Validation
- ◕ **Quality Assurance** – Dependencies, Recommendations, Assumptions, System Under Test (SUT) pre-reqs for certification, Entrance/Exit Criteria, Test Frameworks, Categories, Harness(es), and Tools
- ◔ **Test Results** – Metrics, Measurements, and Respective Certifications and Badges – e.g. pass/fail, measure only, etc

## Forward Looking

- (!) **Future Planning** – Additional considerations, documentation, lab mgmt, tools, or test strategies (GeoRed, DR)
- (!) **Recommendations** – Best practices (placeholder) after initial implementation of RA#1

# Methodology & Goals

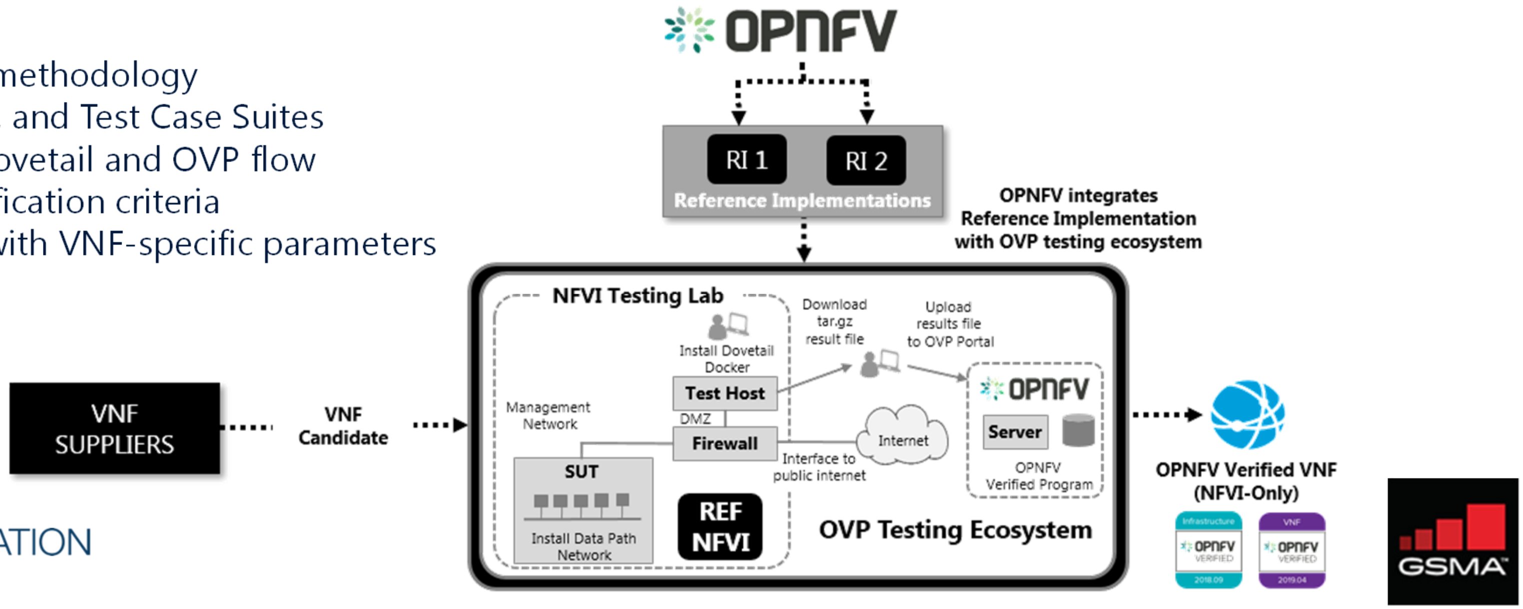
**Methodology:** Perform NFVI validations using CNTT reference architecture, leveraging upstream projects to define features/capabilities, test scenarios, and test cases, to be executed via the OVP Ecosystem.

**OVP/CVC Validation Strategy & Vehicle:**

- **NFVI Verification (*Compliance*)**: NFVI is the SUT, ensuring NFVI is compliant with specs of RM and RA
- **Empirical Validation with Reference VNF (*Validation*)**: NFVI is the SUT, ensuring NFVI runs with Golden VNFs
- **Candidate VNF Validation (*Validation & Performance*):** VNF is the SUT, ensuring VNFs operate with RM and RA
- **Security:** Ensures VNF is free from known security vulnerabilities, utilizing industry standard cyber security frameworks

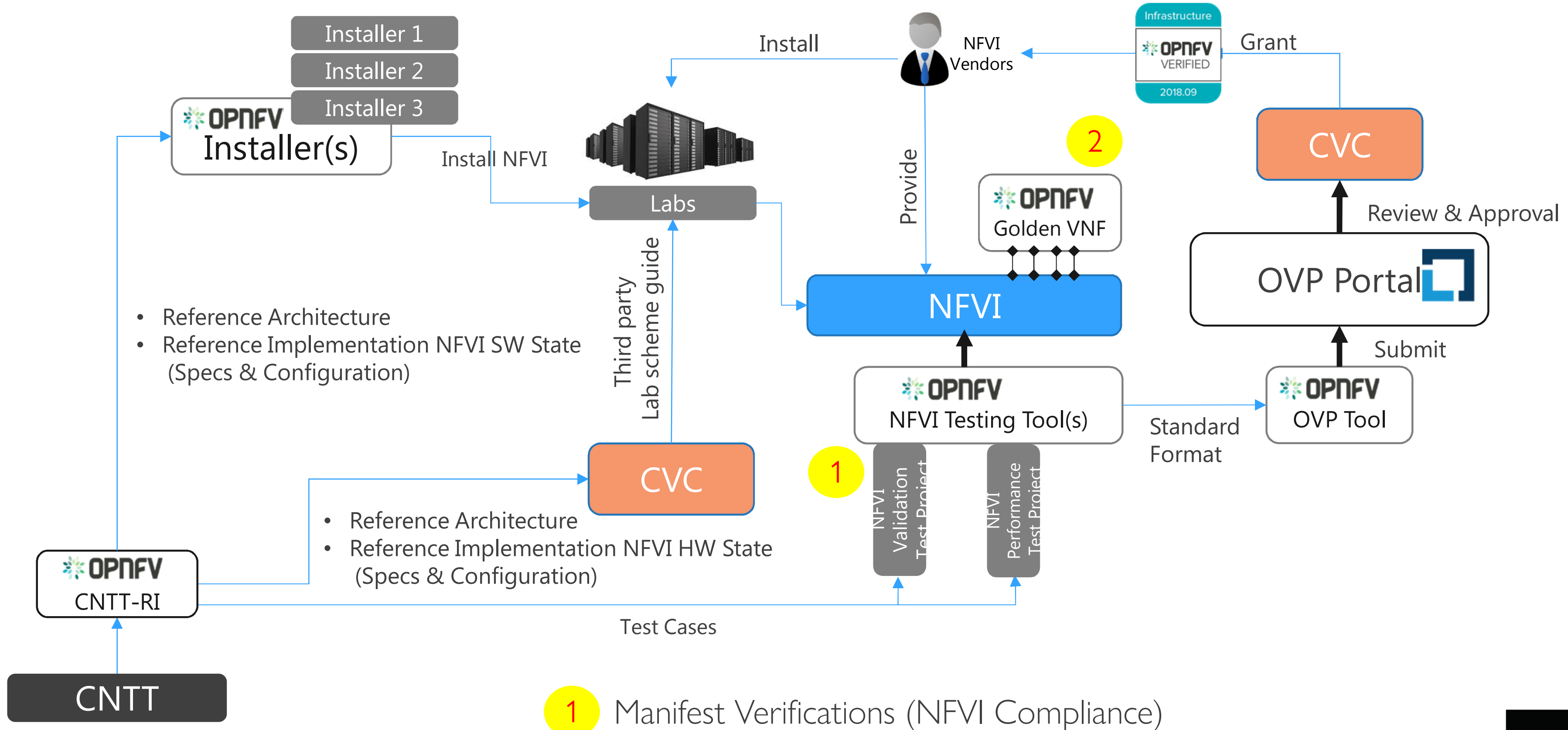
**Best Practices:**

- Standardized test methodology
- Standard Test Plan, and Test Case Suites
- Integration with Dovetail and OVP flow
- Standardized certification criteria
- Leverage models with VNF-specific parameters


