Enabling 5G Network Automation Over AWS with RIFT

Cloud-native Service Orchestrator on AWS

January 2021

Notices

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents current AWS product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from AWS and its affiliates, suppliers or licensors. AWS products or services are provided “as is” without warranties, representations, or conditions of any kind, whether express or implied. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

Contents

Abstract

5G is transforming the connectivity landscape, allowing lower latency and higher bandwidth across a larger scale of devices. To make this possible, 5G embraced the principles of function decomposition and microservice architecture, which results in another challenge: orchestration across multiple functions and services. Operating and managing 5G mobile network functions on Amazon Web Services (AWS) with RIFT's service orchestrator allows implementation of closed loop automation of 5G networks. This whitepaper highlights the best practices for designing RIFT's service orchestrator for managing end-to-end 5G networks on AWS.

This whitepaper is aimed at members of a network operations team comprised of communications service providers (CSPs).

Introduction

The next generation of mobile services promises support for all uses, from smart appliances to interactive, high-resolution gaming. The ultimate vision of 5G not only provides frictionless delivery of traffic in terms of bandwidth, latency, and scalability, but also frictionless delivery of the service itself, from the moment of customer request to the availability of the service.

This combined vision of instant-on, bespoke mobile networks at scale can be achieved only through a combination of technologies that provide:

- Top-to-bottom automation of service fulfilment, from the service provider's customer portal to the infrastructure layers.
- End-to-end automation that works across multiple technology domains and locations, from Radio Access Network (RAN), Transport, and Core, to private, public, and edge clouds.
- Automated management of the 5G service, to maintain high performance and availability, and to ensure the customer's Service Level Agreements (SLAs) are met.

Yet challenges remain in the realization of this vision. While it is possible to build catalogues of applications and network functions and launch them in the cloud, many network functions today are still virtual machine (VM)-based, and not built with cloud-ready techniques in mind. The service-based nature of 5G slices and the evolutionary characteristic of the 5G architecture requires service providers to design and build networks of applications and network functions that span VM-based and container-based clouds in various locations, to meet the customer's service, coverage, and performance needs. AWS and RIFT can overcome these challenges and help to realize the vision.

AWS provides various types of cloud services and application programming interfaces (APIs) which enable the design of cloud-native network functions and microservices applications. AWS can provide single pane of glass unified management tools, including DevOps and CI/CD pipeline, to effectively operate these network functions and applications. However, it is a common requirement in the telecommunications (telecom) industry to have an orchestrator/manager application that meets industry standards. AWS provides the building blocks for programmable operation and automation pipelines.

RIFT is a company that bridges the IT and network domains through the development of RIFT.ware™, a next-generation, model-driven, standards-compliant network functions virtualization (NFV) automation and orchestration product with carrier-grade capabilities. RIFT.ware provides an environment to automate the onboarding and life cycle management of multi-vendor 5G network slices across any cloud environment and technology domain.

RIFT.ware also streamlines the operation of these slices through Closed Loop Day 2 operations, which include zero-touch, end-to-end network service management such as auto scaling or self-healing. (*Day 2 operations* usually means ongoing configuration change and update after *Day 0 deployment* and *Day 1 configuration*.)

Together, AWS and RIFT.ware form an ideal infrastructure for end-to-end automated management and operation of 5G slices. The RIFT.ware solution is a multi-standards automation solution that enables end-to-end service orchestration across hybrid clouds.

RIFT.ware's standards-based APIs and models are built according to TM Forum, Internet Engineering Task Force (IETF), European Telecommunication Standards Institute (ETSI), and 3rd Generation Partnership Project (3GPP) specifications to enable integration with operations support system/business support system (OSS/BSS) for deployment of 5G, Software-Defined Networking in a Wide Area Network (SD-WAN), and multi-access edge computing (MEC) across cloud types and locations. This is to create end-to-end services by chaining cloud deployments across the WAN, and to enable standards-based 5G slice and slice SLA management.

As a multi-standard, compliant, open-source product, RIFT.ware is designed to eliminate vendor lock-in through multi-domain orchestration of any use case across any cloud at any site. RIFT.ware is designed to reduce operating expenses (OPEX) by providing a single pane of glass into the 5G slice operating environment. Designed as a product and not a loose toolset or framework, RIFT.ware also increases service velocity through the incorporation of standards-compliant templates and built-in integrations to the service provider OSS/BSS and NFV ecosystem.

In the following sections, we examine how AWS Services and tools, coupled with RIFT's RIFT.ware Orchestration and Automation suite, can automate a service provider's 5G offerings, and achieve the goals of 5G slice deployments.

