LF NETWORKING

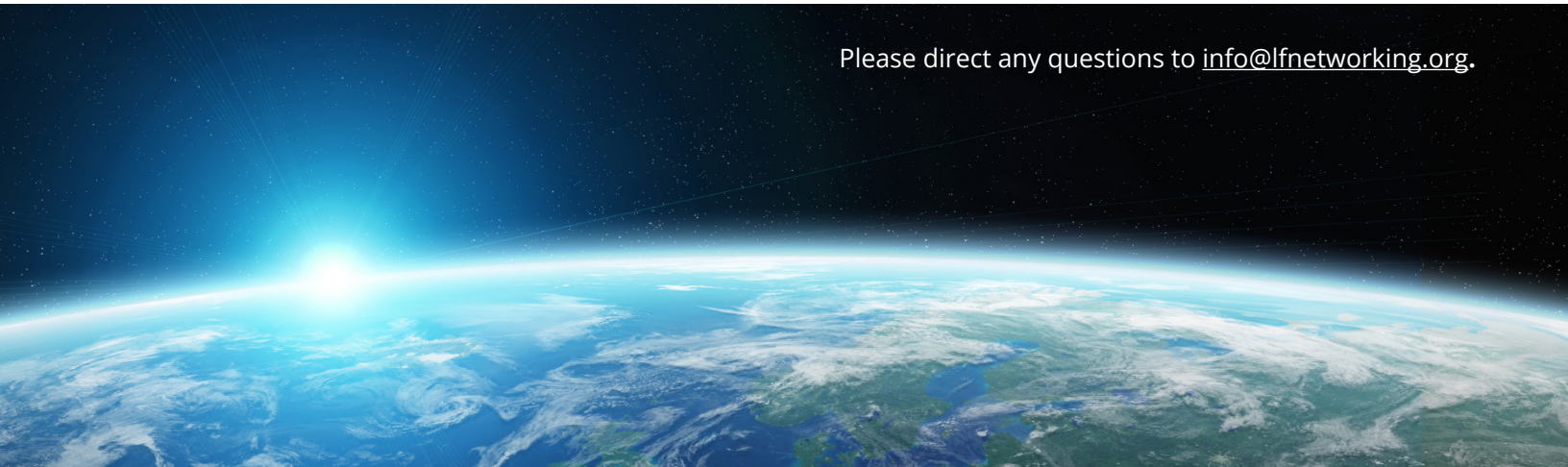
White Paper: NFV Testing and Automation Research and Methodologies

A Telecom Operator's Perspective

Authored by the Members of the Linux Foundation End User Advisory Group

Randy Levensalor (CableLabs)
Lei Huang (CMCC)
Ahmed ElSawaf (STC)
Saad Sheikh (STC)
Cecilia Corbi (TIM)
Massimo Banzi (TIM)
Beth Cohen (Verizon)

For a discussion of the EUAG's recommendations for NFV Testing, please see the accompanying paper: NFV Testing and Automation.



Please direct any questions to info@lfnetworking.org.

LF NETWORKING

Table of Contents

1 Introduction

The companion *White Paper: NFV Testing and Automation*, we recommended changes to NFV testing models and the development of a common understanding of the requirements for automation and more efficient testing methodologies. This would be used for supporting the delivery of Telecom NFV systems and infrastructure across the industry. Some of the key takeaways from that document include:

- To take advantage of rapidly changing technologies, the Telecom industry needs to be able to deliver new NFV (Network Functions Virtualization) services quickly and efficiently.
- Successful scalable NFV deployments require extensive testing, standard testing methodologies and testing automation.
- A shared understanding of NFV testing processes across the industry is needed to optimize interoperability.
- Some systems and testing automation approaches are available today, but more R&D is needed across the industry to establish best practices.
- The open source community and Telecom standard bodies can play a key role in furthering the development of standard testing platforms, frameworks and best practices.
- All stakeholders across the Telecom industry, operators and ecosystem companies are welcome and encouraged to join this critical effort to define the testing projects and requirements in support of NFV deployments.

This companion document takes a deeper dive into how the EUAG came to its conclusions and recommendations. It provides the supporting research and evaluation methods that were distilled into the recommendations made in the white paper. It starts with a discussion of the evaluation criteria used to apply to the findings, followed by an analysis of the existing testing capabilities currently used in the industry and why it is unable to support the new testing requirements imposed by the NFV environments. The final section surveys the testing projects and resources available in the standards and open source communities to date.

