Independent market research and competitive analysis of next-generation business and technology solutions for service providers and vendors

HEAVY READING WHITE PAPER

NETCONF/YANG: What's Holding Back Adoption & How to Accelerate It

A Heavy Reading white paper produced for OneAccess

ONEACCESS
Fast Network Virtualization

AUTHOR: JAMES CRAWSHAW, SENIOR ANALYST, HEAVY READING

INTRODUCTION

Simple Network Management Protocol (SNMP) was developed by the Internet Engineering Task Force (IETF) in 1988 for network management, but was used mainly for monitoring, with configuration normally done using vendor-specific Command Line Interface (CLI). Information was difficult to represent in a detailed and organized manner using SNMP, and its lack of a standard, automatic discovery process to find a device's management information base (MIB) meant that discovery had to be done manually. This rendered configuration management with SNMP hopelessly complicated.

NETCONF/YANG is the industry's latest attempt to crack the network configuration challenge. With the advent of network functions virtualization (NFV), NETCONF/YANG will become a key tool for the configuration of programmable networks and a key enabler of automation. The move to a network configuration management standard is long overdue as communications service providers (CSPs) move into an era of dynamic, self-provisioning network services.

Why SNMP Sucks at Network Configuration

A key advantage of a CLI versus SNMP is that it is human-readable. In contrast, SNMP is encoded using Basic Encoding Rules (BER), which is in binary format. Moreover, unlike CLIs, SNMP does not distinguish between the operational and configuration data stored within a device. An SNMP query can return as many as 500 variables, only a subset of which are configuration-related, and SNMP does not define which variables belong in which subset. Nor does SNMP support clear text, so variables can't be cut and pasted between devices.

CLI Sucks Too, Just in Different Ways

However, CLI needs scripts for daily management, and the content and formatting of the output is unpredictable. Some estimates suggest that 60 percent of outages are caused by humans mistyping or translating things incorrectly from printed documentation into their consoles (see Why use NETCONF/YANG when you can use SNMP and CLI?, SNMP Center). Moreover, the use of vendor-specific CLI can lead to vendor lock-in, as only that vendor's element management system (EMS) can manage their particular network elements.

Common Management Information Protocol (CMIP) was devised as a replacement for SNMP but was too complex to implement and maintain, and thus was poorly supported by vendors.

NETCONF to the Rescue

The latest attempt to crack the network configuration problem is NETCONF, an XML-based protocol initially created by the IETF in 2006. NETCONF supports operations such as installing, querying, editing and deleting configuration data on network devices. It can be conceptualized as four layers:

1. Content – configuration data for the network device
2. Operations – methods for communicating between agent and manager, e.g., <get-config> or <kill-session>
3. RPC – used for encoding the Remote Procedure Call
4. Transport – communications path between server and client using secure protocols such as SSH and SSL

As operations support system (OSS) consultant Ryan Jeffery writes in his blog:

> "In a nutshell, it [NETCONF] allows for network management via a human-readable XML language. Not only does the XML make it human-readable, but it makes it easily diff'able (i.e., great for change management). IETF were the developers of SNMP, which has become ubiquitous for network health and statistics, but NETCONF came about because vendors were using command lines (CLI) for config management. From an OSS perspective, this [CLI] was a disaster. I worked on a tier-1 carrier project which had just about every switch/router platform offered by the world's biggest name in switches/routers. Almost every one of them had CLI/response differences, meaning mediation device variants had to be spec'd and built separately. It was a nightmare, especially for regression testing purposes in the days when automated regression suites weren't readily available."

Historically, the automation of device activation was carried out by OSS vendors building adapters in a proprietary, workflow-driven way. This resulted in a significant cost for operators commissioning new equipment or deploying new services. As Stefan Wallin and Clae Wikstrom wrote in their paper Automating Network and Service Configuration Using NETCONF and YANG:

> "Existing approaches to configuration management such as CLI scripting, device-specific adapters, and entrenched commercial tools are an impediment to meeting these new requirements. Up until recently, there has been no standard way of configuring network devices other then [sic] SNMP and SNMP is not optimal for configuration management. The IETF has released NETCONF and YANG which are standards focusing on Configuration management. We have validated that NETCONF and YANG greatly simplify the configuration management of devices and services and still provide good performance."

