O'REILLY®

Network Automation Roadmap

From Assessing Organizational Readiness to Planning Adoption

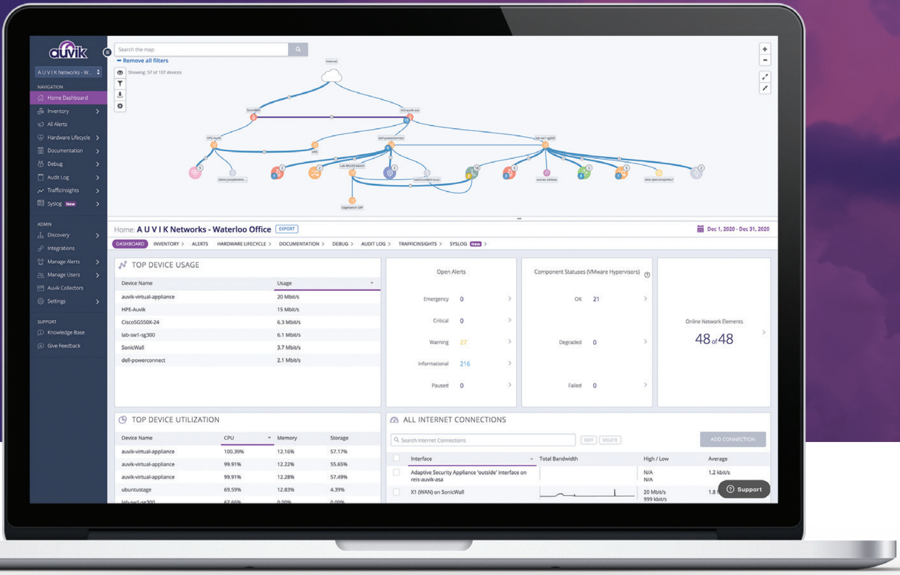
Steve Petryschuk

REPORT

Start automating your time-consuming tasks today, with Auvik.

Network Automation—it's all about efficiency and ease-of-use. With Auvik's cloud-based network monitoring and management software, you'll see both. Be prepared for more proactive IT operations thanks to...

- Automated asset inventory
- Easy configuration backup and restore
- Real-time network mapping
- And more!



Home: A U V I K Networks - Waterloo Office

Dec 1, 2020 - Dec 31, 2020

TOP DEVICE USAGE

Device Name	Usage
auvik-virtual-appliance	20 Mbit/s
HPE-Auvik	15 Mbit/s
CiscoSG550X-24	6.3 Mbit/s
lab-sw1-sg300	6.1 Mbit/s
SonicWall	3.7 Mbit/s
dell-powerconnect	2.1 Mbit/s

Open Alerts

Emergency	0
Critical	0
Warning	27
Informational	216
Paused	0

Component Statuses (VMware Hypervisors)

OK	21
Degraded	0
Failed	0

Online Network Elements

48 of 48

TOP DEVICE UTILIZATION

Device Name	CPU	Memory	Storage
auvik-virtual-appliance	100.39%	12.16%	57.17%
auvik-virtual-appliance	99.91%	12.22%	55.65%
auvik-virtual-appliance	99.91%	12.28%	57.49%
ubuntustage	69.59%	12.83%	4.39%
lab-sw1-sg300	67.66%	0.00%	0.00%

ALL INTERNET CONNECTIONS

	Interface	Total Bandwidth	High / Low	Average
□	Adaptive Security Appliance 'outside' interface on reis-auvik-asa		N/A N/A	1.2 kbit/s
□	X1 (WAN) on SonicWall		20 Mbit/s 999 kbit/s	1.8

Visit **auvik.com/automation** to learn more and start a free trial.

Network Automation Roadmap

From Assessing Organizational Readiness to Planning Adoption

Steve Petryschuk

Beijing · Boston · Farnham · Sebastopol · Tokyo
O'REILLY®

Network Automation Roadmap
by Steve Petryschuk

Printed in the United States of America.

Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472.

O'Reilly books may be purchased for educational, business, or sales promotional use. Online editions are also available for most titles (*http://oreilly.com*). For more information, contact our corporate/institutional sales department: 800-998-9938 or *corporate@oreilly.com*.

Acquisitions Editor: Jennifer Pollock
Development Editor: Michele Cronin
Production Editor: Beth Kelly
Copyeditor: Shannon Turlington

Proofreader: Amnet Systems LLC
Interior Designer: David Futato
Cover Designer: Karen Montgomery
Illustrator: Kate Dullea

September 2021: First Edition

Revision History for the First Edition
2021-09-30: First Release

The views expressed in this work are those of the author, and do not represent the publisher's views. While the publisher and the author have used good faith efforts to ensure that the information and instructions contained in this work are accurate, the publisher and the author disclaim all responsibility for errors or omissions, including without limitation responsibility for damages resulting from the use of or reliance on this work. Use of the information and instructions contained in this work is at your own risk. If any code samples or other technology this work contains or describes is subject to open source licenses or the intellectual property rights of others, it is your responsibility to ensure that your use thereof complies with such licenses and/or rights.

This work is part of a collaboration between O'Reilly and Auvik Networks. See our statement of editorial independence.

978-1-098-11090-1

[LSI]

Table of Contents

CHAPTER 1

What Is Network Automation?

Drivers and Trends in Network Automation

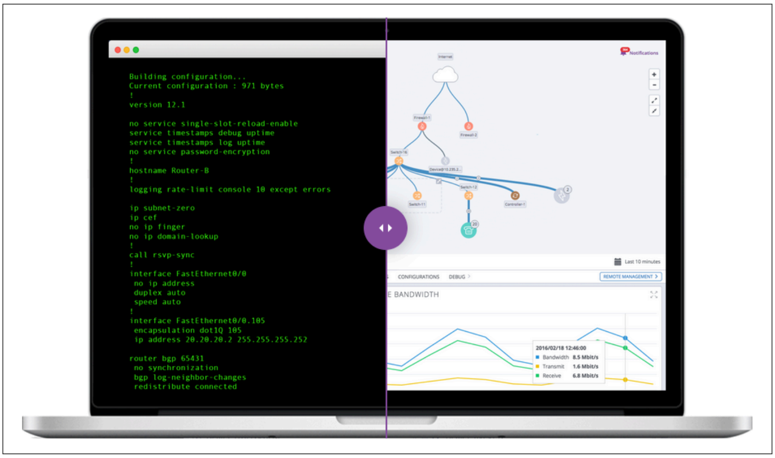
Today's computers, from modern mainframes and hyperconverged infrastructure to smartphones and Internet of Things (IoT) devices, would be almost useless without networks to connect them. Many network components—from routers, switches, and access points to the software that implements virtual private networks (VPNs) and software-defined wide area network (SD-WAN) topologies—are complex mechanisms that, at least from the user's perspective, should "just work" all the time, no matter how loads and user needs change. Theoretically, the best possible performance of a network may be obtained with a manually optimized configuration of all of the network's elements that takes into full account all the components, users, and application usage, along with changes in use of each of these. While this would lead to a perfectly optimized network, working in this way would consume too much time, require so much expertise, and likely still produce enough errors to cause network downtime—after all, there is still room for human error.

Relying on manual network optimization becomes unsustainable in the long run for every organization counting more than a few users or devices under management. In all but the simplest and most stable networks, almost every low-level, manual configuration or maintenance of your network devices represents time and money that could, and really should, be spent somewhere else.

