Cloud-Native DevOps and DOCSIS Networks

How to improve performance and reclaim control of your broadband business

Hardware-to-software inflection point

The cable industry is considering several major upgrades to its hardware-centric DOCSIS infrastructure. A fundamental shift from hardware to a more software-driven model is a major opportunity. DevOps practices, such as those enabled by the Cisco Cloud Native Broadband Router (cnBR), can help mobile service operators (MSOs) overcome existing constraints. They can improve performance and embrace change by building up and tearing down functions, features, and services on their own timetables. In doing so, they can meet heightened expectations for service delivery and efficiency.

Contents

DOCSIS networks and capacity needs

Over the past two decades, the cable industry's DOCSIS networks have helped shape and meet an exponential growth in demand for broadband services. In North America and around the world, as MSOs continue to gain high-speed data subscribers, the business is shifting from triple play (voice, video, and data) to pure data. Broadband has enabled a service convergence at the expense of traditional revenue-generating services. Broadband data continues to absorb voice and video services, substituting the traditional with over-the-top (OTT) services, such as IP telephony and IP video. The decline in traditional service offerings is in part due to OTT competition from highly efficient and agile web-scale companies.

For MSOs, their future resides on their broadband networks. However, that infrastructure is under stress. From a capacity standpoint, not only are subscribers growing, but the number of devices and data consumption per capita are also increasing. Broadband video is an especially strong driver and a good example of how traditional services are jumping aboard the broadband train. Studying the impact of video on network capacity is one way we can better adjust and accommodate for future needs. According to the Cisco Visual Networking Index, IP video will account for 82 percent of all global IP traffic by 2022. Of that, over half will be high-definition video, and 22 percent ultra high-definition, up from 3 percent in 2017.[1] Even with improved video compression and other bandwidth-saving techniques, these trends pose challenges.

To address those challenges and expand capacity and capabilities, MSOs are once again preparing to invest. The industry is juggling several infrastructure upgrades, including migration to a Distributed Access Architecture (DAA), the viability of Full-Duplex DOCSIS (FDX) and Extended Spectrum DOCSIS (ESD), and the even more drastic measure of going full fiber. But even if these investments are necessary, they may prove to be insufficient if MSOs fail to address a second area of stress: operational scalability.

Current operational constraints

Over time, cable DOCSIS networks matured, while OTT providers leveraged the power of software. Performance expectations shifted as web-scale companies replaced the services that were the foundation of cable's business model, changing the standards for service delivery, agility, and stability along the way. The cable industry's hybrid fiber-coaxial (HFC) infrastructure has evolved and will continue to do so. But along with DAA and other upgrades, MSOs aiming to meet higher performance standards will need an operational transformation to overcome embedded obstacles.

The hardware-based nature of cable networks translates into inefficient and manual-intensive operations. Upfront, these tasks include racking-and-stacking individual cable modem termination system (CMTS) or converged cable access platform (CCAP) devices and lengthy, usually manual configurations for each box. Then there are silos and inefficiencies in network management and monitoring, which involve third-party tools, aggregated low-level data, and complicated rules for checking system health. Even if you're not flying totally blind, troubleshooting can still require time and expertise, rooting around in the command-line interface (CLI) for problems and issues. All of these operations are performed manually, box by box, node by node.

Problems in cable networks can be disruptive. With no economical way to ensure services, problems can significantly impact subscribers. One underlying problem is how the current network model limits communication, cohesion, and visibility. Unlike in a modern Converged Interconnect Network (CIN), the connection between headend and nodes is point-to-point, which limits capabilities. Remediation requires visible data. But no correlations are available until you dump data into a collector and begin sifting through it. Then remediation entails another round of manual interventions and procedures.

The current model also limits the ability to leverage improvements in technology. In cloud computing, software is abstracted from the underlying server infrastructure, which can be regularly refreshed. In the conventional headend or network operations center (NOC), however, cable operators must live with

equipment. They need to manage internal components such as ASICs across the lifespan of a device and are locked into the vendor's overall velocity and feature commitments.

Network component vendor velocity has been inherently low. Although agile software development is a well-established IT practice, it is not widely deployed in networking circles. In current operations, software upgrades arrive infrequently and with waterfall-style disruption and risk. An upgrade that goes awry could drastically impact systems and services.

A similar dynamic occurs in feature requests. Following the traditional process, requesting a new feature requires patience. It can take up to 24 months of development and testing before a new feature becomes generally available on a production network. Staying the course, being at the mercy of a vendor's cycle and priorities is risky, especially when that cycle is slow, and customers increasingly expect choice and quick availability.

