ORCHESTRATION WHITE PAPER

Lumina Networks, Inc.
2077 Gateway Place, Suite# 500
San Jose CA 95110

+ 1 (669) 231-3838
+ 1 (800) 930-5144

marketing@luminanetworks.com
www.luminanetworks.com

The Open Network Automation Platform - How Lumina Networks' SDN Controller Helps Deliver Network Intent

The Linux Foundation's Open Network Automation Platform (ONAP) is the networking industry's most broadly supported service orchestration initiative. With deep expertise in OpenDaylight, upon which ONAP is being built, Lumina Networks is ideally suited to provide an open source network control tool-set in a way that supports the capabilities and flexibility that service providers want and need.

Introduction

The trend in carrier-grade networking is toward virtualization and software-defined networking (SDN). Every service provider (SP) is moving to separate the software layer of networking from the underlying hardware to reduce their dependence on expensive, purpose-built equipment and to gain agility in their operations. This abstraction process allows them to move their now-virtualized networking software to general purpose compute platforms—the same kinds of inexpensive white box hardware and bare metal platforms that applications and other services run on today.

When the network itself is virtualized, it runs in software as if it is just another application. This creates a challenge in that the operation of the network must be coordinated closely with the operation of the applications and services that normally run on the network, as well as with the underlying infrastructure. "Orchestration" is the industry term for the coordinated management of all the network and compute elements needed to deliver a virtualized network service. While orchestration may involve functions like provisioning or management, the hallmark feature of orchestration is that it involves the coordination of different elements simultaneously. Because this type of coordination can get exponentially complex with the size of the network, automation tools are critically important to make orchestration functions possible.

Any tool that is going to orchestrate the operation of a software-defined network must be able to support any network the service provider wants to run, as well as any application or service that is to run on that network. If the tool can't support "any network, any application/service," it's really just another proprietary tool, and that's what service providers are trying to avoid. Proprietary commercial tools have limitations in that they cannot be modified easily, even upon discovery of a defect, and there is limited ability to customize for a specific customer's needs.

Network orchestration can only properly be achieved by using an open source platform that can direct any and all of the elements of the network. IDC appears to support this notion in its Technology Spotlight report Why Intent-Based Networking Demands an Open Platform: [1]

[An] open platform uses published application programming interfaces (APIs) to integrate and exchange intelligence with adjacent network and IT services. It integrates with other IT resources, security tools, line of business applications and third-party infrastructure. An open network platform must be able to take inputs from these sources and automatically deliver the network resources needed.

Lumina Networks understands how to take an open platform – OpenDaylight – to create service level abstraction and to orchestrate the operations of disparate network elements. Lumina is ideally suited to provide an open source network orchestration toolset – in particular, the control plane part of it – in a way that supports the capabilities and flexibility that service providers want and need. This white paper outlines industry efforts to collectively create an open source orchestration platform, and also lays out a bit of roadmap and direction on Lumina's efforts to develop a software-defined network controller (SDN-C) to drive this platform. Lumina's SDN controller is fully an open source component which also offers extensions and microservices provided by Lumina to allow service providers additional flexibility with network orchestration.

The Importance of Orchestration

"Orchestration" is a broad set of networking functions required to deploy and manage virtual networking functions (VNFs). Orchestration technology is now considered to be key to the deployment of VNFs and vital to the success of network function virtualization (NFV). While many VNFs came to market without orchestration as part of the solution, it can be observed that when a complete set of orchestration tools is included, acceptance of deployment of the product is accelerated. A recent example is SD-WAN, where the solutions that came to market were deployed quickly, in part because the orchestration functions are built-in as part of the solution.

Orchestration technologies have shifted from a tactical to a strategic part of networking. This sentiment is validated by Jennifer Clark, Vice President of Research at 451 Research:[2]

"Network orchestration … has been a tactical purchase by the operators, focused on managing the virtual network functions within a data center. As the software-defined networking and network functions virtualization implementations expand to multiple datacenters and multiple networks, operators will acknowledge network – and service – orchestration as the most strategic component of their virtualized infrastructure and the first step to a next-generation OSS."

Orchestration is the process of automating the lifecycle management of virtual network devices and making sure that the devices can be created and configured, that they are able to communicate with other devices in the network so that services can be configured, and

---

[1]Brandon Butler, Senior Research Analyst, IDC, Why Intent-Based Networking Demands an Open Platform, June 2018

[2]Jennifer Clark, Vice President, 451 Research, "Thought Leadership – Orchestration and OSS, Beauty Meet Beast," Sept 2015

that traffic can be passed across the network as necessary. The goal here is to automate all the operations of the network, from lifecycle management, to configuration, to monitoring, to sending traffic. All this can be done with the click of a button.

An important element of SDN orchestration is the ability to monitor the network and automate connectivity. Many global service providers are looking for software that integrates orchestration, fulfillment, control, performance, assurance, usage, analytics, security, and policy of enterprise networking services based on open and interoperable standards. This approach is also known as lifecycle service orchestration (LSO).

In the future, SDN orchestration systems will provide the important "glue" between a wide range of technologies that enable cloud-based network and communications services. It is expected that they will provide the coordination and automation technology that bridges the gap between telecom systems, data center resources, OSS systems, and the customers looking to purchase cloud-based technology and network services.

An aspirational goal beyond orchestration is intent-based networking, in which a human tells the network through a high-level language what outcome, or intent, he is looking for. No detailed configuration instructions are provided. Instead, the network determines the configuration and policies required to implement the intended services (See sidebar for more on this topic.)

ONAP: What It Is and How It Came About

The Linux Foundation's Open Network Automation Platform (ONAP) is the industry's most broadly supported service orchestration initiative. ONAP has its origination in early 2017 as the combination of AT&T's Open ECOMP initiative and the Linux Foundation's Open-O project. These two groups combined their efforts to form ONAP.