OPNFV
RI 1
RI 2
Reference Implementations
OPNFV integrates Reference Implementation with OVP testing ecosystem
VNF SUPPLIERS
VNF Candidate
NFVI Testing Lab
Install Dovetail Docker
Test Host
DMZ
Management Network
Firewall
SUT
Install Data Path Network
REF NFVI
Download tar.gz result file
Upload results file to OVP Portal
Internet
Interface to public internet
OPNFV
Server
OPNFV Verified Program
OVP Testing Ecosystem
OPNFV Verified VNF (NFVI-Only)
Infrastructure OPNFV VERIFIED 2018.09
VNF OPNFV VERIFIED 2019.04
GSMA

THE LINUX FOUNDATION

# OVP Framework – NFVI Validation

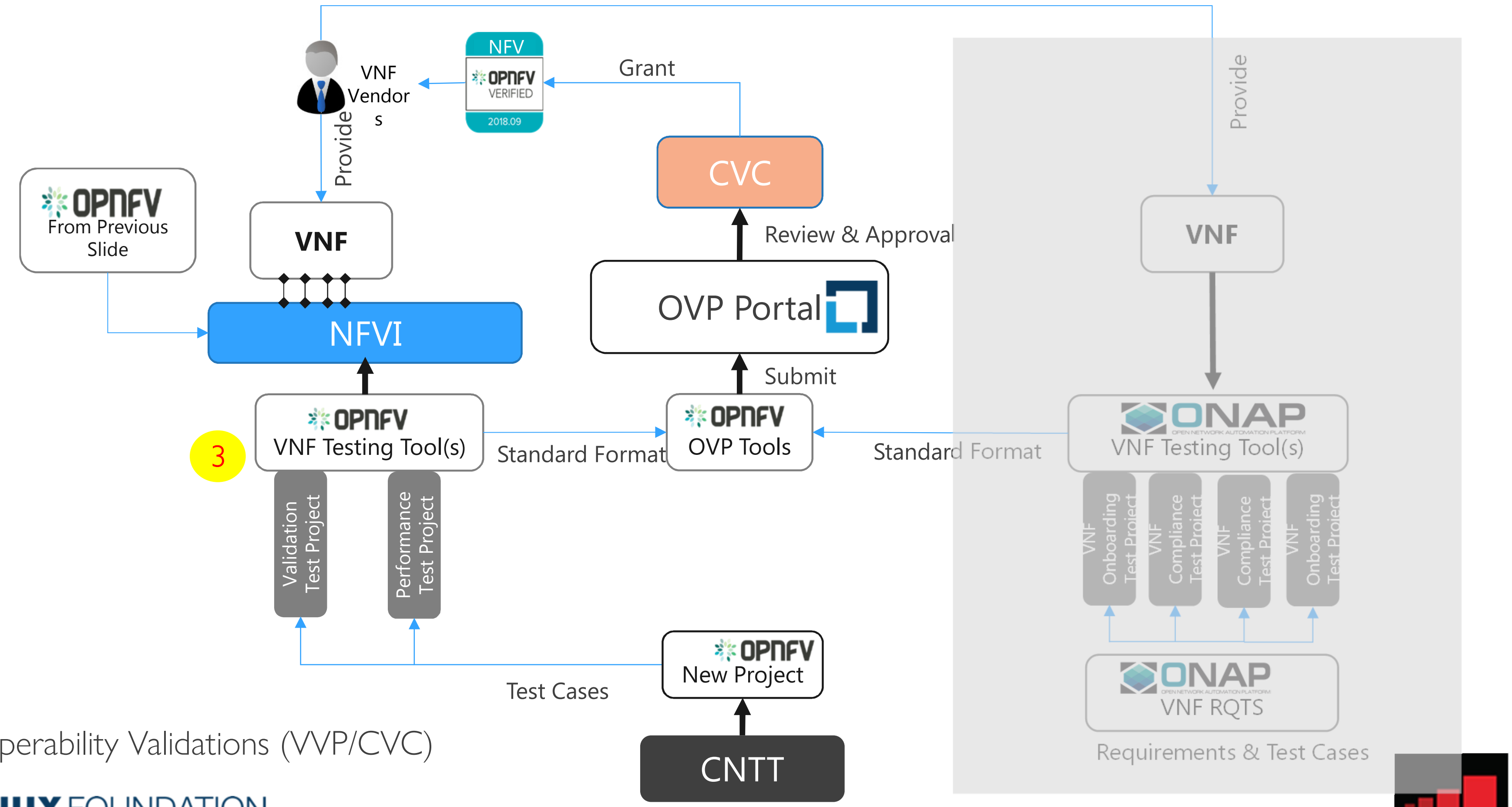

Installer 1
Installer 2
Installer 3
OPNFV Installer(s)
Install NFVI
Install
NFVI Vendors
Infrastructure OPNFV VERIFIED 2018.09
Grant
CVC
Labs
Provide
2
OPNFV Golden VNF
NFVI
Review & Approval
OVP Portal
Submit
Third party Lab scheme guide
Reference Architecture
Reference Implementation NFVI SW State (Specs & Configuration)
OPNFV NFVI Testing Tool(s)
Standard Format
OPNFV OVP Tool
CVC
1
NFVI Validation Test Project
NFVI Performance Test Project
Reference Architecture
Reference Implementation NFVI HW State (Specs & Configuration)
OPNFV CNTT-RI
Test Cases
CNTT

