From MANO to CNA: a new era of declarative and model-driven automation for cloud-native networks

January 2024

Gorkem Yigit

ETSI NFV MANO[1] has shaped network automation over the past decade. The first version of network cloudification, which was based on monolithic virtual network functions (VNF) running on OpenStack IaaS, needed an automation framework like MANO. Early telecoms cloud environments were designed to run bulky virtual machines in large, centralised data centres. While OpenStack provided the cloud resource orchestration capabilities for these data centres, it was not built for end-to-end lifecycle management of an entire network cloud. Consequently, MANO was created for the operationalisation of the network cloud by defining new components such as VNF managers (VNFM) and NFV orchestrators (NFVO). However, these external orchestration and automation mechanisms are imperative, script-based and superimposed on IaaS clouds rather than being an inherent part of them. The resulting architecture is complex, expensive to implement and has delivered limited automation and operational efficiency benefits to communications service providers (CSPs).

In Analysys Mason's new report, *Cloud-native automation framework: redefining network automation for the cloud-native era*, we argue that CSPs need to consider an alternative trajectory by building a new cloud-native automation approach from the ground up.

With the introduction of cloud-native technologies such as decomposed microservices architecture and containerisation, Kubernetes-based container-as-a-service (CaaS) is becoming the primary cloud environment, superseding OpenStack. Originally, Kubernetes was adopted for container orchestration and aligned with MANO-based automation systems. However, Kubernetes is a comprehensive, intent-based automation platform with inherent autonomous capabilities such as self-healing and self-scaling without requiring layers of complex, external systems like OpenStack does. Using its native mechanisms such as the Kubernetes Resource Model (KRM), Custom Resource Definitions (CRD) and Kubernetes operators, Kubernetes can be extended beyond container orchestration to automate any construct of a cloud-native network, including network functions (xNFs), CaaS components and cloud infrastructure. It enables a fundamentally different automation approach that is declarative, model-driven and an intrinsic part of the network cloud. We refer to this new automation model as cloud-native automation (CNA), which means that the automation platform should be of the same fabric as the cloud-native and disaggregated networks they are designed to automate.

CNA is a converged, decentralised automation architecture that is built using open-source components and GitOps

The traditional mode of network cloud operations and automation paradigm has multiple challenges, which include the following.

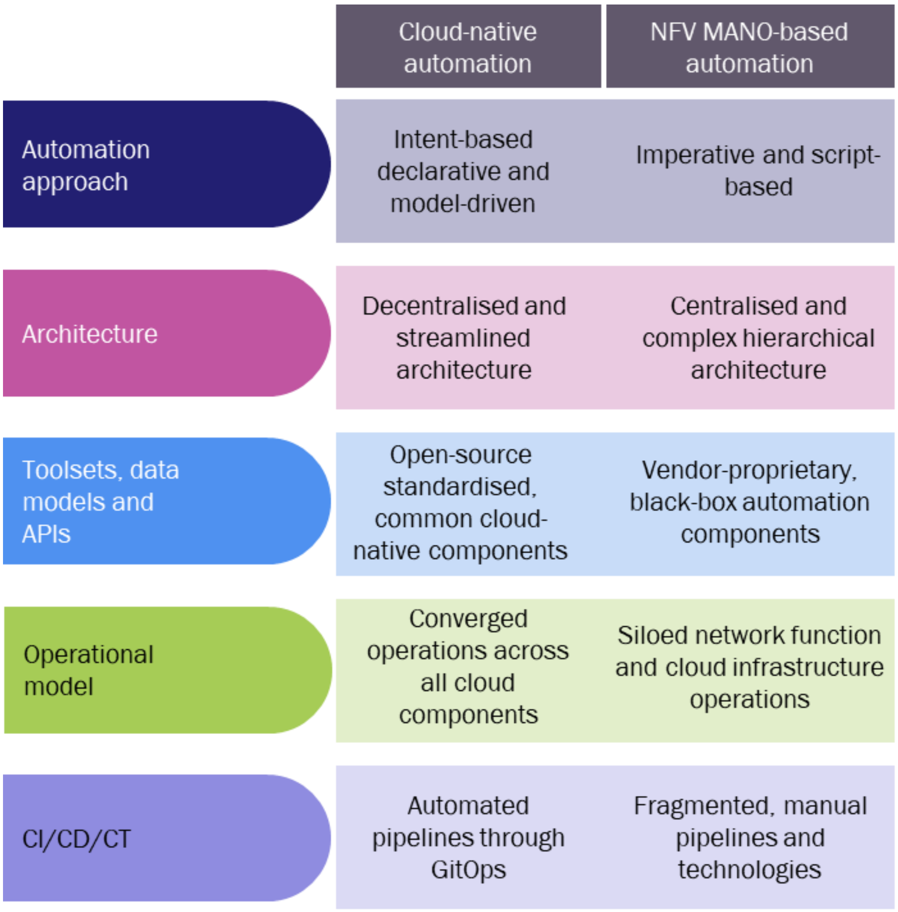
- Cloud infrastructure and xNF lifecycle management are executed in isolation, which conflicts with cloud-native networks' need for integrated communication and collaboration.