The Importance of Transactionality

As Ryan Jeffery describes above, the greatest limitation of CLI scripting is the lack of transaction management. Configuring a device can be a complex task, involving multiple actions. Usually these actions cannot be done partially, as this would leave the device in an undefined state. If any step of the end-to-end provisioning fails, there is a need to roll back all previous actions, to revert to the original configuration. This requires extensive programming when transaction management is not supported, as is the case with CLI.

The Need for a Modeling Language

While NETCONF initially appeared to be the Holy Grail, engineers soon realized that for NETCONF to be useful, they needed a consistent modeling language. Although many modeling languages existed, the telecom sector argued that the idiosyncrasies of the industry required their own domain specific language. The IETF considered XML Schema but ultimately settled on YANG, yet another next-generation modeling language, which was first published in 2010. YANG looks like C or JavaScript and can be used to model both configuration data as well as state data of network elements.

YANG data models that describe devices and services (configuration and state information) are transported via the NETCONF protocol. The configuration is plain text and human-readable. It is easy to copy and paste and compare between devices and services.

THE IMPORTANCE OF NETCONF/YANG FOR NFV

NETCONF/YANG is a key enabler of the new paradigm of NFV, where services can be designed at a high level, independent of the complexities and device dependencies of the underlying infrastructure. This infrastructure will be heterogeneous, assembled "on the fly" from a range of virtualized functions from multiple "best of breed" vendors. Most importantly, these services will be provisioned in minutes and their operation highly automated.

NFV Needs NETCONF/YANG for Automation

The dynamic nature of virtual networks, with functions being spun up and changed continuously, requires highly automated management. The combination of NETCONF and YANG provides operators with greater flexibility in the management of network devices, both physical and virtual.

A wide range of software-defined networking (SDN) projects and industry organizations – such as the Open Networking Foundation (ONF), the OpenDaylight Project, the Open Network Operating System (ONOS) Project, and the Open Platform for NFV (OPNFV) – are now promoting NETCONF/YANG as a universal southbound protocol for the configuration and management of both virtual network functions (VNFs) and physical network devices in SDN environments.

NETCONF/YANG Can Enable Automation Without the Need for Virtualization

SDN and NFV promise to revolutionize network operations through automation and open application programming interfaces (APIs), driving down operating costs and increasing the speed with which new services can be delivered. However, many of these benefits can be achieved today, without the complications of virtualization, simply by adopting a programmatic and standards-based way of writing configurations to any network device from any vendor, replacing the manual configuration of tens or hundreds of devices that CSPs must currently undertake to deliver a service to an individual customer.

Several vendors now support NETCONF/YANG on their latest-generation network devices, enabling CSPs to program the physical network functions (PNFs) residing on these devices and even construct a service chain between these PNFs and cloud-based VNFs.

NETCONF and YANG support the automation of configuration across heterogeneous devices at multiple layers of the network. In particular, NETCONF/YANG supports transactionality, so CSPs can be confident that either all the configurations in a sequence are applied, or the entire update is rolled back.

WHAT'S HOLDING NETCONF/YANG BACK?

Given the weight of the arguments laid out in the introduction in favor of NETCONF/YANG, you might wonder why it isn't more widely used. After all, NETCONF was introduced in 2006 and YANG in 2010. What are CSPs waiting for?

There are a number of issues holding NETCONF/YANG back from more widespread adoption, which we discuss in turn below.

Installed Base Doesn't Support NETCONF/YANG

Legacy equipment (i.e., the clear majority of the installed physical base) doesn't support NETCONF/YANG. Although virtual functions and new physical equipment should support it, operators will need to manage a hybrid of NETCONF/YANG-enabled gear and CLI-managed gear for some time. This can be facilitated using orchestrators such as Cisco's Network Services Orchestrator (NSO) or Ciena's Blue Planet, which can manage both legacy and YANG-enabled functions.