1 Manifest Verifications (NFVI Compliance)

2 Empirical/Baseline Validations (NFVI Compliance)

FQ

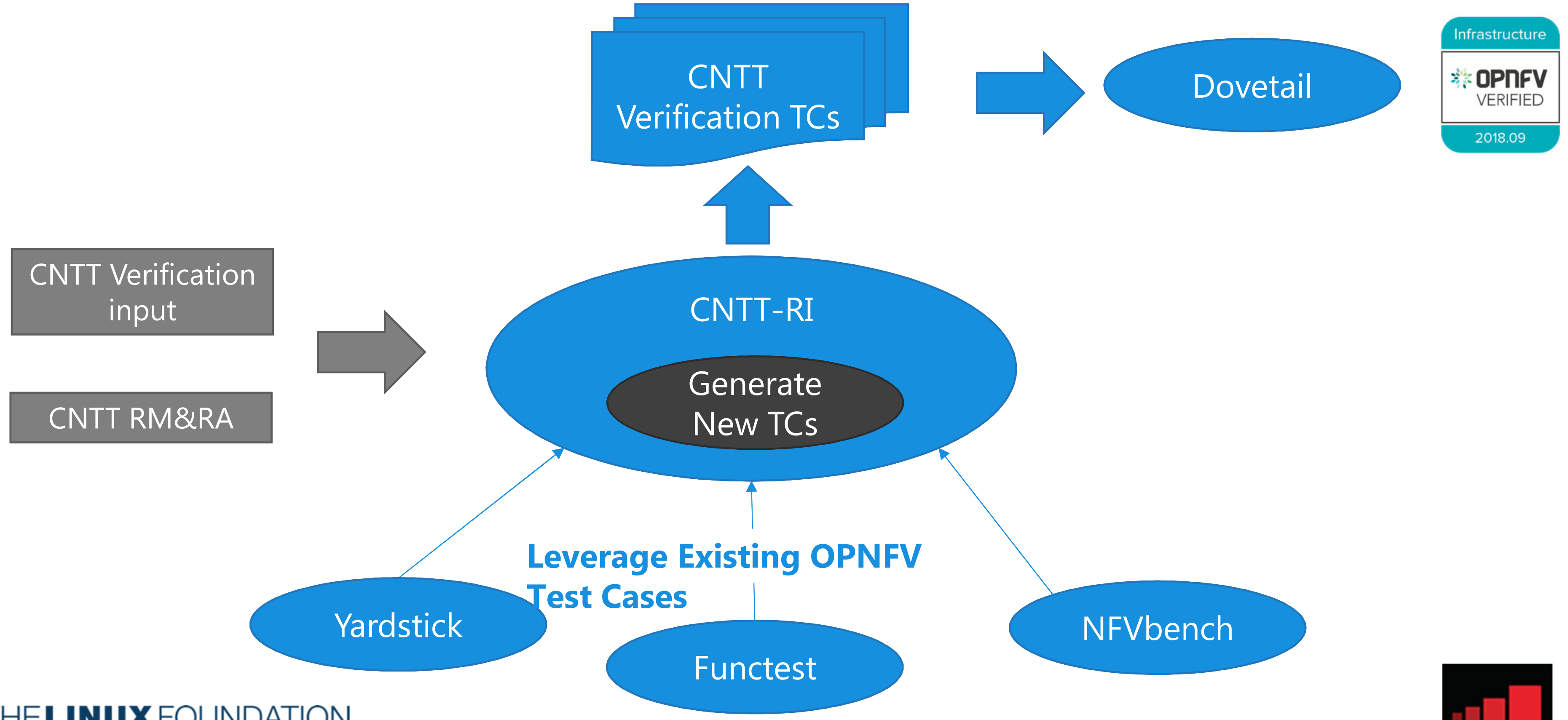
# OVP Framework – NFVI+VNF Validation



3 Interoperability Validations (VVP/CVC)

FQ

# Enhance Intake into OVP Framework



FQ

# OPNFV/OVP Entry & Exit Criteria

## Prior to Testing – VNF Supplier Provided

***OPNFV Entrance criteria must be satisfied before testing starts.***

***This demonstrates implementation of CNTT Reference Architecture.***

### Entrance Criteria

- **Design Details Provided**
  - High/Low Level Design (Config, Features)
- **Environment** (document, secured, connected)
- **Test Requirements**
- Testing schedule
- Completed security review
- Test owners documented
- **Test assets available**
  - Images, configurations, templates, etc
- Additional vendor-specific test cases documented & supplied

## Prior to Telco Handoff – VNF Supplier Verified

***OVP Exit criteria must be satisfied to receive CVC Validation Badges.***

***Validation demonstrates adherence to CNTT NFVI Quality Standards.***

### Exit Criteria

- **All test cases have a valid status**
- **No outstanding high severity issues**
- Known defects and outstanding issues are clearly documented
- **Operationally Stable and Functional**
  - API end-points reachable/working
  - Standard Images present, and operational
  - Snapshots and backups working
  - NFV migration is confirmed working
- **Documentation is available**
  - Deploy, config, admin, user, API guides
  - Release notes

MF

# Test Category / Case Gap Summary

## Projects Identified

- * Airship Installer
- Barometer
- Bottlenecks
- Doctor
- * Dovetail
- * Fuel
- FuncTest
- * High Availability
- NFVBench
- * Pharos
- SampleVNF
- VSPerf
- Yardstick

- **# 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

## 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

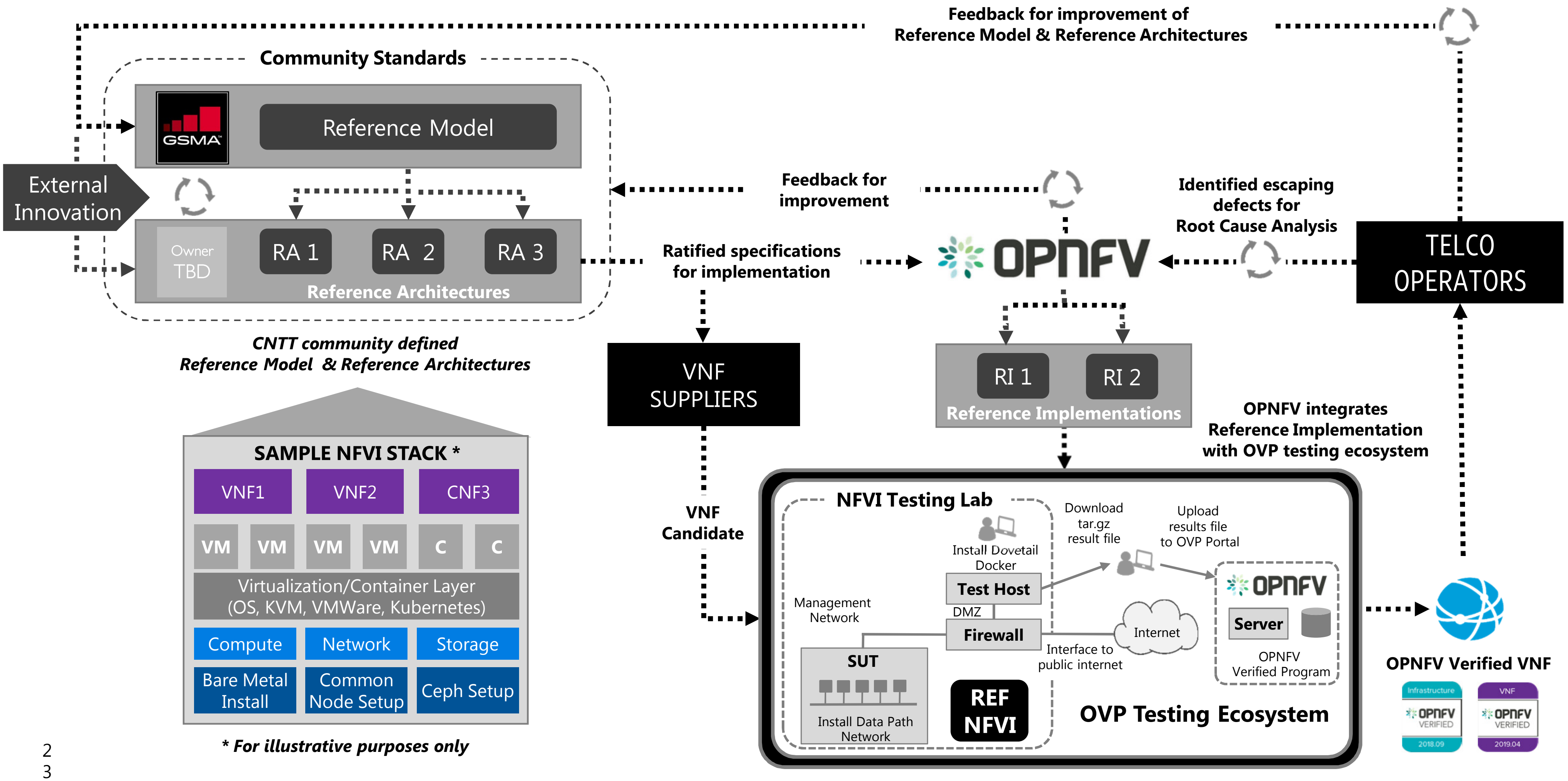
- **Review Projects** Identified during Antwerp not considered
- **Integrate TCs** from these projects into the delivery stream
- **Discuss augmenting existing test project**s, or create new
- **Onboard Spirent Test Case contributions** – **Where?**
- **Finalize** Test-/Use- Case Needs