ONAP provides a comprehensive platform for real-time, policy-driven orchestration and automation of physical and virtual network functions that will enable software, network, IT and cloud providers and developers to rapidly automate new services and support complete lifecycle management.

ONAP provides scaling for NFV over OpenStack using automated orchestration. The orchestration platform coordinates with virtual infrastructure managers (VIMs), such as OpenStack, to provide application and tenant connectivity within and between clouds. In short, ONAP is the platform above the infrastructure layer that automates the network, and it has broad support across the industry with service providers and vendors.

The primary genesis of ONAP is a collective of projects at AT&T known as ECOMP, or Enhanced Control, Orchestration, Management & Policy. Originally intended solely for AT&T's private use, the ECOMP projects aimed to provide a framework to standardize all network interfaces coming in or going out of AT&T's complex network. ECOMP is already in production within the AT&T network.

AT&T chose to give a considerable amount of its ECOMP source code to the open source community, where it can be further expanded and developed by global contributors with a vested interest in an open platform such as this. Since then AT&T's body of work has gone beyond that which was envisioned. In 2016 and 2017, Open ECOMP combined efforts with a similar project called "Open-O" to form the Linux Foundation's ONAP project.

The ONAP set of projects has strong support from at least 60 leading operators and networking companies, including AT&T, Verizon, AMDOCS, China Mobile, Ericsson, Vodafone, Huawei, China Telecom, Nokia, Orange, Lumina Networks, and others. The breadth of industry support behind ONAP underscores the urgent need for operators to find a streamlined way to launch new virtual network functions and services at the click of a button, while also leveraging their existing network investments. Many would like to see ONAP become the de-facto industry standard for automation and orchestration.

To illustrate just how ambitious and extensive the ONAP project is, here is a diagram that reflects the overall architecture of the automation platform.

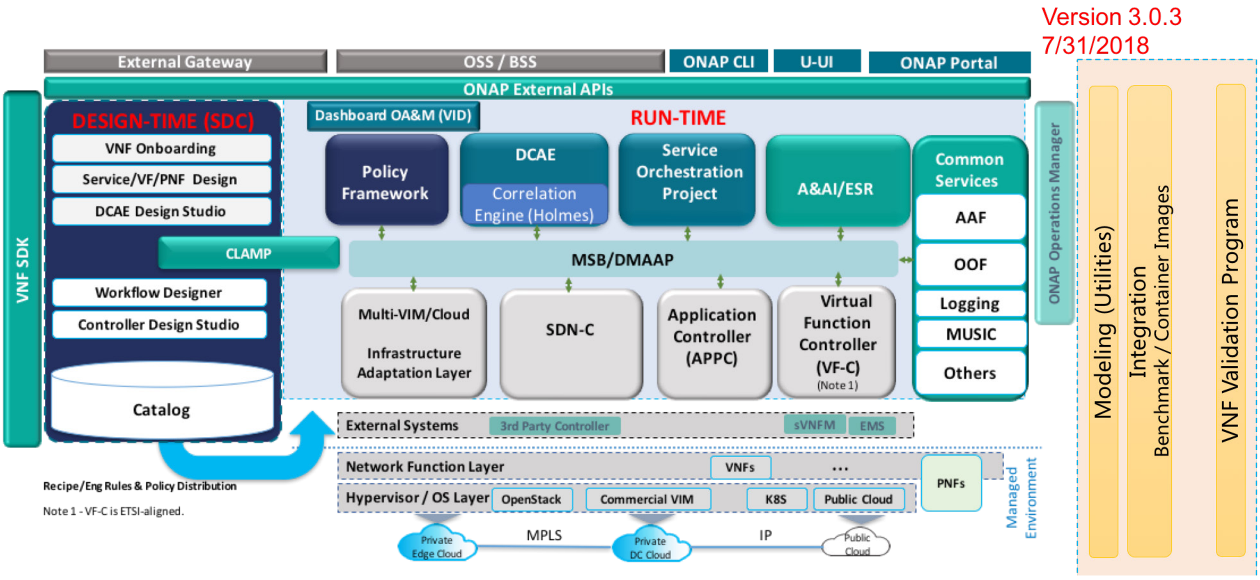


***Figure 1:** The OANP Architecture diagram*
***Source:** The Linux Foundation*

For the purposes of this white paper, Lumina has simplified the onerous ONAP diagram to illustrate many of the components to be discussed herein.

