

Harnessing Network Infrastructure as Code with Alkira & Terraform

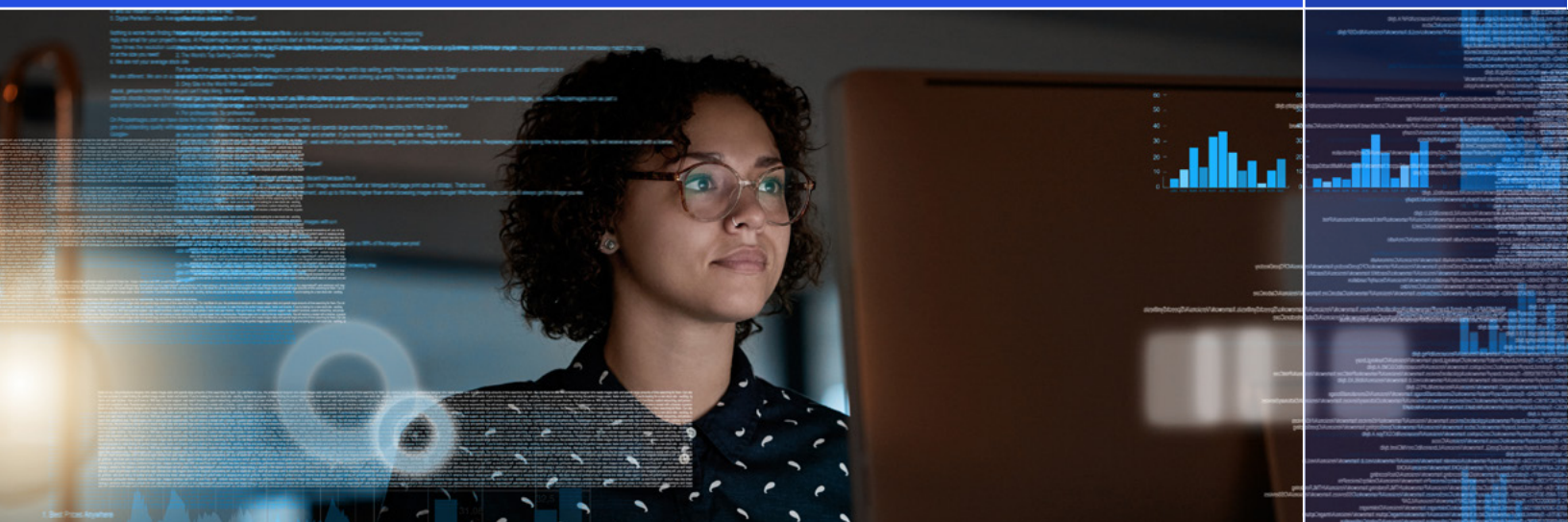


Table of Contents:

Automation is Foundational	03
The State Of Network Automation	03
How has Cloud Transformed Automation?	04
Driving the Alkira Platform with Terraform	05

Glossary of Terms

CXP Cloud Exchange Point: is a fully virtualized point of presence delivering an entire network stack with rich network services

Connector: is a termination point connecting on-prem and cloud networks

CSP Cloud Service Provider: provides a wide range of cloud computing services

MCN Multi-cloud Networking: utilizing multiple CSPs to build, deploy, and manage applications and workloads



Automation is Foundational

Enterprises deploy and manage network infrastructure in many ways using various approaches and tooling. With the rate of change increasing exponentially, network engineers are faced with an increasing number of choices for which tools to use and how to maintain them. Recent years have seen a massive increase in public cloud consumption from enterprises. One of many reasons for this is the ability to sidestep an accumulation of legacy technology, design, culture, and thinking. Effectively automating infrastructure is no longer a luxury but a staple in the enterprise move through future transformation. Despite this, network automation is still lagging behind other technical disciplines.

The [NetDevOps Survey](#) reveals that participants use an average of 5.31 (with a maximum of 24) unique vendor devices/platforms. Additionally, up to 6 different solutions are being used across participants for generating and/or deploying configuration. With those solutions, as many as 6 unique methods are used to interact with devices and platforms.

By adding additional vendor devices, platforms, orchestration tools, and unique protocols and transports over which communication occurs, the opportunity for consistent, repeatable, and reliable automation steadily decreases.

The State Of Network Automation

Before outlining the benefits of using Hashicorp Terraform with the Alkira Platform, it is vital to understand the current state of network automation. As the complexity and challenges become more evident, it becomes easier to understand the practical use-cases of treating the whole domain of networking as Infrastructure-as-Code.

Numerous surveys exist in the realm of DevOps, one such survey being The State of DevOps. A community-driven survey called The NetDevOps Survey aims to take similar criteria and feedback but applied to how network operators and engineers use automation to run their networks.

Causal Explanation

Complex dependencies exist on-premises between software layers, physical hardware, and security components. To respond to market demand and release features quicker and more reliably, end-to-end automation is mandatory.

What are the underlying attributes that enable this for compute and application delivery in the Cloud?



The following attributes provide insight as to why.

