



White Paper

Creating the Programmable Network: The Business Case for NETCONF/YANG in Network Devices

Prepared by

Caroline Chappell
Senior Analyst, *Heavy Reading*
www.heavyreading.com

on behalf of

The logo for tail-f, featuring the lowercase letters "tail-f" in a bold, dark blue, sans-serif font. The "f" is stylized with a horizontal bar that extends to the right and ends in a small vertical tick.

www.tail-f.com

October 2013

Executive Summary

Communications service providers have long wished to drive down the operational cost (opex) of their networks and increase the agility and speed with which they deliver new services. Their concern to achieve a "programmable" network that can be reconfigured quickly, easily and cost-effectively to meet new customer and service demands is driving huge interest in technologies such as software-defined networking (SDN) and network functions virtualization (NFV). These technologies promise to revolutionize network operations through automation and application programming interfaces (APIs).

But there is one means of dramatically reducing opex and accelerating service delivery that can be adopted today, delivering SDN advantages into the existing physical network, as well as preparing the way for OpenFlow-based SDN and NFV in the future. This is the adoption of 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 service providers must undertake to deliver a service to an individual customer. The Network Configuration Protocol (NETCONF) is an Internet Engineering Task Force (IETF) standard protocol for reading and writing network configurations and, together with its associated data-modeling language, YANG, it can make a powerful contribution to network opex reduction and speed of service turn-up.

NETCONF and YANG support the automation of sequences of configurations across heterogeneous devices at multiple layers of the network – with transactionality, so service providers can be confident that either all the configurations in a sequence are applied or the entire transaction is rolled back. NETCONF/YANG are specifically designed for the task of network configuration, so they enforce configuration best practices and make it quicker and easier for service providers to carry out configuration tasks. As a result, leading service providers are beginning to mandate support for NETCONF/YANG on their network equipment vendors: They want to buy devices that have formal data models expressed in YANG and support for NETCONF out of the box.

The largest network equipment vendors are already behind these standards, and momentum is growing for their adoption. Vendors can currently promote their support as a point of competitive differentiation, but in the future, NETCONF will be as ubiquitous as SNMP, and without it vendors risk being excluded from network procurement activities. The days of vendors making money from their equipment's proprietary interfaces, element management systems and tools are numbered, and service providers are intent on ending them. The move to a network configuration management standard is long overdue and inevitable as service providers move into an era of dynamic, self-provisioning network services.

Section II examines the market drivers for an improved approach to network configuration that reduces opex and accelerates service delivery.

Section III describes the history of NETCONF and YANG standardization and the features that make them so compelling.

Section IV discusses the market momentum behind NETCONF/YANG adoption and why leading service providers and vendors are supporting these standards.

Market Drivers for the Programmable Network

Service Provider Demand for a "Programmable" Network Is Growing

There is no doubt that the telecom industry is on the cusp of huge change. Demand for the raw product of telecom networks – bandwidth – is exploding at an unstoppable rate. But the cost of meeting that demand is trailing further and further behind the revenues network operators can earn both from bandwidth itself and from the relatively few services they own and can offer over it. A number of communications service providers are already experiencing massive pressures on their revenues and margins, as demonstrated by the state of their balance sheets, as well as growing divestiture and M&A activity in the industry.

Service providers are hobbled by two major impediments. The first is the opex cost of their networks, which, unlike their capex, remains constant or rises year on year, especially as networks become larger and more complex. Network device configuration – at 45 percent of total cost of device ownership over five years according to some industry sources – is the greatest contributor to opex. As one of the last, unautomated frontiers of network management, network configuration requires a small army of network operations staff to carry it out and recover the network from misconfiguration when this inevitably occurs through human error.

The second impediment is the difficulty of creating and/or onboarding new services that allow service providers to generate more revenues from their networks. The need to manually configure large numbers of network devices involved in turning up a single service means that each service can take weeks or months to deliver. In today's fast-paced and highly competitive environment where customers expect almost instant access to the services they order, such delay is increasingly unacceptable.

The common denominator between these impediments is the physical and heterogeneous nature of network devices, which not only makes them expensive to operate using current manual interfaces, but also time-consuming to change to support new services.