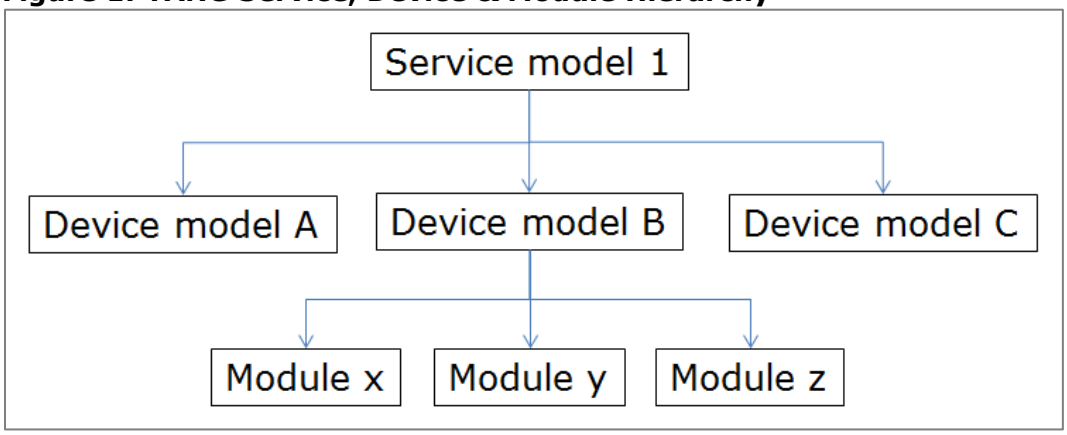
It is important that operators insist that any new gear they put in their network is NETCONF/YANG-capable to reduce their dependence on YANG-CLI "translators," such as the orchestration platforms mentioned above. While some vendors might argue that one can configure their devices more easily using their proprietary tools, the downside of such an approach is reduced automation and increased vendor lock-in.

YANG Module Standardization Is Immature

Although NETCONF and YANG have been standardized, the problem then shifts to the standardization of modules. A module is one YANG file, while a model comprises several YANG modules (see **Figure 1**). A model could be for a router, while its component modules could be for static routing, BGP, interfaces, etc. For a more thorough discussion of YANG modules, see the IETF's YANG Module Classification.

A service (e.g., Layer 3 VPN) is implemented by mapping YANG models of the underlying network functions (physical or virtual) to the standardized YANG model of the network service. The introduction of a service abstraction level hides the complexity of the network function level from the higher orchestration, MANO and/or OSS levels. Consequently, changes at the network level do not have knock-on impacts on the higher-level, management applications.

Figure 1: YANG Service, Device & Module Hierarchy



Source: Heavy Reading

While NETCONF and YANG are standards, the modules that vendors use to describe their appliances can vary significantly and efforts to standardize them have been slow. This diversity impedes interoperability. The IETF is working on this, establishing definitions for interfaces such as Ethernet and optical (see A YANG Data Model for Interface Management). Once implemented, vendors are naturally keen to defend their current module implementations and resist change. It is also hard to create normalized modules that are both simple and widely useful. Thus, the standards-setting process has been slow.

OpenConfig was an attempt led by Google, AT&T, BT, Deutsche Telekom, SK Telecom, Verizon and others to break the logjam of the IETF standards-setting process and create a set of models that they could then point vendors to and say "comply." However, there was little focus on interoperability between different CSPs' YANG model definitions. The work of OpenConfig appears to be winding down; the last news item on its website is a review of the organization's work in 2015. The YANG model standardization process is likely to now shift back to the slower but more rigorous standards-setting process within the IETF.

