**SPIRENT**

# VNF Lifecycle Management

## Customer Profile

**Customer:** Cloud Service Provider (CSP)

**Industry:** Telecommunications

**Employees:** 22,500 (2016)

**Customers:** 3+ Million

## The Challenge

A CSP finds that rolling out new services, utilizing new client device types, requires a short configuration window. If dedicated network hardware is utilized, business opportunities can be lost. The CSP wishes to move to a software-defined network based on off-the-shelf servers. An appropriate software architecture allows service modules to be added and removed as effortlessly as Lego blocks. Network Functions Virtualization (NFV) transfers the network functions of dedicated hardware devices to software-based applications.

Virtualized Network Functions (VNF) are managed and orchestrated through the NFV Infrastructure by the NFV MANO, allowing a VNF to be instantiated, managed, scaled up or down and in or out, and terminated when no longer required. The efficiency of VNF lifecycle practices can be a critical determinant as to how successfully VNFs of different types are created and managed. For instance, Service Providers often receive upgrades, bug fixes and enhancements for any of the different NFV components. Even a small modification can cause a major instability issue when not detected early by the VNF Lifecycle Management test case system. The system used must be agile and adaptable while development is still in progress and, in addition, there are often many API differences amongst vendors. This lack of API consistency and the continuous development evolution of components adds to VNF an increasing complexity. The transitory nature of APIs virtually requires initial and ongoing testing of the VNF elements.

## VNF Lifecycle Management

The VNF Lifecycle verification architecture described on the next page assumes the existence of a well-structured NFV system that includes management and orchestration components. The NFVO MANO is the NFV management framework. The NFV Orchestrator (NFVO) is responsible for VNF orchestration, and gives the instruction of VNF instantiation and termination to the VNF Manager. The management of the virtualized resources of network, memory, compute and storage is provided by the Virtualized Infrastructure Manager (VIM). Validation of the VNF Lifecycle Management encompasses all major NFV components and is therefore closely tied to the NFVi, the VNF Manager, the VIM, and the VNF function.
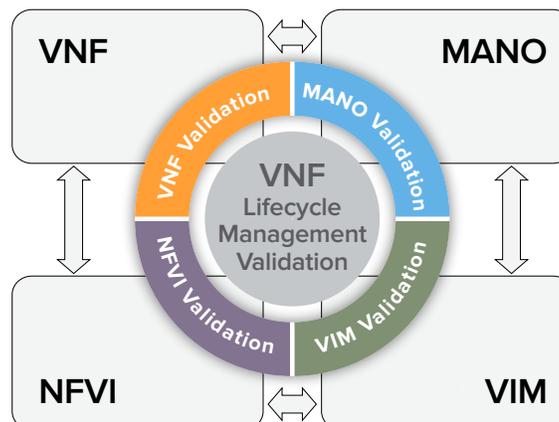


When validating a fully populated NFV system, the customer should also consider validating each component separately. Validating the NFVi and VIM ensures a stable and high performance infrastructure, which is essential for a solid foundation as the entire NFV system is built upon it. A thorough test suite tests each element of the infrastructure such as vCPU, vStorage, and vMemory. NFVi Benchmarking is one of the test types. The VNF validation focuses on the actual function or service the VNF provides. The MANO validation focuses on the feature set and scalability aspect of the NFVO and VNF Manager. Finally, the VNF Lifecycle Management is a systemwide test where the four NFV components orchestrate all the services provided by the VNF.