Network dilemmas, web-scale solution

The cable industry may have grown used to these constraints on operations and device management, but they are reaching their limits so proportionally and profitably scaling broadband services is no longer feasible.

To satisfy customer demand, you must increase capabilities on all fronts. You need more capacity, more service offerings, quicker time to market, and faster response times. If the network's transformation is not carefully planned, some options that are aimed at increasing capacity, such as DAA, FDX and ESD, can actually increase complexity and create more challenges for operators to meet customer demands. Digitizing deep access with Remote PHY, for instance, taught us lessons in how to build a DAA. Processes that seem like manual effort upfront can begin to scale, which allows operators to reap the benefits down the road.

In a traditional, hardware-centric, manually driven system, providing more services and operating more efficiently means deploying significant engineering resources. Even then, the time it takes to decide,

develop, install, test, and deliver new services can be extensive.

Given these constraints, it makes sense to consider another model. One alternative is to follow the lead of web-scale companies. By fully leveraging cloud computing and adopting software development and IT operations (DevOps) practices, MSOs can regain control and become more agile and efficient (see Figure 1). Cisco has taken this approach in reimagining the headend with the Cloud Native Broadband Router (cnBR).

Figure 1. Cloud-native vs. existing operations
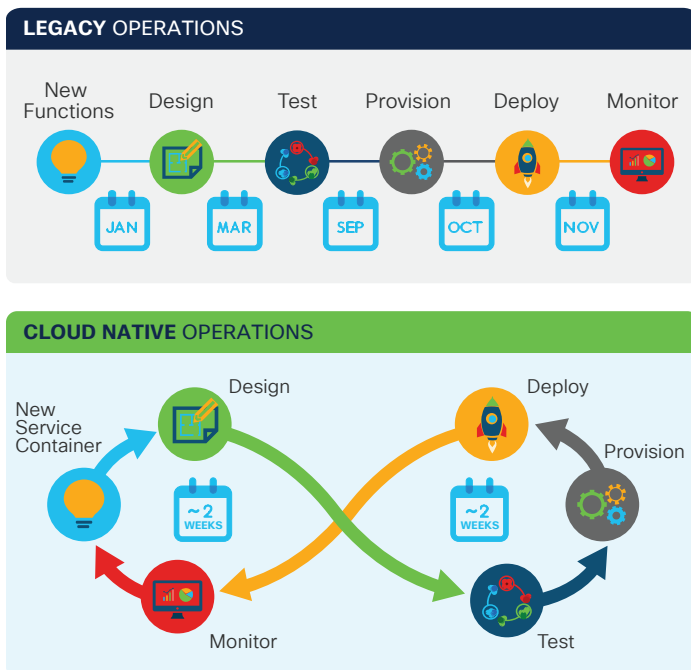
The cnBR and cloud-native benefits

A decade ago, it would have been hard to embark on this journey because existing systems weren't near the end of their useful life. But with the passage of time came a need for more accurate scaling of services and operating processes and a search for better resource utilization. The concepts of software-defined networking (SDN) and network functions virtualization (NFV) also gained steam, along with the cable industry's DAA proposal.

With a DAA, operators can distribute and virtualize CCAP functionality, replacing existing hardware with virtual network functions (VNFs). At Cisco, we saw early on that lifting and shifting existing software from a hardware-based platform and loading it on a server didn't lead to real efficiencies. Learning from initial attempts at virtualization, we focused on enabling the network to become a platform for natively hosting functions, services, and applications. As early attempts at virtualization failed to meet high expectations and hype, we decided to go beyond virtualization and more completely take advantage of the cloud. We refactored our CCAP platform from the ground up to create building blocks out of loosely coupled, container-based microservices, which are important tenets of being truly cloud-native.

Software containers and microservices are central to the software architecture. A high-performance container networking layer sits in the middle of the cnBR stack (see Figure 2). The networking layer also is how the virtual functions connect to the real-world functions and fulfill the original expectations of virtualization.