Resistance to Change & Skill Shortage in CSPs

It is hard to persuade some network management teams to change the way they operate and move to YANG. If one has spent one's career obtaining vendor-specific, CLI-related certifications and scripting experience, there is little incentive to start to learn YANG or XML for a Linux environment. As Antoine Lecomte of OneAccess writes here:

> "CLI engineers... see the change coming, but as they are buried by the demands of their day-to-day activities, they have not yet had the time to study and experiment with NETCONF. So, this protocol along with YANG data modeling remains very abstract, if not confusing to them. Of course, they understand quite clearly that NETCONF is about a programmatic approach to the process of service creation. Network engineers thus understand they must acquire new skills in programming, but it is certainly not their comfort zone today."

The argument for updating one's skills is well made in Benoit Claise's blog post Why Should Networking Engineers Learn Python?, where he explains how, as a network engineer with more than 20 years of CCIE qualification, he decided to learn Python. For a specific guide to learning YANG, check out Carl Moberg's tutorial on GitHub.

In general, a lack of familiarity with NETCONF/YANG is holding back adoption in CSPs. It is difficult to create YANG service models, and there is a general difficulty with thinking more abstractly about network management. There are simply not enough programmers in CSP engineering departments. CSPs need to make new hires or retrain people in key skills such as YANG, XML and Python.

Below we provide some examples of key technologies CSPs should consider in their training programs.

Figure 2: Network Automation Technology Stack

Layer	Examples
Software provisioning automation engines	Ansible, Chef, Salt, Puppet
Configuration templates	Juju, HOT, TOSCA
Configuration protocols	NETCONF
Encoding	XML, JSON
Modeling languages	UML, YANG
Programming languages	Python, Ruby

Source: Heavy Reading

Confusion Between TOSCA & YANG

There remains some confusion about the overlap of management responsibility between NETCONF/YANG and Topology and Orchestration Specification for Cloud Applications (TOSCA) in virtualized networks. While network operators have embraced YANG, cloud application developers generally prefer TOSCA.

OASIS created TOSCA to facilitate the creation of cloud applications and services requiring coordination (orchestration) among diverse resources (Web servers in different hosts, DNS servers, load balancers, firewalls) in complex application environments. In the telecom domain, TOSCA can be used to orchestrate end-to-end services, including legacy and SDN components, in a similar vein to OpenStack's HOT or Canonical's Juju Charms.

While VNFs can be deployed using cloud application lifecycle management templates such as TOSCA, they are likely to need further configuration at runtime, e.g., to fulfill a change in service such as a new customer-specific firewall rule. This is where NETCONF/YANG comes in to deliver programmatic configuration and support the automated lifecycle management of NFV.

TOSCA and NETCONF/YANG are thus complementary within network management, with TOSCA handling higher-level orchestration aspects while NETCONF/YANG deals with lower-level device configurations. As an example, AT&T has decided to use TOSCA to describe the VNF resource packaging required for onboarding to the Open Network Automation Platform (ONAP), its open-source management platform, and YANG to describe the device's (i.e., VNF) data mode.

Confusion Between NETCONF/YANG, REST API & RESTCONF

IT developers from outside the telecom industry might look at NETCONF/YANG and wonder what all the fuss is about. Why not just use REST APIs to talk to network functions? This simplistic view is a bit like comparing an envelope with a formatted letter template.

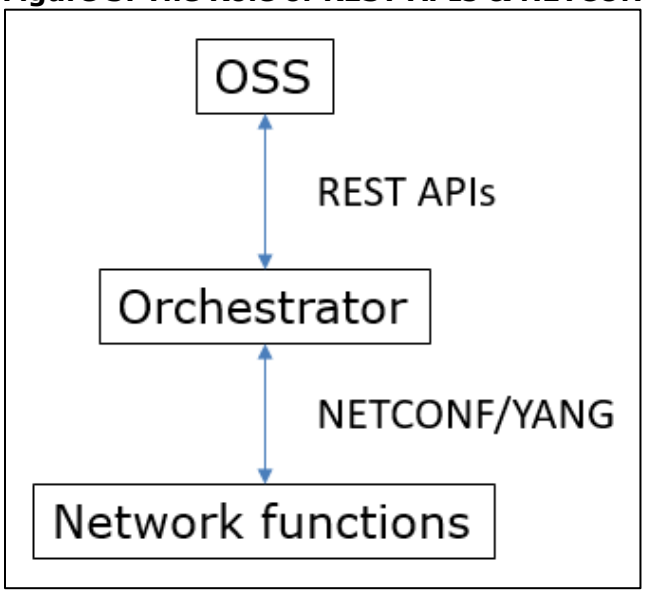
REST is a principle for Remote Procedure Calls (RPC). There is no REST standard to test a system for compliance. REST is like an unstructured envelope; the content and format of the payload is not defined, and there are no rules on compatibility. A REST-based interface would require custom code to integrate into OSS and enable automation. However, since REST is a loose RPC mechanism, it is simpler for vendors to implement than RESTCONF/YANG, and hence some may try to promote it as an alternative.