2 Assumptions and Evaluation Model

The first step in any research project is to identify the criteria for evaluating the findings and establish the assumptions that are part of that evaluation. The following are the criteria that were used to evaluate the data.

- **Resource skills and capacity:** Testers are required to have an overall grasp of NFV system architectures, familiarity with the tested network elements and interfaces, and corresponding specifications (in the initial stage, 5G core network main network elements such as AMF/SMF/UPF), the testing capabilities of open source and commercial tools required for NFV testing, and the development capabilities of automated scripting languages (Restful API, Python, etc.).

A reasonable test framework design mapped to skills required to perform the expected tasks forms the following ability matrix. The matrix can be used to find a suitable project starting point. As the industry ecosystem tools, solutions or technologies change the matrix can be expanded to add other criteria and requirements.

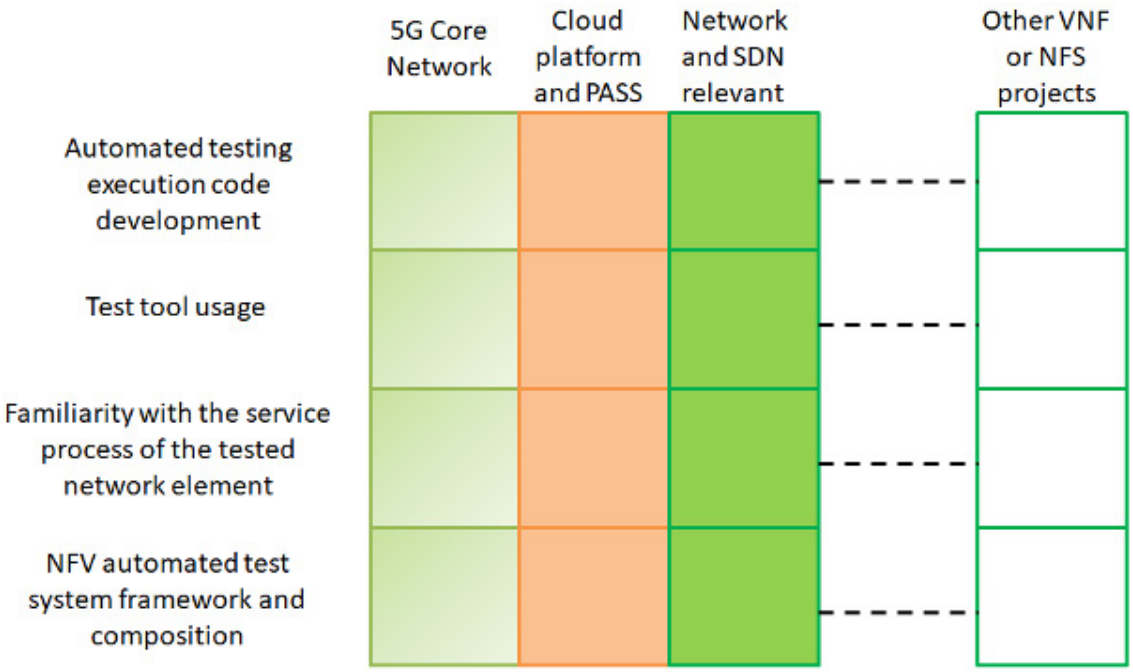


Figure 1: Sample mapping of skill to NFV environments

- **Basic environmental capabilities:** The ability to do comprehensive NFV testing is inseparable from the need for a resource-rich, stable and reliable test platform. The flexibility and dynamic scalability required by NFV testing need to be considered during the preparation of the basic environment.

Given that OpenStack is currently the most widely used cloud platform for NFV workloads, it is best to consider creating a test bed that includes a server and switch networking structure that is as consistent as possible with the production network environment to fully validate any issues found during the testing phase. This will avoid issues related to test results deviation caused by differences between the test environment and the production environment.

Considering the multiple software and hardware combinations tested in the NFV lab, it critically important to plan for different hardware devices in different resource pools, including automated server configuration and network switching equipment configuration change capabilities. Any differences in test results exposes potential problems caused by platform compatibility. A topology management mechanism needs to be introduced between resource pools to improve the automated execution efficiency and the ability to automate the testing of multiple combinations of software and hardware.