Figure 2. Layered cnBR architecture
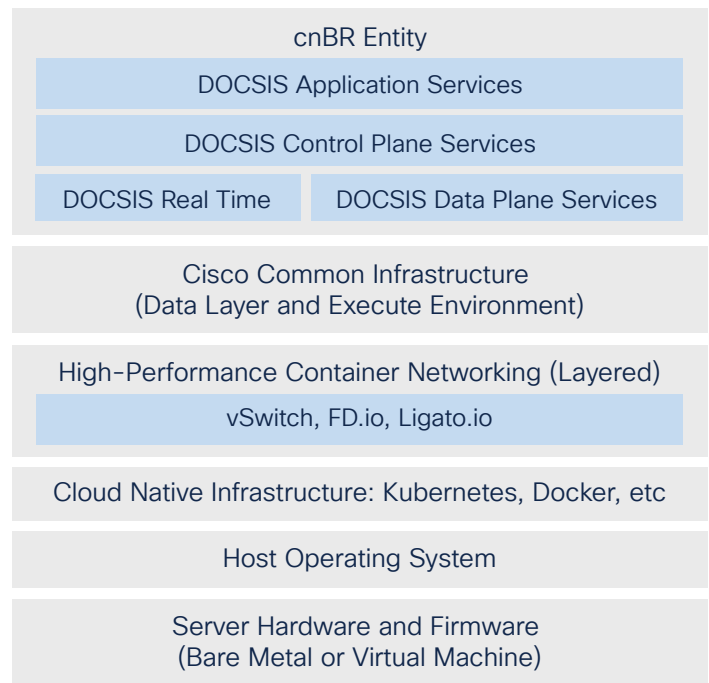
The benefits of microservices that are organized and developed by containers is at the heart of cloud-native implementations such as the cnBR. The benefits include:

- Resilience. Cloud-native applications are inherently resilient because each microservice is independent. If one fails, it doesn't bring down the overall system. It simply fails, reboots, and reattaches. Each service is highly available and can survive infrastructure failures with cloud resource redundancies, which results in reduced risk and increased stability. Compared to existing systems, the constantly on and improving cloud-native environment delivers a new paradigm for high availability.

- Elasticity. Container-based microservices also exhibit elasticity, which is another cloud-native tenet. In other words, container-based microservices can scale up or scale down, manually, or in automated fashion, independent of other services. And because cloud-computing costs are typically based on usage, the ability to dynamically manage scalability in a granular manner enables efficient use of the underlying resources. The cnBR removes the risk of over- and under-provisioning.

- Composability. Cloud-native microservices are composable. Designed to attach to other applications, microservices come with uniform and discoverable APIs, along with well-defined behaviors for registration, discovery, and request management. Their entire makeup lends itself toward automation. Composability is fundamental to DevOps. It allows you to regain control of your network, operating it as you prefer with the services that make most sense.

Cloud-native DevOps: alignment, automation, and continuity

For MSOs that are more attuned to operations than software development, DevOps can appear new and threatening. It may be a new skill set for MSOs to learn, but the concepts are more than a decade old and address a common challenge. Like the rift between IP and RF engineering within MSOs, developers and IT operations traditionally have stood on opposite ends of the IT spectrum. Because developers and operations have different tasks, they speak in distinct dialects. At best they mocked each other, at worst they were pointing fingers and passing blame.

Putting these two groups of people together to adopt DevOps practices changes the game. With DevOps, developers build and commit small batches of code. Then IT operations monitors and offers feedback. Finally, the developers update the code, and a new cycle begins. One of the goals of this closer alignment was to break the common occurrence of large software releases leading to big operational failures. Ideally, more frequent and stable deployments should also result in a healthier workplace with fewer long nights and more normal working hours for the operations team. For workplace improvements to occur, however, operations needs to interact more closely with their digital counterparts.

Instead of a threat, DevOps is an opportunity for cable operators to marry the expertise of the hardware and analog world with that of the digital and automated world. This rethinking requires both understanding how

things were done in the past and finding ways to make them better in the future. Above all, DevOps aims to improve performance for both the pace of deployment and the stability of code, but it also requires more automated and continuous operations (see Figure 3).

To illustrate the character of these operations, let's walk through what a cycle looks like for Feature X. First, our Developers design, build, and test Feature (or Function or Service) X, with or without automated tools. If it passes the test, Feature X is compiled and sent to be installed. IT operations then takes the code and initiates the installation. That process is another one that could either be automated or cascaded automatically through the network once IT operations initiates it.

With Feature X running, the system to which it attached itself to can be actively monitored. If demand increases, it will scale using additional cloud resources. However, if demand decreases, it will scale down and reduce its resource use. As the cycle has begun, developers now get feedback and analytics and can begin to update and revise Feature X. IT operations then takes the revised code, installs it, and allows updates to run automatically throughout the system without disruption. The update is monitored and analyzed, and the developers continue to improve upon Feature X, repeating the cycle.

