Unleashing Network Potential through SDN in Virtual Networking Lab Environments

A Technical Paper prepared for the Society of Cable Telecommunications Engineers
By:

Pilar Somohano
General Manager – Virtual Junos Business Unit
Juniper Networks
10 Technology Park Drive, Westford, MA
978-589-0320
psomohano@juniper.net

and by:

Brian Mutcherson
Regional Operations Engineering Integration
Comcast
1800 Bishops Gate Blvd., Mt. Laurel, NJ

Boon Thau Loo
Research Lead
University of Pennsylvania
3451 Walnut Street, Philadelphia, Pennsylvania

Introduction

Service providers are faced with competitive challenges that require them to optimize their operations and maximize profitability. To gain a leading edge against new low-cost and fast delivery alternative services, MSOs need methods for fast deployment of differentiated services with increased efficiency. This requires a two-pronged approach to meet the challenge. First, they need to reduce costs by improving efficiency. Secondly, they need to maximize revenue streams by offering differentiated services to more subscribers to maximize ARPU and win in the market place while expanding profitably.

Some of the most promising technologies that will help achieve these goals are:

- Software Defined Networking / Network Function Virtualization
- Enhanced use of Automation
- Virtual Networking Labs and Virtual Network Capture for lab environments

Undoubtedly, the flexibility to enable new services and increasingly smarter implementations of the control plane enabled by SDN and NFV opens a rich potential of opportunities and efficiencies. To fully unleash this potential, it is necessary to pair this intelligence with adequate verification tools in the lab environment to design and model the operation of these new applications before they get into production.

This paper describes a methodology to use captures of a production network in a virtual networking test environment to boost the potential of software defined networking and automation methods to create new services and improve network efficiency.

Overview of SDN and Benefits

Principles of SDN

The enabling principle of Software Defined Networking (SDN) is a clear separation of networking functional blocks to create a modular approach to service development and chaining, network evolution, and dynamic management through balanced centralized/distributed deployment of these functional elements. The table below shows the four functional blocks that constitute SDN.

	Functional Goals
Network Management	Unified Network Operation Smart discovery and orchestration
Services Plane	Accelerate creation of differentiated services Layer 7 applications supporting multi-source platforms
Control Plane	Enable adaptive networking applications Protocol management and dynamic routing
Forwarding Functions	Maximize bitrate forwarding performance Minimize power consumption and footprint

Table 1 – Key functional blocks of a Software Defined Network

Each of these blocks has clearly established functions, and therefore, demands different capabilities. The forwarding tier, shown on the table above, is usually based on hardware platforms, and its evolution is focused on the exponential capacity growth experienced by service providers. The services and control planes, which are the focus of this paper, are the enablers of new services and network optimization policies; and are the most dynamic part of the carrier's day to day operations. Finally, the management system is the centralized brain that coordinates the functional blocks, orchestrating the centralized and distributed components of the network operation. This system provides access to the information contained in each block of the network, and this functionality is essential for a complete understanding of a particular network setup.

Benefits of SDN – software-defined service chaining

By separating the network control and forwarding planes and centralizing management under a global controller, SDN allows you to optimize every element of the network and simplify network design, ultimately lowering the operating cost of the network

Today, service chaining is setup manually. Using SDN to enable software-based service chaining is a new concept. For example, network traffic can first go through a stateful firewall to insure security and then the traffic can go through an Application Delivery Controller (ADC) to distribute the load across servers. In the physical networking world,

that service chain is manually configured. In the new SDN approach, the service chain is configured and administered all in software that can adapt as the services need to adapt.

The SDN architecture opens the opportunity to create and automate the implementation of new services that can be activated faster than ever to reduce costs, accelerate revenue generation and increase market differentiation.

The combination of SDN technology and automation boosts the efficiency and performance for MSO networks:

- SDN facilitates smart distribution, and maximizes the utilization efficiency of infrastructure.

- Software automation accelerates configuration of new nodes, users and services, and can also contribute to reduce implementation errors. For all the benefits it provides, automation must be planned and implemented carefully since an automation error can cause fast propagation of problems in the network, and thus, the need to refine these methods before implementation is paramount.

New practices for design, test and implementation of SDN Services

The need to test SDN services before deployment

SDN service enablement opens the door for optimized operations. Because SDN is the foundation for automated service chaining, operators must test that the defined services work properly, that the chaining is executed as desired, and that these processes can be repeated for multiple regions and segments of the network reliably.

