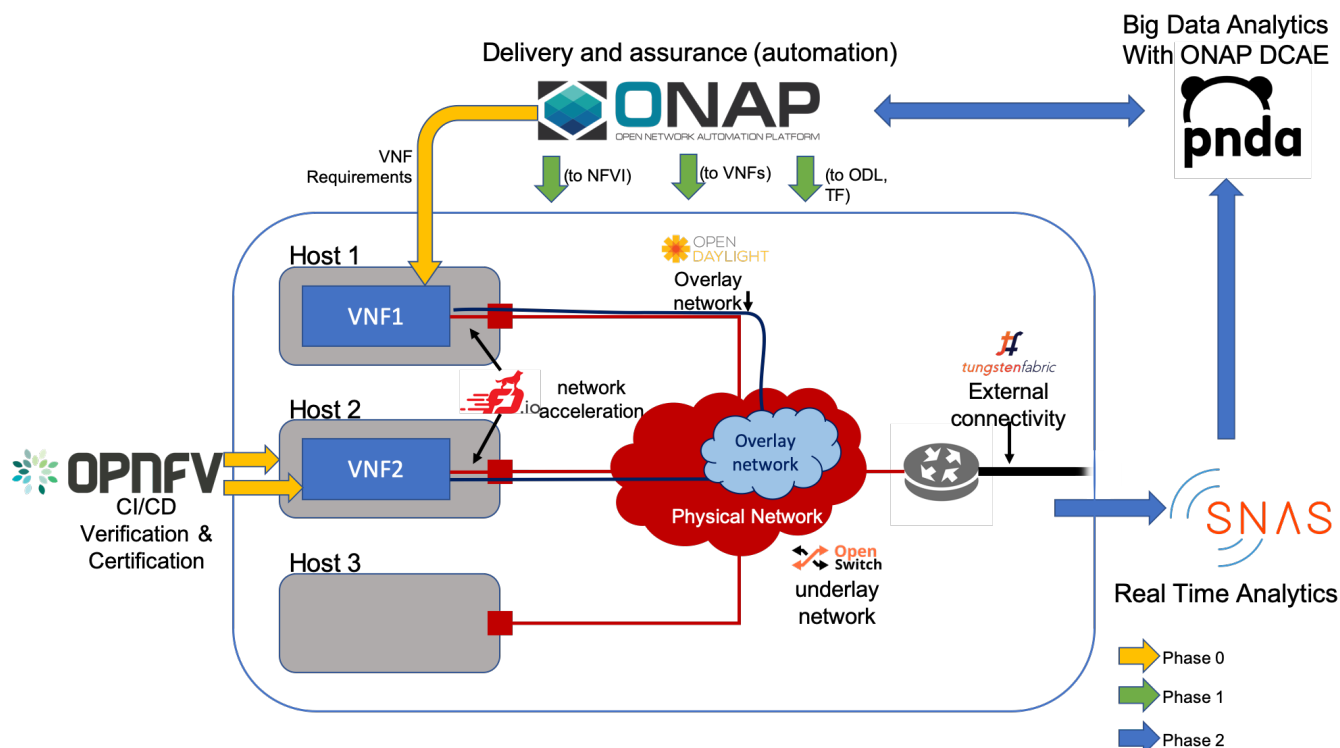


4. LFN Projects Symphony (E2E Use Case and integration points)

As highlighted in the previous chapters, LFN projects are designed to be part of an E2E modern network. As such, each of the projects has integration points with other LFN projects, as well as external open source projects. this section will highlight some of those integration points and the value they create.

This section aims to present an end-to-end use case where the 8 LFN projects work in harmony to deliver a "service" that includes VNFs, connectivity and analytics-powered assurance as shown in the following picture:

Picture needs an update: VES - VNF Event Stream - from VNF directly to ONAP for VNF feedback loop



In this example, 2 VNFs (for the sake of simplicity, provided by the same vendor) must be deployed on top of an NFVI (e.g., OpenStack), be interconnected and provided with external connectivity to the Internet. Moreover, the VNFs require network acceleration and the whole service must be assured using Analytics driven closed loop operations.