On-Premises Challenges:

- **Tight Coupling:** Components have a heavy dependence on each other, causing cascading impact. The blast radius is large in outage events, causing a decrease in overall changes leading to less frequent releases.
- **Mutable Infrastructure:** Physical or virtual network devices are configured, updated, or modified in place at runtime. This causes significant configuration drift over time, leading to environment inconsistency.
- **Vertical Scaling (Scale-Up):** Legacy network infrastructure scales by increasing the capacity of individual nodes on the network. Scaling-up focuses on maximizing the power of individual devices like a single set of firewalls or routers.
- **Inconsistent Interaction Surfaces:** Enterprise networks consist of numerous vendors, hardware models, and software versions. CLI and API interaction surfaces vary in functionality and interoperability, leading to a combination of manual intervention, scripts, and disjointed tooling.

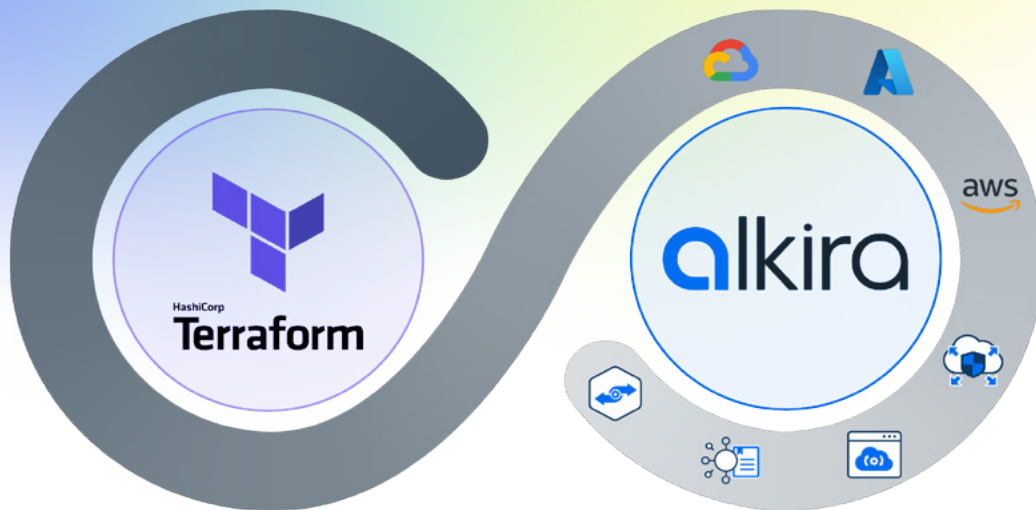


How has Cloud Transformed Automation?

Cloud has driven a shift to thinking in terms of fully software-defined, automated, and abstracted infrastructure:

- **Loose Coupling:** Components are detached, enabling them to work independently of each other as part of a larger group of systems. The blast radius is reduced significantly, causing an increase of changes which leads to more frequent releases.
- **Immutable Infrastructure:** Cloud infrastructure is configured at build time. This reduces the number of moving pieces at run time, increasing environment consistency, reliability, and grounded DR capabilities.
- **Horizontal Scaling (Scale-Out):** In the cloud, overall capacity is increased by adding additional nodes, usually of equivalent capacity. Scaling-out shifts focus to combining the resources of many nodes together.
- **Consistent Interaction Surfaces:** Cloud infrastructure is provisioned by a console, proprietary provisioning tool, or cloud-agnostic tool. In AWS, Terraform would use a provider plugin that leverages the AWS GO SDK to build infrastructure. This provides the same interaction experience across all AWS infrastructure products.





Driving the Alkira Platform with Terraform?

Why Terraform?

Managing cloud infrastructure with Terraform by Hashicorp has steadily gained popularity as more organizations move to the cloud. As multi-cloud becomes more prominent, securing, governing, and auditing infrastructure is critical. Having a single workflow to manage heterogeneous infrastructure enhances and simplifies these capabilities. In addition, treating all aspects of a given system as code provides an excellent opportunity for standardization and repeatability.

The Alkira Platform

The Alkira Platform, like the public cloud providers, presents a consistent interaction surface for the entire network. Segmentation and policy work consistently across all networking, creating a single architecture across all clouds and traditional networks. For the first time, core, edge, and cloud networking can be provisioned with a common workflow. Adding in version control, coupled with the immutable nature of Terraform, and your whole network from end-to-end becomes manageable, just like the public cloud.

Organizations that mature or seek additional maturity surrounding automation capabilities in the cloud aim to leverage Continuous Integration / Continuous Delivery (CI/CD) to deploy cloud infrastructure. While Alkira’s platform already reduces build time significantly, leveraging Terraform to drive Alkira’s platform can be added to complement existing workflows and pipelines that already exist.

The following benefits can be achieved:

- Complete codification of core, edge and cloud networking deployed and maintained with cloud-grade infrastructure-as-code.
- One interaction surface for the whole domain of networking creating a natural evolution to immutable infrastructure for the network.
- Entire network infrastructure hosting an application or supporting a business imperative is delivered intact between remote sites, public cloud, data center, network services, and SaaS/Internet connectivity.
- Elastic network infrastructure deployed at scale, driven by modern CI/CD platforms, with the same velocity as public cloud.





Summary

Alkira's platform provides robust solutions for the inherent challenges of IoT deployments, ensuring scalable, secure, and efficient multi-cloud IoT infrastructure. This white paper emphasized Alkira's role in enhancing IoT deployments through segmentation, policy consistency, and multi-cloud connectivity.