YANG is a layer above RPC mechanisms such as REST. It defines the semantics of operations on top of an RPC (in this case NETCONF). The YANG layer embeds decades of experience in rules for a good configuration of network devices and functions. YANG defines the payload of the operations data, data types and structure. In contrast, REST is clueless on such matters; that is up to the client to deal with *ad hoc*.

Note there is a NETCONF mapping to REST called RESTCONF. The IETF standardized RESTCONF as a REST-like API to access YANG data models. RESTCONF is still a relatively new protocol, and the ecosystem is comparatively small. Some orchestrator vendors are using RESTCONF in their northbound API to give access to their abstracted YANG object models. For southbound communication, between the orchestrator and network functions, the NETCONF/YANG approach is used.

The rationale for SDN controllers and NMS/EMS (or any network connectivity management-related software) to use REST and/or RESTCONF northbound (to connect to IT/OSS systems) is to reach a wide audience of developers that already know REST APIs well and have tool-sets and frameworks to handle this type of integration. REST is an API style that is the current *de facto* standard for IT system integration, with a huge existing ecosystem.

Figure 3: The Role of REST APIs & NETCONF/YANG



Source: Heavy Reading

Confusion Between YANG & Information Models

There is some confusion over the hierarchy of models needed in next-generation network management. Many vendors talk about compliance with the TM Forum's Shared Information/Data model, SID. The SID is a useful, high-level model of the business and operational environment within a CSP. It can help engineers conceptualize the information flows between B/OSS systems, but it does not enable one to program or automate them.

To transform SID concepts, expressed in Unified Modeling Language (UML) diagrams, into network configuration actions, programmers need to build lower-level models, drivers and interfaces that can talk to devices. So even if a vendor is "SID-compliant," if its product is not NETCONF/YANG-compatible, it will require significant integration work to manage within a network. As a formal service modeling language that can interpret SID constructs in a standardized way, YANG bridges the semantic gap and enables the creation of service and device models.