While an automated environment saves time on deployment, untested modifications and additions can also propagate much faster than manual network changes and could cause operational difficulties in multiple points of the network posing a challenge. Therefore, it is necessary to test the new service implementations to prevent the introduction of incorrect configurations in the production network.

The role of a virtual lab environment to test SDN services

Adequate testing of new SDN services and automation procedures boosts the efficiency and performance for MSO networks. A traditional lab environment, based on hardware, lacks the flexibility and scalability needed on the new dynamic architecture of SDN.

Using virtual networking Labs, operators can test new services on a large scale environment. Also, using this large scale virtual environment, operators can also test software automation tools such as configuration templates and upload scripts.

The topologies for the virtual lab can be synchronized to the production network to maximize the efficiency and realism of the tests. The method proposed for this synchronization is the Network Capture tool described on the next section.

When this lab environment is closely integrated to the operational procedures and when it can obtain real-time information from the production network via a capture tool, it becomes a unifying tool that simplifies handoff of network information between teams from the planning stage to design and into implementation and operations.

While SDN and Automation will play a key role for MSOs, the virtual lab environment and its close reproduction of the production network will make the implementation of the new technologies a true possibility by mitigating the risks associated with these complex methods and by providing a reliable framework to test and refine these developments.

Questions to Consider:

- How much time does it take you to set a test environment to verify new services or automation scripts?
- How often do you test these elements at on a large scale environment?
- Is it feasible to test at a large scale on a network with the same properties as the production environment… and do so efficiently?
- What do you do when the network experiences problems?... do you have time to verify the fix before pushing it into production?
- Can different groups collaborate?

For most service providers, the answers to these questions are subject to the innate limitations of today's physical lab environments.

First, physical lab setup is time-consuming, and because of the cost, there is limited availability of resources. The table below shows a comparison between physical and virtual labs to summarize how these environments supplement each other to maximize results.

	Physical Lab	Virtual Lab
Scalability	Difficulty to set up and limited availability of lab hardware make scalability tests rare on physical environments	The design of virtual labs is focused on fast and cost-effective activation of large networks, making the virtual lab ideal for scalability tests
Duration	Due to the scarcity of physical lab resources, test intervals are kept short, and the depth of tests is often capped in physical environments	Because of its cost-effectiveness, the virtual lab allows for long run tests, and additionally for simultaneous run of multiple "what-if" scenarios giving depth and breadth of results
Team Collaboration	Often times, one team is the "owner" of the lab, and passes implementation procedures to others. Teams receiving information do not have a way to review or correct this in a lab when they find issues. It is difficult to replicate initial test conditions in the physical lab.	The setup information for topologies, configurations, scripts and other definitions is stored as efficient files that can be launched and closed multiple times. All this information can be shared among teams, reproduced, analyzed and modified as many times as needed allowing for review of original conditions.
Access	A group of users has access to a lab, and the access cannot easily be restricted o particular clusters. As multiple users share this lab, it is common that one user performs accidental changes to a particular topology, impacting the project of another colleague.	The virtual environment allows for hierarchical permissions and topology insulation, avoiding accidental interference between users. Furthermore, if a user makes an irreparable mistake, it is always possible to resort to the original topology files.
Test Reproducibility	Most of the setup requires manual intervention, thus, test reproducibility is hard, and often imprecise.	All topology and configuration information is saved on compressed files that allow for quick and exact reproduction of any given topology, by any user or group at any time.
Throughput Testing	Physical labs have the same forwarding plane performance of production equipment, and therefore, they are best suited for capacity and transport performance testing.	As virtual labs are optimized for volume efficiency, they do not provide the same performance of physical nodes. The best practice combines virtual + physical labs through a "hybridge" connector that links physical with virtual elements. This setup takes advantage of scalability and performance qualities of both environments.

Table 2 – Comparison of test practices in physical vs. virtual networking labs

Team collaboration via virtual topology files

On the virtual lab, the topologies are defined and configured by files that can be stored, shared and copied inside the virtual environment. The topology files can be stored and activated at any time. These files can be passed from one functional team to another. One topology file can be copied multiple times, and variations of it can be tested simultaneously to evaluate “want if” scenarios. An active topology instance can be accessed by a single user or by a group of users.

With this flexibility, a planning team can communicate clearly to an engineering team the characteristics of a new deployment; two technology groups can share their points of view topology approaches or configurations; and all teams can keep centralized change control records.

This characteristic of the virtual labs creates a smoother operational flow between departments and facilitates collaboration.