The obvious solution to the problems presented by manual network optimization is network automation, which can occur in several areas. Throughout this report, we will explore these areas of automation available to IT teams and discuss a path toward automating various tasks that today are done manually. If we were to think about each area of tasks that we can automate as distinct "buckets of automation," we could then measure to which degree we automate each bucket. It is important to recognize that some organizations will never need to go to the maximum degree, or maximum automation, in each area. Other organizations may find that different departments or different network segments may experience different degrees of automation as time goes on.

To try to define the degree of automation, let's look at a quick example: a user needing VPN access to a new network for a specific project. We can first look at the simplest and oldest type of automation: minimizing the amount of typing required to add a user to the VPN by replacing typing in a command-line interface (CLI) with single clicks in a graphical interface (Figure 1-1). Although some manual work is still required to perform this routine task, the task has become much simpler and, in a sense, some of the work has been automated. The next step is automating the task initiation. Rather than requiring a person from the IT team to click in a graphical interface, the action is initiated automatically based on certain conditions or actions. To a certain extent, this is the "outsourcing" of some operations. In our example of a user needing VPN access for a new project, the user could access a self-serve portal to initiate creating the new VPN directly from their device, rather than submit a ticket to IT and wait for the IT team to complete those few clicks in the graphical interface. In this example, automation nirvana, or the ultimate degree of automation, is where the VPN is automatically configured, enabled, and connected with no human intervention. This could occur as a "just-in-time" provisioning as the user attempts to access the VPN they need to use. In this example, we've defined three degrees of automation, and although the number of degrees and exact steps for automating this task will vary, we can see that each degree takes one step toward a fully automated task.

Although our example looked at a user-initiated request, a parallel trend has been to automate more of the manual maintenance tasks that the staff in charge of configuring and maintaining a network routinely complete. The natural evolution has been automatically executing some of the simplest operations that occur on a daily basis, like resetting remote machines or reconfiguring some of their parameters from a central dashboard.



```
Building configuration...
Current configuration : 971 bytes
!
version 12.1

no service single-slot-reload-enable
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Router-B
!
logging rate-limit console 10 except errors

ip subnet-zero
ip cef
no ip finger
no ip domain-lookup
!
call rsvp-sync
!
interface FastEthernet0/0
 no ip address
 duplex auto
 speed auto
!
interface FastEthernet0/0.105
 encapsulation dot1Q 105
 ip address 20.20.20.2 255.255.255.252

router bgp 65431
 no synchronization
 bgp log-neighbor-changes
 redistribute connected
```

Figure 1-1. The first step toward automation: the old-school CLI on the left and the "new," faster, easier-to-use (but still manually initiated) GUI on the right.

In recent years, the arrival of cloud computing, either inside company-owned data centers as private clouds or by external providers in public clouds, has made it possible for many network staff to provide not just single services but entire infrastructure and software-defined networks on demand, in real time. This is accomplished in two main ways. One is automation, which is sometimes simply the integration of most of the underlying sequential operations that previously happened manually. The second is through outsourcing, as the automation capabilities provided through software-defined networks increase the number and variety of operations that are safe to outsource. The result is higher user satisfaction, much better utilization of the skills and time of network staff, and much less risk.

Today, the digital transformation that is continuously taking place across businesses of all sizes increases the pressure on networks in ways that call not just for even more automation but also for deep changes in its nature. Some of the main factors that contribute to this trend are the accelerated adoption of ecommerce, video conferencing, and remote work in general. Some organizations may add thousands of devices to their networks in a very short time by adopting IoT technologies, for tracking company fleets or movement of goods through their supply chains, for example.

All of these changes are opportunities for business transformation, which are discussed in more detail in Chapter 2, and make networks much more complex and dynamic, introducing problems and needs almost unheard of before. Even when they are not huge in size, today's networks can quickly become so heterogeneous and so variable in their loads and conditions of use as to resemble some sort of unknowable black holes: places, that is, where it is extremely hard to know what the main problems really are, or even to just *detect* that there are problems!

These facts shine the light on two areas of network management that are relevant to the question of network automation. One is to make visible what was hidden via much better automated discovery and data analysis so as to make it easier to see which actions or decisions should be taken. The other is to perform at least some of those actions. At the furthest degree of automation, this means to make networks self-healing, or able to reconfigure themselves, when faults happen, new subnetworks are added, or usage conditions change.

This short overview shows how network automation makes any organization with a network (which, if we're being honest, is every organization these days!) much more efficient. It is now time to examine the main areas in which this automation should happen inside a network, one at a time.

Network Design

Network automation starts with the design of a network. A good network is, first of all, one whose composition, topology, and configuration (its design) are always completely known. To achieve this goal, we may think that we need to automate the design of a new network itself. But it can also start with automating the representation of an already existing one through automated network documentation. With new networks, design automation gives the network designers the responsibility to *describe* the desired results, such as how many network segments they want, how they are connected, what their connection with the internet is, and how they are protected (for example, allowing only email or web access), as well as the minimum bandwidth to reserve for video conferencing services and so on. With such information, the automation system could then take care of all the low-level details—for example, map out how many switches or routers should be deployed and describe how to connect and configure them. For existing networks, design automation would entail certifying or auditing their actual topology and producing a similar map by probing all the devices active on the network to extract their configuration parameters and infer from them topology and other information.

When a network's entire actual structure and composition are completely known, every part of the network becomes easy to upgrade, replace, or restore after faults, at the smallest overall cost. As obvious as this thesis may seem, in the real world, "completely known" is exactly where the problems start. System administration folklore abounds with stories of ghost servers or switches, long forgotten in some basement but still running and connected to the internet—that is, equipment that hosts content that should not be there, runs software that begs to be attacked, or, in the best possible case, wastes bandwidth and electricity for no reason at all. Even when no ghost devices are present, company reorganizations, acquisitions, or relocations to new facilities can cause unpleasant surprises, creating designs like those depicted in Figure 1-2. In all such circumstances, if the network's inventories and maps do not match reality, administrators will consume precious time just to be sure of what they should do first.

In addition to not knowing of a device's existence and its place within the network, it can be equally easy to forget how some devices are actually configured and *why* configuration choices were made. When network requirements change and static legacy network device configurations remain in place, bandwidth bottlenecks and other performance degradation (read: unnecessary costs and unnecessary poor user experience) may remain hidden for years. Moving content and services to the cloud, either public or private, may increase the likelihood of such events.