# VNF Lifecycle Management

## VNF Lifecycle Validation Overview

Test cases for VNF Lifecycle Validation can be divided into six domains: VNF Instance, Virtualized Resource, VNF Fault and Recovery, VNF Package, VNF Snapshot and VNF Performance Management.

| Domain | | Description |
| --- | --- | --- |
| VNF On-Boarding | VNF Instance Management | Validation of VNF Instance Management includes VNF State Management and VNF instantiation and termination |
| | Virtualized Resource Management | Validation of Virtualized Resource Management includes<br>■ Scaling in/out the VNF or VNFC(s)<br>■ Scaling up/down the VNF resources<br>■ Scaling of virtualized specialized hardware acceleration<br>■ Scaling of network bandwidth<br>■ Scaling of Storage |
| | VNF Package Management | Validation of VNF Package Management include<br>■ VNF Package Content and Signature<br>■ VNF Update and Upgrade<br>■ VNFD (Virtual Network Function Descriptor) validation |
| VNF Snapshot Management | | Validation of VNF Snapshot Management include<br>■ Creation of VNF Snapshot<br>■ Deletion of VNF Snapshot<br>■ Instantiation of a VNF Snapshot |
| VNF Fault and Recovery Management | | Validation of VNF Fault and Recovery Management includes<br>■ VNF Recovery or Healing<br>■ VNF Migration Management |
| VNF Performance Management | | Validation of Performance Management |
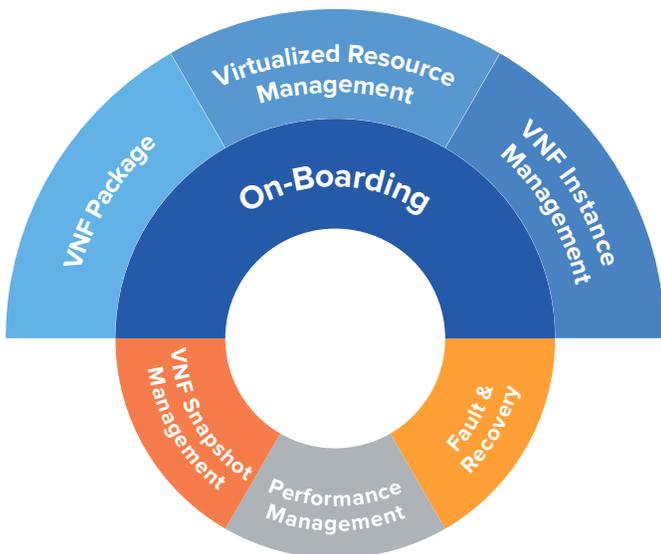


Some of the above domains are still in definition or refinement by the ETSI NFV working groups.

VNF Lifecycle Validation Test Automation takes advantage of the intuitive interface of the open source project Twister. Its framework allows for the complex setup of VNF Lifecycle Management to run with ease. As with any Linux-based environment, there are benefits to using it such as rapid deployment and it allows for a high degree of customization. The web-based GUI provides straightforward configuration, control, monitoring and reporting and is compatible with any browser or mobile platform. The reporting is integrated with Open Source project Kibana in the same graphical framework. Kibana is a data visualization and exploration tool and enables the user for an interactive data visualization that can be explored through customized dashboards (such as vResource statistics, VNF Lifecycle Performances, VNF Scaling Statistics). The backend of Kibana is ElasticSearch, a search engine designed for Big Data.

## MANO Challenges

Many companies combine the functionality of the NFVO and VNFM into a single entity, named MANO. This is different from the NFVO MANO as it includes VIM, VNFM and NFVO. Therefore, this use case applies the term "MANO" as an abbreviation for a module that combines the functionality of the NFVO and VNFM.

MANO vendors have started to adapt their implementations to the ETSI standard. By combining the NFVO and VNFM in a single module, the MANO vendor can deliver features quicker, though it carries the expense of eliminating the VNFM interface. Once stability and feature richness has been reached, the actual usage and POC tests may determine if a VNFM is required or not. Currently, MANOs comply only partially to the ETSI NFV standard.

As a result of the early stage of MANO compliance with ETSI, Spirent's VNF Lifecycle Validation test is built to be adaptable via Spirent's MANO API layer. MANO interface differences between MANO products result in interface inconsistencies. MANO vendors still add features, resulting in API customized enhancements. The MANO vendors also add proprietary features leading to differences across APIs. Therefore, a service for porting the code is required until we either have implemented libraries for vendor specific APIs to cover most relevant MANO providers or providers agreed to a common interface. Spirent's MANO API is quickly portable and adapts to the vendor's specific selection of MANO functionality.