- **Network element and test tool capabilities:** Unlike traditional networking workload testing, NFV testing is looking at different network elements and needs different types of testing tools. The good news is that some existing tools can meet basic functional testing requirements. A combination of open source and commercial tools are available to meet the requirements for stability and accuracy needed for performance test scenarios with large numbers of users and high throughput traffic. It is possible to perform functional tests of some network elements with a combination of open source tools and commercial tools.

The deployment capability of tools also needs to be considered in the design of NFV automated test frameworks. Commercial test tools commonly rely on dedicated hardware to provide the desired stability and high performance, which directly contradicts with the desire for the software and elastic characteristics of NFV testing. Dynamic topology management tools are needed to bridge the gap to achieve the rapid networking topology construction capabilities required for flexible test environment construction.

Any software-based testing tools that will be used for NFV workloads must have “cloud native” features. At a minimum, that means they can be deployed flexibly across various virtual machine types. Since virtualized test tools usually have little

problem with hardware compatibility (except in the case of hardware accelerators), they should be part of the basic test resource pool. These reserved resources can be used to support the deployment of cloud-based test tools that can even be deployed together with the tested network elements in a resource pool during automated testing to avoid network performance bottlenecks in a "multi-cloud" environment.

- **Test tool and system ability to support cloud environments:** Cloud deployment of test tools is usually not a problem, but the network planning between the NFV test framework and the tested environment may introduce some additional complexity. During the testing execution phase that relies on the test execution machine and test machine instance, there is a certain degree of system integration needed between the license manager, controller suite or other management components, and the tested system or network elements. The resources needed to execute the tests across the various software and hardware components requires the NFV test framework to have the ability to execute multiple tests in parallel. Resource allocation planning for elements such as network hardware, topology, IP, VLAN, etc. is a primary condition for the smooth execution of automated NFV tests.

3 Common NFV Testing Methodologies

While every operator approaches testing a bit differently. This section covers some of the more common types of testing processes and methodologies that are used by the operators to validate the vendor solutions in their own network infrastructures. Not all of these scenarios will apply to all operators or vendors; however, it is still useful to capture this information so there is a better understanding of at least some of the testing requirements needed to support NFV architectures.

3.1 NFV Testing Process Overview

Common NFV testing stages includes test topology design, test environment deployment, test execution, test result analysis, and finally validation.

- **Test topology design:** The test topology needs to be designed before executing any test sequence. The design commonly includes the component under test, test equipment and instruments, surrounding components and the network (wired or wireless) connection between them, etc. In addition, the test topology needs to include the software and virtualized networking components as well as the hardware configurations. The topology designs that incorporate software components might be different than the more traditional hardware based ones.
- **Test environment deployment:** The test environment deployment includes the processes for building the hardware environment and software environments. The hardware environment refers to the required physical servers, clients, network connection equipment, etc. The software environment refers to the virtual infrastructure, hypervisors, cloud managers, operating systems, databases and other applications used when the NFV software under test is running.
- **Test execution:** This is the heart of the testing process. During the execution the system runs a series of operations procedures in the test environment to simulate specific scenarios such as service processes, information interactions or equipment failures that might occur in a real production environment to validate if the system under test or function under test meets the specified functions, performance, reliability, interface compliance and any other requirements needed in support of production workloads.

- **Test result analysis and validation:** After the tests are completed, the final step is to analyze test execution results to identify any issues or potential problems. If any flaws are found, the systems will need to be analyzed to find the root causes, then rectify them, before repeating the tests to confirm they have been fixed. Once the systems complete the tests successfully, only then are the systems validated to be considered ready for use in production.

3.2 Network Access Scenario

Network access testing is a type of test used to validate if a new system or software release are compatible with the existing network. The test method of network access testing is similar to procurement and acceptance testing, but is concerned specifically with the ability of the new component to work with existing systems.

3.3 Procurement Scenario

Procurement testing is used to validate whether a system conforms to the technical requirements identified by the procurement team. The test results can then be used as the technical basis for the procurement bidding evaluation. Some operators encourage vendors to conduct testing according to a pre-developed test plan before the procurement bidding commences to ensure that the vendor is bidding a service that will meet the expectations of the operator. This is an area that would benefit from input from the open source and standards communities to level the playing field across the operators.