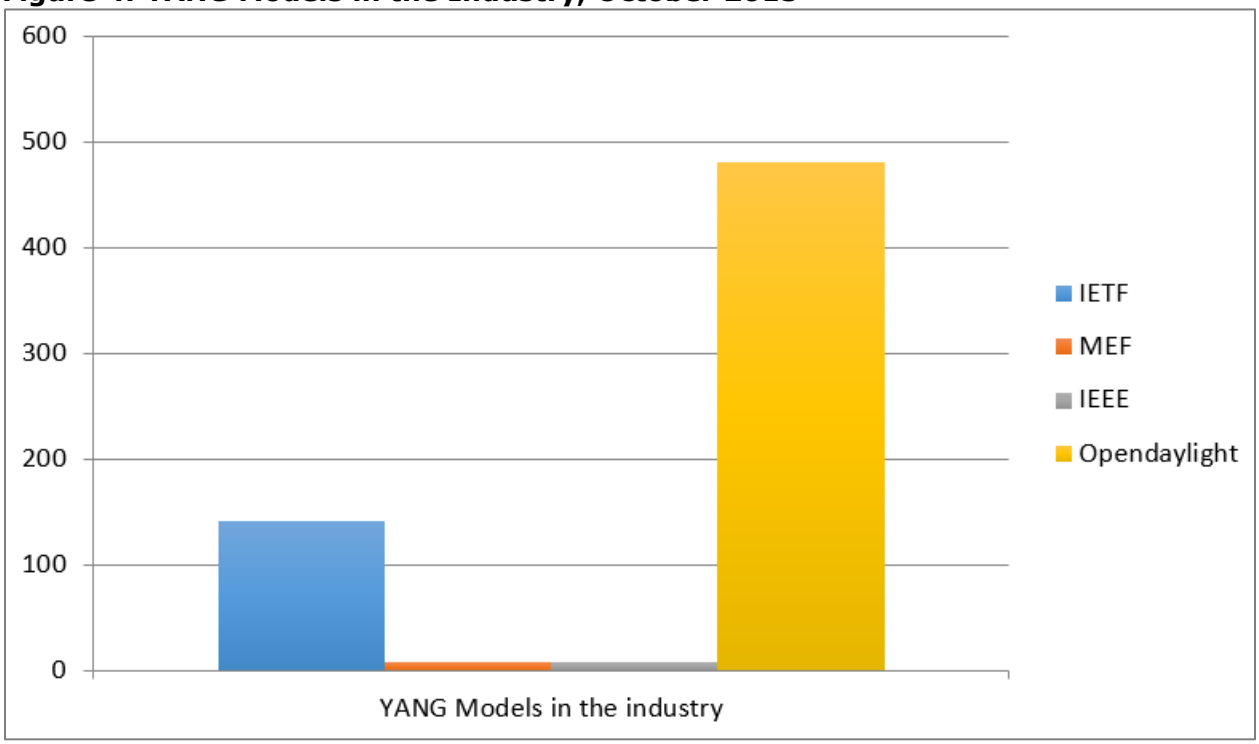
Lack of Interoperability

Since standards can be interpreted and implemented in proprietary, or even incorrect ways, there is a need for NETCONF/YANG validation tools that prove that vendors' models are truly standards-compliant. Vendors supporting NETCONF/YANG can increase confidence in their models by participating in interoperability tests with one another. Interoperability is both in a North-South direction (i.e., between the network function and its orchestrator) as well as in an East-West sense (i.e., between different VNFs and PNFs).

TRANSITION TO NEXT-GENERATION MANAGEMENT

For all the reasons expressed in the previous section, we are still quite early in the adoption of NETCONF/YANG for network management. **Figure 4** shows the number of public YANG models in the industry as of October 2015. By April 2017, the number of IETF YANG models had reached 221, while the number of OpenDaylight YANG models had reached 703 (Beryllium release). The IETF is working on L3VPN service models, while the MEF is working on the modeling of carrier Ethernet services such as E-Line, E-LAN and E-ACCESS.

Figure 4: YANG Models in the Industry, October 2015



Source: Benoit Claise, www.claise.be

NETCONF/YANG for Hybrid Physical & Virtualized Networks

The migration to NFV won't take place over night. Operators will need solutions that work in a hybrid physical/virtual environment for many years. There will be a transition from existing management systems (e.g., CLI and TR-069 for CPE) to NETCONF/YANG. The migration from network element-centric to software-centric operations will drive fundamental changes in the network operating model across multiple dimensions, from tighter integration across network, IT and architecture teams to new processes and tools to manage the network.

Future-Proof Your CPE

Adding NETCONF to an existing router is challenging. Modeling the router's operational models in YANG is a significant endeavor requiring the router's software to be re-architected and, although support for the protocol itself is relatively easy, NETCONF/YANG requires more memory resources than most classical customer premises equipment (CPE) has been shipped with. As a result, an upgrade of the installed base of CPE is rarely feasible. At the same

time, moving to X86 white-box CPE is premature for most operators and unlikely to be cost-effective, given the maturity of existing CPE technology.