## The Solution:

### VNF Lifecycle Validation Architecture

Spirent's VNF Lifecycle Validation architecture has been designed with portability and adaptability in mind. The diagram below illustrates the overall architecture. Spirent's Lifecycle Validation Test Cases have been implemented to call well defined and generic APIs: MANO API Layer, VNF API Layer, VIM API Layer, Traffic Automation API Layer, Element Management API layer, MANO API layer, and Metric Service API. By basing test suite on consistent and well-defined APIs, Spirent tools can help overcome differences in APIs offered by MANO vendors.
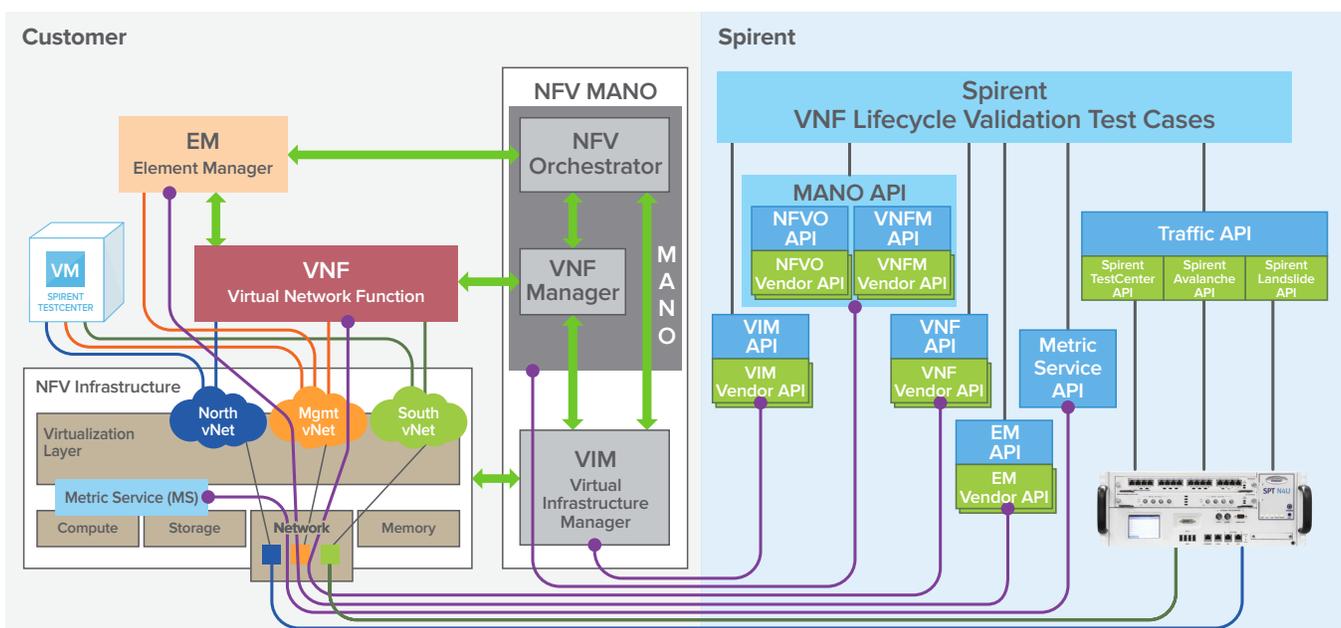
# VNF Lifecycle Management

Lifecycle verification begins with a test sequence implemented in the **VNF Lifecycle Validation Test Case Layer**. The test cases call well-defined APIs of four specific inter-related layers: the VNF API Layer, the MANO API Layer, the VIM API Layer, and Traffic Automation Layer. The APIs in these four layers configure, construct, control, manage, and retrieve information or results. The **MANO API** Layer exposes a generic API interacting with the NFVO and VNFM. It abstracts the vendor-specific lower layer and resolves and encapsulates any vendor-specific impacts. The vendor-specific access to the MANO is coded in the MANO vendor API, represented in green as NFVO API and VNFM API. This allows us to adapt the API to different vendor specific MANO interfaces without impacting the overall architecture and test sequence.