In a hardware-dominated model, the process is different. After software is launched, updates are disruptive, rare, and even feared. The cable industrytraditionally did not hire developers, so the handoff was more of an over-the-fence lob from the vendor. In contrast, with a

cloud-native DevOps lifecycle, updates are ongoing and ordinary. They can occur as frequently as you like. For a web-scale operator, the velocity of updates could be as high as hundreds per day. The ability to update and then roll back updates dramatically reduces risk because mistakes can be rolled back at once without disruption.

cnBR: Service velocity and innovation

What do DevOps practices look like in the context of the cnBR? As in a web-scale business, cloud-native DevOps draws operations and development teams in an MSO environment closer together. Whereas developers used to work exclusively for the vendor, now they can be integrated as part of the MSO. Or they can be contracted out or held in partnership with the vendor. As the line between Dev and Ops is blurred, power and influence shift back to the operator.

By adopting DevOps practices, operators can regain control of their networks and services. Cloud-native composability, for instance, means someone without coding expertise can specify new workflows and features. In this way, operators can innovate and meet residential and business broadband customers on their own terms.

Without DevOps, operators tend to rely on hardware vendors to develop new features, then spend months testing them. Then, should the software be flawed, the timeline extends, and the process begins again. Often

Figure 3. Cloud-native DevOps lifecycle
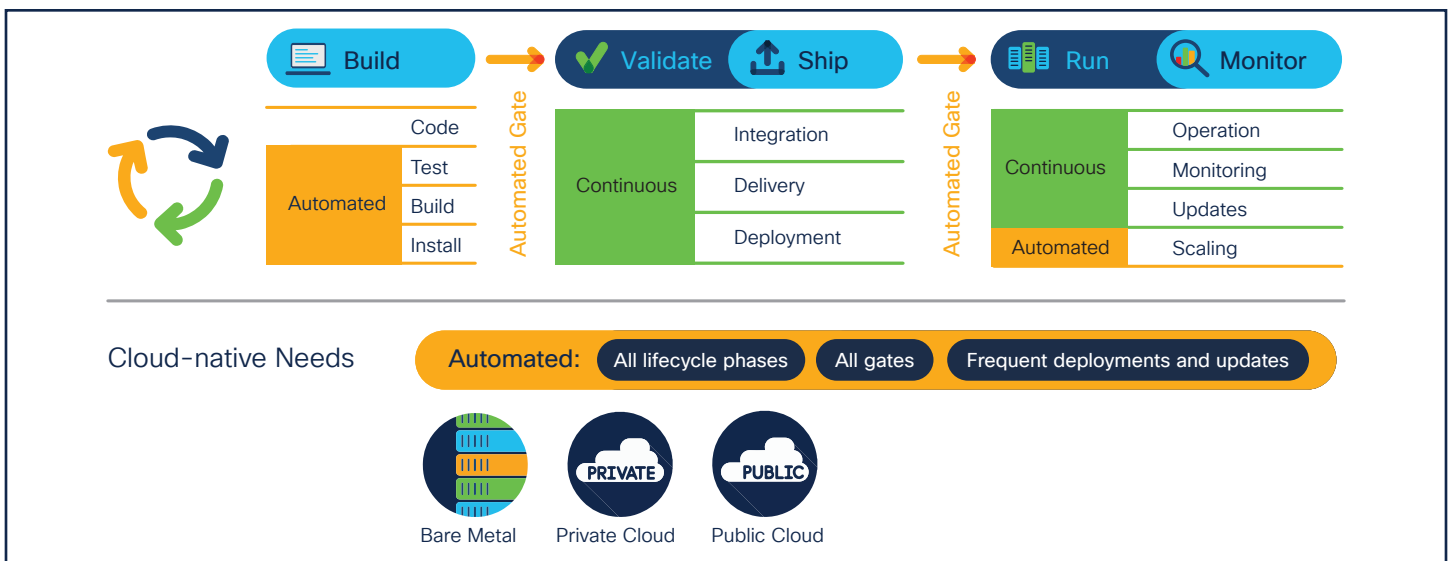
deploying new broadband services becomes a real undertaking. In contrast, the time to market with cloud-native functions reduces to a matter of days, not months or years, even as reliability increases.

Increased service velocity can benefit even those operators who are less inclined to make requests for features or to innovate new functions themselves. Software that will drive the new capabilities within DOCSIS itself or handle tasks such as deploying RPDs within a DAA framework will be part of the continual and increasingly automated update process.

cnBR: Workflow automation

The concept of network automation was around before DevOps and automated software development processes existed. But it has taken years for network operators, not just cable operators, to begin adopting and adapting workflow automation practices that were pioneered by the OTT and web-scale businesses. This lack of adoption wasn't because operators did not want it, but because the technology and underlying network infrastructures weren't ready.