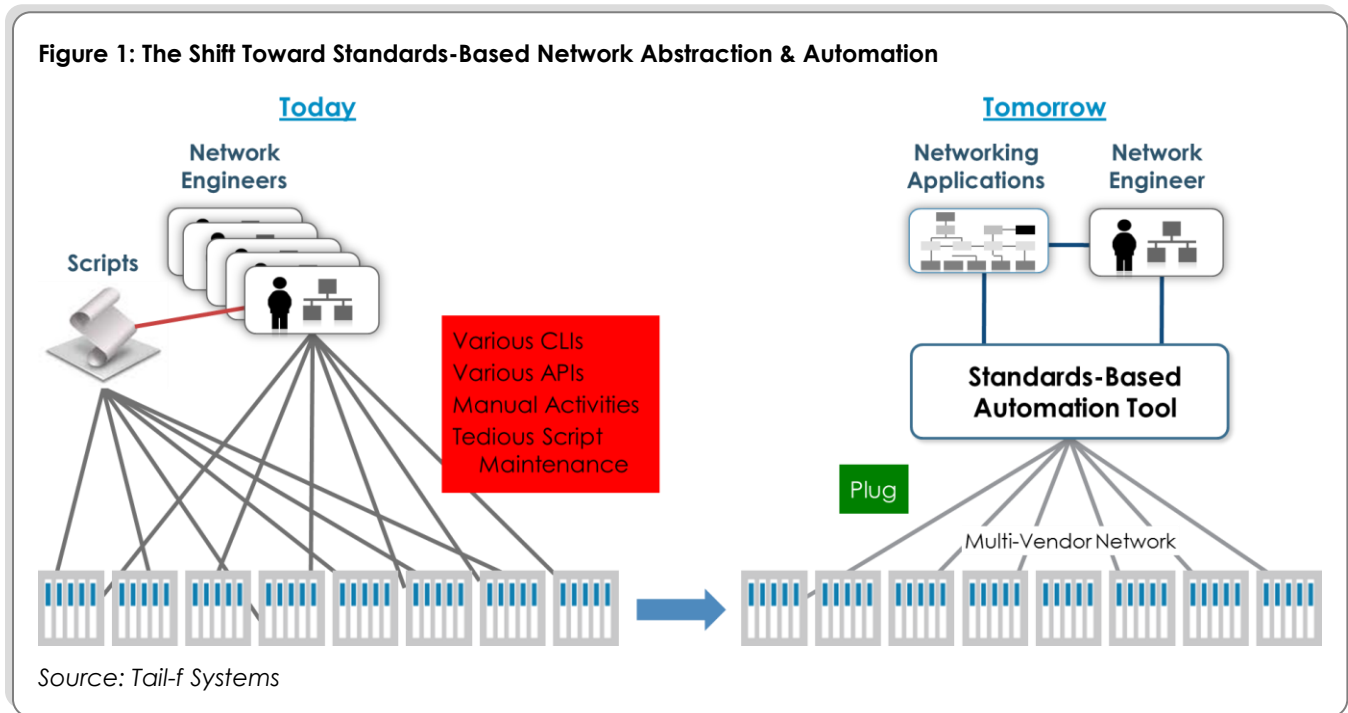
SDN and NFV are promising new approaches to building and managing networks that, among other benefits, push more network functionality into hardware-independent software and introduce standardized, programmatic interfaces to that software. The network can be represented as a software-based abstraction, the configuration of which can be automated, just as the configuration of virtual compute and storage resources are automated in the cloud.

SDN and NFV are expected to have a massive impact on the cost and speed of service delivery and network operations once they have revolutionized service provider networks. However, they are also highly disruptive. The fact that service providers are showing considerable enthusiasm for such technologies, despite the amount of change they involve, is a measure of how hard-pressed service providers are by high opex and their inability to deliver innovative services in a timely manner.

SDN and NFV have shone a spotlight on the shortcomings of traditional approaches to network management, highlighting the need for network programmability – in other words, an automated, standardized and rapid way of configuring the network. But the full adoption of SDN and NFV will take years. Service providers can't wait that long – they are beginning to demand, and even mandate, solutions that will work with the existing, physical network today.

Operator Requirements for a "Programmable" Network Solution

Such a solution will need to replace the multiple, proprietary command line interfaces (CLIs) service providers have in their networks (see **Figure 1**).