Both the **VNF API Layer** and the **VIM API Layer** resemble the NFVO and VNFM sublayers, in that they are both divided into two layers for common API abstraction and vendor-specific lower-layer functions. The top layer VNF or VIM API is the common layer and exposes a well-defined common interface to the upper layer. It abstracts the lower layer and resolves any vendor specific impacts. The vendor-specific access to the VNF or VIM is coded in the vendor API (green). This allows us to adapt to different vendor-specific VNF interfaces without impacting the overall architecture and test sequence. Note that in the three API layers, Spirent's separation of functions into two layers allows the test suite to abstract elements of the API that are unique to specific vendors.

Finally, the **Traffic API Layer** helps to validate the correctness of VNF functionality, a requirement of Lifecycle verification. The VNF Lifecycle verification can use Spirent TestCenter and Spirent TestCenter Virtual to generate and analyze traffic. With the Spirent TestCenter precision we are able to detect traffic packet loss and generate traffic up to line rate with ease. For circumstances where stateful traffic beyond Spirent TestCenter or Spirent TestCenter Virtual TCP traffic load is not sufficient, the Avalanche product (hardware or virtual) can be used.

Spirent can optimize its test tools to match expected protocol loads in both stateful and stateless environments. Because of the different types of traffic demands by the VNF to be tested, the Traffic API module gets separated into two layers, similar to the dual-layer architecture of the three previous APIs. In the case of the Traffic API, the common traffic API layer gets called by the VNF Lifecycle Validation Test Cases. It provides a generic and common interface to the upper layer. The common traffic API layer takes advantage of the Spirent TestCenter traffic layer, the Avalanche traffic layer or the Landslide traffic layer. Each layer will have its own configuration file to drive its functionality. The traffic API layer determines the appropriateness of Spirent TestCenter, Avalanche and Landslide for traffic generation, based on the configurations provided. It is important to emphasize that the architecture does not disallow a combination of traffic generated by Spirent TestCenter (hardware and virtual), Avalanche (hardware and virtual), and Landslide (hardware and virtual). This feature could be used to validate VNF service chains.

## Lifecycle Observations

Independent manufacturers and open-source projects are using ETSI NFV standards as a baseline architecture with which to develop new products. Feedback from these efforts, as well as requests for new features, reach the ETSI NFV working groups and result in standard modifications and additions. Proof-of-Concept NFV studies contribute to this process as well. Overall, the ETSI NFV standards have reached a mature baseline, but development has to catch up with these standards. Areas like VNF Descriptor and MANO undergo heavy development and modifications. MANO providers incorporate the features to fulfill the ETSI NFV standard. Feature set differences from vendor to vendor result in differences between MANOs, enhanced by the fact that no vendor is in full compliance with ETSI NFV. Spirent's VNF Lifecycle Validation Test Cases are highly adaptable and uniquely capable of addressing differences in MANO products across vendors.

ETSI NFV working groups will enhance and push the architecture forward. These additions and modifications require new or modified VNF Lifecycle Validation Test Cases. Spirent is committed to maintain and enhance the VNF Lifecycle Validation Test Cases.

## Benefits

Spirent provides a comprehensive set of test traffic methodologies in Spirent TestCenter, Spirent TestCenter Virtual, Spirent Avalanche, and Spirent Landslide to perform all Lifecycle Validation functional tests. The generic and flexible architecture of the VNF Lifecycle Validation allows the adaption and validation of a wide variety of VNF types via the different Spirent test products, whether it is virtual or dedicated hardware. The portability of the VNF Lifecycle Validation Test Cases does not end with the validation of ETSI-compliant features. Spirent's architecture, with its unique dual-layer API, allows adding new or non-standard based test cases with ease.