- **Services and service models:** Identifies the service processes involved, their proportions, and service volumes.
- **Test case:** States the test objectives, preset conditions, test steps, and expected results.
- **Test basis:** Includes the technology and test specifications of system under consideration, and the evaluation criteria for any test results.
- **Test topology:** includes the system under test and surrounding network elements.
- **Testing tools:** Specification for the instruments and tools used to perform the tests.

3.4 NFV Acceptance Scenario

Acceptance testing is used to verify whether a system can meet defined service delivery and service assurance parameter expectations. Acceptance testing is sometimes known as delivery testing, or ATP (Acceptance Test Procedure), or UAT (User Acceptance Test). These include testing modules associated with anything that is related to the decision to accept the system into production.

Prior to running acceptance testing, vendors usually conduct unit tests. These tests generally cover all acceptance test cases (except for the acceptance test cases added by the operators), but they do not test the integration of the various systems.

As it relates to NFV workloads, a common scenario is that acceptance testing is often led by operators, but the testing work is often done by the vendors. The testing objects include but are not limited to hardware, infrastructure, infrastructure managers, MANO, VNF workloads, etc. Acceptance testing modules includes but are not limited to functional testing, performance testing, reliability testing, abnormal testing, etc. The number of test cases for acceptance testing varies from dozens to hundreds depending on test objectives.

4 NFV Testing Challenges

With the introduction of NFV architectures, the resource pool and network element tests need to be conducted separately. Because pairing tests are needed between resource pools and network elements, this increases both the test types and the frequency of the testing.

Another factor that affects NFV testing is that due to the requirements for the rapid introduction of new network or service functions and the need to upgrade the software more often, the upgrade cycle of NFV software-based network elements is often substantially shortened to 2-3 weeks or 1-2 months compared with the more traditional half-year upgrade for physical network elements. Unless this is accounted for, this can put considerable stress on the staff and lab resources needed to complete the NFV testing cycles. Some of the major limiting factors making this requirement difficult to achieve include:

- Operators' test environments are limited resources
- Long approval processes and intervals for access tests
- Insufficient stability of the production environment
- Personnel, management, and equipment testing costs are high

At the same time, from the perspective of equipment, instrument, environment, personnel, etc., adding the NFV requirements for diversification of network elements and interfaces means the need for more complex instruments, problem location, large testing capacity, high demand for skilled staff, and more. To help demonstrate the complexities and challenges, below is an example of a 5G core test.

Dimensions	Challenges
Equipment	Many network elements, interfaces, service processes; complex service models, large service volume, complex network element configuration, and difficult to locate problems.
Instrument	Complex operation, high degree of professionalism, difficult to locate problems.
Environment	The test capacity is large and requires ample resources. Material preparation, environment and networking construction takes time.
Personnel	Multi-vendors, multi-network elements, multi-instruments, and multi-processes lead to high requirements for personnel and technical skills.

5 NFV Testing Automation Feasibility

Using the analysis criteria, from the automated test evaluation index model, NFV automated testing needs to automate the following elements to be successful: test environment deployment, configuration, test cases and execution, process observation, test reports output, etc. Specific test frameworks/tools can provide further automation of related test phases, making automated testing feasible:

- The application of software and hardware products with interfaces defined by open standards makes it possible to automate the control of these products, to complete the control of different devices through a unified interface, and to give automation test cases have a good replication usability.
- The existence of various open source or professional testing instruments and tools makes automated testing possible. Some examples include, SpecCpu-CPU performance testing tool, Stream-memory performance testing tool, FIO-storage performance testing tool, IxNetwork-professional network performance testing tool, Cloudpeak-expanded automated test suite based on Yardstick NFV test standard.
- Using CI/CD automation solutions and platforms make automated testing as well as continuous development and integration possible.
- Jenkins/pytest can drive the execution of test cases, GitLab can help with project management and code hosting.
- The universal automated test framework integrates the capabilities of third-party and open source test tools/instruments, which can then automate test topology design, deployment of test environments, execution of test tasks, analysis, and certification of test results.

6 NFV Automated Testing Tools and Framework Research

The following section is an overview of the NFV automated testing approach and procedures as defined by Standards Developing Organizations (SDOs) and open source Communities. This is by no means a comprehensive list, but it does touch on the best known ones.