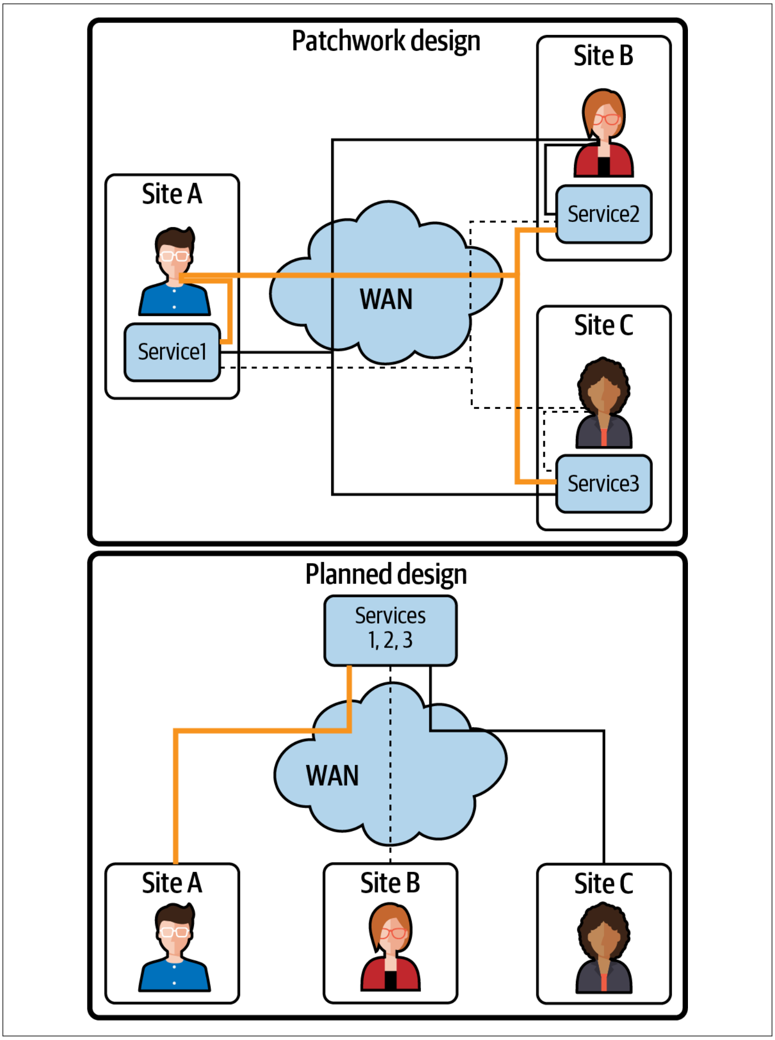


Figure 1-2. Patchwork versus planned design. In the patchwork design at the top, as would be created through M&A or unplanned organic growth, it's a patchwork where users go all over the place to access the services they need. However, in the planned design at the bottom, the services are centralized and access paths are more clearly defined. It's not perfect, but it is much more well thought out.

This lack of visibility and inefficient network design should never happen in a good network. The basic methods and approaches to proper network design and visibility are well known, in principle, and are all based on open technologies. There are many ways to collect make, model, serial number, virtual local area networks (VLANs) and IP addresses, address resolution protocol (ARP) tables, and all other details for every device on the network. Standard protocols and procedures, from wire tracing and port mapping to Cisco Discovery Protocol (CDP) and Link Layer Discovery Protocol (LLDP), can gather all the data that a technician needs to infer and draw a full diagram of the whole network and understand the entire network design. The point is, neither the collection of that raw data nor its formatting and presentation should ever be done, or kept current, manually. Doing so would surely cost more in staff time, without any guarantee that errors would not be made, than using solutions already tested in many other organizations. The same is true, in almost all cases, for carefully crafted in-house custom scripts and tools that invariably end up consuming much more time in maintenance than originally expected.

Asset inventories and everything else that is necessary to have complete network visibility and control in real time should be a given, not something that requires constant intervention or dedicated manual efforts. Maps that can show the exact design of a network, with detailed visibility into the location or switchport connections of every device, should constantly update by themselves, as soon as the network changes. Even higher-level operations—for example, partitioning a network in semi-independent zones that can be independently managed or updated one at a time, without affecting all the others—should happen with as little manual work as possible, following consistent but automatic procedures.

Network Configuration: Policies

A perfectly mapped network is still an ugly place without rules on how to use it and adapt it to its users' needs that are easy to set and follow without ambiguities by all interested parties. The most common but by no means the only examples of such rules and procedures are those used to define bandwidth caps, access-control lists (ACLs), user quotas, password policies, and firewall rules. As we think about automating these rules and procedures, we can think about automation both in terms of policy definition and policy enforcement. Automating policy enforcement is exactly what network devices like firewalls are designed for, so when we speak of automating network policy, it is very much about the policy definition or configuration. When putting together the description and enforcement of exactly how users must or can use assets, it's important that those tasks be automated to consume as little IT staff time as possible. Besides reducing the daily load on network administrators, this automation of policy definition brings two other big advantages: consistency and (self) documentation. To expand on these, if policy definition has been automated, all policies will follow the same format and structure and will look consistent to the readers of your network design and documentation. Along with this, the documentation of policy configuration and changes can be automated at the time of policy creation, so your network documentation is always up to date.

Network Configuration: Provisioning

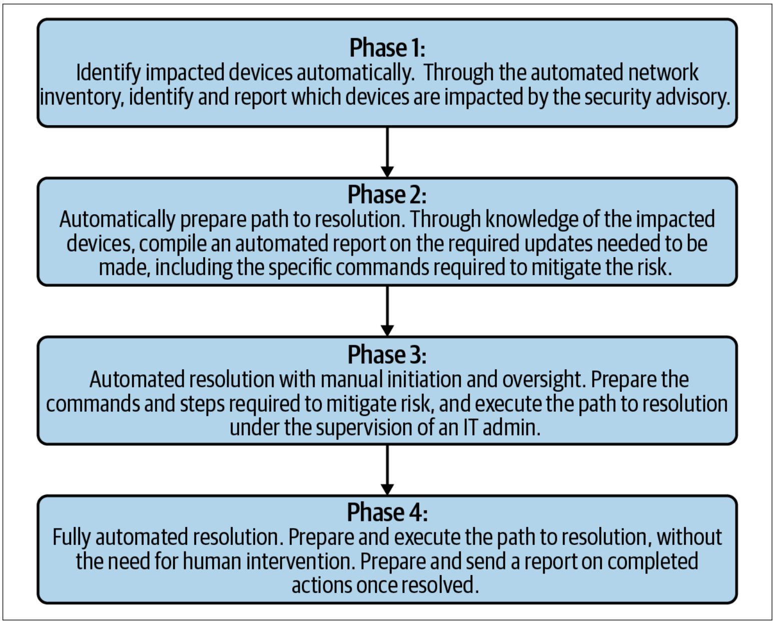
Clear, concretely enforceable rules on how to use the network are of little use if the network itself, or the services it makes accessible, are hard to change. With infrastructure as a service (IaaS), distributed teams, and work-from-anywhere becoming increasingly common, it becomes necessary to provide applications, services, and general connectivity to any combination of local and remote hardware and virtual platforms. To understand when, how, and why this provisioning could happen in practice, it is easiest to consider a software development application: let's consider the developers of some real-time collaboration software that need to reproduce a reported bug. In such a situation, those developers would work much better if they could reproduce the issue exactly as the client sees it, in the exact same network where the bug was first noticed, and at minimum cost. They would need a virtual network to play in, maybe for just a few hours or days, but with virtual switches, virtual firewalls, and so on that both reproduce the desired conditions and keep that area completely isolated from the rest of the network. Other examples may be a company that needs to set up a product demo at a conference or a university that must run final exams in a temporary but tightly controlled network to avoid cheating. Both would have very similar needs and would benefit from streamlined, automated provisioning.