# Test Category / Case Gap Summary.. Continued

| Project | Purpose | Recommendation |
|---|---|---|
| **FuncTest** | ▪ Functional interoperability validations | ✓ **ADOPT,** as an RI suite. Covers 2k+ Openstack Interoperability Validations<br>✓ **Augment** to include Baremetal testing for Manifest Validations |
| **Yardstick** | ▪ VNF/Payload performance validations | ✓ **ADOPT,** 62 TCs, leverages Shaker and YAML for test-case development<br>✓ **Augment** to perform POD restarts and HA for Maria/Ceph restarts |
| **VSPerf** | ▪ vSwitch perf testing | ✓ **ADOPT,** for OVS-DPDK validations with 32 perf and functional TCs<br>✓ **Setup** external packet generator to avoid latency caused by the tool. |
| **DoveTail** | ▪ Automation framework | ✓ **ADOPT,** with large number of test cases for conformance evaluation |
| **Barometer** | ▪ Platform availability and NW usage validations | ✓ **ADOPT,** for use of NFVI+VNF validations capturing Telemetry data<br>✓ **Augment** to include device specific resiliency testing and monitoring. |
| "**NEW**" | ▪ **Baremetal Validations** | ✓ **CREATE New** Baremetal Validations to verify engineering packages |
| "**Augment**" | ▪ **Spirent Validations** | ✓ **Augment** projects with 240 TC adds for load, scaling, cloud migration. |
| "**NEW**" | ▪ **Chaos Toolkit** | ✓ **CREATE New**, project to test POD resiliency by injecting chaos (failover) |
| **Bottlenecks** | • Stress Testing | ✗ **Not recommended** with limited test sets and results categorization |
| **Doctor** | • Computer NFVI Fault Mgmt validations | ✗ **Not recommended** with limited/no coverage for SDN, KVM, or containers |
| **XTesting** | • CICD tool chaining in CNTT validations. | ? **REQUIRES POC** if CNTT NFVI requires ADOPTION and USE of tool chaining.<br>? **There are no specific TCs**, as XTesting is for chaining together CICD test projects, and not for NFVI validation. |
| **NFVBench** | • NFVI Perf Measurements (at physical hardware/host level) | ✓ **ADOPT,** as a complement to vsperf and yardstick<br>✓ **Augment** to expand SRIOV and/or OVS-DPDK test cases. |

# Backup Slides

# COMMON NFVI LIFECYCLE FRAMEWORK



Feedback for improvement of
Reference Model & Reference Architectures

Community Standards

GSMA

Reference Model

External
Innovation

Owner
TBD

RA 1

RA 2

RA 3

Reference Architectures

*CNTT community defined*
*Reference Model & Reference Architectures*

Feedback for
improvement

Identified escaping
defects for
Root Cause Analysis

Ratified specifications
for implementation

OPNFV

TELCO
OPERATORS

VNF
SUPPLIERS

RI 1

RI 2

Reference Implementations

OPNFV integrates
Reference Implementation
with OVP testing ecosystem

VNF
Candidate

SAMPLE NFVI STACK *

| VNF1 | VNF2 | CNF3 |
|---|---|---|

| VM | VM | VM | VM | C | C |
|---|---|---|---|---|---|

Virtualization/Container Layer
(OS, KVM, VMWare, Kubernetes)

| Compute | Network | Storage |
|---|---|---|
| Bare Metal Install | Common Node Setup | Ceph Setup |