6.1 Standards Developing Organizations NFV Testing Automation Resources

As the network disaggregation evolution (SDN, NFV, Cloud Native paradigms) has changed Telecom environments, the model of traditional monolithic network functions and single vendor solutions has evolved into looking more like a set of virtualized functions from multiple vendors deployed on a virtualized infrastructure, with dynamic automated lifecycle management. Because these new technologies are so different from before, testing methodologies and tools need to evolve to support the broader scope of configurations and production scenarios. Several standards communities have been working on different testing approaches and methodologies. Below are some of the more common resources used in the industry.

6.1.1 ETSI

- **ETSI ISG NFV:** ETSI ISG NFV started in 2012, has been the originator of the network transformation activities within ETSI, being at the core of NFV technology, definition and standardization. It is the reference group that has been developing testing specifications related to the ETSI NFV architecture, defined as a virtualized infrastructure often provided by the service provider as a managed cloud, representing the ETSI NFV NFVI+VIM, a set of VNFs, all using a management function to manage the lifecycle of the VNFs.

All these components need to be tested individually, then tested for component interoperability and integration. Another element that needs testing is the conformance of their interfaces against a set of standard implementations, such as the VIM OpenStack implementation. Performance testing needs to be done to provide vital infrastructure and resource requirements criteria that will be used in the design phase, as well as after deployment to validate that the allocation of resources meets the requirements and that the VNFs deliver the expected performance.

- **ETSI ISG ENI:** ETSI ISG ENI specifies an architecture to enable closed-loop network operations and management leveraging AI. The need for close-loop operations at any domain of the network and cross-domains requires ENI to be deployed and operate either in one network domain, and/or cooperatively across several different domains.

The ISG ENI focuses on improving the operator experience, adding closed-loop artificial intelligence mechanisms based on context-aware, metadata-driven policies to recognize and incorporate new and changed knowledge, and hence, make actionable decisions more quickly. To date no specific work item on testing processes has been defined within the ENI framework.

- **ETSI ISG ZSM:** ETSI ISG ZSM is primarily concerned on providing a framework that enables zero-touch automated network and service management in a multivendor environment. While NFV and ENI have defined management capabilities in their respective focus areas, ETSI ZSM aims to define a holistic end-to-end network and service management concept which, among others, enables the integration of ENI, NFV and MEC management demands. ZSM is building a flexible service-based network and service management framework that supports cross-domain end-to-end management and provides enablers that support closed loop automation and data-driven management algorithms that can be used to enable machine learning and artificial intelligence of networks. The table below is a short summary of the ETSI ISG NFV, ZSM and ENI and a list of the relevant specifications.

Work Group	Description
ETSI NFV TST	Since its establishment in 2014, the NFV TST group has provided specifications and analysis of testing methods, tools and frameworks. The main contents of the TST working group are as follows: • Maintain and develop the PoC framework, and expand testing activities to cover interoperability based on ISG NFV specifications. • Develop test specifications and test methods. • Coordinate the experimentation and demonstration of NFV solutions (e.g. PoC exhibition area, etc.) • Generate PoC case studies (for example, VNF life cycle), and record/report the results in various forms, such as white papers, wikis, etc.

ETSI NFV TST (cont.)	(cont.) • Provide feature requests to open source projects, provide implementation experience results to open source communities, and develop reports based on implementation experience (for example, guidelines, best practices, etc.) • Pass the results to other ISG NFV working groups to ensure consistent delivery of specifications through actual implementation and testing.
ETSI ISG ZSM	The ZSM Zero Touch Network and Service Management Industry Specification Group (ZSM ISG) focuses on 5G end-to-end network and service management. The objective of ZSM ISG is to define work practices and systems to achieve agile, efficient and qualitative management and automation of emerging and future networks and services, etc. The goal is to enable all operational processes and tasks (such as delivery, deployment, configuration, assurance and optimization) to ideally achieve 100% automation. Initially, it will focus on 5G end-to-end network and service management (such as network fragmentation management), with the intention to expand to future network management. • ZSM published the following specifications on Requirements and Architecture where some scenario of system test in a fully automated manner are describedZSM001: Use Cases and requirements of ZSM • ZSM002: Reference framework of ZSM

6.1.2 3GPP SA5

3GPP SA5 Is focused on defining the Management, Orchestration and Charging for 3GPP systems. The testing activities for this group are not directly in scope of SA5. There are some work items where the testing activities are collateral.