Instead, operators are perhaps best served by continuing to purchase classical CPE with their cost-effective, network-processor-based designs and diversity of network interfaces (DSL, 4G, fiber, TDM, etc.), but insisting that these come with integrated NETCONF/YANG support alongside the traditional CLI/TR-069. This will ensure integration with next-generation OSS (NFV-MANO) that requires NETCONF and, though less flexible than the X86 white-box CPE approach, can be more cost-effective.

Insist on Openness

A lack of flexible and open access solutions has been a longstanding complaint from many operators. For example, some SD-WAN vendors have proprietary schemes, and some have wrapped proprietary extensions around OpenFlow. One European enterprise service provider we know of was concerned by the proprietary nature of the SD-WAN solution it had purchased, and so insisted that the vendor's VNFs were deployed on a white box sourced directly by the operator, as well as ensuring that its orchestration and management system was back-ended to its OSS. It also insisted the vendor make its SD-WAN interoperable with VNFs from other vendors. Insisting on support for NETCONF/YANG is another key tool to avoid vendor lock-in.

CONCLUSIONS

NETCONF's adoption had been slow, but with the emergence of SDN/NFV, NETCONF is the best candidate to replace CLI for configuration management of programmable networks. Leading CSPs are already mandating support for NETCONF/YANG from their vendors. Most large network equipment vendors already support these standards. In the future, NETCONF will be as ubiquitous as SNMP, and without it vendors risk being excluded from RFPs. The days of vendors making money from their equipment's proprietary interfaces, EMSs and tools are over. The move to a network configuration management standard is long overdue as CSPs move into an era of dynamic, self-provisioning network services.

Key recommendations:

1. Operators should select vendors that have good support for NETCONF/YANG, mature data models, and embrace openness. Device and VNF vendors need to add NETCONF support to their products to remain relevant.
2. CSPs should take an open, multivendor approach that avoids vendor lock-in. Ensure that the vendors you select are taking a truly open approach, and not just paying lip-service to technologies such as NETCONF/YANG. (For more, see Netconf & Yang Go Mainstream.)
3. CSPs must take a pragmatic approach to service migration that provides flexibility over the timing of the move to virtualized delivery of services and minimizes its complexity.
4. Good YANG models take a lot of effort to write. The current manual approach to writing them is leading to poor interoperability. This gives rise to the need for tools to help build YANG models using a machine learning-powered schema generator with reduced human intervention.

5. CSPs should undertake interoperability testing to identify immature data models and demand that vendors fix them.
6. NETCONF/YANG will take many years to spread across the network. Legacy devices are unlikely to be made NETCONF/YANG-compatible – hence the need for a middleware platform, orchestrator or CLI translator to connect to and configure old kit.
7. The skill set of operations teams needs to be expanded to include scripting capabilities (or their equivalent via GUI-based tools) to be able to effectively create "recipes" for managing software VNFs.

To learn more about NETCONF/YANG, you can participate in a webinar sponsored by OneAccess on June 13, 2017: www.oneaccess-net.com.

Pravin Mirchandani, CMO for OneAccess, will be participating in a panel ("Accelerating NFV and SDN deployment") at Network Virtualization Europe in Madrid on May 30, 2017. If you would like to meet with Pravin, please send an email to marketing@oneaccess-net.com.

ABOUT ONEACCESS

OneAccess Networks is a leading provider of physical and virtual network functions enabling cloud and other managed services to SMB and enterprise customers around the world. For over a decade, OneAccess has been delivering enterprise network functions designed specifically for service providers' needs: performance in compact multiservice access routers.

Our approach to virtualization takes this philosophy forward. Network operators and CSPs have told us they want a catalog of compact, efficient, carrier-grade VNFs delivered on an open, standards-based platform that allows them to stay in control. They want the ability to integrate third-party network functions as they see fit. Plus they need a phased approach to NFV migration, at a pace that they define. So, that's what we deliver today: fast network virtualization.

Twenty-nine of the top 100 service providers in the world have already chosen OneAccess to enhance their competitive edge. Now they can do so using state-of-the-art, open virtualization technology.