** For illustrative purposes only*

NFVI Testing Lab

Install Dovetail
Docker

Download
tar.gz
result file

Upload
results file
to OVP Portal

Test Host

DMZ

Firewall

Management
Network

Internet

Interface to
public internet

SUT

Install Data Path
Network

REF
NFVI

OPNFV

Server

OPNFV
Verified Program

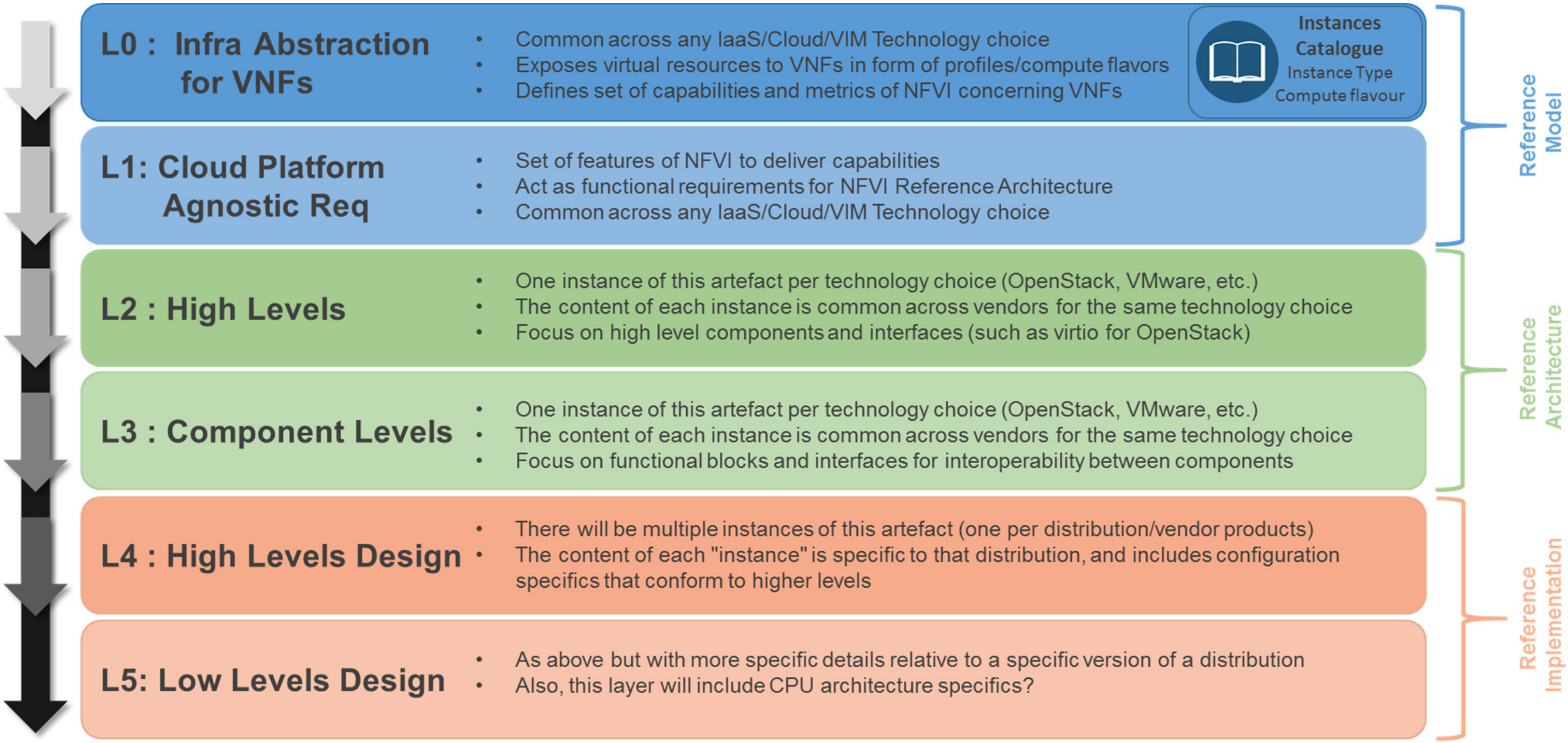
OVP Testing Ecosystem

OPNFV Verified VNF

Infrastructure
OPNFV
VERIFIED
2018.09

VNF
OPNFV
VERIFIED
2019.04

# CNTT NFVI Reference Levels



**L0 : Infra Abstraction for VNFs**

- Common across any IaaS/Cloud/VIM Technology choice
- Exposes virtual resources to VNFs in form of profiles/compute flavors
- Defines set of capabilities and metrics of NFVI concerning VNFs

**Instances Catalogue**
Instance Type
Compute flavour

**L1: Cloud Platform Agnostic Req**

- Set of features of NFVI to deliver capabilities
- Act as functional requirements for NFVI Reference Architecture
- Common across any IaaS/Cloud/VIM Technology choice

**L2 : High Levels**

- One instance of this artefact per technology choice (OpenStack, VMware, etc.)
- The content of each instance is common across vendors for the same technology choice
- Focus on high level components and interfaces (such as virtio for OpenStack)

**L3 : Component Levels**

- One instance of this artefact per technology choice (OpenStack, VMware, etc.)
- The content of each instance is common across vendors for the same technology choice
- Focus on functional blocks and interfaces for interoperability between components

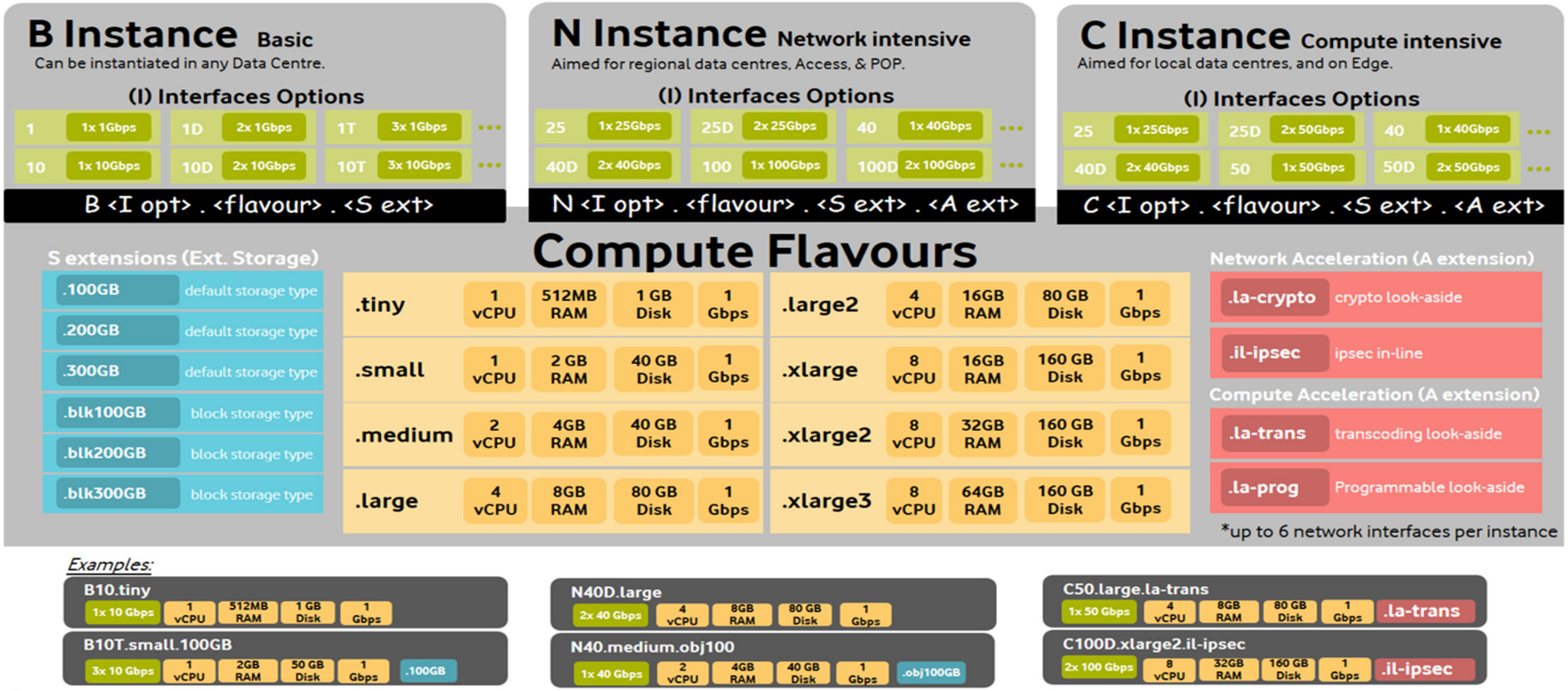
**L4 : High Levels Design**

- There will be multiple instances of this artefact (one per distribution/vendor products)
- The content of each "instance" is specific to that distribution, and includes configuration specifics that conform to higher levels

**L5: Low Levels Design**

- As above but with more specific details relative to a specific version of a distribution
- Also, this layer will include CPU architecture specifics?

# Instance, Flavor, Acceleration Options – B / N / C



## B Instance Basic

Can be instantiated in any Data Centre.

### (I) Interfaces Options

| | | | | | | |
|---|---|---|---|---|---|---|
| 1 | 1x 1Gbps | 1D | 2x 1Gbps | 1T | 3x 1Gbps | ••• |
| 10 | 1x 10Gbps | 10D | 2x 10Gbps | 10T | 3x 10Gbps | ••• |

B <I opt> . <flavour> . <S ext>

## N Instance Network intensive

Aimed for regional data centres, Access, & POP.

### (I) Interfaces Options

| | | | | | | |
|---|---|---|---|---|---|---|
| 25 | 1x 25Gbps | 25D | 2x 25Gbps | 40 | 1x 40Gbps | ••• |
| 40D | 2x 40Gbps | 100 | 1x 100Gbps | 100D | 2x 100Gbps | ••• |

N <I opt> . <flavour> . <S ext> . <A ext>

## C Instance Compute intensive

Aimed for local data centres, and on Edge.