Around 95 percent of network devices currently on the market assume a human interface and, therefore, must be configured manually. If service providers want a programmatic interface to such devices, they typically have to build it themselves or deploy – and then integrate – multiple vendor-specific tools or implement proprietary multi-vendor tools that rely on expensive adapters that have to be maintained and/or systems integration work whenever a new device needs to be added. Large operators, such as Deutsche Telekom and fellow members of the Next Generation Mobile Network (NGMN) Alliance's Next Generation Converged Operations Requirement (NGCOR) working group, have reached the limits of their tolerance for CLIs and proprietary configuration approaches.

Service providers want a programmable network solution that will automate sequences of configuration changes across multiple, heterogeneous devices to help them provision services quickly and accurately. The solution should be based on standards – particularly on a standard that can write configurations to devices, not just read and display them. Current standards supported by devices, such as SNMP, deal with operational rather than configuration data, while network change and configuration management (NCCM) tools do a reasonable job of extracting configuration data, but they are ill-suited to the bi-directional, programmatic communication with devices needed to support configuration automation.

Where sequences of configurations can be automated, they will also need to be transactional so service providers can be confident that either all changes have been committed successfully, or, in the case of failure at any point, none of them have been applied and the network is not left in a partially-changed state.

Such a programmable network solution should be model-driven. Service providers are beginning to demand that network element behavior is modeled in a formal, standards-based way so that they can better understand how to modify it through configuration. They also want more continuity between formal device models and their higher-level service models, so that the latter can drive configuration programmatically into devices, without the expensive and time-consuming requirement for developers to write proprietary transformations between the two. Service providers are becoming more conscious of the benefits of formal data models written in a standard language because such models will be able to "plug and play" together, avoiding high integration costs and enabling service providers to bring both new equipment into the network and new services to the market faster.

Happily, a standard approach to network configuration that supports writing configurations to multiple vendor devices and transactionality across configuration sequences, and which provides a standard language for modeling both devices and services, does exist. The standard, developed by the IETF, is stable and has been implemented in network devices and NCCM tools. Momentum is growing for the adoption of the IETF's NETCONF protocol and YANG data modeling language, and *Heavy Reading* expects them to become widely known and used over the next few years because they resolve so many of the network configuration management and automation issues that plague service providers today.

Supporting the Programmable Network

The History of NETCONF & YANG

In 2003, the IETF started an effort to develop and standardize a network configuration management protocol, NETCONF. It was widely recognized that existing interfaces used for configuring network devices were unsatisfactory.

CLIs do have some good features, such as the ability to separate out a box's configuration data from its operational data and present this data in clear text. This means that the configuration can be clearly understood and "cut and pasted" to similar boxes, saving a certain amount of configuration time. But the downside of CLIs is that they are highly proprietary and they rely on a human to interpret their text-based specifications. They can't easily be programmed, so configuration can't be automated.

And it is time-consuming for a network engineer familiar with one vendor's interface to learn the CLI for another vendor's device.

SNMP had also failed as a network configuration mechanism. 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.

So the search was on for an alternative protocol that would be as ubiquitous and standardized as SNMP but that would provide a programmatic interface to network devices through which configurations could be safely written. The IETF drew on a rich and long-standing vein of expertise in network management, protocol development and data modeling languages from the vendor and service provider communities to develop NETCONF. Its standardization effort was guided by a stringent set of service provider requirements that drew on the best aspects of CLIs (see **Sidebar**).

The specification for NETCONF was published in 2006. It was clear that a common data modeling language would also be needed to express the structure and semantics of configuration information in a vendor-neutral format. A NETCONF data modeling language – YANG – was proposed and is similarly an IETF standard.

Key Features of NETCONF

NETCONF has three key features that make it superior to other network configuration management approaches: domain-specific knowledge; support for transactionality; and vendor device independence.

Sidebar: Operator Requirements for NETCONF

The new protocol needed to satisfy the following requirements:

- Operator ease of use
- Separate collection of configuration data, operational state data and statistics from individual devices and the ability to correlate across devices
- Configuration of network and services, not individual devices, i.e., support for automated sequences of configuration changes
- Support for configuration transactions across multiple devices
- Ability to minimize the number of configuration changes with devices responsible for the order in which they apply configurations internally
- Support for backup and restoration of configurations
- Validation (consistency checking) of configurations, both within a single device over time and across a network
- Support for text processing tools such as diff and version management tools, such as RCS and CVS
- Support for standardized data models
- Support for multiple configurations within a device at the same time
- Role-based access control, including the ability to check access control lists (ACLs) across multiple devices, and supporting both data and task-oriented access control

Domain-Specific Knowledge

NETCONF was developed to support network configuration, so it has built-in functionality specific to this task. For example, NETCONF requires the implementation and standardized use of domain-specific operations, such as "get-config."