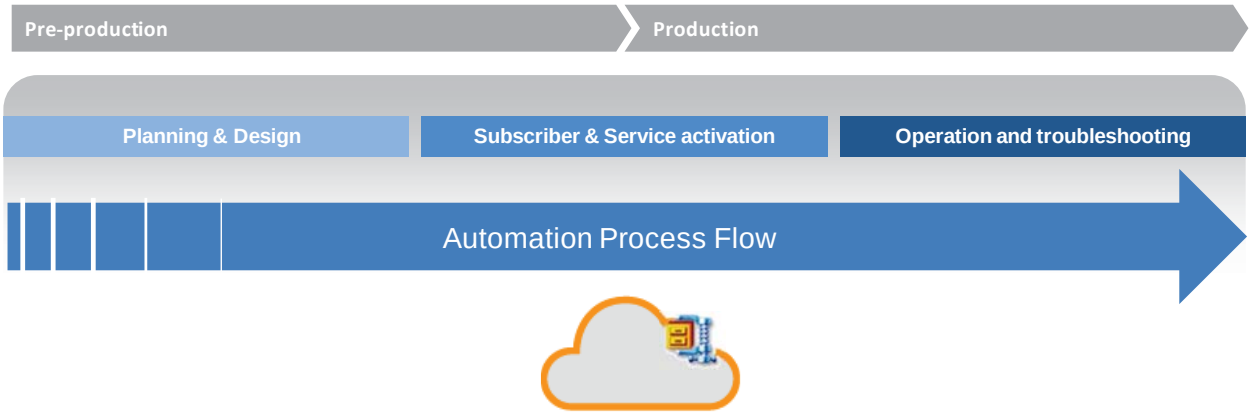

Figure 1 – Inter-department information flow

Virtual Network Lab Structure and Functions

Virtual Lab Architecture

A virtual network lab is a cloud environment that supports virtual applications testing and consists of:

- Interfaces that enable:
  - individualized access for multiple users (multi-tenant architecture)
  - creation and use of virtual topologies
  - administrative tasks such as cloud capacity management and allocation
- Sandboxes:
  - where users create their custom topologies and modify test configurations
  - that enable insulation between topologies
- Hybridge Connector:
  - establishes a hybrid connection between the virtual lab and a physical one
- Virtual Ecosystem:
  - a set of virtual routers, servers and diverse applications that run in the virtual lab and can be connected in diverse ways to form custom topologies. A rich the ecosystem allows a more realistic virtual reproduction of a physical network