### (I) Interfaces Options

| | | | | | | |
|---|---|---|---|---|---|---|
| 25 | 1x 25Gbps | 25D | 2x 50Gbps | 40 | 1x 40Gbps | ••• |
| 40D | 2x 40Gbps | 50 | 1x 50Gbps | 50D | 2x 50Gbps | ••• |

C <I opt> . <flavour> . <S ext> . <A ext>

## S extensions (Ext. Storage)

| | |
|---|---|
| .100GB | default storage type |
| .200GB | default storage type |
| .300GB | default storage type |
| .blk100GB | block storage type |
| .blk200GB | block storage type |
| .blk300GB | block storage type |

## Compute Flavours

| | | | | |
|---|---|---|---|---|
| .tiny | 1 vCPU | 512MB RAM | 1 GB Disk | 1 Gbps |
| .small | 1 vCPU | 2 GB RAM | 40 GB Disk | 1 Gbps |
| .medium | 2 vCPU | 4GB RAM | 40 GB Disk | 1 Gbps |
| .large | 4 vCPU | 8GB RAM | 80 GB Disk | 1 Gbps |
| .large2 | 4 vCPU | 16GB RAM | 80 GB Disk | 1 Gbps |
| .xlarge | 8 vCPU | 16GB RAM | 160 GB Disk | 1 Gbps |
| .xlarge2 | 8 vCPU | 32GB RAM | 160 GB Disk | 1 Gbps |
| .xlarge3 | 8 vCPU | 64GB RAM | 160 GB Disk | 1 Gbps |

## Network Acceleration (A extension)

| | |
|---|---|
| .la-crypto | crypto look-aside |
| .il-ipsec | ipsec in-line |

## Compute Acceleration (A extension)

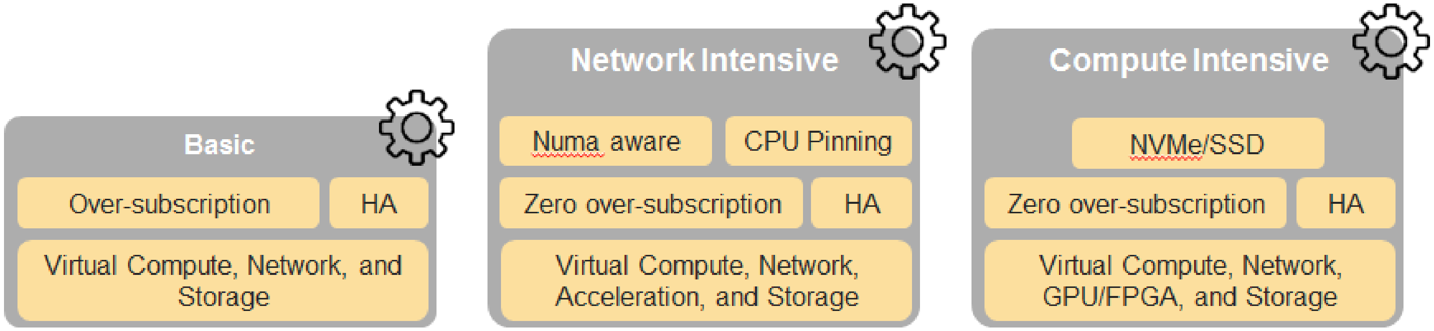
| | |
|---|---|
| .la-trans | transcoding look-aside |
| .la-prog | Programmable look-aside |

*up to 6 network interfaces per instance

*Examples:*

| | | | | | | |
|---|---|---|---|---|---|---|
| B10.tiny | 1x 10 Gbps | 1 vCPU | 512MB RAM | 1 GB Disk | 1 Gbps | |
| B10T.small.100GB | 3x 10 Gbps | 1 vCPU | 2GB RAM | 50 GB Disk | 1 Gbps | .100GB |
| N40D.large | 2x 40 Gbps | 4 vCPU | 8GB RAM | 80 GB Disk | 1 Gbps | |
| N40.medium.obj100 | 1x 40 Gbps | 2 vCPU | 4GB RAM | 40 GB Disk | 1 Gbps | .obj100GB |
| C50.large.la-trans | 1x 50 Gbps | 4 vCPU | 8GB RAM | 80 GB Disk | 1 Gbps | .la-trans |
| C100D.xlarge2.il-ipsec | 2x 100 Gbps | 8 vCPU | 32GB RAM | 160 GB Disk | 1 Gbps | .il-ipsec |

# Software & Hardware Profiles



Basic

Over-subscription | HA

Virtual Compute, Network, and Storage

Network Intensive

Numa aware | CPU Pinning

Zero over-subscription | HA

Virtual Compute, Network, Acceleration, and Storage

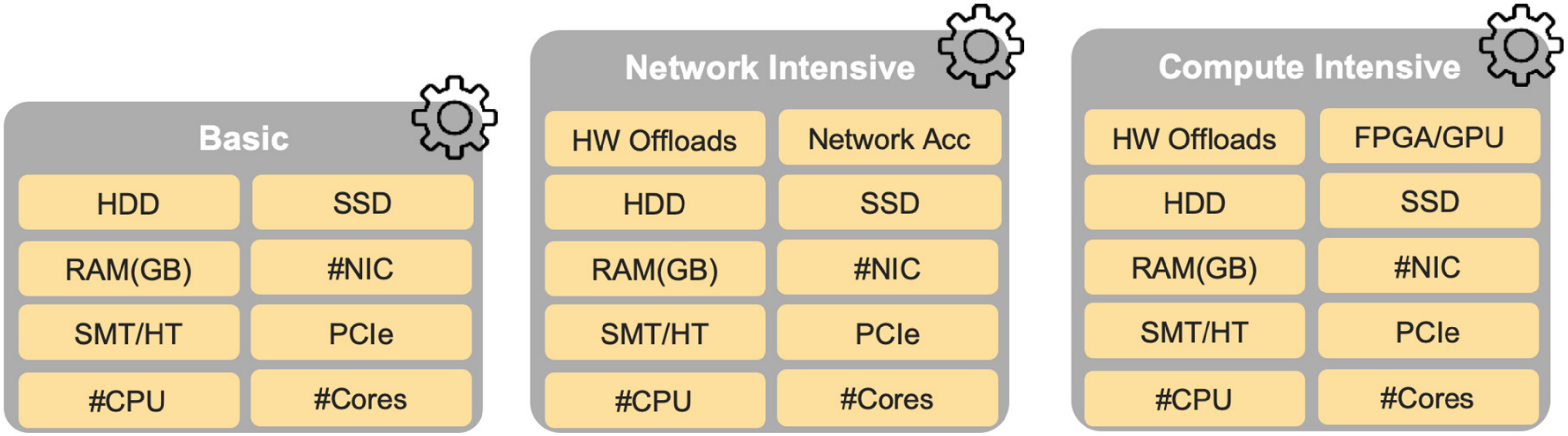
Compute Intensive

NVMe/SSD

Zero over-subscription | HA

Virtual Compute, Network, GPU/FPGA, and Storage

**Figure 5-3:** NFVI software profiles.



Basic

| | |
|---|---|
| HDD | SSD |
| RAM(GB) | #NIC |
| SMT/HT | PCIe |
| #CPU | #Cores |

Network Intensive

| | |
|---|---|
| HW Offloads | Network Acc |
| HDD | SSD |
| RAM(GB) | #NIC |
| SMT/HT | PCIe |
| #CPU | #Cores |

Compute Intensive

| | |
|---|---|
| HW Offloads | FPGA/GPU |
| HDD | SSD |
| RAM(GB) | #NIC |
| SMT/HT | PCIe |
| #CPU | #Cores |

**Figure 5-4:** NFVI hardware profiles and host associated capabilities.

# Current OPNFV/OVP Certification Process

**CVC Compliance, Verification, and Certification governing framework:**

- **Compliance testing**: compares the system under test against the specifications / standards
- **Validation testing:** ensures the system under test is operating according to its intended / required purpose
- **Performance testing**: measures how well the system under test performs its specific purpose(s)

**OVP certifications are accomplished as a two-part process:**

1. OPNFV provides Test Tools and Test Cases to OVP.
2. OVP provides vendor NFVI products a "badge" claiming "OPNFV-certified" once OVP testing passes successfully (via Dovetail).

