Automation and Orchestration of Multiple Platforms to Offer a B2B Self-Service Cloud Platform

A technical paper prepared for presentation at SCTE TechExpo24

David Camilo Olea
Sr. Manager B2B Technology Design
Liberty Latin America
David.olea@lla.com

Table of Contents

List of Figures

1. Introduction

In a multiplatform telco environment, to offer a private cloud service requires many different platforms orchestrated and automated to provide a single interface for the B2B customers in the most cost effectively way. This paper describes how using automation tools such as Ansible in conjunction with other open-source platforms a whole solution could be developed and be part of the DevOps telecommunications environment to ensure that the continuous customer needs could be met.

The end-to-end architecture was developed including several tools: GIT repository, Ansible engine, back end databases and ManageIQ as customized front for B2B customers. This environment allows us to offer a self-service platform including security and interfaces to billing.

The new environment is fully DevOps oriented. This means that the definitions meet the IaC (Infrastructure as Code) requirements and the architecture is future-proof granting growth functionalities, adjusting to the new version and quickly features adding.

Most important benefits from this new architecture are Support Opex reductions (about 70%); improving the delivery time (from 2 days to 1 hour); new features available like VPN, metering by customer use; opportunities to execute new customizations according to the market needs.

The other important topic with automation is the cultural change in the organization. The success of any kind of project is to have engineers specialized in each field and with the disposition to work coding. This is the unique way to take a real advantage of the tools and it is a huge cultural change for many of our network, security and cloud experts to become in a new automation era engineer. It is possible in a short period of time.

2. The Challenge: Unifying Infrastructure for Agility and Efficiency

The demand for flexible computing capabilities and rapid service deployment has pushed telecommunication companies to implement new platforms to interact directly with their customers. This initiative aims to unify their data centers and virtual networks into a single solution while reducing operational costs. By integrating the different platforms and orchestrating them into a single pane of glass for the customers, telco companies can provide a more cohesive and streamlined experience.

Modern businesses require rapid and efficient resource utilization. In Latin America, most telecommunication providers face the challenge of supporting self-managed infrastructure services with stability, scalability, and flexibility. These providers must accommodate new features, integrate with third-party solutions, and enhance user experience. Additionally, gaining knowledge, reducing costs, and optimizing activities through automation are crucial for managing their virtual data centers and associated virtual networks via a unified layer.

Network automation plays a pivotal role in addressing these challenges. By automating routine tasks, telco companies can significantly reduce operational expenses (OPEX) and improve efficiency. Automation tools and platforms can help streamline processes such as network provisioning, configuration management, and fault detection. This reduces the Mean Time to Install (MTTI) and enables faster deployment of new services and features.

At the same time, there is pressure to increase the operational efficiency of networks and services. This involves reducing the MTTI and offering a wider range of features, options, and flavors to meet business needs in a continuous and agile manner. To achieve this, companies must leverage their existing infrastructure and legacy systems, which have evolved through acquisitions and company growth.

3. Background and Concepts

3.1. Key Points to Design the Automation Architecture

To address the various challenges in a private cloud environment where a fully self-managed solution by the client is desired, an advanced automation architecture was developed with the following key features:

i. **Complete Reuse and Integration of Existing Platform:**
   - The architecture ensures seamless integration with the current computing infrastructure. This means leveraging the existing hardware, software, and network resources without requiring significant overhauls or replacements.
   - By utilizing the existing platform, the solution maximizes the return on investment (ROI) and minimizes the disruption to ongoing operations.

ii. **Integration with Internal Organizational Systems:**
   - The automation architecture is designed to coordinate with various internal business and support systems. This includes CRM, BSS, and other enterprise applications essential for daily operations.
   - Such integration ensures that business processes remain uninterrupted and that data flows smoothly between different departments, enhancing overall organizational efficiency.

iii. **Low Cost (Open-Source):**
   - The solution leverages open-source technologies, which significantly reduce costs while providing robust and reliable performance. Open-source tools are often developed and maintained by a large community of developers, ensuring continuous improvements and updates.
   - By adopting open-source solutions, the organization can avoid expensive licensing fees and vendor lock-in, providing greater flexibility and control over the technology stack.

iv. **Future-Proof:**
   - The architecture is designed to be future proof, allowing for the integration of new solutions and network expansion as needed. This means it can easily adapt to emerging technologies and evolving business requirements.
   - Scalability is a core aspect, enabling the system to grow with the organization and accommodate increasing workloads without compromising performance or reliability.

v. **Cultural Transformation:**
   - One of the critical aspects of the architecture is its focus on facilitating the transition of network engineers to automation engineers. This involves providing the necessary training and resources to help engineers develop new skills in automation.
   - The architecture also promotes a cultural shift within the organization, encouraging a mindset that embraces innovation, continuous improvement, and collaboration. By fostering this cultural transformation, the organization can better adapt to the rapidly changing technological landscape.

3.2. Key Technology Points

OpenStack Platform is an open-source cloud computing platform designed to manage and control large pools of compute, storage, and networking resources within a data center. It offers infrastructure-as-a-service (IaaS) capabilities, allowing users to deploy and manage cloud environments through a web-based dashboard, command-line tools, or a RESTful API.