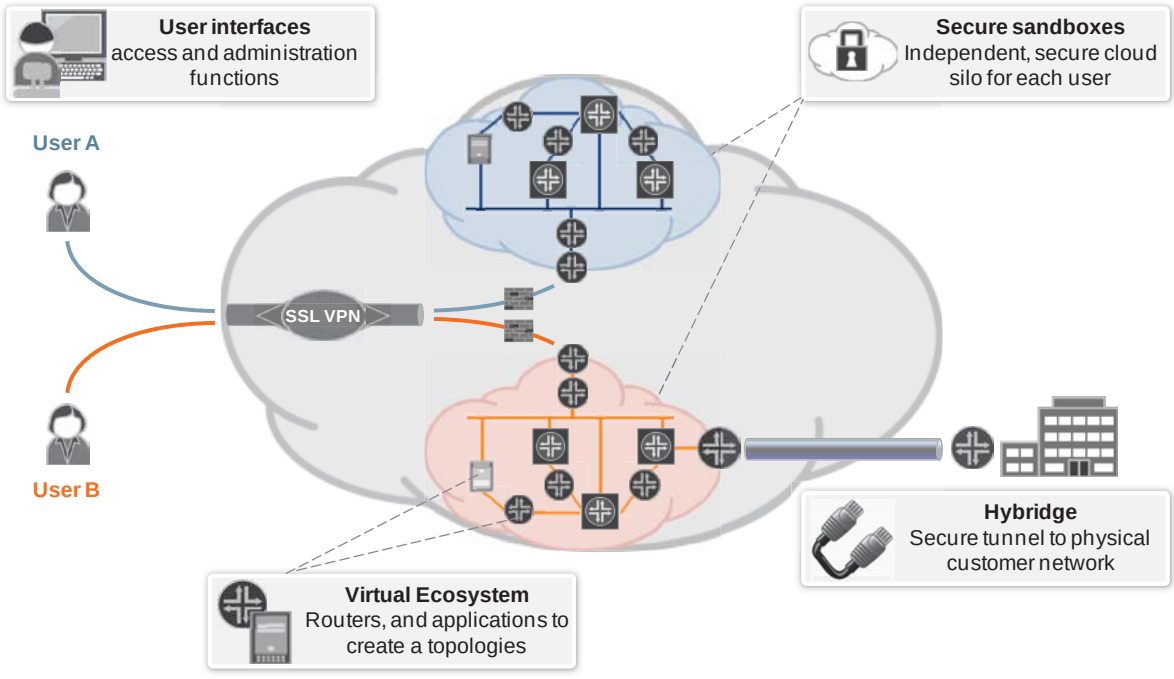

Figure 2 – Basic architecture of the virtual lab environment

Hardware and Software components

The virtual lab environment is built on a datacenter. The capacity of the servers used depends on the size and number of topologies that will be used. The memory, processing and storage are defined also by the virtual “images” that will run in the topologies.

Software components:

- On top of this hardware infrastructure, there is the virtualization layer. This layer allocates compute resources to create instances of virtual machines in a topology.

- For networked virtualization, it is necessary to connect virtual machine interfaces to other virtual machines. For this application, virtual switching technology is used to establish plain bridges among the interfaces that constitute a topology.

- For the creation of topologies, it is required to have virtual images. These “images” are pre-defined models that allow for creation of multiple virtual instances of a machine with identical initial settings.

Network Capture functionality on the virtual lab

Figure 3 shows a new approach that combines the intelligence available on the network management system and the flexibility of a virtual networking environment. The lab will enable us to create a virtual network capture of the production environment. The capture allows for realistic "what-if" tests. Operators can test new services, applications, and dynamic network control algorithms. These tests permit corrections before production.

Using this method, the node state information is used to define the properties of nodes in the virtual environment which are logically identical to the physical nodes on the production network. Because this method is automated, an operator can instantly create multiple virtual replicas, and multiple test scenarios can be conducted simultaneously, efficiently, and cost-effectively.

Since the network capture on the virtual environment has the same logical properties of the production network and it is scalable, it allows for a realistic test of the new applications., Potential problems are detected and corrected early, before impacting the production environment. This approach reduces guesswork that results in operational problems such as introducing errors on the production environment, and then, attempting to debug them in that live network.

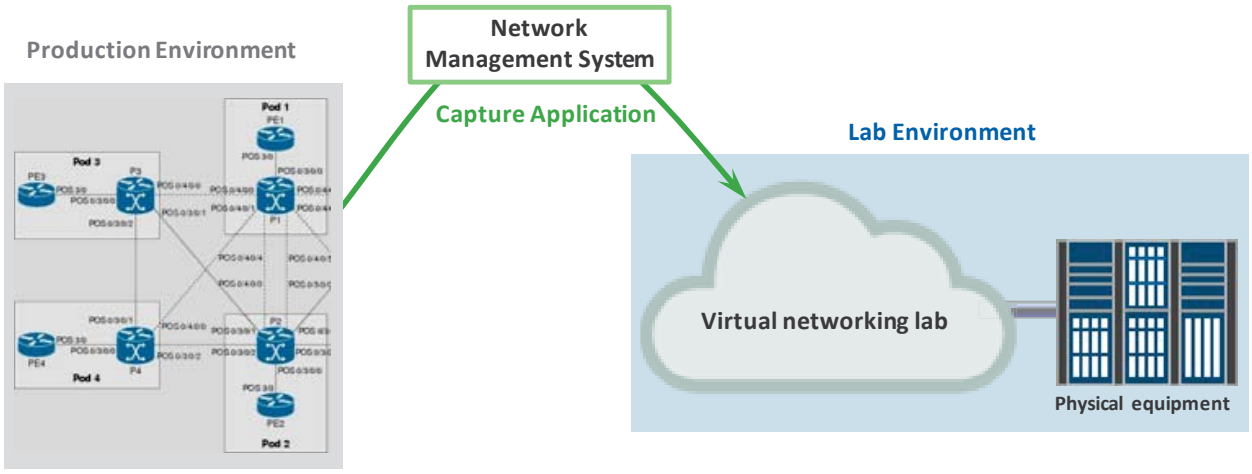

Figure 3 – Network Capture Functionality

Application Cases

CASE I:
Automation – proper test before incorporation of automated methods

Automating network processes reduces implementation times for new services and enables new subscribers to be turned up faster, accelerating revenue generation cost-effectively. This section shows application cases used by providers today.

Testing new configuration templates

These templates, used in configuration automation, are validated in the virtual lab to ensure they don't break the production network when they are used.