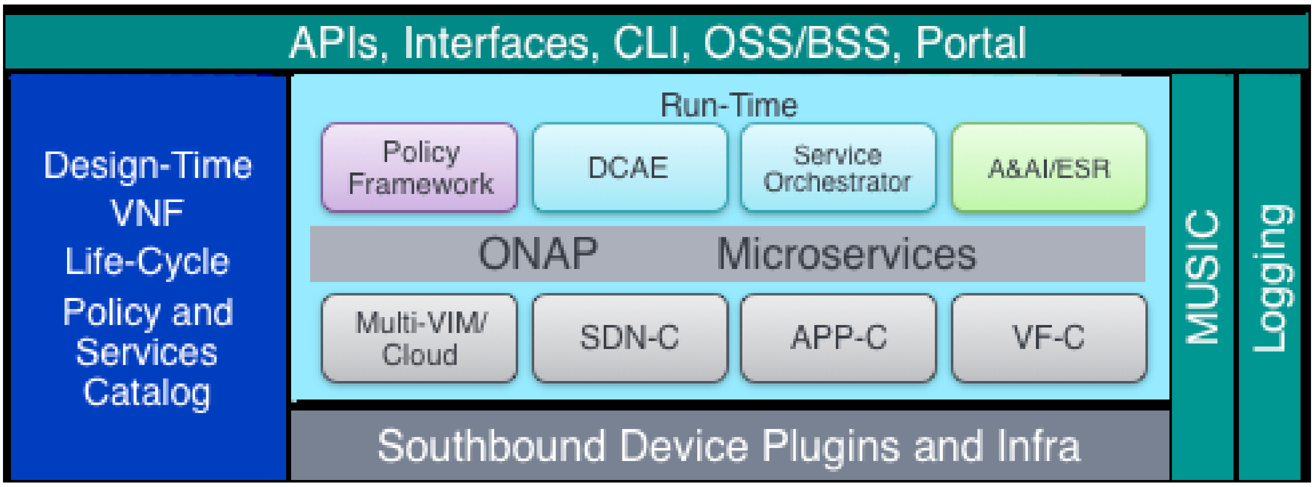


***Figure 2:** Simplified ONAP diagram*
***Source:** Lumina Networks*

An Overview of Key ONAP Projects

There are several ONAP projects that are key to the overall development of the platform, summarized as below. Awareness of these projects is helpful to understanding how ONAP will work. Lumina Networks is participating in the development of some of these projects, bringing our deep expertise in OpenDaylight to the working teams.

SDN-C

The SDN-C (software-defined network controller) project provides a global network controller, built on the Common Controller Framework, which manages, assigns and provisions network resources. As a “global” controller, the SDN-C project is intended to run as one logical instance per enterprise, with potentially multiple geographically diverse virtual machines / Docker containers in clusters to provide high availability. The project also supports the ability to invoke other local SDN controllers, including third party SDN controllers.

According to project lead Dan Timoney of AT&T, the SDN controller is based on the OpenDaylight platform with a few notable additions. One addition is the Service Logic Interpreter, which is code that allows the execution of directed graphs (DGs). A DG allows you to capture the behavior that you want the controller to implement without having to write code in Java. Directed graphs can be loaded in real time and it’s easy to make changes to them.

In ONAP SDN-C, a tool is used to create the directed graphs using IBM’s Node-RED, which is an open source project, also used for Internet of Things (IOT) Applications.

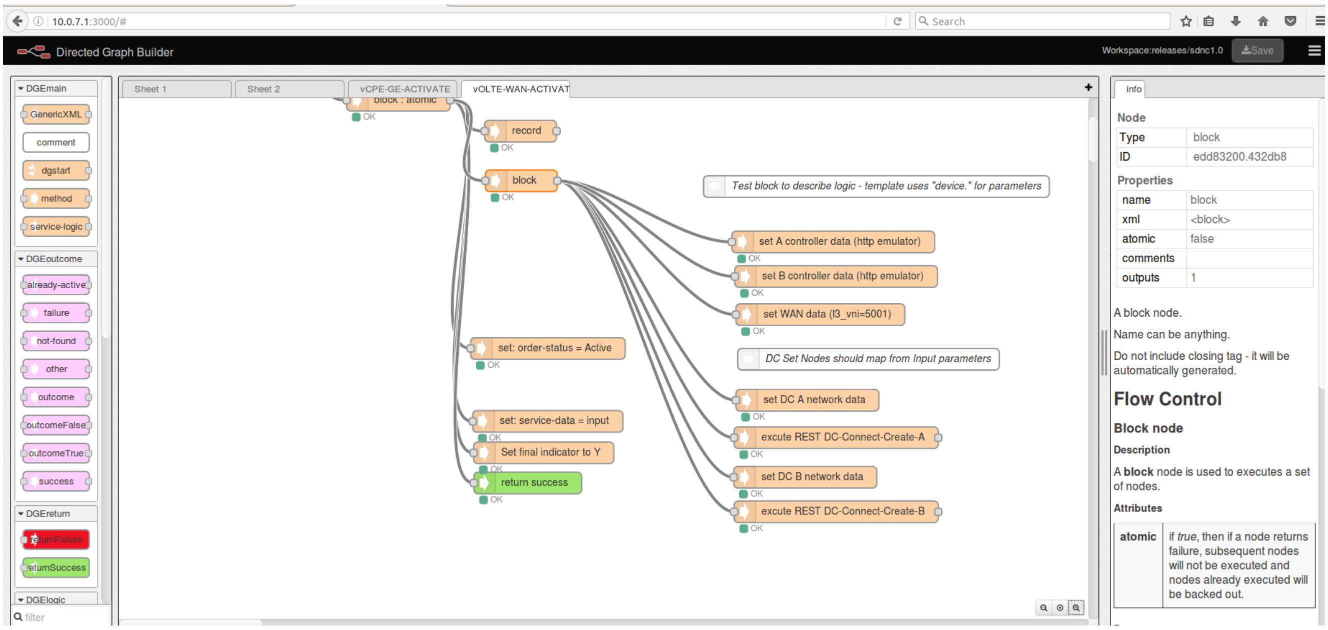


***Figure 3:** Screenshot example of a Directed Graph using IBM's Node-RED tool. VoLTE Use Case.*

***Source:** Reefwing Robotics Website, Author David Such. http://reefwingrobotics.blogspot.com/2017/04/node-red-dashboard-for-raspberry-pi.html*

The main problem the SDN-C is trying to solve is to simplify the process of setting up network connections. This has been a very labor-intensive process in the past, but now an SDN controller automates all that activity, making it less labor-intensive, less costly, and less error-prone. Messages that come into the SDN-C platform come in through a north-bound API. The code for this interface is typically generated by YANG models generated by the OpenDaylight YANG tools compiler.