AWS Services and RIFT.ware for Management and Orchestration

As described in a previous whitepaper called *5G Network Evolution with AWS*, one of the most common challenges of CSPs in the era of NFV and 5G is Management and Orchestration (MANO) defined by ETSI Industry Specification Group (ISG) NFV across all networks and infrastructure resources, as well as end-to-end service configuration. In the case of the 5G mobile network, network slicing is leveraged to create a dedicated network for enterprise and specific service use cases. This means that orchestration will have to solve another layer of complexity. In the referenced paper, three different options for implementing the orchestration layer for 5G networks have been explored:

- AWS native management and orchestration
- Extended orchestration using AWS tools
- Application-based orchestration by API interworking

The third approach provides the MANO layer on AWS by deploying RIFT.ware as Service Orchestrator, NFV Orchestrator (NFVO), and Generic Virtual Network Function Manager (VNFM) into the AWS platform, and integrating with AWS as a Virtualized Infrastructure Manager (VIM) through the native AWS APIs. It is also beneficial to leverage AWS Services natively, and develop using container services, such as Amazon Elastic Container Services (Amazon ECS) and Amazon Elastic Kubernetes Service (Amazon EKS), AWS App Mesh, serverless (AWS Lambda), and Amazon API Gateway, as well as Continuous Integration / Continuous Delivery (CI/CD) DevOps tools.

This approach results in the creation of an orchestrator that is fully built based on microservice-based flexible architecture. This is a critical aspect, because the complexity of the 5G orchestration tool will continue to increase as new service cases get added. Microservice-based architecture will ensure that the orchestrator can be further developed and enhanced in an agile and flexible manner.

RIFT.ware

RIFT's RIFT.ware is a carrier-grade Orchestration and Automation platform that delivers management and life cycle automation of virtual network services, applications, and functions with scale. Designed specifically for deployment of service provider use cases,

RIFT.ware simplifies the day-to-day operations of network functions, and the composition and management of complex network services.

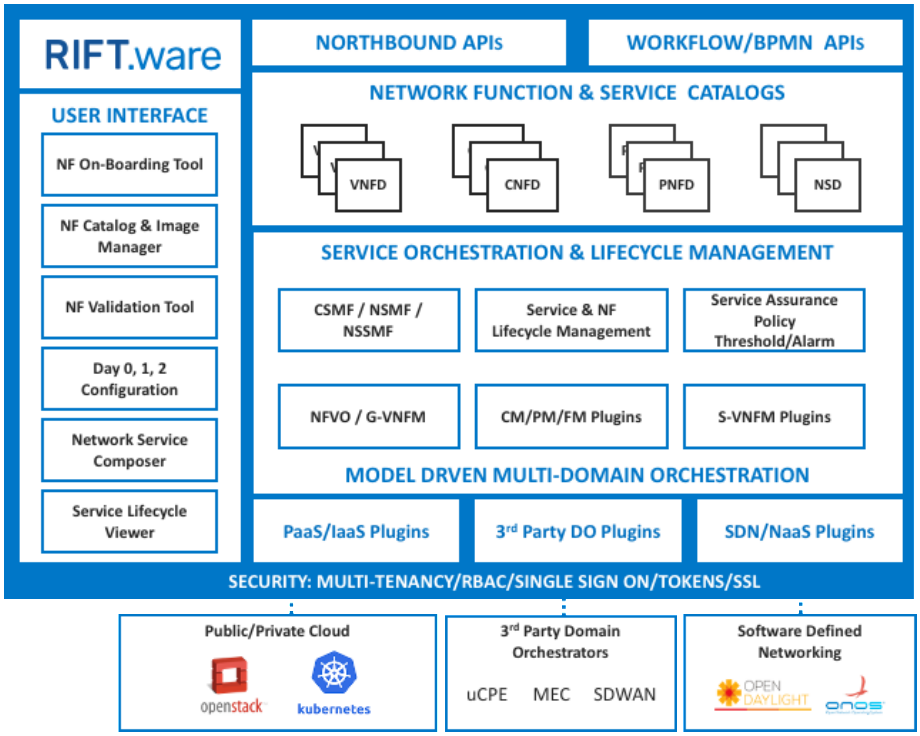

Figure 1 – RIFT.ware architectural framework

Architected to adapt to any orchestration environment, RIFT.ware is a cloud native, modular, and scalable system that is capable of providing orchestration and automation services according to the following common architectures:

- 3GPP Slice Management Functions: Communication Service Management Function (CSMF), Network Slice Management Function (NSMF), Network Slice Subnet Management Function (NSSMF)
- ETSI NFV Management and Orchestration (MANO)
- TM Forum Open Digital Architecture – Service Orchestration and Domain Orchestration
- Open Network Automation Platform (ONAP)
- ETSI Open Source MANO (OSM) – Service Orchestration and Resource Orchestration
- MEF 55 – Lifecycle Services Orchestration Reference Architecture

RIFT.ware's orchestration modules provide full virtual service, virtual function, and virtual network life cycle management and automation, enabling service providers to rapidly onboard, deploy, manage, and automate an end-to-end service spanning multiple clouds and locations in a standards-based, repeatable manner. RIFT.ware's open interfaces and plugin-driven architecture enable operators to deploy multi-domain services to satisfy multiple markets, including:

- 5G network slices and network slice subnets
- Multi-site enterprise virtual private network (VPN) and SD-WAN
- Multi-access edge computing

The multi-tenant nature of RIFT.ware provides secure separation of access based on the service provider's internal organizational boundaries, suppliers, and customers. To enable distributed RIFT.ware deployments in the service provider's private cloud or in the AWS Cloud, RIFT.ware provides integration with the service provider's single sign on (SSO) system to ensure security.