These are just a few examples of why, to keep up with the pace of business, adding users, LANs, VPNs, virtual switches or firewalls, and more must be possible in real time, in ways that are transparent to end users and, to some extent, also to the network staff. In a fully automated workflow, for every situation like those just described, the users should ideally be able to describe what they *need to do* and under which high-level conditions without having to configure intricate technical details manually. For example, "emulate a running website with up to a hundred simultaneous users, each with at least upstream X bandwidth, but isolated from the real internet" is a description of a high-level network to provision, without the need to include all the details. In other words, as far as provisioning is concerned, network automation must make it possible to perform and coordinate all these tasks always in the same way, from the same interface, regardless of where the interested software and physical devices are, and by describing the desired outcome—that is, the final status the network should be in—rather than which options should be set to get there.

Life Cycle

Networks are most valuable when they are reliable, and a reliable network depends on managing the full life cycle—from initial deployment to end of life—of all of the underlying infrastructure that keeps the network up and running. To start, the predictable, regular updates of firmware and software are the simplest of several life cycle issues to consider. Company acquisitions or opening new remote offices are much more complex, but they are likely to happen—in most cases, at least—with enough notice to allow proper planning of how the network should be expanded or redesigned.

A number of less predictable updates occur throughout a device's life cycle as well. Take identified vulnerabilities and the subsequent security patches as an example. This example is well positioned for automation, as security advisories are released without notice and often need near immediate reaction—little time to plan manual activities. Let's look at security patching as another example of a progressive automation. As a first step, a properly automated network should spot and report automatically every security advisory or software update that affects any of its devices as soon as it is announced. This is an incremental process, as shown in Figure 1-3, as the maturity of the life cycle automation increases. Similar notifications and reports should be issued for ordinary new releases of firmware or software, indicating which specific devices should be updated but at first leaving to the administrators the responsibility to push those updates manually. As you increase the degree to which the security patching is automated, these manual updates become automated: first by enabling the IT administrator to simply initiate the process and confirm the result, and eventually without any human intervention or oversight at all. It must be stressed that all of this monitoring should happen regularly, by itself: effective, real-world automation is not a series of fire-and-forget actions but a self-sustaining, incremental process. Even nontechnical notifications, like approaching expiration of support contracts or the mandatory phaseout of some product, should be issued and reported by the network automation system, in one place and one coherent format, to give full visibility of what lies ahead. Ideally, network managers should always have available, in any moment, the exact, complete answer to questions like: if one of my devices fails, am I able to replace it with a similar device, or are those devices no longer available for purchase? As far as it is concerned, the network automation system should contribute to the answer by being able to list, in addition to all the parameters mentioned previously, the exact capabilities of each device.



Phase 1:
Identify impacted devices automatically. Through the automated network inventory, identify and report which devices are impacted by the security advisory.

Phase 2:
Automatically prepare path to resolution. Through knowledge of the impacted devices, compile an automated report on the required updates needed to be made, including the specific commands required to mitigate the risk.

Phase 3:
Automated resolution with manual initiation and oversight. Prepare the commands and steps required to mitigate risk, and execute the path to resolution under the supervision of an IT admin.

Phase 4:
Fully automated resolution. Prepare and execute the path to resolution, without the need for human intervention. Prepare and send a report on completed actions once resolved.

Figure 1-3. An example of incremental progression toward automated patching of network device security vulnerabilities.

As an extension of life-cycle automation, there are continuous changes to compliance requirements with new regulations for privacy, data protection, employee safety, and financial transparency. General Data Protection Regulation (GDPR), the Sarbanes–Oxley (SOX) Act, and the Health Insurance Portability and Accountability Act (HIPAA) are only three of the many regulations that put concrete obligations on company networks in the United States, the European Union, and beyond.

While we often think of these frameworks as putting obligations on data, it is worth noting that these acts have an impact on networks as well. They routinely mandate what a network must guarantee (i.e., uptime) or prevent (i.e., reduce risk) and also how to *present* the corresponding data about the network—for example, through reports in the Information Technology Infrastructure Library (ITIL) standard format.

All of these reports should not be prepared only when an audit is coming. They should always be there, already ready for external or internal audits, courtesy of the network automation services. The same services should also continuously work to *maintain* compliance, refusing—or at least warning against—any change to the configuration of the network that would end compliance with some regulation.

CHAPTER 2

Laying the Groundwork for Network Automation

Stakeholders in Network Automation

Any large or medium-sized organization is composed of many, usually very different parts—manufacturing, software development, IT support, network support, finance, logistics, training, sales, customer service, and so on—and each includes individual contributors all the way up to senior management. Whatever the mission of the organization is, the administrators of its network devote all their energies to making things "just work," in the safest and most effective way possible, for each one of these groups and each individual stakeholder. While the customers for a network admin are those end users, to the end users the network must be invisible while ensuring adequate access to every service they may need, at all times. This includes reliable and consistent video conferencing, safe and secure sharing of data with coworkers or clients, and uninterrupted access to the business applications they need to perform their jobs. All of these services are expected to "just work." For managers, the technology that enables their team, including the network, is just one of many components that must support evolution and growth of the business by enabling all their teams to cooperate in the most efficient and cost-effective way.

In the context of this report, there is one thing that all the components of this picture have in common, no matter how diverse they are: they all take for granted that an invisible but solid network will help them do their jobs. In other words, all of the departments of an organization are stakeholders in its network and therefore influence how its automation should happen and should be managed.

The problem is that, besides different responsibilities, each of those groups has very different skill sets and very diverse, sometimes contrasting, ways of working and priorities. Even among the technical teams, for example, network engineers may not understand software development methodologies and the real-world requirements of mission-critical operations like version control or continuous integration. Software developers, in turn, may not always see all the necessary interdependencies between their own recurring deadlines and the longer-term obligations coming from apparently separate realities, like regulatory compliance, that are among the top concerns of senior management. Technical professionals may also expect much more freedom to configure and use their own devices than security and compliance managers can accept. On the opposite side of the spectrum, nontechnical employees and managers may not see and appreciate all the complexities and hard constraints that make good networks not just useful but also simple to use.

While it is important to identify all of the stakeholders in network automation (which, as we can see, are many!), it is equally important to manage collecting and prioritizing the input from all competing stakeholders. The input from stakeholders will range from high-level business requirements down to technical implementation details, depending on the group the input is collected from, and it must be properly weighted as you're planning your network automation journey.

The intent of collecting this information up front is to enable the network team to plan a path toward automation that supports the entire business, not just the plans for IT. Even if it may seem a bit odd to ask finance or sales what their requirements are for an automated network, having their input early in the process will reduce the likelihood of surprises as network automation is implemented.

This brings us to two simple but crucial consequences. First, the more diverse the organizational requirements are, the more important it is to involve all stakeholders in the planning of a path toward network automation. Second, keeping so many diverse stakeholders happy means that, in order to cope with all of their contrasting priorities, network automation must be simple to implement and flexible to evolve along with organizational priorities.

Industry Adoption of Network Automation

Adoption of network automation has not been uniform across all industries to date. In fact, at first glance it may seem that serious network automation is only really necessary or easy to implement at high-tech startups or for organizations that are born in the cloud or whose networks either already migrated or are about to migrate from company-owned hardware to a private or public cloud. Indeed, running a network in the cloud makes it possible to automate its management without significant investments in networking hardware, which can be quite complex and expensive. These platforms were designed with automation in mind—with the APIs and workflows required to make network automation easy. Besides, working in the cloud leaves fewer things to automate. Instead of dealing with potentially hundreds of hardware and software providers of unique devices, commands, or programming interfaces, there are fewer already well-defined ways of managing the network infrastructure, both in public clouds by major providers like Microsoft Azure, Amazon Web Services (AWS), or Google Cloud Platform (GCP) and in fully private ones. In every sector of the economy, organizations can become leaders in network automation not because they explicitly wanted to but simply as a consequence of their move to increasingly convenient, and sometimes unavoidable, cloud-based infrastructures and services. It is also true that automation is much easier to adopt at high-tech startups, and it's simultaneously more necessary for them. These companies by definition have early-adopter mindsets and no legacy equipment to worry about. Besides, while they are small, they need to be ready to grow really rapidly, without conspicuous investments up front.