Another challenge faced by broadband networks is how they were built. Broadband networks were often constructed in sprints that were designed to increase capacity and bandwidth. Features and functions were added by layering one on top of another. The race to enable new services and keep existing services running resulted in limited thought as to how systems would need to interact in order to scale. Now we are reviewing the mistakes made in haste. Today, operating networks require scalable processes. To best scale, you need intelligent cross-device communication, which is a prerequisite for automation.

It is possible to automate some things on a cBR-8 today by logging into the CLI and mimicking a user. The downside is that you are limited to a single cBR-8 because replicating the exercise across multiple boxes entails considerable time and risk. With the cnBR, you are not limited by a physical and logical proximity. And with direct APIs and REST interfaces for configuring and performing practically any operation, it only takes a few clicks to specify what you want to automate, how

you want it to happen, and where it should happen. Connecting various tasks in a toolchain then creates a repeatable template.

First, you need to capture the workflow you're trying to automate. Senior technical operations leaders on the MSO side who can detail the steps in any particular process are critical to the DevOps proposition. They are likely to be more familiar with processes than younger and less operationally experienced DevOps team members. By using standard business process modeling notation (BPMN), these technical operations veterans can initiate an automation project by simply documenting current practices.

The operations team members don't need to become fluent in writing scripts and code. An API could have a million lines of code behind it, but all you may need to know is that it says, "create a service." By using a BPMN model to call APIs (as opposed to using it in a view-only mode) members of operations teams can use it to translate well-notated business tasks into automated routines.

Going forward, operators will need to understand, record, and simplify existing processes, no matter what they are or how many domains they cross. In a cloud-native world, workflow automation will be a critical skill for making DevOps the streamlined and innovative methodology that it has the potential to become.

cnBR: Better DOCSIS monitoring

Existing DOCSIS monitoring can be both complicated and skewed toward low-level data. With cloud-native applications, you have an opportunity to simplify and go straight from functionality to data visualization and alerting. Both of these functions are important to network management and maintenance. Although the cnBR supports Simple Network Management Protocol (SNMP) polling, IP Detail Record (IPDR) data collection, and other traditional techniques, it also collects information using more modern and efficient REST APIs. These APIs are adaptable so operators can integrate with them and gain greater network insights.

At the highest level, the cnBR offers a new central point of information. It provides a birds-eye view of system

Learn more

To learn more about the Cisco Cloud Native Broadband Router, visit this page:

www.cisco.com/go/cnbr

health, potentially across the entire access plant. The cnBR's REST APIs also provide engineers and technicians with data-driven dashboards. These dashboards are attractive alternatives to using CLI for detailed drill-downs, troubleshooting, and other technical operations tasks (see Figure 4).

Figure 4. Shifting CMTS operations to a web model

Today	cnBR
CMTS is a **central point** for a **"small" segment** of cable access plant	**Operations Hub** is now the **central point** potentially for **entire** cable access plant (or at least a region)
DOCSIS Engineering interface directly with the CMTS (CLI, SNMP)	**DOCSIS Engineering** needs to change how they gather data - Rest APIs, Dashboard
Capacity Planning leverage IPDR data generated by the CMTS to grow services	**Capacity Planning** does not change, IPDR is supported
CSRs rely on tools written in-house that leverage SNMP (or worse CLI scraping)	**Specific APIs** have been built to support **CSR tools**
Field Technicians rely on weekly reports from CLI to trend HFC issues and trends	**Field Technicians** can leverage dashboard for reports, initially some APIs have been developed

The liberating potential of DevOps

As MSOs consider their next moves for evolving their infrastructure and processes to scale, it's important to remember how much OTT offerings and web-scale businesses have changed the standards for service delivery, agility, and stability. As at previous inflection points, the industry has thrived by importing and adopting new technologies. Cloud-native functions and related DevOps practices, such as those enabled by the Cisco cnBR, are one such adoption. Through continuous and automated processes, cloud-native DevOps can boost performance, provide operations teams with improved monitoring and workflow automation capabilities, and enable MSOs to regain control and innovate their own networks.

1. Cisco Visual Networking Index Forecast and Trends, 2017–2022
   https://www.cisco.com/c/en/us/solutions/service-provider/visual-networking-index-vni/index.html