**Test frameworks and supported test cases for OVP Certifications include (OPNFV and ONAP):**

1. **OPNFV – NFVI Testing and Certification**

   **CVC Category: Validation, Compliance**

   **Purpose: NFVI platform is checked against the Open Stack requirements**

   Test framework, test result database and Web UI

   **Dovetail** project

   Test tools, test cases and test execution

   API testing (**FuncTest**)

   Performance and HA (**Yardstick**)

   Load testing (**Bottleneck**)

2. **ONAP – VNF Testing and Certification**

   **CVC Category: Compliance, Performance**

   **Purpose: VNF template is checked against the ONAP Requirements**

   - VNF Validation/Packaging Compliance (HEAT and TOSCA/CSAR/VFD)
   - Web front-end integrated with OPNFV Dovetail Web UI
   - Future plan: testing of VNF lifecycle, VNF functions, VNF performance

**VNF Validation Testing** - CVC and ONAP teams are working to develop the initial life-cycle tests for VNF devices. The first release will run on the "basic open stack" meeting ONAP requirements. Future versions would require this to be the reference NFVI defined by CNTT

THE LINUX FOUNDATION

GSMA

# Test & Validation: OPNFV/OVP Entry/Exit Criteria

## ENTRANCE CRITERIA

**OPNFV entrance criteria must be satisfied for VNF certification.**

***Demonstration of Reference Architecture Implementation***

**Design & Requirements**

- Design, Configuration, Features, SLAs, and Capability documentation complete
- Users stories / Adherence to CNTT Model principles and guidelines
- Chosen RA Matches a RA from the product catalog

**Environment**

- Lab / Flavor, component s/w rev levels specified, with confirmation of compatibility with external systems
- Tenant needs identified
- All connectivity, network, image, VMs, delivered with successful pairwise tests
- Lab instrumented for proper monitoring

**Planning & Delivery**

- Kickoff / Acceptance Criteria reviews performed
- Delivery commitments, timelines, and cadence accepted
- Confirm backward compat. across software/flavor rev levels

**Data / VNFs / Security**

- Images, Heat Templates, Preload Sheets available
- Images uploaded to tenant space
- External system test data needs identified
- Owners (NFVI, VNF, PTL, etc) documented
- Security Compliance Satisfied (see Ch. 8 – scans, vulnerabilities)

**Test Case Contributions**

- VNF Developer/Supplier validations to be performed documented and supplied
- NFVI validations to be performed supplied (e.g. echo, manifest)
- Test to ensure users are added and have correct privileges for the tenant
- Test to ensure quota against submitted request for the respective tenants
- Test to ensure custom flavors against submitted request for respective tenants

## EXIT CRITERIA

**CERTIFICATION TENANTS (via OVP Ecosystem)**

- All Test Cases have a status of "Passed", "Failed", "N/A" or "Out-scoped".
- All Severity 1 and Severity 2 issues are resolved.
- All Issues have been Resolved or the Project/Component Team has voted unanimously for a Conditional Certification.
- Known defects, or issues, are clearly documented and furnished to Telco providers with certification notes.
- Certification Notification(s) issued.

**OPERATIONAL CONSIDERATIONS (TELCO PERSPECTIVE)**

- Orchestration capabilities verified to be working as expected
- Fabric setup/configuration validations successfully passed
- Openstack API endpoint is reachable and working for that zone
- Compute zones and cinder types verified
- Standard images verified to exist (and usable)
- Network object created (and working, as in IPs are bindable and usable)
- Resolver overlay/DNS traffic/port 53 overlay on gateway is working properly
- Designate is working, domain preferably created, and maybe test A record created/verified to be resolvable
- Standard NTP servers are working and verified (using tenant's CIDR source IP)
- NFVI/VNF is tested at steady state and high load
- Continuously monitored to ensure SLAs are met and used as feedback to load/perf tests

**END USER CONSIDERATIONS (TELCO PERSPECTIVE)**

- Component redundancy to ensure graceful updates without disruption of services
- Thin provisioning storage should handle actual full quota usage cases
- Load balancing should support elasticity
- SRIOV Network configuration via SDN must be aware of all VMs on a host (and their network config)
- Auto-healing databases (any component related db) when out of sync
- Obvious, but, supports all required network functionality (all protocols, service chaining, VLAN trunking, QoS marking, probe/mirror, etc)
- Supports NFV migration
- Supports snapshots and backups of large volumes
- Pre-check or audit failures during NFV deployment should allow follow-up mitigation, when possible, rather than killing deployment and rolling back

# Test Category / Case Gap Summary.. Continued

## Recommend Adopt and/or Augment

| Project | Gaps | Recommendation |
|---|---|---|
| **FuncTest –** functional interoperability validations | ▪ Doesn't directly verify Baremetal and Operating System for Manifest Validations | ✓ **ADOPT** as an RI suite. Covers large (2k) Openstack Interoperability Validations including test support of Dovetail, Pharos, AirShip, and Fuel, for example.<br>✓ **Augment** to include Baremetal testing, or stand up separate test project to cover Baremetal which supports "Manifest Validations".<br>✓ **Reuse** functest-smoke (functional test mostly based on tempest), functest-benchmarking (rally_full and rally_jobs) and possibly the other tiers. |
| **Yardstick -** VNF/Payload performance validations | ▪ SRIOV/DPDK: No frame-size (MTU) for validating VNF perf parameters<br>▪ No TCs defined for emerging OVS-DPDK | ✓ **ADOPT,** 62 TCs, leverages Shaker and YAML for test-case/scenario development.<br>✓ **Augment**<br>✓ (tc025 node down HA check) to perform POD restarts on nova, neutron, etc.<br>✓ HA framework to check MariaDB and CEPH during node restarts.<br>✓ **Reuse** as TCs are broad across core components, resiliency, HA, compute, network, and storage. TC test general VNF requirements of: latency, throughput, packet-loss, IOPS. |
| **VSPerf** – vswitch perf testing | ▪ None | ✓ **ADOPT** for OVS-DPDK validations with 32 TCs covering performance and functional testing, validating Throughput, Scalability, Memory, NIC acceleration, etc.<br>✓ **Setup** external packet generator tool from the SUT to avoid latency caused by the tool. |
| **DoveTail** – automation framework | ▪ None | ✓ **ADOPT,** as an automation framework with access to large number of test cases for conformance evaluation tests such as security flaws in OpenStack(VIM), K8s, Tenant HA and various other conformance areas. |