Using the 8 LFN project, an end user (e.g., a carrier) can realize the above as follows:

Phase 0 - Building the network infrastructure and preparing the network functions

Following the CNTT Reference model, the operator decides which CNTT [OpenStack based Reference Architecture](#) may best suit its needs. This is followed by picking a set of infrastructure components that fit a CNTT Reference Implementation of choice. The infrastructure is built using the deployment tools and CI/CD provided by OPNFV. Next, the infrastructure is certified using the CNTT RC and OPNFV CVC.

Several LFN projects may be used as infrastructure building blocks for addressing the needs of network functions, such as high throughput/low latency networking:

- **OpenDaylight and Tungsten Fabric** can be used as [3rd party SDN solutions](#) to provide network connectivity.
- **Open Switch (OPX)** can be used to configure the physical (underlay) network that connects the physical hosts used to deploy OpenStack. The network topology may follow the leaf and spine topology as a physical infrastructure is recommended in the [requirements of physical infrastructure](#) of the CNTT Reference Architecture.
- **FD.io** provides data plane network acceleration through its Vector Packet Processor (VPP).

VNFs are prepared for deployment and inclusion in network services:

- An NFV vendor pre-validates and certifies a couple of VNFs (i.e., VNF1 and VNF2) through the **OPNFV** Verification Program (OVP). The OVP program cover the following certification aspects of VNFs:
 - Compliancy of VNF Packaging to industry standards such as ETSI and ONAP.
 - The ability to onboard and life cycle manage VNFs on a given cloud platform.
 - Limited performance characterisation of VNFs.

- The NFV vendor ensures that the VNF complies with the [ONAP VNF requirements](#). This will enable ONAP to properly control the lifecycle of the VNF as part of a network service.

Phase 1 - Network service design and deployment

At design time, ONAP is used to onboard the VNFs that are compliant with the ONAP requirements and pre-certified using the OPNFV CVC . Those compliant resources can later be used to design any E2E service using **ONAP** Service Design and Creation (ONAP SDC).

At runtime, ONAP orchestrates the deployment of the whole service either through ONAP internal functions/components or leveraging the capability to interwork with 3rd party components.

In particular, ONAP Service Orchestrator (ONAP SO) instructs the underlying ONAP functions in order to deploy all of the elements that compose the end-to-end service.

ONAP deploys the VNFs in the available NFVI and the overlay network connecting them using ONAP SDN-C. SDN-C uses its **OpenDaylight** based architecture to model and deploy the L1-L3 network. Next the ONAP APP-C is used to configure the network functions and their L4-L7 functionality. This is also done leveraging the **OpenDaylight** architecture.

OpenDaylight may be used to stitch together the physical switch fabric of the infrastructure with the virtual networking in the NFVI (e.g. OpenStack Neutron). Through the **OpenDaylight** Northbound Interface, ONAP-SDNC is able to instruct the OpenDaylight SDN controller for underlay network management. The southbound interfaces (e.g. NETCONF, etc) support interactions with OpenSwitch running on the leaf and spine fabric switches in the NFVI.

By leveraging its SDN-C southbound interface, ONAP instructs **Tungsten Fabric** to create the external connectivity that will enable customers to "consume" the services offered by the VNFs. The predefined policies that will control the lifecycle of the network service are designed using the ONAP design-time components such as SDC and CLAMP.

Phase 2 - Network service operation

Naturally, being a Network Automation Platform, **ONAP** plays a central role in the delivery and assurance of the service. The VNFs report their performance and fault data to the ONAP DCAE using the VNF Event Streaming (VES) interface. This information is constantly analyzed and may trigger predefined policies that were created at design-time. The policies are used to invoke closed loop automation actions such as scaling and healing of service components in order to assure the required SLA and respond to changing demands and network conditions.

Finally, closed loop operations may be further enriched by combining LFN Real Time Analytics capabilities of **SNAS.io** and the synergies offered by ONAP and **PNDA.io**. Information about changes in the network topology gathered by **SNAS** can be used to trigger **ONAP** policies that will spawn more instances of packet routing network functions. The data analytics capabilities of **PNDA** may be used to trigger **ONAP** policies based on data streams produced by all layers of the infrastructure as well as the network functions. **ONAP** may respond to an infrastructure issue detected by **PNDA** by migrating VNFs from an affected location to one that is healthy and has the available resources.