As the Network Engineers develop the service templates, they can validate them using the virtual lab. This reduces the time and effort associated with organizing and pre-configuring physical lab gear. Instead, virtual devices are activated using APIs (Application Programming Interfaces). The virtual devices are pre-configured, and the same configuration can be used as many times as necessary, then the results of the template are applied, and the team can perform the validation tests.

Runtime validation of custom configurations

For the implementation of non-standard configurations, runtime validation on a virtual network capture of the production environment, ensures proper performance before implementation and reduces margin for errors. With this, the operations team can focus on creation and activation activities, reducing the need for repairs while saving subscribers from the quality impact resulting from errors.

Development test of automation platforms

Typical lab environments count with a limited number of physical devices. Virtual labs provide the scalability required to ensure proper behavior of new automation programs to conduct functions such as new deployments. The virtual lab can be easily modified and allows for test for large network applications.

Policy management

When policies are updated, the virtual lab can be used to ensure that new inbound and outbound policies to customers are consistent with existing deployments. Even when a configuration template is not available, the virtual lab allows function validation of the new policies so they become the new template. In this application, the master configuration templates can always be stored in the virtual lab for ulterior reference, use and modification.

Simultaneous test of what-if scenarios

To determine the most efficient way to apply new policies, multiple variations of the same topology can be created on separate sandboxes and run simultaneously. The operators observe the results of diverse criteria, such as route learning performance, to help to defining the most efficient maintenance strategies for the network.

CASE II:
SDN - Declarative and adaptive networking

At the University of Pennsylvania (UPenn), there is ongoing research work on developing new programming tools and platforms for software-defined networks. The technology called declarative networking, enables advanced intelligence on the network using SDN to dynamically optimize a network in several domains. Some examples are:

- adaptive network optimization based on traffic and QoS
- network response to faults according to tolerance criteria
- network security functions
- wireless channel selection
- network configuration management

This new SDN-based technology allows operators to maximize the throughput of their installed base while maximizing its robustness to faults, intrusion, and other common problems.

In this case, the researchers at UPenn use a virtual lab environment to test multiple functions and applications of this new technology at a scale:

- Creating large-scale networks, emulating diverse traffic patterns, and evaluating the response of adaptive networking tools
- Simulating network faults and optimizing declarative networking programs
- Testing security functions under simulated attacks and verifying the learning capabilities of the adaptive tools

Summary

Network operators are faced with competitive challenges that require them to optimize operations and maximize profitability.

Software Defined Networking and Software Automation open new avenues for creation of differentiated services, streamlined operations and efficient usage of the installed base.

Virtual Networking Labs enable operators to design and test new SDN services and automation software in a safe environment. Testing before implementation improves reliability, saves time and reduces operational costs by minimizing time spent on lab setting and post-deployment troubleshooting of configurations.

The power of virtual lab environments resides on their scalability and simple setup. New network capture tools enable operators to replicate quickly portions of a production network in the virtual lab. Using the virtual lab, diverse workgroups in a service provider can share information about network designs and configurations in an efficient way.

Today, leading service providers and technology corporations are incorporating the use of virtual labs to test automation scripts, configuration templates and network optimization methods. In doing so, they optimize the use of their scarce physical lab resources while boosting the reliability of their new services.

Bibliography

1) Virtual Network Optimizing A Physical Network
P.Somohano, B.O'Sullivan, H.Stern; M.Yip; A.Mints
Patent application Ref. No. JNP-2039

2) Recent Advances in Declarative Networking
Boon Thau Loo, Harjot Gill, Changbin Liu, Yun Mao, William R. Marczak, Micah Sherr, Anduo Wang, and Wenchao Zhou.
Fourteenth International Symposium on Practical Aspects of Declarative Languages (PADL), co-located with POPL, Jan 2012.

3) RapidNet declarative networking platform.
http://netdb.cis.upenn.edu/rapidnet/

Abbreviations and Acronyms

Abbreviation / Acronym	Description
API	Application Programming Interface
ARIN	American Registry for Internet Numbers
AS	Autonomous System
BGP-TE	Border Gateway Protocol – Traffic Engineering
IETF	Internet Engineering Task Force
LSP	Label-Switched Path
MSO	Multi-Service Operator Refers typically to a cable carrier company
NFV	Network Functions Virtualization
QoS	Quality of Service
RFC	Request for Comments A recommended procedure documented on IETF
RSVP	Resource Reservation Protocol
SDN	Software Defined Networking