[1] European Telecommunications Standards Institute network functions virtualisation management and orchestration (ETSI NFV MANO).

- Vendors providing OSS/orchestration, network function and cloud infrastructure are pushing automation solutions that rely on proprietary tools, components and data models. These 'black box' solutions lead to vendor/domain-specific snowflake automations and limit CSPs' control over automation and lifecycle processes.
- Retrofitting Kubernetes-based automation onto traditional systems like NFVO and VNFM leads to fundamental conflicts and inefficient use of the cloud capabilities due their differing operational paradigms.

Several advanced CSPs and pioneering vendors are actively working towards a CNA model that addresses these challenges. Our recent report outlines a framework that embodies the main principles and requirements of CNA which reflects the shared vision of these CSPs and vendors. In this CNA framework, automation comes from the network cloud as an embedded feature and any complementary or adjacent automation systems should be aligned with it.

Figure 1: Comparison of CNA with traditional automation



	Cloud-native automation	NFV MANO-based automation
Automation approach	Intent-based declarative and model-driven	Imperative and script-based
Architecture	Decentralised and streamlined architecture	Centralised and complex hierarchical architecture
Toolsets, data models and APIs	Open-source standardised, common cloud-native components	Vendor-proprietary, black-box automation components
Operational model	Converged operations across all cloud components	Siloed network function and cloud infrastructure operations
CI/CD/CT	Automated pipelines through GitOps	Fragmented, manual pipelines and technologies

Source: Analysys Mason

At the heart of the CNA framework is a converged operational model offering unified management for all network cloud resources, including xNFs, CaaS and cloud infrastructure. CNA's use of CRDs and the Configuration as Data (CaD) approach enables modelling and integration of cloud resources within Kubernetes for lifecycle management using open-source, standardised components for Day 0, 1 and 2 operations. Nephio is pioneering these capabilities and Project Sylva is also adopting similar principles. Even if Nephio does not gain expected traction, industry efforts are likely to continue evolving this model in various forms, or leading CNA vendors will establish their own methods as standards.

Kubernetes is built on a declarative model, where the desired state of the system is defined in configuration files, which lays the foundation for intent-based automation. The use of a GitOps approach fits neatly into this declarative nature of Kubernetes, enhancing its capabilities by providing a single, immutable source of truth, storing intent/desired state and artifacts that are defined in a declarative way. A CNA-based solution can pull the desired state from Git repositories, which become the new inventory system for cloud-native networks, and reconcile the current state autonomously to reach and maintain the desired state using Kubernetes' operators/custom controllers.

Our CNA framework also foresees an evolved, lightweight NFVO as many automation and closed-loop processes will be decentralised and performed near-autonomously by a Kubernetes-based automation layer. It will act as a higher-order control plane system that actively understands a network domain holistically with all the components functioning as a network using observability and testing data. This will enable it to generate modifications and optimisations in the intents as required, with the help of artificial intelligence/machine learning (AI/ML), based on a whole-network domain context, external inputs (for example, new service/slice) as well as the monitoring and self-regulating role.

Cloud-native automation will redefine the network cloud automation landscape and influence CSP 6G strategies

CNA will influence operators' automation strategies and reshape the market over the next 10 years. It will challenge traditional solutions and could lead to established vendors being replaced by emerging alternatives and in-house developments.

AWS, Google and Microsoft are pioneering the new cloud-native automation paradigm but they are following distinct approaches, each leveraging their unique strengths and ecosystems. Network function vendors such as Ericsson, Huawei and Nokia have the most to gain or lose from CNA as they dominate the network cloud automation market. Nokia leads the field of xNF vendors in the development of CNA. It has a solution that is currently in trials with operators and deep involvement in Nephio. CaaS vendors such as Red Hat, VMware and Wind River as well as OSS/orchestration vendors such as Amdocs, Netcracker and Oracle will also be affected by the CNA; the extent of it depending on their strategy to move up or down the automation stack. New entrants such as Aarna Networks are also emerging to take advantage of the shift to CNA.

We expect that the winners will be determined by several key factors. First, their mastery of cloud-native networking and operations, using open, cloud-native constructs, is crucial as the CNA is part of the network cloud, not an external OSS/orchestration system. Second, the propensity for innovative risk-taking and the pace of adopting this radical approach is vital, despite its potential to disrupt and cannibalise vendors' existing solutions. Finally, providing flexible migration paths from traditional automation is essential. This involves offering versatile solutions with a modular, reusable set of components, designed to accommodate each operator's starting point and adoption strategy.