"Get-config" fulfills the basic service provider requirement for the extraction of configuration data separately from statistics, alarms and other operational data. The ability only to read and write config data is critically important not just for config manipulation and automation, but also for key activities such as configuration backup and restore.

NETCONF also supports other domain-specific operations that help make configuration management more efficient and that can similarly be programmed so they can take place without human intervention. In order to deal with large configurations, the protocol supports filtering mechanisms that allow only a subset of the configuration to be retrieved.

Support for Transactionality

It is critical that NETCONF support Atomicity, Consistency, Isolation, Durability (ACID) transactionality if configuration changes are to be automated. NETCONF supports changes to the live network, possibly involving several devices where configuration change may lead to transient connectivity problems. Therefore, either the entire transaction must succeed, or if it fails at any point, the whole transaction should be rolled back. Service providers also must be sure that only valid data is written to devices and that transactions are audited and backed up. Both of these requirements are satisfied by ACID transactionality.

NETCONF supports several different configuration change transaction models. The simplest model requires a running configuration data store in the device that describes the currently active configuration; all subsequent configuration changes are applied directly to the running configuration. There is an optional, distinct, startup model that assumes the existence of a special startup configuration data store that is loaded by the device as part of its initialization. A third, optional transaction model introduces a candidate configuration data store that can be used as a "scratchpad" to test device configurations before they are either committed to the running data store or deleted.

Vendor Device Independence

So that NETCONF can be used as a standard configuration protocol for any type of vendor device, the protocol leaves it up to individual vendors to determine how to apply configuration instructions, and in what order, within their devices.

Each vendor best understands the dependencies between configuration variables within their boxes. They currently expose these dependencies to the outside world through their CLI specifications and make service providers responsible for managing ordering logic within and across multiple pieces of network equipment. However, this is costly for service providers as they – and the NCCM tools vendors they use – must maintain hard-coded rules about the order in which to configure variables within many different types of network boxes. Given the complexity involved in following all the rules, it is also extremely difficult to automate configuration transactions in a multi-vendor environment. Service providers, therefore, can't reap the benefits of full network "programmability" and automation.

When implementing NETCONF, network equipment vendors will face additional development costs as they support the handling of configuration change requests in any order. The trade-off, however, is the removal of a large total cost of ownership/opex headache for service providers and a guarantee that the vendors' devices will plug into service providers' network automation scenarios.

In the short term, a device manufacturer that can demonstrate lower opex and support for automation through the use of NETCONF will have a strong competitive advantage. In the longer term, as NETCONF is widely adopted across the industry, network equipment vendors will have to support it or risk not being able to sell their products at all.

NETCONF vs. REST APIs

It is worth noting that these three features differentiate NETCONF from general API mechanisms, such as REST. It is certainly possible to build a REST-based interface using NETCONF principles; however, a vendor would have to employ programmers with a deep knowledge of network management to build an API with NETCONF's richness and level of capability. Such an API would be vendor-proprietary at a time when service providers are increasingly demanding, and in some cases mandating, the use of standards.

The IETF is at a very early stage of mapping NETCONF to REST to facilitate its wider adoption in future.

Key Features of YANG

The YANG data modeling language is also domain specific, designed for modeling the data needed for network management. YANG incorporates precise modeling syntax needed to configure pieces of network equipment. When creating a YANG data model, for example, architects are explicitly required to specify which are configuration items and which are operational items. YANG was designed so that its structure mirrors the way that configuration data is represented on real devices. YANG is not itself XML-based, but it does have an associated XML representation, YIN, to which it exactly maps.

Although some vendors have used their brightest and best people to create XML-based data models for their equipment, the vast majority have informal models that must be "intuited" through their CLIs. YANG makes such models explicit, concrete and able to be programmed through NETCONF quickly and easily.

Vendors that use YANG data models in their equipment will help service providers drive configuration from formal, YANG-based service models. YANG is rich enough to model both devices and services. Many service providers have imprecise, abstract and *ad hoc* service models because the knowledge needed to interpret the models and apply them to network devices resides in humans: the network engineers that implement services.