# Test Category / Case Gap Summary.. Continued

## Recommend Adopt and/or Augment

| Project | Gaps | Recommendation |
|---|---|---|
| **Barometer –** platform availability and NW usage validations | ▪ Limited coverage with VNF traffic monitoring, or network usage.  Includes, but not limited to monitoring VNF when traffic is introduced, handling of that traffic, or reacting to faults to confirm resiliency of the device.. | ✓ **ADOPT**. for use of NFVI+VNF validations given the broad breadth of evaluations performed with the Barometer framework:  CPU utilization, Monitoring, Telemetry, etc.<br>✓ **Augment** to include device specific resiliency testing and monitoring. |
| "**NEW**" – baremetal validations | ▪ Baremetal Validations lacking from any known project. | ✓ **CREATE New, or confirm** if Baremetal Validations can be inserted into an existing project to validate baremetal delivery matches the manifest, or engineering package provided.<br>✓ e.g. validations to include, but not limited to: NUMA config checks, NIC frame size (MTU), Huge Page configuration, BIOS, Firmware, checks etc. |
| "**NEW**" – Spirent Partner | ▪ Gaps in NFVI Assurance, VIM/OpenStack Assurance, and VNF & NS LCM (Life Cycle Mgmt) | ✓ **Augment** existing test projects, or possibly **Create New,** with the following additional test cases:<br>✓ **NFVI Assurance** – add 30 TCs, extreme scale load generation, cloud migration assistance<br>✓ VIM/OpenStack Assurance – add 70 TCs, VIM scale testing and consistency checks at scale<br>✓ VNF & NS LCM (Life Cycle Mgmt) – add 140 TCs, VNF/NS LCM autoscaling testing |
| "**NEW**" – Chaos Toolkit | ▪ Lack of tool/project to measure uncertainty of distributed systems at scale to discover platform weakness | ✓ **CREATE New**, or confirm existence of, a tool to test the Kubernetes PODs for resiliency by injecting chaos and test if the PODs recover from chaos.<br>✓ e.g. scenarios include:  Node failover testing, RabbitMQ resiliency, impacts during CEPH outage, container POD eviction and replication after shutdown, etc. |

THE LINUX FOUNDATION

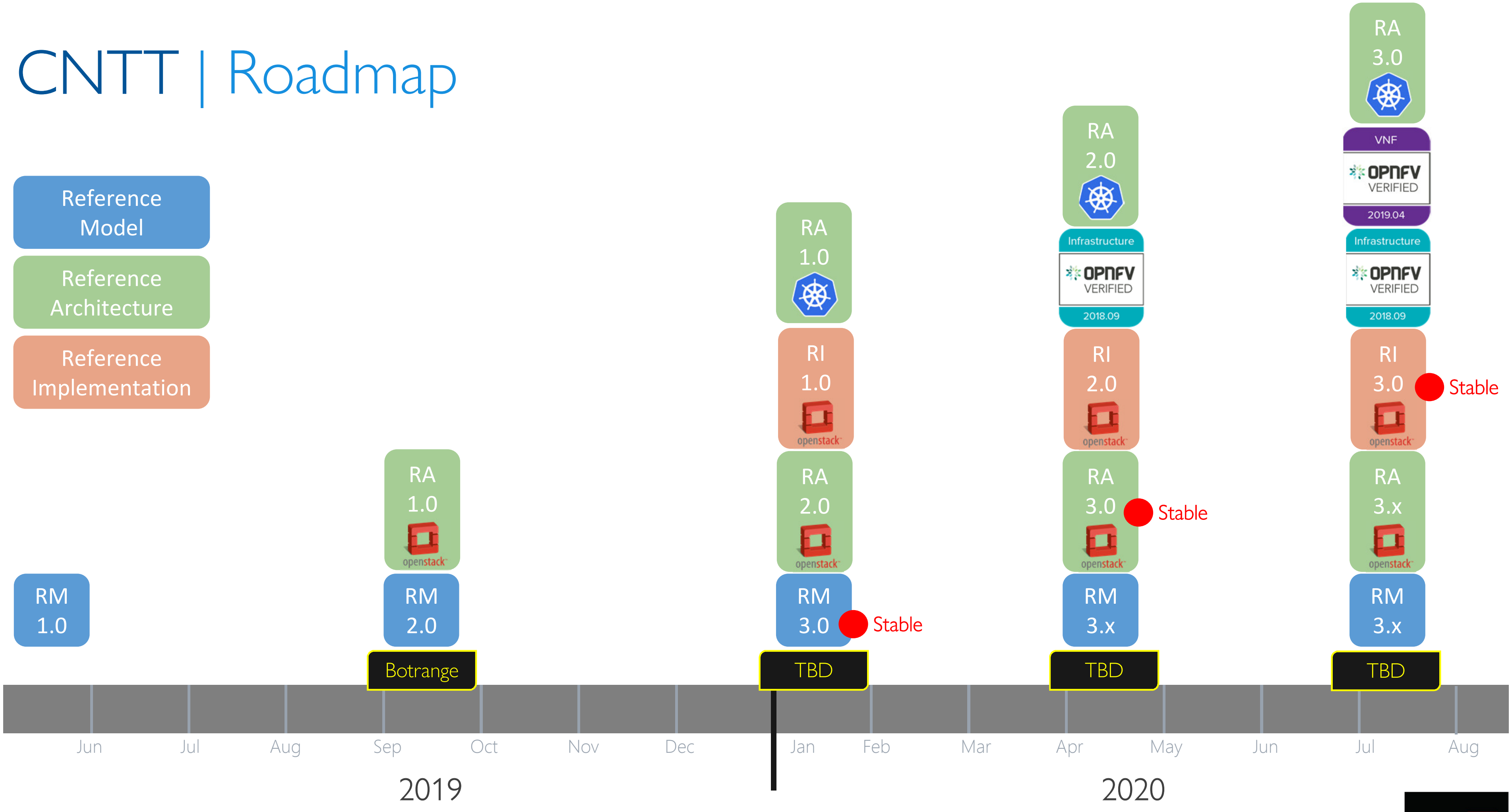
GSMA

# Test Category / Case Gap Summary.. Continued

## Not Recommended

| Project | Gaps | Recommendation |
|---|---|---|
| **Bottlenecks –** stress testing | • Limited test-suites, needing stability, robustness, and SLA test suites/capabilities<br>• Not built out as a tool that considers different hardware and middleware capabilities along with true VNF load testing. | ✗ **Not recommended** as a CNTT test suite as it relies on other test projects for test cases.<br>✗ With limited test sets, categorization of results will not paint the whole picture and as such, other tools (albeit outputs without categorization) may be better suited. |
| **Doctor** – compute/NFVI fault mgmt. | • Few parameters based on which the fault mgmt. alarms are triggered such as VM status(whatever nova provides), Port issues affecting connectivity. | ✗ **Not recommended** as TCs need to be enhanced to cover more fault scenarios – SDN, KVM, etc.<br>✗ **Recommendation** is for Doctor PTL to extended fault monitoring to container infrastructure, and not be limited to VIM/OpenStack. |
| **XTesting** – CICD tool chaining | • None; however, PoC required to evaluate need-for CICD tool chaining in CNTT validations. | ? **REQUIRES POC** if CNTT NFVI requires ADOPTION and USE of tool chaining.<br>? **There are no specific TCs**, as XTesting is for chaining together CICD test projects, and not for NFVI validation. |
| **NFVBench** – NFVI perf (blackbox) | • Limited SRIOV and/or OVS-DPDK validations for NIC virtualization testing. | ✓ **ADOPT,** as a complement to vsperf and yardstick which measure VM level NFVI perf, whereas NFVBench evaluates NFVI from the physical hardware/host level, and includes an extensive test coverage around VxLAN.<br>✓ **Augment** to expand SRIOV and/or OVS-DPDK test cases. |

# CNTT | Roadmap



Reference Model

Reference Architecture

Reference Implementation

RM 1.0

Botrange

RM 2.0

RA 1.0

TBD

RM 3.0 ● Stable

RA 2.0

RI 1.0

RA 1.0

TBD

RM 3.x

RA 3.0 ● Stable

RI 2.0

Infrastructure OPNFV VERIFIED 2018.09

RA 2.0

TBD

RM 3.x

RA 3.x

RI 3.0 ● Stable

Infrastructure OPNFV VERIFIED 2018.09

VNF OPNFV VERIFIED 2019.04

RA 3.0

THE LINUX FOUNDATION

GSMA