APPC

The Application Controller (APPC) performs functions to manage the lifecycle of VNFs and their components providing model driven configuration. It abstracts cloud/VNF interfaces for repeatable actions, uses vendor agnostic mechanisms (NETCONF, Chef via Chef Server and Ansible) and enables automation. The APPC term comes from the fact that virtual network functions are managed as applications in a virtual network. Monitoring the lifecycle of VNFs and applications in a virtual network is absolutely critical since there is no other way to know if a function is present and how to remediate the situation if the function discontinues for some reason. Lifecycle includes instantiation, operational state monitoring and discontinuation for no longer needed functions.

ccSDK

The common controller software development kit (ccSDK) project provides a common set of reusable code that can be used across multiple controllers. For example, the SDN-C; the APPC; the Data Collection, Analytics, and Events (DCAE) subsystem; ONAP Operations Manager; and ONAP controller can reuse common pieces from this framework.

While controllers are encouraged to use the common controller SDK libraries, usage of this common code is optional. The goal is to provide code that is sufficiently flexible that there is no need for controllers to implement their own custom solutions, although it is recognized that there are valid reasons why specific controllers might need to implement their own solutions.

VNF-SDK

VNF onboarding is a challenge across the industry because of the lack of a standard format and information model for VNFs. The VNF software development kit (VNF-SDK) project is building an ecosystem for ONAP-compatible VNFs by developing tools for vendor continuous integration / continuous delivery (CI/CD) toolchains and as well as tools for validation and testing.

This SDK will deliver automation tools for VNF product specification, packaging, publication and ingestion, as well as for package validation, lifecycle test (framework), and functional test (framework). There will be a Reference Repository for VNFs to enable CI/CD without dependency on service provider ingestion. The functionality supplied by this project is intended for use by NFV operators, VNF product developers and VNF product DevOps teams.

DCAE

DCAE is the umbrella name for a number of components collectively fulfilling the role of Data Collection, Analytics, and Events generation for ONAP. The architecture of DCAE targets flexible, plug-able, microservice oriented, model-based component deployment and service composition. DCAE also supports multi-site collection and analytics operations which are essential for large ONAP deployments.

SDN-R

The SDN-R (R stands for “Radio”) subproject adds features/functionality to the OpenDaylight-based controller SDN-C that is built on the Common Controller Framework to configure and control wireless resources. The overall objective is to develop and certify an OpenDaylight-based controller capable of supporting simplified 5G use cases, including a meaningful subset of features and functionality to configure and control wireless resources, both PNF and VNF. Controlling the radio endpoint for network slices (virtual network segments) is a key part of what the SDN-R will do.

Multi-VIM/Cloud

ONAP needs underlying virtualized infrastructure to deploy, run and manage network services and VNFs. Service providers always look for flexibility and choice in selecting virtual and cloud infrastructure implementations; for example, on-premise private cloud, public cloud, mobile edge, or hybrid cloud implementations, and related network backends. It’s important that ONAP provide compatibility with the various cloud infrastructure managers needed for common orchestration use cases.

The Multi-VIM/Cloud ("VIM" is virtual infrastructure manager) project aims to enable ONAP to deploy and run on multiple infrastructure environments; for example, OpenStack and its different distributions (e.g. vanilla OpenStack, Wind River, etc...), public and private clouds (e.g. VMware, Azure), and microservices containers, etc.

The Multi-VIM/Cloud project will provide a Cloud Mediation Layer supporting multiple infrastructures and network backends so as to effectively prevent vendor lock-in. This project decouples the evolution of ONAP platform from the evolution of underlying cloud infrastructure, and minimizes the impact on the deployed ONAP while evolving the underlying cloud infrastructures independently.

MUSIC

The Multi-site State Coordination Service (MUSIC) is aimed at achieving high availability, or "five 9's" reliability as it's often called by service providers. ONAP components need to work in a reliable, active-active manner across multiple sites. A fundamental aspect of this is state management across geo-distributed sites in a reliable, scalable, highly available and efficient manner. MUSIC provides a scalable sharded eventually-consistent data-store (Cassandra) wherein the access to the keys can be protected using a locking service that is tightly coupled with the data-store. ONAP components can use the MUSIC API directly to store and access their state across geo-distributed sites. This API enables ONAP components to achieve fine-grained flexible consistency on their state.

Lumina Networks' ONAP-Compliant Components

Lumina Networks has several existing technology components that have a projected role in the ONAP platform. While they are not fully ONAP-compliant at this writing, engineers are actively working to adapt the components so they can become some of the "Lego-block pieces" that fit into the overall ONAP scheme. The following information should be considered roadmap direction that is subject to change.

Lumina is adapting the Lumina SDN Controller (LSC) into an SDN-C compliant package that can plug into ONAP. LSC is a true non-forked upstream distribution of OpenDaylight that has been thoroughly tested and quality assured. It is in active use today in non-ONAP applications such as network configuration, SD-Core and telemetry. Following successful deployment at several large service providers, Lumina is currently in the process of ensuring LSC can operate as an ONAP SDN controller. Specifically, Lumina is developing a containerized version of LSC, and once it is ready, we will recertify the virtualized versions of switches and routers (i.e., VNFs) as well as the physical devices we support with the SDN-C.

Lumina will ensure that our clustering works with MUSIC because the SDN-C may need to support multiple locations, interacting with LSC elements all over the world or operating in all sorts of different clouds. Those elements need to be coordinated together – which is what MUSIC does – so Lumina will make sure that our version of the SDN-C works with MUSIC.

Lumina will ensure that the telemetry and alarming services that we have today work with the DCAE, which is ONAP's monitoring system, and that our existing VNF Manager works with ONAP applications.