In the low-cost, automated future service providers both want and need to achieve, formally-specified service models will drive device configuration. It is, therefore, extremely helpful if both service and device models "speak" the same language so that one can natively be translated into the other without an intervening data transformation step. A common modeling language also allows service providers to "plug and play" any devices with YANG data models into their service models.

Vendors will have to accept that in the coming standards-based world of automated network management, there is no longer any mileage in trying to lock service providers into their equipment through proprietary interfaces. Such a strategy will increasingly backfire, especially as growing numbers of service providers are making total cost of ownership, including operational and integration cost, a key criterion for hardware procurement.

Service & Device Models vs. Information Models

There is currently market confusion over the hierarchy of models needed in the network management environment. Many vendors talk about product compliance with the TM Forum's SID information framework. The SID is a very useful, high-level model of the business and operational system landscape within a service provider organization that has transformed the way the industry thinks and talks about telecom management in the past decade. The SID helps humans conceptualize and model the information flows between systems in support of business and operational processes, but it does not help to program or automate them.

Transforming SID concepts into actions – for example, configuration actions on the network – requires programmers to build low-level models, drivers and interfaces capable of talking to devices. There is a large gap between the semantics of a high-level information model represented in UML diagrams, like the SID, and the semantics needed to talk, concretely, to devices. So even if a vendor is "SID-compliant" at a low level where its product hits the network, it always has to carry out a large amount of proprietary integration work.

As a formal service modeling language that can interpret SID constructs in a standardized way, YANG bridges this semantic gap. It is a key enabler for the TM Forum's early objectives, which were to create a more standardized, automated management environment for service providers, and it helps to remove the large costs that are associated with creating and maintaining proprietary integration with the network.

What's Next for NETCONF & YANG

For NETCONF and YANG to fulfill their potential as standards for the "programmable" network, vendors must create and publish YANG models of their own devices. This would be a huge step forward for the industry. Since standards can be interpreted and implemented in proprietary, or even incorrect ways, there is also an urgent requirement for NETCONF/YANG validation tools that prove that such models truly are standards-compliant and complete. Vendors supporting NETCONF/YANG can increase confidence in their models by participating in interoperability tests with one another.

The IETF is currently working on standardized ways of modeling fundamental device attributes, such as their interfaces. Service providers are likely to require that their vendors comply with such standard models in the future. The fact that OpenFlow devices use NETCONF as a configuration protocol will also put pressure on legacy equipment vendors to join the growing NETCONF/ YANG ecosystem.

Building an Ecosystem Around NETCONF/YANG

Implications of NETCONF & YANG for Network Device Vendors

If service providers are to reap the benefits of the programmable network, their equipment vendors will need to support NETCONF and YANG in their systems, replacing CLIs, proprietary APIs, SNMP Management Information Bases or any other configuration approach they may use for writing configurations. SNMP will continue to be useful for other purposes, such as read-only monitoring, so it will still have an important role to play in its own right or bridged over NETCONF/YANG.

Large equipment vendors, including Juniper, Cisco and Ericsson, have historically led the market here and are already incorporating NETCONF and YANG in their devices. These companies see compelling reasons for adopting the IETF standards:

- **The need to stay in business.** Unless service providers remain profitable, network equipment providers will suffer. Although many equipment vendors make money out of their equipment's proprietary interfaces, element management systems and tools, this gravy train is about to run off the rails, and a powerful lobby – NGMN's NGCOR working group – wants to give it a final push. The move to a network configuration management standard is long overdue and inevitable as service providers move into an era of dynamic, self-provisioning network services.
- **The ability to participate in market-leading RFPs.** The market is watching Deutsche Telekom closely as an early adopter here, and momentum will quickly build for NETCONF and YANG as other service providers see hard evidence for their success. In the past six months, *Heavy Reading* has seen rising interest in the IETF standards from network equipment vendors anxious not to be shut out of future network procurement contracts.
- **Short-term differentiation.** Long term, *Heavy Reading* expects NETCONF to become as ubiquitous a protocol for network configuration as SNMP is for network monitoring. The ability to provide a YANG model of network function/virtual network function behavior that can plug and play with whatever (YANG-based) network provisioning system a service provider has in place will be table stakes in any deal. In the short term, equipment providers that want to give their equipment a competitive edge would be wise to have NETCONF and YANG in their roadmaps, or better still, to demonstrate an early implementation.
- **Reduced internal development costs.** Having a formal device model and standard configuration protocol doesn't just benefit service providers – it can also significantly reduce development and maintenance costs for network equipment vendors, especially those with multiple lines and generations of equipment.
- **Readiness for NFV.** Automation and standardization are key tenets of NFV. Although there is as yet no consensus over the way in which virtual network functions will be modeled and configured in NFV, NETCONF and YANG support the standardization and high-scale automation needed in a cloud environment, and a formal, abstracted model of a network function can easily be reused when that function is virtualized. It is likely that service providers that have invested in NETCONF/YANG support in their physical networks will want continuity as they adopt NFV. Equipment providers that implement these standards today will, therefore, future-proof themselves as the migration to NFV gathers pace.