Key components of OpenStack include Nova (compute), Swift (object storage), Cinder (block storage), Neutron (networking), Keystone (identity services), Glance (image management), Horizon (dashboard), Heat (orchestration), and Ceilometer (telemetry).

OpenStack is highly scalable, flexible, and customizable, making it suitable for private, public, and hybrid cloud environments. It is supported by a global community of developers, ensuring continuous updates and innovation. OpenStack is widely used by enterprises, telecommunications companies, and service providers to build and manage cloud infrastructures cost-effectively while avoiding vendor lock-in.

ManageIQ is an open-source management platform designed for hybrid IT environment. This platform enables organizations to manage private, public, and hybrid clouds. It provides capabilities such as self-service provisioning, resource management, and policy enforcement, helping organizations gain visibility and control over their cloud environments. In the architecture acts as a self-service portal, allowing clients to manage their virtual machines and networks autonomously.

Ansible Automation Platform: is a powerful automation tool that allows for the automation of IT tasks such as configuration management, application deployment, and task automation. Ansible uses a simple, agentless architecture, making it easy to manage complex environments. Configurations are defined in human-readable YAML files called playbooks. Ansible ensures consistent changes with idempotency and includes a wide range of built-in modules. By automating repetitive tasks, Ansible helps to improve efficiency and reduce the risk of human error.

4. Implementation: A Unified Approach

4.1. Proposed Architecture

For network functions within Neutron, external network and security elements (such as switches and physical firewalls) were managed using Ansible playbooks. This approach offers flexible task creation, including additional features like VPN services, seamlessly integrated into ManageIQ's open-source module and made available via the portal.

All Ansible code is stored in a GIT repository, ensuring proper governance, easy management, and updates when deploying updates to physical devices or adding new service features.

This platform management architecture was concurrently used to replace the private cloud hypervisor. The Figure 1 shows four functional blocks:

i. The user layer includes the portal and database with information replicas, essential for managing client virtual inventory.

ii. The second block encompasses existing infrastructure, either migrated or coexisting scenarios with the new hypervisor. Adjacent to this block is the new hypervisor and automation platform connected to a GIT repository storing all generated code. This block integrates private cloud computing capabilities and interfaces with external platforms such as firewalls or backup elements. It also facilitates integration with billing systems (e.g., usage-based billing), adjusting consumption data generated by ManageIQ.

iii. The control plane oversees each of the other blocks, crucially separated to ensure proper operation and functional isolation across the solution.

This architecture emphasizes automation and the creation of a self-service interface for clients based on open-source software, reducing costs while enabling extensive customization and reusing existing licensed or unlicensed systems and platforms.

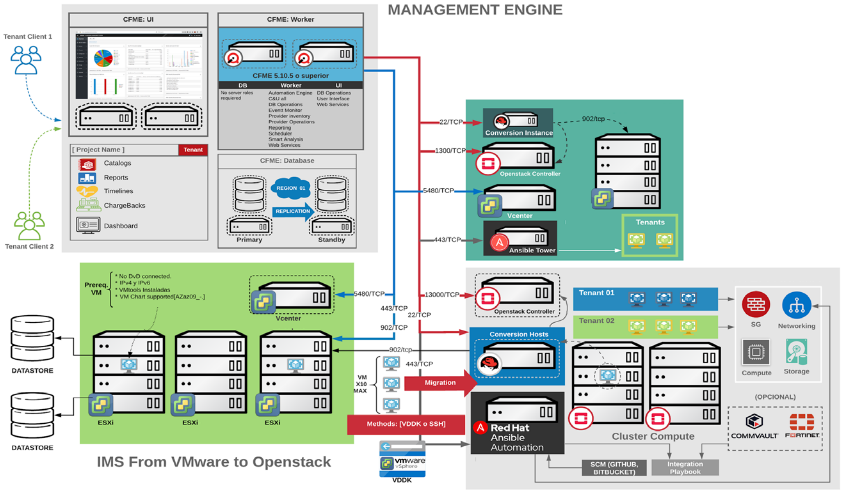

Figure 1 - proposed architecture

4.2. Journey to Orchestrate the Cloud Platform

A cloud platform encompasses much more than just the hypervisor; it includes numerous additional components to deliver a complete service. Typically, these platforms consist of the following key elements:

- **Physical infrastructure layer**: The foundational hardware resources.
- **Hypervisor**: Manages the computing resources.
- **Storage**: Integrated within the hypervisor in HCI architectures.
- **Internal network (LAN)**: Often integrated with the hypervisor, which is recommended.
- **External network**: Connects to the data center.
- **Security management**: Includes external firewalls.
- **Additional services**: VPN, load balancing, WAF (Web Application Firewall), backups, and object/file storage services.

Moreover, the platform must ensure seamless integration with BSS (Business Support Systems) and OSS (Operational Support Systems) while providing the flexibility to generate detailed usage information and reports.