The highly adaptable Spirent VNF Lifecycle Validation supports different customers' demands. For a service provider, this system validates all components of the NFV system. Here many different VNFs must be validated on the same MANO, VIM and NFVi. Spirent's VNF Lifecycle Validation architecture addresses this via the VNF API layer. A MANO vendor is more interested in the execution of the VNF Lifecycle Validation against different VIM vendors in conjunctions with different VNFs. The primary test goal for the MANO component vendor is to validate the MANO. Through the VIM API layer and the virtual products Spirent TestCenter Virtual and CloudStress, the VNF Lifecycle Validation is capable of supporting these test cases. Finally, a VNF Vendor is interested in validating a VNF against different NFVi/VIMs and MANOs. The following table summarizes the customer's demands on the VNF Lifecycle Validation. See Appendix for detailed test topologies.

| Customer | MANO | VIM & NFVI | VNF |
|---|---|---|---|
| Service Provider | Fixed | Fixed | Multiple VNF |
| MANO vendor | Fixed | Multiple | Multiple VNF |
| VNF Vendor | Multiple MANO | Multiple | Single VNF |

The highly portable VNF Lifecycle Validation system in combination with a Spirent Service option to port the code onto the customer's NFV ecosystem allows the customer to accelerate testing and ensures the end product is high quality.

# VNF Lifecycle Management

## Appendix

Use Case Service Provider

This Service Provider selects a specific system of NFVi, VIM and MANO. To provide new features quickly to customers, they will rely on vendors to deliver many different VNFs with specific functionality. Here there are a large number of VNFs that require to be validated by the VNF Lifecycle Management. Spirent's VNF Lifecycle Validation allows support for many VNF vendor APIs and corresponding Traffic APIs. With this system, the service provider can ensure a high quality of VNFs when receiving new or updated VNFs. Figure 1 illustrates the basic system blocks for this type of use case.
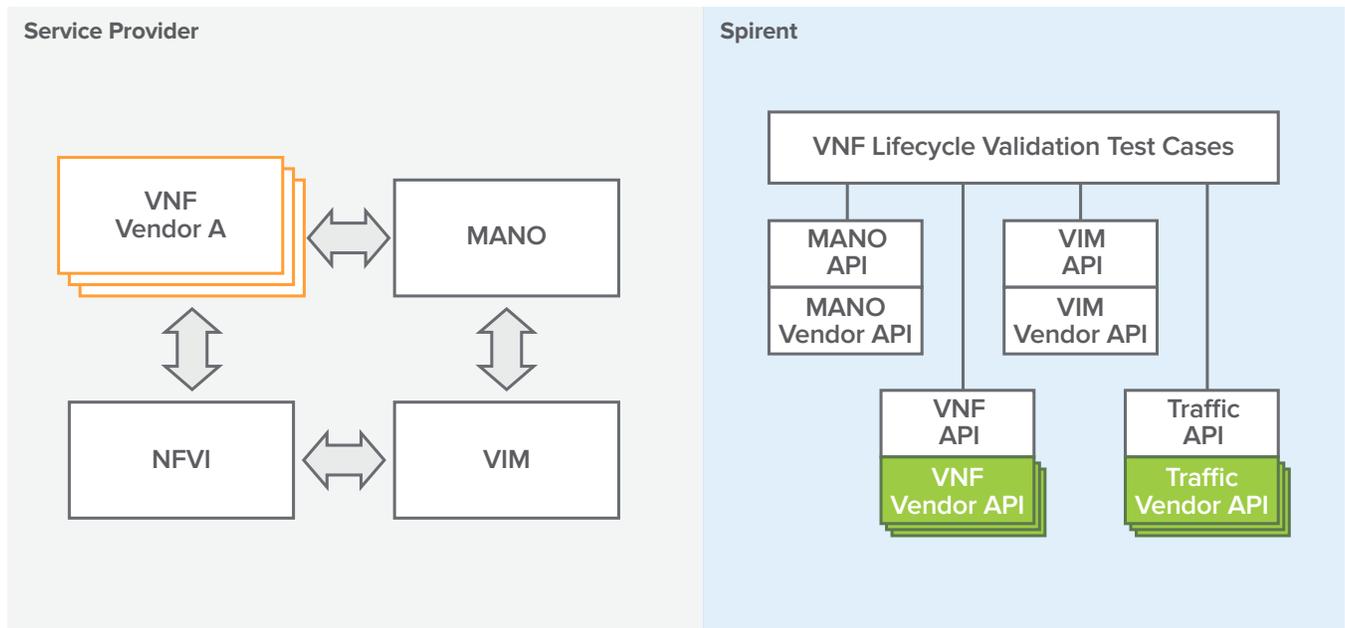