Two Key Use Cases of Orchestration

Ultimately there will be numerous use cases for orchestration enabled by the ONAP platform. Given the scope and complexity of the ONAP projects as well as the number of companies involved in contributing their code, there would have to be significant benefits associated with the existing and future use cases to make the investment worthwhile. Keep in mind that ONAP is in its infancy, and developers are taking their time with proof of concept projects to validate the overall architecture, design, and many interface points. There also are relatively new open source projects such as DANOS (Disaggregated Network Operating System) and Akraino Edge Stack that will standardize parts of the following use cases. Nevertheless, it is useful to consider generalized use cases of ONAP orchestration.

In the most recent release of ONAP, two important use cases have been put forward: 5G Network Deployment, Slicing, Optimization and Automation, and Virtual Customer Premise Equipment.

5G Network Deployment, Slicing, Optimization and Automation

Although there are similar automation projects for the telecom industry sponsored by The European Telecommunications Standards Institute (ETSI) and TM Forum, the Linux Foundation's ONAP project is drawing significant attention from communications service providers and wireless operators around the world. In fact, operators committed in some way to ONAP now account for 60% of the global mobile subscriber base. As one executive from Ericsson noted, "Service agility, network agility and automation are key for NFV and SDN transformation leading to 5G networks." Given this strong interest from operators, the recent Beijing release of ONAP (i.e., Release 2) includes platform capabilities to support operators' 5G requirements.

There are actually three sub-cases in the proposed 5G use case for ONAP. Each one identifies how it can leverage unique ONAP capabilities and identify potential gaps or challenges. Demonstrating ONAP capabilities in these areas will increase service providers' confidence in ONAP and could identify potential areas of improvement in ONAP. The three sub-cases are as follows:

ONAP Support for 5G Radio Access Network (RAN) Deployment – Disaggregated 5G RAN consists of hybrid network elements (PNF and VNF) and will require a cloud infrastructure deployment at the edge. This sub-case will expand and enhance ONAP capabilities to support different lifecycle management aspects of PNF, including enhancements in SDK/certification, PNF onboarding, and SO, DCAE, and Active & Available Inventory (AAI) support for PNF. Moreover, ONAP's multi-cloud layer's scope needs to be expanded to support Edge Cloud, which could have a different flavor of hardware, different networking requirements and support.

ONAP Support for Network Slice Segments, Network Slices, Services riding on various network slices – Many providers are interested in using ONAP to manage the general concept of Slicing—extending the cloud notion of sharing network/

compute/storage to sharing network functions and services implemented across PNFs and VNFs. Supporting such a generic slicing concept could require ONAP enhancements in the areas of SDC/AAI modeling and service definition, lifecycle management of slices (whether done within the existing SO/ controller context or via a new layer of control such as a slice controller) since slices could have their own state, metrics and scaling procedures different and distinct from those of the underlying network functions (e.g., service/slice across multiple network functions from different providers).

ONAP Support for Network Optimization and Optimization Framework – Many providers are interested in flexibly deploying network optimization functions from various vendors into the ONAP framework. An ONAP Optimization Framework (OOF) is needed to support two aspects. The first one is for the optimal placement of 5G virtual network functions. The second one is for performance optimization, where Self-Organizing Networks (SON) is a key functionality. Since optimization problems in general have a common structure, a framework exploiting this commonality would render significant efficiency in creating optimization functions in these two areas. Also, when different optimization capabilities are deployed, there is a need for potential conflict resolution across these different schemes (both at design and run time). Hence the OOF should support policy-driven optimization capabilities that can be quickly developed and on-boarded. To support OOF, the data collection/processing in DCAE needs to satisfy the needed latency requirements based on various use cases.

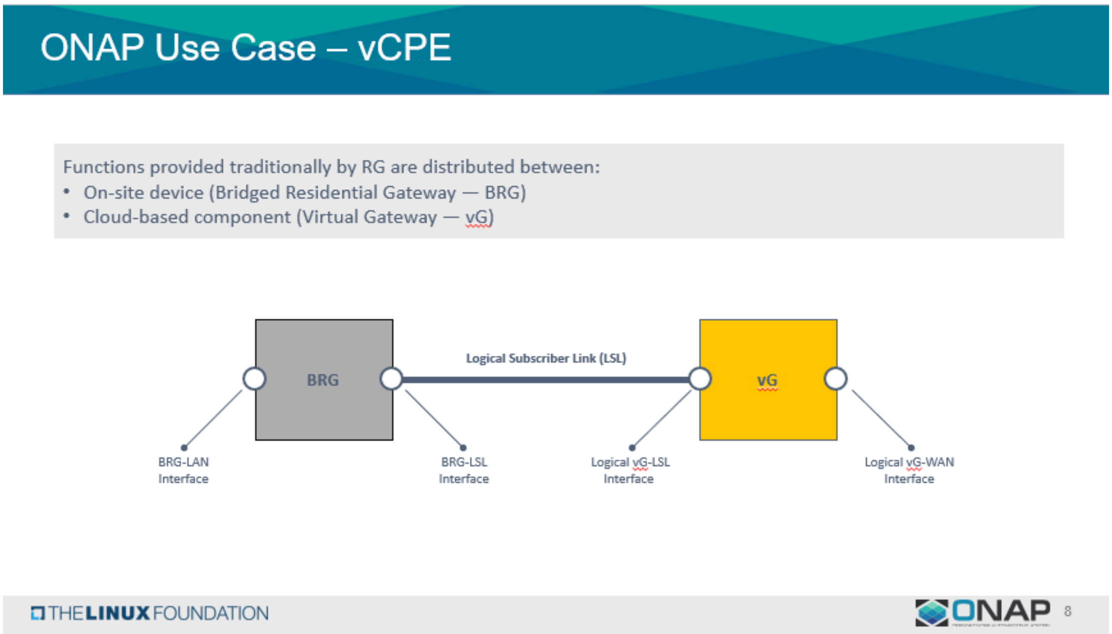


Figure 4: *ONAP use case example of a virtual CPE.*
Source: *Linux Foundation, 2018*

Deployment of Customer Premise Equipment (CPE) is one of the key elements in provider services. CPE provisioning and support is a significant expense and an area of high risk for service providers. In recent years there have been significant changes within the CPE space, for example, the evolution toward SD-WAN.