This does not mean that network automation is only needed and beneficial in those cases! Network automation can benefit organizations of all types and all sizes to help improve efficiency, compliance, security, and productivity. For any organization, especially those with legacy on-premises networks, getting started would be more complex, for obvious reasons. Almost always, actual hardware components inside company data centers and offices are much more different from one another than their virtual or hardware counterparts inside a cloud. However, even when it involves substantial investments in rewriting network management practices, there still are many scenarios where it makes more sense to keep and properly automate "legacy" networks than to leave them for any cloud. Some companies may have hardware that still works perfectly and is a long way from paying for itself or legacy, mission-critical applications that cannot run on virtual machines. Others may, due to security or data-protection constraints related to the specific nature of their business, be bound to keep some of their files and applications accessible only from inside their own premises. Even in those cases, however, automation will make network management easier, more reliable, and therefore more secure.

Inside or outside the cloud, the first, most obvious driver of network automation is the increasing, ubiquitous digitization or automation of every ordinary activity, from programming and manufacturing to meetings, sales, and customer service. Besides, thanks to 5G mobile networks and edge computing, even companies that have only one actual office may begin to handle much more traffic from external locations in the coming years than they do today. For some companies, the data exchanged with all sorts of traditional mobile or fixed terminals may become the smallest part of the total network load, falling far behind the data coming from IoT devices deployed for services like shipping, inventory management, or environmental monitoring. As size, complexity, and entropy in networks increase over time, there is an ever-increasing need for automation.

Automation's potential to enable the predictive analysis of a network will increase in the near future, thanks to increasingly common artificial intelligence and machine learning techniques. This will make it much easier to study how a network actually behaves under real-world, highly variable loads in order to estimate where faults or disservice are more likely to happen and what could be done to improve overall performance while minimizing both costs and human errors.

In the next few years, end users of modern networks will demand more automation, albeit not explicitly, in several ways. One will be the growing requests, if not happening already, to support work-from-anywhere practices on any device. Standardization of personal computing devices is no longer uniformly possible. Educational institutions, retailers, coffee shops, and other providers of WiFi access in more or less public spaces may be the most common sources of this kind of demand but by no means the only ones. In general, any organization that relies on highly interactive online services, from ecommerce to gaming to ehealth, would have to face the same general challenge: the more diverse user devices can be, the higher the need to automate everything related to how they should or could interact with the network.

Even inside relatively stable groups of corporate users, both telecommuters and truly mobile workers will become more common, and mere connectivity to the company network will be only the first, most basic thing that will need automation. On top of that, these users will also need much more support for, to cite a few examples, single sign-on, VPNs, and transparent but secure access to every document or network-dependent service, regardless of where they are.

Telecommuting, mobile work, and bring your own device (BYOD) practices will also impose enforcement of more policies ensuring that network resources are always used in the optimal way and never abused. Regardless of how or from where their users will connect to their network, its administrators will need to guarantee quality of service to guarantee the user experience. In technical terms, these are parameters like minimum bandwidth, maximum latency, and mandatory encryption procedures. The same administrators will need to make sure all their equipment gives the right priority to certain applications (e.g., interactive points of sale or remote customer service), while limiting or completely excluding others (e.g., music streaming). In general, any network that for these reasons must frequently but securely update their firewall settings, VLAN topology, VPNs, or any other part of its architecture will have to automate all of the corresponding procedures completely.

Last but not least, it is not just the management but also the initial installation and configuration of the network devices that should support these services—and this may have to happen in the most cost-effective way in order to increase capacity and efficiency for an organization's team of highly specialized network admins. There are many firewalls, switches, WiFi access points, and similar products that can fetch their full configurations from the network management console practically by themselves the first time they are powered on. However, they are of little practical benefit if that network is not capable of taking control, thanks to configuration templates, interfaces, and other tools that allow the administrators to specify all of the relevant data without errors but with the smallest possible effort.

Taken together, all the factors and industry trends introduced so far are important enough to drive the global network automation market to be "worth USD 22.58 billion by 2027 while exhibiting a stellar CAGR [compound annual growth rate] of 24.2% CAGR between 2020 and 2017," which Marketwatch attributes to rising investment in automated solutions in North America. A nonnegligible part of this growth will be due to trends like hyperautomation, defined by Gartner as "a business-driven, disciplined approach that organizations use to rapidly identify, vet and automate as many business and IT processes as possible [through] the orchestrated use of multiple technologies, tools or platforms." That doesn't just include every business process but *integrates* them. As a consequence, every organization will soon require at least some degree of advanced network automation, not just those with the most stringent needs for highly scalable, dynamic, and remote management.

All of this means that automation, including network automation, is on an ever-increasing path. Automating network functions will become a requirement for businesses and the network admins that manage their networks in order to stay competitive in their respective markets. Whether your industry is at the forefront of network automation or just starting to dip its toes into it, it's coming, and the time to start planning for your adoption of network automation is now.

CHAPTER 3

Assessing Organizational Readiness for Network Automation

The first two chapters of this report described how and why serious network automation is not a challenge but a necessity for any medium or large organization. Only companies with a network that can quickly, automatically, and proactively adapt to new conditions are always ready to seize new business opportunities or recover from incidents. Before embarking on this network automation journey, however, network administrators and their leadership teams need to really know what they know about their network and possibly what they do not know. This process of assessing organizational readiness starts with a clear picture of the current state. Only then is it possible to lay out the areas where a specific network can be automated, along with the ways and times to do it that are optimal for that network.

On a related note, it is worth pointing out that preparing for network automation can also expose the strengths and weaknesses of an organization itself along with those of its network, which can provide important insights on improving its way of working.

It is not uncommon for an organization to ignore or forget how and why some of its own in-house tools were developed, how they evolved over the years, and their implications for the future of its network. To put together a simple but concrete example, consider the very common case of a script created years ago to monitor or reconfigure remote network devices via Secure Shell (SSH). If that code, as it often happens, had been expanded in many small but hasty steps, possibly by different people at different points in time who cut and pasted commands without enough attention to performance and robustness, then the current script may be launching 10 distinct SSH connections at every run, each performing just one single operation. Such a script would not just be much slower than one that connects once, runs all of those 10 commands in one fell swoop, and exits, but it would also be much more fragile. Any temporary loss of connectivity during execution could corrupt the outcome of the remaining commands, thus leaving the remote system in an unknown state or not working at all. Not to mention that the documentation of the actions performed by such scripts is probably only known to one or two network admins—if known at all! Scripts like that may be just one of several categories of "dark objects" that a thorough assessment for network automation readiness should discover and aim to fix.