To ensure the correct orchestration of systems, the following steps were followed:

i. **Modernize the hypervisor:** This fundamental step enables the creation of new services, facilitates the integration of external platforms, and keeps systems updated. It ensures the availability of open interfaces (REST-API). In this case, OpenStack was used.

ii. **Verify that additional systems have management interfaces:** Typically, these are REST APIs or, alternatively, command-line interfaces. This includes interactions with systems such as Cisco, Fortinet, and Commvault.

iii. **Deploy the automation platform:** This includes Ansible as the automation engine, GIT as a code repository, and connections between a wide range of elements.

iv. **Deploy the user layer:** This layer allows customer access and interaction with the infrastructure. The portal should be highly integrated with automation and flexible enough to meet business needs. In this case, ManageIQ was used.

v. **Ensure network:** Secure all connections, ports, and flows to allow proper communication between elements.

vi. **Adhere to company security policies:** Adjust the infrastructure and platforms at the logical level to ensure compliance with regulations and security pillars.

vii. **Clearly define processes, tasks, and involved areas for each orchestrated service:** This is essential for ensuring that the automation initiative is shared within the organization.

viii. **Design, execute, test, and adjust playbooks:** Perform these activities according to the service requirements.

5. Implementation Results

5.1. Significant Cost Savings and Product Improvements

Using the above architecture in a telecommunications service provider, the following results can be achieved:

60% Reduction in Operational time: Automation and unified management significantly reduce the need for extensive manual intervention. This reduction brings correlated benefits that should be analyzed:

i. **Increased Capacity without OPEX Increase**: With less time required for operations, the same team can support more customers and platforms without any increase in operational expenditures (OPEX).

ii. **Process Improvement and Innovation**: The team can invest the saved time in improving support processes and creating new use cases, which is crucial for evolving the services offered by the company.

iii. **Improved Employee Management and Documentation**: From the company's perspective, handling staff rotation becomes easier. Additionally, technical documentation for each change is clearly described and tracked in the repository, ensuring consistency and accountability.

50% Reduction in MTTI for the new customers: Automation has streamlined processes, allowing for faster project execution. Additionally, this automation can be orchestrated by CRM or BSS systems to provide foundational services ready for use.

Improved Deployment and Integration: Enhanced capabilities allow for offering on-demand services through seamless integration with other platforms. This speeds up the time to market, enhances product robustness and usability, and eliminates operational mistakes. Ensuring that all deployment steps are followed correctly avoids friction between different teams responsible for each network segment, achieving the best Operational Level Agreements (OLAs).

5.2. Developing an Automation/DevOps Culture

DevOps is a set of practices that integrates software development (Dev) and IT operations (Ops). The primary goal is to shorten the development lifecycle and achieve continuous delivery of high-quality software. Embracing a DevOps culture enables telecommunications companies to foster collaboration between development and operations teams, automate workflows, and ensure swift and dependable software deployment. This cultural shift is pivotal for harnessing automation's full potential and attaining enhanced operational efficiency.

As a result of this automation, an interesting transformation occurred within the team leading the project and among the surrounding teams. Network engineers and cloud solutions specialists began focusing on understanding the end-to-end aspects of each service and seeing a broader landscape of possibilities for new services and products. They also emphasized continuous maintenance and improvement of the platforms.

Despite having only basic knowledge of programming or coding, they had two clear principles: a service-oriented approach and a deep understanding of the platform's workings, tasks to be executed, and the ease of developing code (playbooks) for everyday situations using a common framework.

6. Conclusion

The implementation of the platform has transformed the company’s infrastructure management. This unified solution has led to significant cost savings, enhanced efficiency, and improved service delivery, enabling the company to meet evolving client demands and maintain competitiveness in the telecommunications sector.

This path demonstrates that it's possible to construct a functional architecture at low cost that allows for the deployment and management of cloud solutions. Open-source options are evolving technologically faster, maintaining their philosophy of interoperability.

In telecommunications companies, there is significant potential to transition many network engineers or solution experts into automation engineers. This shift aims to create new features while evolving functionalities such as self-diagnosis, self-repair, and zero-touch provisioning.

This architecture offers flexibility for integrating with Business Support Systems (BSS) and can adapt to organizational standards. Automation is not aimed at reducing the number of engineers but rather at leveraging their experience, process knowledge, and time availability. This approach simplifies the path to code generation.

To achieve this, orchestration is essential to free up engineers' time involved in the solution. This allows for more time to observe the solution's behavior and trends, thereby maintaining the development cycle effectively.

Abbreviations

API	Application Programming Interface
B2B	Business-to-Business
BSS	Business Support Systems
DevOps	Development and Operations
HCI	Hyper-Converged Infrastructure
IaC	Infrastructure as Code
IT	Information Technology
MTTI	Mean Time to Install
Opex	Operational Expenditure
OSS	Operational Support Systems
REST-API	Representational State Transfer Application Programming Interface
VPN	Virtual Private Network
WAF	Web Application Firewall

Bibliography & References

https://www.ansible.com/

https://www.manageiq.org/

https://www.openstack.org/

https://devops.com/

https://openpracticelibrary.com/

https://www.redhat.com/es/topics/devops/what-is-sre

Ansible for DevOps: Server and Configuration Management for Humans by Jeff Geerling