CPEs typically are used to deliver one dedicated service to the end customer. In the case of a business customer, the service might be a firewall to protect the company's local network, or a virtual private network to support remote workers. In the case of a consumer/residence, the service might be cable television service or broadband Internet access from the home. With only one service enabled per CPE, and the equipment being installed at the customer's location, it's difficult for service providers to deploy a new service over the same appliance and for end customers to maintain the equipment themselves.

Virtualized CPE (vCPE) is a way to deliver network services to customers by using software on a white box or on a universal CPE (uCPE) rather than dedicated hardware devices. By virtualizing CPE, providers can dramatically simplify and accelerate service delivery, remotely configuring and managing devices and allowing customers to order new services or adjust existing ones on demand.

For this particular use case, the ONAP project used the Network Enhanced Residential Gateway (NERG) concept put forth by the Broadband Forum to deliver virtualized services to consumers at their homes. The functionality of the traditional residential gateway device is split into two parts. The part that goes into the home to provide connectivity is a light weight device called the Bridged Residential Gateway (BRG). Most of the functionality resides in a cloud-based component called the virtual gateway (vG) hosted by the service provider. The vG is the set of virtual functions that can deliver all kinds of services to the residence that weren't possible before with the legacy residential gateway.

Through the Logical Subscriber Link (LSL), there is Layer 2 connectivity between the BRG and the vG. The service provider can use this connection to deliver software-based functions and services – otherwise known as virtual network functions – to the residence. The BRG allows for multiple services over that single device; for example, streaming content, gaming services, home security services, home automation services, broadband connectivity, etc.

The service provider is able to provide enhanced management and troubleshooting from its cloud. This allows an operator to offer a support service for anything in the home. What's more, the service provider can offer specific services on a per home device basis. For example, say the homeowner has a home office where she works on a daily basis. She can have the service provider deliver a higher quality of service to the computing devices in the office than to the gaming and entertainment devices in the home. These types of services were not possible using legacy CPE devices.

vCPE promises to open up a wide range of business and residential services that are enabled by the ONAP technologies and platform. For commercial/enterprise customers, functions and services such as switching, routing, load balancing, firewall, virtual private network and others can be delivered via vCPE. Cases such as this show that ONAP has the capability to slice generic virtual network services and design end-to-end services, enabling service providers to get innovative in the new services they offer.

About ONAP VNF Management

The ONAP platform is the part of the larger NFV/SDN ecosystem that is responsible for the efficient control, operation and management of VNF capabilities and functions. The vCPE use case above discusses the creation of a virtual appliance, through which the functions and services are delivered to the customer. However, just creating that appliance is not enough because the network conditions could vary, and this is what VNF Management addresses.

Any given service can only cater to the demands of so many customers. After that maximum demand capacity is reached, it's necessary to grow that capacity by provisioning more virtual machines to cater to that demand and prevent the service from crashing. A service provider would need to scale the service to address the demand. This could be through manual or auto scaling.

With manual scaling, the service provider has clear control of sizing the service to meet demand. ONAP has the Virtual Infrastructure Delivery (VID) component, through which more instances could be scheduled manually. Doing an auto scaling takes a few more steps, but it's certainly feasible. Auto scaling basically involves the following process:

1. Onboard the VNF by getting the VNF image and deploying it by uploading this artifact into ONAP. The Service Orchestrator (SO) creates the actual instances.

2. Once the VMs start running, they need to be configured for their functionality, typically using NETCONF. The configurations are handled by the APPC and SDN-C components. APPC takes care of Layer 4 to Layer 7 VNFs, and SDN-C handles Layer 2 and Layer 3 routing and switching configurations. The Lumina SDN Controller could handle the work of the SDN-C.

3. Once the VNF is running, it is monitored to ensure it is functioning as expected. The DCAE component of ONAP collects data to evaluate the VNF, and initiate recreating it if necessary. This is known as closed loop monitoring, and the purpose is to keep tabs on the VNF to make sure it is able to service all users as expected

This handles the process of scaling out, which is necessary when there is a growth in demand for the service—whether that growth is long-term or short-term (i.e., a burst of traffic for a short time).

There's also the concept of scaling in once the traffic level comes down, such as following a peak-usage burst. Maybe the instances that were created are no longer needed, and they should be killed because they consume resources. Typically, a scale in policy would remove all the instances that are no longer required when the traffic reaches a normal level once again. At this writing, ONAP isn't supporting a scale in process, but it is planned for a future release of the platform.

Conclusion

The Linux Foundation's Open Network Automation Platform is the industry's most broadly supported service orchestration initiative. The functions served by ONAP and described in this paper are critical to building and deploying next generation services. ONAP will change and evolve rapidly over the next several years as the project matures. Lumina Network provides a critical piece of the ONAP solution, the SDN-C, which serves as the control and abstraction function for virtual and physical network elements. Based on OpenDaylight, the SDN-C, Lumina SDN Controller, is one of the more mature components of the ONAP architecture and is widely deployed today.

Take Action

Do you have a challenging orchestration project that demands an ONAP-compliant SDN controller? Do you want to get extra value from additional services and capabilities for your network? Contact Lumina Networks today to discuss your needs with a network engineer to learn how open source platforms and agile deployment services deliver what matters to you. The following is a summary of some of the products and services offered by Lumina Networks that can help you begin to deploy the SDN-C and ONAP today.