Figure 1. Well-defined, single NFVi, VIM and MANO with VNFs from different vendors

## Use Case MANO Vendor

This MANO vendor focuses on the interactions and compatibility of their product with many other different VIM/NFVi vendors. Here the VNF Lifecycle Validation system needs to work with multiple VIM vendors. A Spirent Test VNF enables the system to validate different types of VNFs against the MANO. Also supported are Scalability test cases for concurrent VNF instantiation using Spirent VNF Lifecycle Validation. Figure 2 represents an overview for this test case.
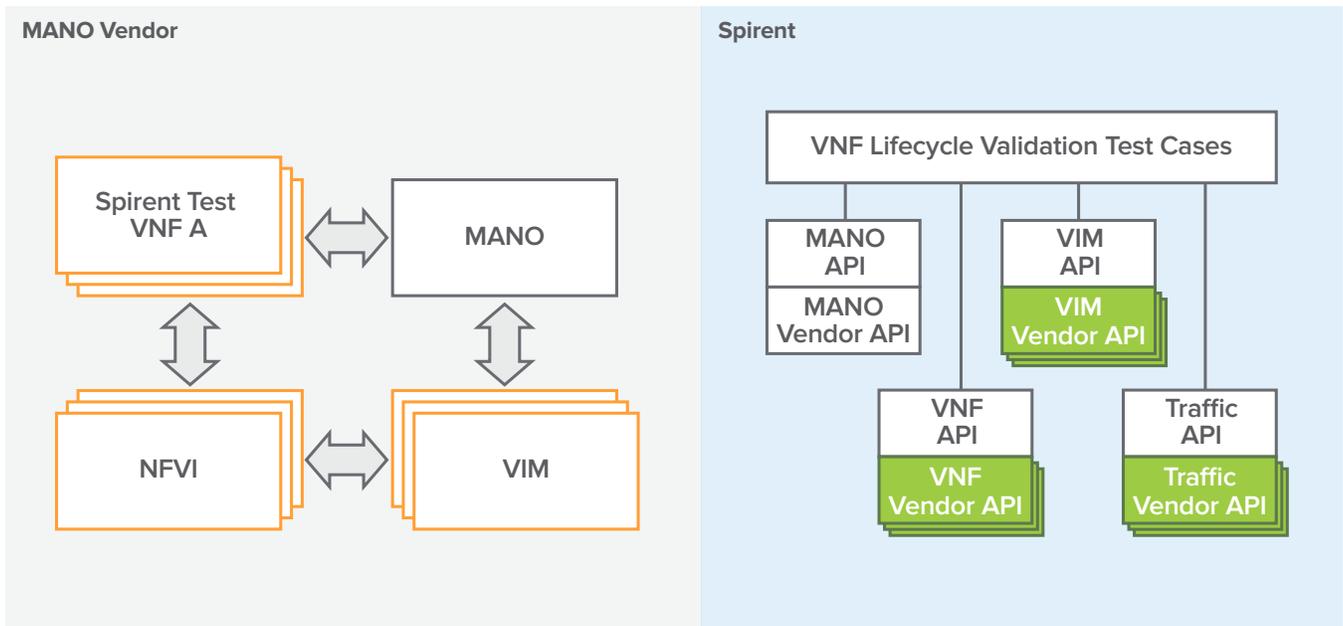