Defining the Parameters for Success

How can IT managers define and recognize successful network automation and measure how far they are from achieving it? A single-sentence answer may be simply "peace of mind." Ultimately, network automation is what gives both managers and IT staff peace of mind, confidence that they are not wasting precious resources, and adequate preparation to face even serious network damages. However, measuring peace of mind is too subjective, so in practice, we must establish some basic quantitative parameters that define and measure the success of a network automation program. Other goals, like our peace-of-mind goal, are more qualitative in nature and can vary greatly from one company to another or in different moments in the life of the same company.

Let's explore some quantitative measures. The first category includes both "reactive" metrics, like mean time to repair, recovery, or resolution (MTTR) from a software or hardware failure, and "proactive" ones—that is, measurements of network uptime or of the time and money it takes to, for example, make each type of addition or change to the network (new firewalls, LANs, servers, and so on) or to resolve end-user network issues. The number of network-health parameters (e.g., the average load of every router) and the percentage of configuration options that can all be checked or set through a single “network dashboard” are other useful indicators for how far from successful automation an IT team is.

Specific measurements like these are the basis for measuring the progress of automation in a network, but this only describes its lower-level benefits. These lower-level quantitative measures will lead to higher-level qualitative outcomes that can still be relatively simple to define, if not to achieve. One example of a higher-level qualitative outcome is the capability to detect and remediate equipment faults or performance issues *before* the users report them, through proper traffic analysis tools.

One clear indicator of early success is knowing where all answers to questions about the network are—that is, being sure not only that your network automation system collects all the relevant data but also that the data is stored and presented in coherent formats, ready for fault analysis and performance optimization. Success at later stages may include statements like “it is possible with just one click or command at the prompt to make each device fall back to its previous known good state” whenever a reconfiguration or software update goes wrong for any reason. In general, automation is successful if it really minimizes two things: first, the number of events that require manual intervention, like resets or reinstallation of devices or services; and second, the number of nonstandard or different ways to perform the same task. Configuring the password or routing table of a router should always happen in the same way, even if a company has, for whatever reason, many different models of routers. If, in the main network administration console, it is necessary to remember the make or model of a router to set its routing table, that is a sure sign that there is room and need for more automation. Having instead, thanks to a successful implementation of network automation, a single interface that always asks for the same parameters regardless of what hardware it controls will:

- Reduce both the time needed to make changes and the probability of errors
- Enable more people to perform a task that maybe only one person knew how to do before
- Add standardized checks and reporting capabilities to the workflow, at little or no extra cost and effort

The metrics and services described in the previous sections and in Chapter 1 prepare network teams to face both predictable and unpredictable events. As an example of being ready for predictable events, consider hardware life cycle. Readiness is more than just knowing some equipment will have to be phased out in one year but also includes having reliable roadmaps and cost estimates for how to actually refresh and replace the hardware well before that moment comes. With unpredictable events, like the destruction of a server room, automation makes it possible to have a reliable plan and a complete set of cost and time estimates to restore the network somewhere else.

More ambitious network automation results have less to do with technology and more to do with people, company culture, and other nontechnological factors, like regulatory constraints. One thing that makes networks brittle, even when they are automated, is depending on too few people to make that automation always work. Does the network team include people who cannot afford to take days off because if something happened during those days, nobody else would be able to handle a fault with the same efficiency? If there are such figures in a team, that is a very strong sign that network automation is not being used as much as it should or that what has been implemented is too complex. After all, one of the benefits of network automation is that it should enable network admins of varying levels of experience to perform network functions.

In the long run, successful automation reduces the interruption of services to the absolute minimum due to maintenance work or provisioning requests and makes these tasks as efficient as possible for the staff. This could mean offloading requests completely, like giving internal clients a web interface from which the system can handle tasks like provisioning a virtual machine or a file server. The general goal should be to reach a point at which all minor changes to network configuration can happen automatically, during production hours, and without interrupting services. This, in turn, would make automation proactive. These days, very few organizations think they want, or could handle, automated fault responses. But an automatic return to safe conditions after faulty operations, such as the self-healing networks mentioned in Chapter 1, is something that should be seriously considered for inclusion in any long-term network automation plan.

Assessing the Current Level of Maturity for Existing Processes

After deciding what your parameters for success will be, how those parameters for success map to your own network, and in which order they should be pursued, it is possible to evaluate how much effort, and where to direct that effort, will be necessary to get there. This assessment starts with understanding the state and maturity of the existing processes.

A major obstacle to starting down this path may be company culture, from both senior management, who may underestimate the benefits of automation, and technicians. Even though technicians may argue that the company network is too unique to be automated or may not see the full benefits of automation, network automation will ultimately improve the efficiency and effectiveness of available resources. This will also allow the technical team to grow professionally and learn new technologies, enabling them to stay skilled-up in network automation technologies and improving future career prospects.

Ultimately, these objections will need to be overcome, so it is up to the leaders of the IT organization to figure out the correct responses to them. This comes from gathering the right industry trends, tying the parameters to business outcomes, and establishing a positive return on investment for any network automation program. In virtually all cases, the answer will favor automation.

The starting point for assessing current process maturity is to ask questions about the current process, such as:

- What are all the processes that we have for network management, and are those processes documented?
- How many of our processes (provisioning, user management, software upgrades, etc.) are already automated and in which way?
- How often are there “exceptions” to the standard process? Are these exceptions well known and well documented?
- Which processes have dependencies on other automated processes or on manual steps?

- Can those processes that are already automated be further integrated with one another or inside future, more sophisticated solutions?
- How many of the manually performed processes can be automated and how?

A great start to assessing current processes is to build a table like the one in Table 3-1 that includes details of all the known processes.

Table 3-1. A sample assessment of the maturity of existing processes

Process name	Documented (Y/N)	Location of documentation	Process "owner"	Process last updated/ reviewed	Automated (Y/N)
Rebooting a switch	N	N/A	John	N/A	N
Adding a firewall rule	Y	<file>	Annie	Sept. 1, 2021	N
...					

What you will likely find is that your existing processes may not be as mature as you'd like. And when it comes to automating network functions, you'll want to automate solid, well-defined processes. Running through this assessment is a critical part of your path toward network automation. Keep in mind that when you get to the stage of automating existing processes, they must have a few things in common: they must be hardware and software independent and future proof, and they must remain at least partially valid even if new, different devices are added to the network or its usage changes substantially.

Identifying Achievable Milestones for Automation

Once the status of the network and the expectations about it are known, it is finally time for planning and then implementing automation with both short (e.g., three to six months), medium (12 to 24 months), and long-term well-defined milestones.

The actual number, content, and timing of all network automation milestones may greatly vary, of course, depending on the initial situation and the goals that were chosen. Organizations with just a few hundred employees, homogeneous hardware, and a good initial situation may be able to do most of the work in less than one year. Companies with a much larger workforce, heterogeneous equipment in multiple locations, or aims to deploy network automation as just one part of a much larger reorganization would take much longer. What is possible in both these extreme cases, however, is to outline a succession of milestones that almost every organization will have to go through, obviously at its own pace.

The following is by no means intended to be the ironclad timeline toward network automation, but it is useful to depict the process with an example. So, as an example, a company with a thousand office workers, distributed over three separate locations in as many cities, each with its own network administration team, may aim to:

- Make a complete inventory, choose a new network automation platform, and test it in one of the locations (the one with the most homogeneous local network or the smallest number of employees or devices), starting with the automation of reporting and simple, frequent operations *inside the local network offices*, **in six to nine months**.
- Extend the same automation, according to the results and guidelines emerged from the initial testing, to the other two facilities while using the first one to test automation of *telecommuting*, in the following **six months** (12–15 months total).
- Completely standardize all maintenance of the network, as well as provisioning of new services, from just one central location, **in the following 12 to 24 months** (24–39 months total).