Service Providers Start to Drive the Ecosystem

But the largest pressure for standards adoption by the entire ecosystem of network device vendors must come from service providers themselves. A powerful lobby of service providers is emerging, spearheaded by Deutsche Telekom (DT).

DT is at the forefront of service providers mandating the IETF's NETCONF protocol and associated YANG data modeling language, which are cornerstones of DT's next-generation TeraStream network architecture. Device manufacturers that don't support NETCONF/YANG will not be allowed to participate in DT RFPs, per the company's TeraStream Device Management Interface Requirements:

"A common, standard configuration management protocol with a common data modeling language is needed in order to support the level of automation required by TeraStream. NETCONF and YANG are suitable because they are self-contained, in the sense that no additional offline knowledge about the syntax and transport of configuration exchange is needed. A CLI falls short since there is no formal data model and no formal vendor-neutral syntax. SNMP (Simple Network Management Protocol) falls short because it requires the application to have knowledge about in which order things can be created or modified."

Where DT leads, *Heavy Reading* expects others to follow. DT has considerable market influence as leader of the NGCOR working group, which is now officially liaising with ETSI NFV ISG. It is worth noting that NGMN now has 22 service provider members and intends to be a powerful lobby for industry standardization where other bodies have failed.

Conclusion

Service providers urgently need to address configuration management as one of the last and unautomated frontiers of network management. Operational costs are rising, and service providers are also struggling to deploy new services in the network in a timely manner. SDN and NFV are promising technologies that may provide solutions in the future, but service providers need a means of abstracting and automating their existing physical networks and making them more "programmable" today.

NETCONF is a stable standard for writing network configurations, with outstanding features for automating sequences of configurations and driving out the costs associated with manual manipulation of devices. These features include domain-specific knowledge, support for transactionality and vendor device independence. NETCONF is also future-proof: It is the configuration protocol used by OpenFlow devices, so it will continue to be relevant to SDNs as these are rolled out, and it supports the standardization and high scale automation needed in an NFV environment.

NETCONF's associated data modeling language, YANG, provides vendors with a formal and precise way of modeling physical and/or virtual device functionality and provides service providers with a means of formally expressing their services, which can be directly mapped onto devices without the need for human intervention. This enables a programmatic approach to configuring all the devices associated with a service, which can considerably shorten the time to provision.

Together, NETCONF and YANG provide a superior solution to other approaches that attempt to read and write configuration data using existing protocols, such as SNMP or REST APIs.

These IETF standards can help to transform the speed and cost of network configuration. Momentum is building for their use: They are already being adopted by leading network equipment vendors and beginning to be mandated by the world's largest service providers. *Heavy Reading* expects them to become widely known and used over the next few years because they resolve so many of the network configuration management and automation issues that plague service providers today.

Today, network equipment vendors can use their implementation of NETCONF and YANG as a point of competitive differentiation, but over time, service providers are unlikely to buy a device without them, just as they expect support for SNMP as table stakes today. Of course, implementations of NETCONF must be standards-compliant and correct, and vendors must demonstrate this through interop tests or through the use of validation tools. Equipment vendors that do so will put themselves in a stronger position to win business as growing numbers of service providers become aware of NETCONF/YANG's potential.

About Tail-f Systems

[Tail-f Systems](#) is a leading provider of network service programmability solutions for traditional and software-defined networks (SDN). Headquartered in Stockholm, Sweden, Tail-f is a Red Herring Top 100 company, a Stratecast Global OSS/BSS 10 to Watch Company, and a Pipeline Network Innovations Award Finalist.