Work Item	Percent Complete	Target time	Planned output	Remarks
Intent driven management services for mobile networks (810027 - IDMS_MN)	55%	SA#87 - Mar 2020	New TR 28.812 Study on scenarios for Intent driven management services for mobile networks. New TS 28.312 Intent driven management services for mobile networks.	This work item continues to R17.
Self-Organizing Networks (SON) for 5G networks (850030-SON_5G)	75%	SA#88 - Jun 2020	New TS 28.313 Self-Organizing Networks (SON) for 5G networks. Update 28.541	This work item is currently postponed for one meeting, but R17 has a new topic about SON.
Study on autonomous network levels (850032 FS_ ANL)	85%	SA#89 - Sep 2020	New TR 28.810 Study on concept, requirements and solutions for levels of autonomous network.	Currently postponed for one meeting, but R17 has a new WID.

6.1.3 TM Forum

Testing is not directly in scope of the TM Forum activities, but the topic has been investigated especially related to VNFs due to the introduction of virtualization into Telecom operational infrastructures. The main contribution to this topic is the IG1137-Jont Agile Delivery Suite.

NFV testing is mentioned within the context of a transformation from NetOps to DevOps. The intent of that transformative process was to break the silos between Development and Operations in order to work efficiently in a value fabric. In Release 16.5, IG1137A introduced the Development and Verification portions of Joint Agile Delivery (JAD), a process which breaks down silos among suppliers, integrators, partners, and the service provider in order to create a streamlined multi-party continuous delivery pipeline with rapid customer feedback and continuous improvement.

IG1137B, covers the Service Assurance portion of JAD which spans the steps from Deployment Readiness to Ongoing Operations. The challenge of Service Assurance in a value fabric is understanding how end user Service Level Agreements (SLAs) are measured and remedied across contributions from multiple contributors and how problems are attributed to a certain contributor(s) and resolved in a timely and efficient manner. In Cap. 2.8 Open APIs and Testing,

> "In Network Service testing (involving multiple VNF's), the ability to share/reuse test cases and a to evaluate the test results consistently are essential. The JAD contributions of a domain specific test language, a testing metamodel, and testing APIs help to achieve this goal. In JAD Phase 3, Test Management and Test Execution API's will be contributed to complement TM Forum Open API's."

The document was also supported by a TMF Catalyst (Joint Agile Delivery - 2017 - https://www.tmforum.org/catalysts/joint-agility-delivery-phase-iii/) where they wanted to demonstrate a Joint Agile Delivery Ecosystem: a standardized and platform-based approach to developing and delivering world-class software that capitalizes on cross-organizational (cross-value fabric) synergies to dramatically improve TTM, quality, and cost. Mainly led by Huawei and Infosys, its key principles were:

- Joint Requirements and Solution Test Strategy On A Unified Collaboration Platform
- Continuous Integration at the Network Service Level
- Standardized Test Cases and Test Language
- Joint Solution (Service) Verification Environment

- Continuous Rapid Delivery at the Solution (Service) Level
- Real-time Customer Feedback Throughout The Development and Deployment Cycle

The AI & Data Analytics Project, within the stream "Artificial Intelligence for Operations - AIOps" has an Acceptance Testing activity. It does not specifically refer to VNFs, but these are assumed to be part of a modern Telcom operational environment. More information can be found at IG1190 AIOps Service Management suite.

6.2 Open Source Automated Testing Resources

6.2.1 ONAP

In the ONAP Guilin release, China Mobile led the development of an automated testing requirement. This effort focused on the establishment of a general automated testing platform by introducing automated testing tools and processes in each testing phases. By enhancing each module of ONAP, the overall framework of automated testing is realized.

1. Enhance SDC to implement the auto design of test topology
2. Enhance VF-C to implement test environment auto deploy
3. Enhance VTP to implement test task auto execution
4. Enhance VTP and integrate with OVP to implement test result auto analysis and certification