Part Number / SKU	Description
LU-9500-1NODE-SVV-SW-1	Lumina SDN Controller / OpenDaylight Node License
LU-9511-SVV-SW-1	Lumina SDN Controller /OpenDaylight Developer Edition
LU-9500-SOURCE	Lumina SDN Controller Source Code, Governed by Eclipse Public License v1.0
IMPL-SDNCTRL-CUST	Contract Lumina SDN Controller implementation. SOW
APPDEV-SDN-CUST	Contract Application testing and verification Lumina SDN Controller customers.
SVC-CONTROLLER-APPDEV	Contract Network Management Application Development for Lumina SDN Controller customers.
ASSESS-SDN-NFV-CUST	Assessment and Design Services.

Call now:
+1 (669) 231-3838
+1 (800) 930-5144

Visit Us Online:
www.luminanetworks.com

Glossary

The following is a glossary of terms found in this paper.

ONAP	Open Network Automation Platform	ONAP provides a comprehensive platform for real-time, policy-driven orchestration and automation of physical and virtual network functions that will enable software, network, IT and cloud providers and developers to rapidly automate new services and support complete lifecycle management.
SDN	Software Defined Network or Software Defined Networking	Software-defined networking is an approach to designing, building, and managing networks that separates the network's control or SDN network policy (brains) and forwarding (muscle) planes, thus enabling the network control to become directly programmable and the underlying infrastructure to be abstracted for applications and network services.
SDN-C	Software Defined Network Controller Project	The SDN-C project provides a global network controller, built on the Common Controller Framework, which manages, assigns and provisions network resources. As a "global" controller, the SDN-C project is intended to run as one logical instance per enterprise, with potentially multiple geographically diverse virtual machines / docker containers in clusters to provide high availability. The project also supports the ability to invoke other local SDN controllers, including third party SDN controllers.
LSC	Lumina SDN Controller	The Lumina SDN Controller powered by OpenDaylight (ODL) provides a common, open platform for developers, giving providers direct control over their SDN development and implementation, thus eliminating lock-in.
VNF	Virtual Network Function	In a Network Functions Virtualization architecture, a virtualized network function, or VNF, is responsible for handling specific network functions that run in one or more virtual machines on top of the hardware networking infrastructure, which can include routers, switches, servers, cloud computing systems and more.

Glossary

NFV	Network Functions Virtualization	Network functions virtualization defines standards for compute, storage, and networking resources that can be used to build virtualized network functions.
VIM	Virtualized Infrastructure Manager	The virtualized infrastructure manager in a Network Functions Virtualization (NFV) implementation manages the hardware and software resources that the service provider uses to create service chains and deliver network services to customers.
VNFM	Virtual Network Functions Manager	The virtual network functions manager (VNFM) is a key component of the NFV management and organization (MANO) architectural framework. The VNFM works in concert with other NFV-MANO functional blocks, such as the virtualized infrastructure manager (VIM) and NFV orchestrator (NFVO), to help standardize the functions of virtual networking and increase the interoperability of software-defined networking elements.
YANG model	Yet Another Next Generation model	YANG is a data modeling language used to model configuration and state data manipulated by the Network Configuration Protocol (NETCONF), NETCONF remote procedure calls, and NETCONF notifications.
TOSCA	Topology and Orchestration Specification for Cloud Applications	TOSCA is a data model that can be used by telecom carriers for creating templates or data descriptions of applications and infrastructure for cloud services. It can also be used to define the relationships among these services, as well as their operational behavior.
container		A container is a form of virtualization in which the operating system is virtualized. A container image is a lightweight, standalone, executable package of software that includes everything needed to run an application on the virtualized OS: code, runtime, system tools, system libraries and settings.

OOF	ONAP Optimization Framework	ONAP Optimization Framework (OOF) provides a policy-driven and model-driven framework for creating optimization applications for a broad range of use cases.
microservices		Microservice architecture, or simply microservices, is a distinctive method of developing software systems that tries to focus on building single-function modules with well-defined interfaces and operations. Microservices can help create scalable, testable software that can be delivered weekly, not yearly.
OOM	ONAP Operations Manager	The ONAP Operations Manager (OOM) is responsible for orchestrating the end-to-end lifecycle management and monitoring of ONAP components. OOM uses Kubernetes to provide CPU efficiency and platform deployment. In addition, OOM helps enhance ONAP platform maturity by providing scalability and resiliency enhancements to the components it manages.

ABOUT LUMINA NETWORKS

Lumina Networks believes the future is open software networks that give providers control over how they implement their ideas and priorities for change. Lumina is the catalyst to bring open software networking out of the lab and into live networks—to ensure that providers can use open source in critical use cases, not just experimental ones.

Yet, delivering technology by itself is not enough. Lumina's customers are learning and doing the work with Lumina NetDev engineers so that they acquire the skills, tools and practices needed to develop and manage the unique platforms that are jointly deployed. Lumina becomes a partner, mentor, and custom developer to help service providers move their technology to the open source world.

Lumina Networks has a clear strategy to disrupt the networking industry status quo with open source and agile deployment. By developing open source platforms and enabling innovation and transformation through NetDev Services, Lumina is the catalyst to open software networks.

Sidebars

What is Cloud-Native?

The current and future requirements of telco applications call for rearchitecting the current telco network and deployment models. "Cloud Native" is one of the enablers and helps to realize the next generation telco infrastructure. According to the Cloud Native Computing Foundation, Cloud Native uses an open source software stack to be: [3]

1. **Containerized** – Each part (applications, processes, etc.) is packaged in its own container. This facilitates reproducibility, transparency and resource isolation.

2. **Dynamically orchestrated – Containers are actively scheduled and managed to optimize resource utilization.**

3. **Microservices-oriented** – Applications are segmented into microservices. This significantly increases the overall agility and maintainability of applications.