No matter the timeline, the first steps of this journey should include an inventory of all tools, including those installed but not used anymore, and a staff audit, to verify their skills and the need for training or external support. In parallel, the staff should identify which operations, services, or even departments would be the best candidates for "quick wins"—that is, the places where the first visible improvements could be reached in a short time frame and with relatively little effort. Another outcome of these investigations could be defining which departments may serve as test beds for the whole company in each successive phase of automation.

On the users' side, their input will be necessary to understand exactly what their major or most frequent complaints really are. Is the network too slow or too unstable? Are all services affected or only those with more stringent requirements (e.g., video conferencing)? Are the problems constant, or do they happen only in certain moments, such as when software developers start testing a new release of a product? What changes and how when users telecommute?

After this prestudy phase, network managers will have a complete list of actions and deliverables and a detailed time schedule to follow (see Figure 3-1). The exact content of that plan will vary enough from one organization to another that it would not be possible to investigate it in more detail in this report. Two metamilestones, or higher goals, that should be present in every plan, however, are worth mentioning.

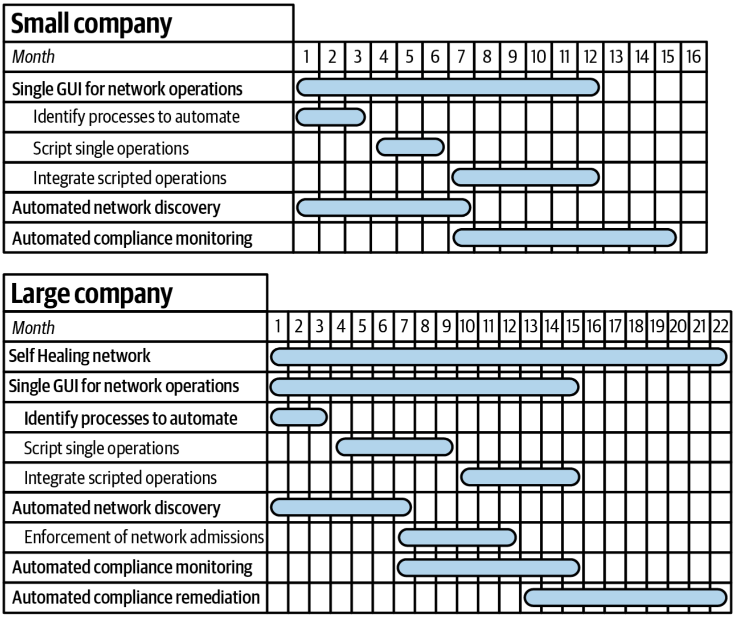


Figure 3-1. Examples of actions and delivery timelines on the path towards network automation. Note that the deliverables and timeline in your organization will vary based on a number of factors.

The simplest, less powerful automation is the one that proceeds bottom-up, writing scripts for each simple task and attempting to integrate them as time goes by. Automation becomes really useful, sustainable, and scalable, however, when it always allows administrators to work top-down, by describing the desired, high-level *result* of some procedure (e.g., "create a VPN for these users with minimum guaranteed bandwidth equal to 10 MB/sec"), rather than specifying how to configure specific firewalls or routers to achieve that result.

Whatever the long-term goals of an organization are and whatever the initial status of its network is, there is one long-term milestone that should be seriously considered for inclusion in any network automation strategy. This is the achievement, already mentioned earlier in this chapter, of a level of automation that allows all minor changes to happen during production hours, *without* interrupting services, involving the network administrators only as controllers. The goal should be to give internal clients a web interface to make tasks like provisioning a virtual machine or a file server happen automatically, without interruption of services but also without the manual intervention of network administrators. From this perspective, the network automation plan should include, at least as a medium-term milestone, the virtualization of at least some services on an internal cloud, if they already aren't. This would make the automatic provisioning of those services much easier, not just on the company network but, above all, on public clouds, should the need for such a move arise in the future.

Measuring Results of Network Automation

As with any other project, it is critical to stop and reflect on the success of any network automation program. As previously discussed, it's important to define the parameters for success because no plan would be complete without concrete metrics that allow checking if certain equally concrete goals have been reached or not. While your specific parameters for success will be unique to your business, the action of measuring those parameters will be universal. When it comes to network automation, the things to compare against the expected results may be divided into three categories.

The first category is about checking if, how, and how much what was already working has *improved*. The basis for these kind of assessments are data about network uptime or, for example, how much *less* time and money it takes on average, compared to 6 or 12 months before, to serve user requests, close a trouble report, or update the firmware of all routers.

The second category includes *reducing unplanned events*, like a decrease in the number of incident reports to reduce the amount of unplanned overtime (and overnight!) hours among all the administrators.

The third category is about *preparedness* for major negative or positive events. These could be anything from actual disasters to deadlines to comply with some new regulation, significant changes to company structure, or business opportunities coming with very little notice. Measurable results in this category consist of reliable estimates and detailed answers to questions such as:

- How long would it take and at what cost to restore full network functionality after certain major accidents (such as a server physical crash, fire in the data center, etc.)?
- If we had an audit tomorrow, would we pass it?
- How many employees could start telecommuting right away, if management decided so?
- How much time and money would we need to open a new office in another town that is fully connected to the rest of the network but manageable remotely?
- If we had to build from scratch the same network we have today, how much time would it take to reconfigure every device exactly as it is now?

Indeed, for some organizations, studying how to be prepared for some of these events may be worth doing just as a thought exercise.

No matter how you choose to measure the results of your network automation, ensure that the results are tied to measurable business outcomes. This will enable the IT leadership team to properly quantify and communicate to the business all of your team's hard work.

CHAPTER 4

Planning Your Adoption of Network Automation

If going from manual to automated network management is surely good and worth the effort, the problem becomes selecting which kinds of products and management platforms should be used to achieve that goal. This is a problem that, paradoxically, has been *complicated* by the very importance and usefulness of network automation. Since the arrival of digital communication networks, so many solutions to automate components of network management have been developed, from monolithic platforms to full custom software to tiny shell scripts in many different languages, that it can be really hard to choose which way to go or how to integrate separate tools. Let's then look at some practical considerations for teams who are wanting to implement or improve their automation practices.

A single console that can be supported by automatic scripts but lets the administrators issue any command that the network supports while always presenting one coherent, interlined view of all relevant data is an essential component of every network automation solution. Over the past few years, this assumption has led to a few "turnkey" graphical dashboards promising network automation. Although the convenience of such user interfaces is without question, we all know the value of a platform goes well beyond its looks, and looks alone do not guarantee that the platforms are all equally efficient and sustainable in the long run. The potential problem is not in the dashboard itself but in how the automation is implemented in the backend. If a network automation platform has (regardless of its license!) a monolithic architecture that is hard to expand or customize, it may be unnecessarily difficult to "attach" it to networks whose design principles were very different from those imagined by the developers of that platform.