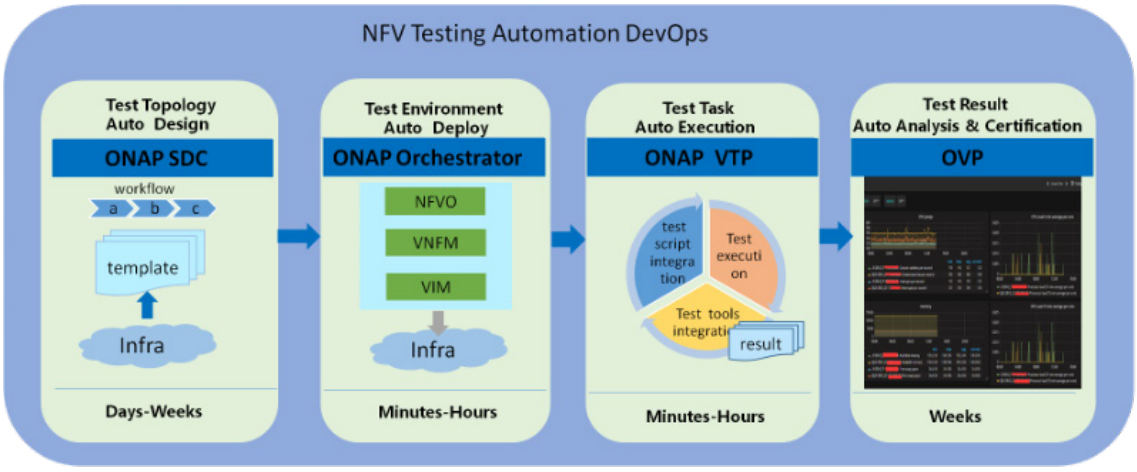


Figure 2: Sample Common Testing Framework

6.2.2 Anuket/OPNFV

The LFN's OPNFV oroject defined a number of tools to perform NFV Testing as part of its charter to develop testing tools and badging for NFV type workloads. When it merged with the CNTT NFV infrastructure taskforce to form the Anuket project, the charter expanded to include both NFV and NFV infrastructure testing frameworks and code, which includes unit, feature, and component, system level testing for development, automated deployment, performance characterization and stress testing.

As described in https://docs.opnfv.org/en/stable-gambia/testing/ecosystem/overview.html, there are several projects related to the Anuket testing ecosystem.

6.3 Available Tools and Interface Standards

6.3.1 ETSI TAP Test Platform

The TAP platform development is led by the ETSI organization and deployed in the ETSI Plugtest interoperability and verification center laboratory. The test scripts in this platform are based on Robot Framework and operate in an interface method. The functions of the TAP test platform include testing session creating, starting, executing, and terminating. Organizations participating in API conformance testing need to deploy test products to their local laboratories, apply for VPN, and connect to HIVE remotely, then log in to the TAP system through plugtests.

6.3.2 Other Open Source Test Tools

Testing tools in open source communities	Description
VTP	VTP is a vertical common test platform for various VNF tests, which can be used in different test stages, including CI/CD, LFN OVP certification testing, uploading, design, and active/passive testing. The platform provides the execution verification function of test cases, mainly including controller, CLI test scheduling center and test runner three functions.
Xtesting	Xtesting was originally proposed by Functest to achieve smoke and integration testing. Now it is mostly used to build CI/CD tool chains. The test platform supports a variety of test cases, including Python, Bash, unittest, Robot Framework and VNF, etc. The generated docker container is lightweight and can be easily used with any CI/CD tool chain.

6.4 Third Party Tools and Resources

While seeing rapid growth in open source test tools as outlined in previous sections, there are still many gaps that are not covered by open source projects. There are three classes of test tools that are often better covered by proprietary solutions.

- Test management
- Security testing
- Functional test

Test management

These test management tools are used to reproduce an environment with a test failure, track coverage and results over time. A broad set of tests cases need to be run for NFV applications across several configurations. Different test cases can require complex changes to the hardware and software. While there are open source tools to manage this, many of proprietary stacks provide a rich set of features and are tightly tied to existing test processes.

Security Testing

Many of the industry accepted tool for static analysis, fuzzing and penetration testing are proprietary solutions. Without using these tools, there is a risk of releasing software with known exploits.

Functional Tests

Hardware based test tools can still play a role within the test frameworks, since they provide a known baseline for performance and can provide additional instrumentation on the physical layer that are not always available with off the shelf hardware. For instance, with testing over an optical link, specialized hardware can detect different types of errors in cables and lasers. Some standards and interfaces require an intellectual property rights agreement to use them. Tools which rely on specific intellectual property may not be available as open source.

7 Glossary

OSC	Open Source communities
SDO	Standards Developing Organizations
CSP	Communications Service Providers
LFN	Linux Foundation Networking
EUAG	End User Advisory Group, a working group within the LFN