*Figure 2. Well-defined, single MANO, Spirent Test VNFs running on multiple NFVi & VIM*

## Use Case VNF Vendor

In this example, the VNF Vendor shows a great interest in the interoperability of their VNF against many MANOs and NFVi's. The test automation of VNF Lifecycle Management in conjunction with the support of many MANOs and NFVi's highly reduces the possibility of failing at the Service Providers premise. Spirent VNF Lifecycle Validation supports this type of testing with ease. Figure 3 illustrates the overall VNF Lifecycle system for a VNF vendor.
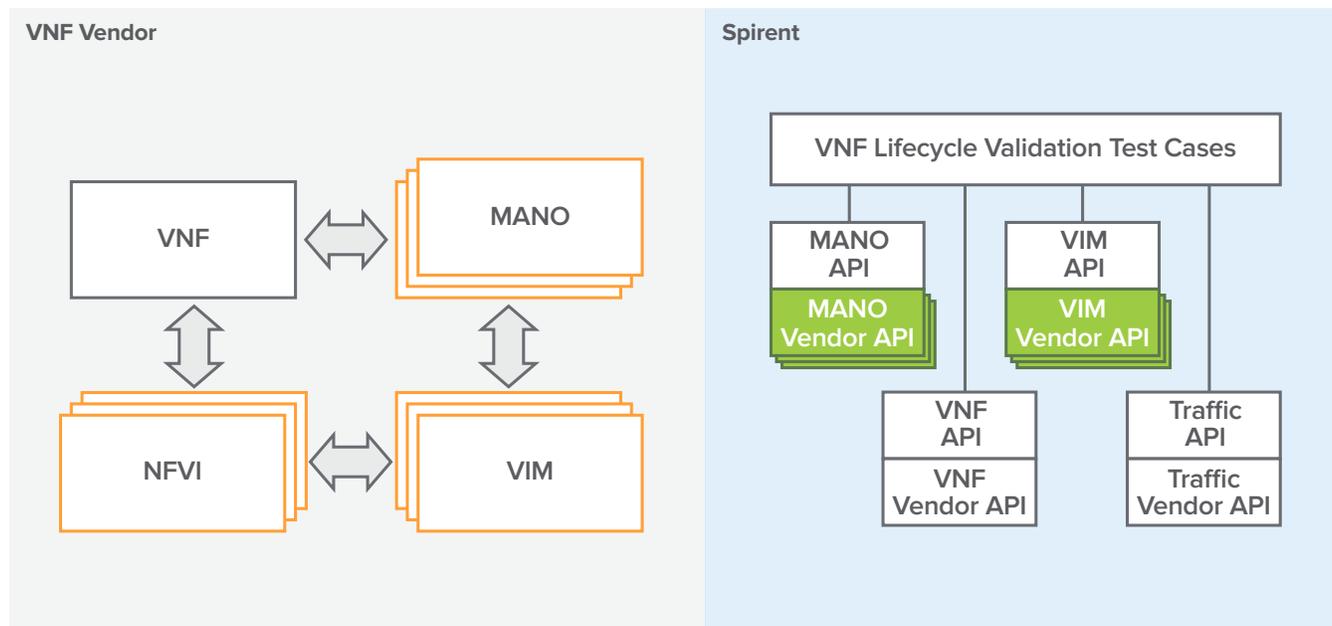
*Figure 3. Validation of VNF on multiple NFVi & VIM and MANOs*

### About Spirent

At Spirent Communications we help the world communicate and collaborate faster, better and more often. The world's leading communications companies rely on Spirent to help design, develop and deliver world-class network, devices and services. Spirent's lab test solutions are used to evaluate performance of the latest technologies. As new communication services and applications are introduced in the market, Spirent provides tools for service management and field test to improve troubleshooting and quality. Spirent also enables enterprises, institutions and government agencies to secure and manage their networks.

From wireline to wireless to satellite, Spirent offers a complete portfolio of solutions to enhance customer Quality of Experience.