Similarly, there have been a number of networking technologies that include an "all-or-nothing" platform requiring all network hardware to be homogeneous and locked into a specific vendor. Although these types of platforms have an appeal for some smaller, simplistic networks, they will severely limit their own real-world efficiency in the long run as the business grows, no matter how convenient and easy they appear to be today—unless, of course, the users could be confident that the network they have to manage with them will not undergo any significant changes for years, which is rarely the case.

As we have seen, almost any modern network must continuously evolve with its users. It must be always ready to support, for example, the addition of remote servers or entire offices, possibly with very different hardware, usage patterns, and automation capabilities than the existing ones. Accordingly, its automation-management system should support both its own initial installation and future events like these with minimal training of the existing network teams and without hiring external professionals every time a major change is needed.

The consequence is that, rather than monolithic platforms, it makes more sense to select a solution stack that fully supports both these expectations and the best practices summarized in the final part of this section: a solution stack that is flexible, easy to understand, and capable of providing an out-of-the-box experience from the first moment for most of your already existing needs but is also ready to grow and scale without restarting from scratch every time. It is important to note that looking for network automation solutions of this kind also makes it easier to trust them enough to get the most out of their adoption, as distrust of network automation vendors has been defined as "one of the primary reasons hampering the growth of the [network automation] market" and, consequently, of all the benefits automation can bring.

Best Practices for Network Automation

Every network is different, but whatever network you have or want to build, it is safe to say its automation must support the following best practices, or capabilities, which summarize the previous parts of this report:

Do things once
: As obvious as it may seem, this bears repeating. As a very basic but adequate example, the full configuration procedure of a switch with one click or one command in a script is an activity that ideally should be taught to the system just once, by creating one template with sensible defaults that could then be applied to any make and model of switch (including future ones!) by just changing those default values when needed.

Choose solutions that adapt
: Solutions should adapt to your existing network, budget, and capabilities and the experience of your staff instead of demanding that you adapt to them.

Make the best of your staff skills, without ever overloading them
: Network automation must do more than minimize the mere number of total working hours required by the whole network support staff. It must make it easier to distribute both workloads and responsibilities among them, in the safest and most cost-effective way, while *increasing* their skills. At the same time, automation should give the peace of mind that comes from knowing that there are no critical dependencies on any single expert if some particular incident happens because the system can help whoever happens to be on call during that moment to do the right thing quickly. At the bare minimum, the automation system should support instant fallback to the previous working configuration of some device if some update did not leave it in a working state or generated a fault.

Choose solutions that are self-documenting
: Self-documenting solutions should save and present all the data you need but not more. Diagrams or real-time animations are great for quickly understanding and documenting what happened in the network but cannot be the only resources of this kind. Audit trails and log entries automatically documenting what network changes are made, together with approval

workflows with logged approval entries, are what enables an automation platform to be self-documenting and fulfill both industry best practices and some compliance requirements.

Finding Your Solution on the Open Source Versus Proprietary Spectrum

Automation does not necessarily mean giving up control or ending up locked into a monolithic platform that becomes more difficult or expensive to maintain each year! The contrary is true, actually. Automation means gaining and maintaining over time not just full control of your network but also the freedom to change your automation practices.

As true as it is, this definition needs careful consideration and further qualification, or its results might be a bit misleading. At first glance, it may even give the impression that the best, if not the only, "true way" to effective network automation goes through doing everything in-house.

Instead, what really matters is to find and adopt solutions so good that you do not want to abandon them but that still leave you free to do so without unnecessary complications if you choose to, no matter who provided the solutions.

At a lower, more pragmatic level, automation must also mask the differences between equipment providers. The natural answer to such a concern is to build a network based as much as possible on open standards for network configuration and automation. This guarantees that networking equipment from different vendors does not expect different configuration commands to perform the same function (such as opening a firewall port), as this would make things unnecessarily difficult while increasing the probability of misconfigurations and network outages. Although many network vendors strive to support open standards, not all do—so naturally, when selecting a solution stack for automating network functions, you need to ensure the solution is flexible enough to support proprietary standards in the future.

A very important quality of open standards is that, if chosen properly, they avoid reinventing the wheel. Open standards often have community support that drives development of functions relevant to your network as well, so you don't have to write scripts and code from scratch in-house every time. From this perspective, as a reference for evaluating how open standards can and should help your adoption of network automation, the rest of this section briefly mentions some high-level general concepts that constitute foundations on which it is possible to build automation that is both easier and free of lock-in. We'll briefly explore network functions virtualization (NFV), software-defined networks (SDNs), and the Open Network Automation Platform (ONAP).

NFV is the decoupling of network functions from proprietary hardware appliances and running them as software in virtual machines on whatever host platform may be most convenient. The specific functions provided by an NFV environment are called virtual network functions (VNFs) and include, but are not limited to, firewalls, traffic control, and virtual routing. Thanks to them, NFV brings benefits similar to those of server virtualization: increased hardware utilization and flexibility at lower costs.

SDNs can use NFVs, but they go a step further. They abstract the control and management planes from the underlying hardware with a software layer that can be managed through physical or virtual controllers. Rather than simply making automation simpler and cheaper, this separation enables things that would not be possible otherwise, by making the networks both hardware agnostic and programmable—that is, capable of being partitioned with great precision with a few simple commands from either a script or a graphical console.

These concepts are enshrined in open standards under ONAP. This platform, under the governance of the Linux Foundation, enables VNFs and other network functions and services to be interoperable in an automated, policy-driven, real-time environment. This provides everyone (as ONAP is fully open sourced and the code is free for everyone to see and consume) the ability to fully create, design, and deploy for automated network services.

Conclusion

Open standards can play a crucial role in your path toward network automation. When exploring the market for vendors to assist you, look for vendors that have embraced those standards, together with the other concepts and best practices described in this report, as opposed to reinventing the wheel in proprietary solutions. Network automation is still a relatively new and evolving field, so while there may not be an out-of-the-box solution that perfectly matches your network today, vendors that align with these concepts will converge on a solution that works for your business as the network automation market matures.

Align with network automation platforms and vendors that allow you to focus on your goals without wasting countless hours in low-value, low-level, and error-prone operations. Remember, network automation is an ongoing process. Therefore, whatever the structure of your network and your expectations about it are, it's important to keep some key considerations in mind:

- Identify where you're at in your automation journey, and partner with solution providers that will support you at every stage along your journey toward a fully automated network.
- Make sure that you can always really count on reliable, affordable support (either inside or outside your organization).
- Put together a realistic plan to get you to a more "automated" state, and stick to your plan.
- Execute the plan, setting concrete milestones and ensuring that you pause to measure progress against your success parameters.
- Continuously evaluate the network automation market, watching for changes and new developments in this evolving field that can help you on your journey.

At this level, perhaps the only guideline that is always valid is that incorporating open standards and automation will nearly always benefit your business.

About the Author

Steve Petryschuk is a product strategy director at Auvik Networks, a provider of award-winning cloud-based network management software. Steve works with prospects, clients, and the IT community at large to identify, analyze, and solve complex network management challenges, while helping guide the Auvik roadmap to better service the IT community. Steve holds a Bachelor of Engineering and Management from McMaster University and is a registered Professional Engineer in Ontario.

Outside of work, you'll find Steve spending time with his kids at the hockey rink or soccer field, exploring the lakes, rivers, waterfalls, and mountains across North America (or wherever the travels take him!), or busy repairing whatever it is the kids have managed to break around the house.