An infrastructure built in this way is intrinsically agile and can be scaled on demand. This opens up the concept of "composable infrastructure," which is the ability to define the elements of the network as "Lego blocks" that are brought together with a "glue layer."

What is Intent-Based Networking?

Intent-based networking is a concept wherein the service provider or IT department defines a service intent using a high-level language or model. The resultant intent defines what the network will do and how it will behave without having to explicitly define how it will happen. Instead, a tool will translate the declared intent into the device-level configurations required to realize the desired outcome. The service defined will set parameters for the connected users, including security, performance, reliability and other high-level attributes.

Gartner defines an intent-based networking system as having four key things: [4]

Translation and Validation – The system takes a higher-level business policy (what) as input from end users and converts it to the necessary network configuration (how). The system then generates and validates the resulting design and configuration for correctness.

Automated Implementation – The system can configure the appropriate network changes (how) across existing network infrastructure. This is typically done via network automation and/or network orchestration.

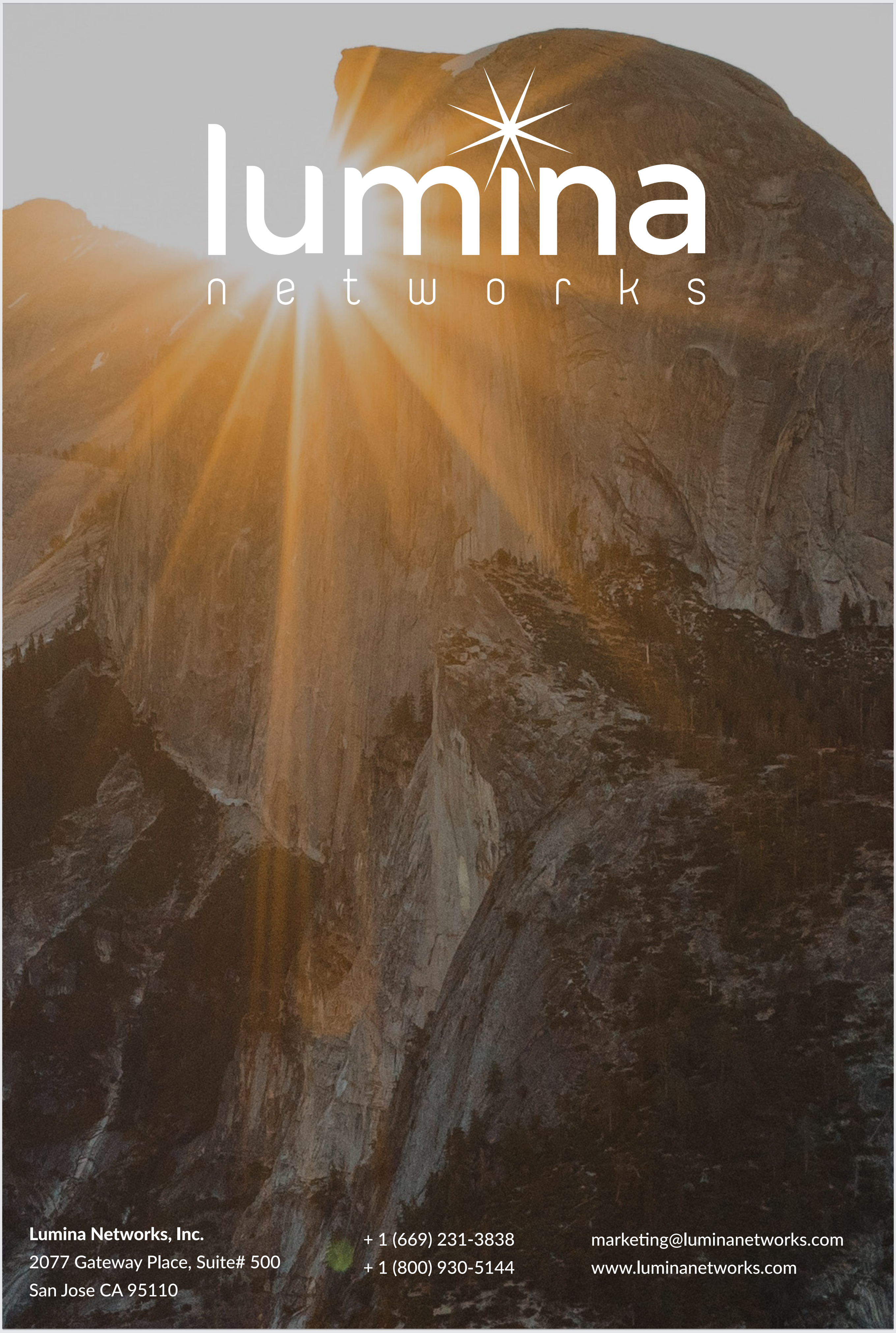
Awareness of Network State – The system ingests real-time network status for systems under its administrative control and is protocol- and transport-agnostic.

Assurance and Dynamic Optimization/Remediation– The system continuously validates (in real time) that the original business intent of the system is being met and can take corrective actions (such as blocking traffic, modifying network capacity or notifying) when desired intent is not met.

Right now, it is very early days and intent-based networking more of an ideal than an actual tool. At best, this concept will not be mainstream for several years, but solutions (such as those in the ONAP projects) are now emerging that provide value as the industry works toward the goal of full intent-based networking.

[3] Jennifer Clark, Vice President, 451 Research, "Thought Leadership – Orchestration and OSS, Beauty Meet Beast," Sept 2015

[4] Andrew Lerner, Gartner Blog Network, "Intent-based Networking," February 7, 2017



Lumina Networks, Inc.
2077 Gateway Place, Suite# 500
San Jose CA 95110

+ 1 (669) 231-3838
+ 1 (800) 930-5144

marketing@luminanetworks.com
www.luminanetworks.com