To ensure high availability and reliability, RIFT.ware is architected to run over Amazon EKS, including the use of stateless worker tasks and task resiliency mechanisms to ensure restarts in the case of local fault events, or load-balanced, state synchronized multi-AZ resiliency. Coupled with Elastic Load Balancing service and placement in different AWS Regions and Availability Zones (AZs), RIFT.ware provides recovery even from a total AZ outage (Figure 2).

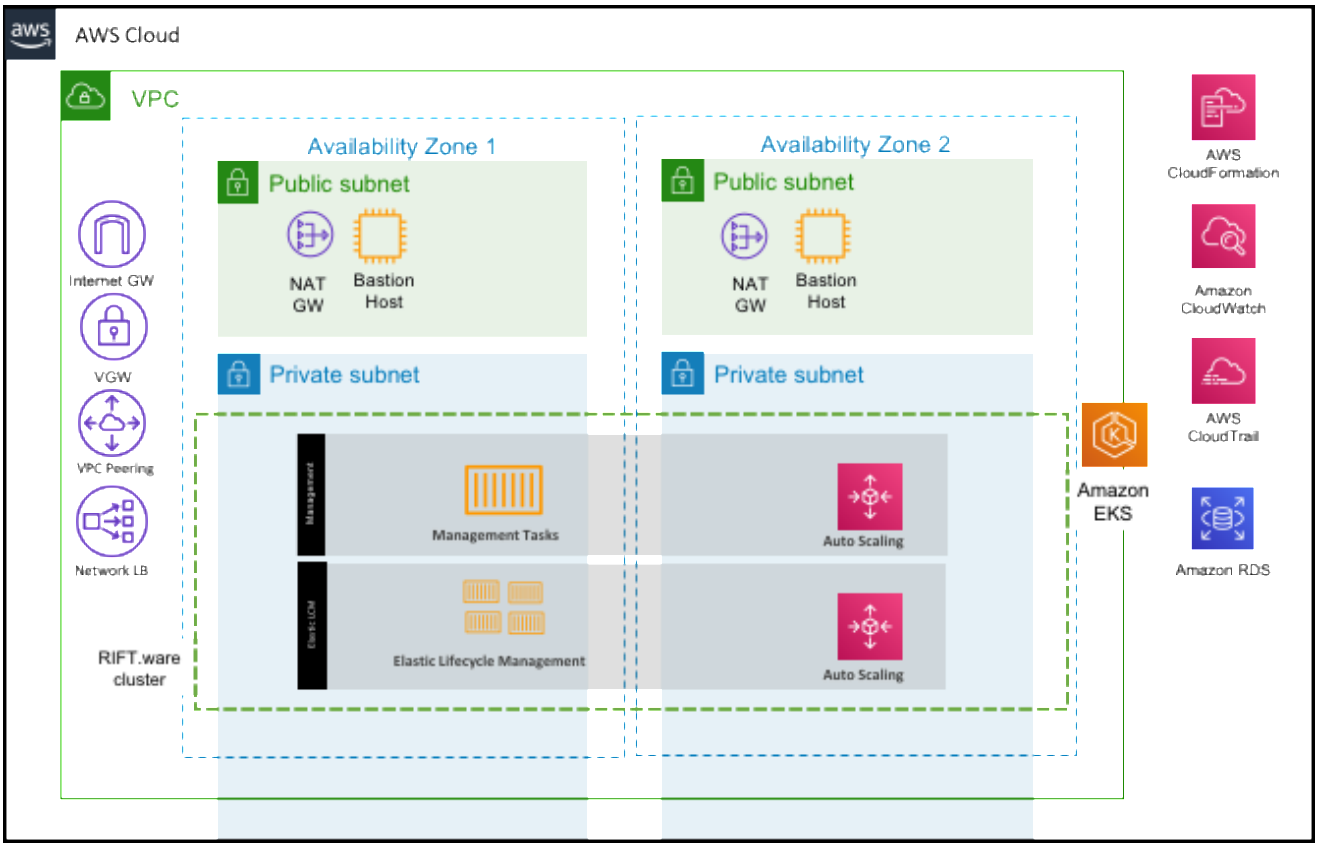

Figure 2 – Multi-AZ redundant RIFT.ware on Amazon EKS

As a high scale, low footprint automation engine, RIFT.ware's deployment overhead is low. Coupled with the economies of scale of the AWS Cloud, such as the ability to scale, right sizing, capacity on demand, and the removal of hardware ownership and obsolescence issues, RIFT.ware on AWS provides significant cost of ownership benefits for service providers.

Key Concepts of the 5G Architecture

5G networks are intended to provide connectivity and data processing services tailored to a variety of business use cases, from Augmented Reality / Virtual Reality (AR/VR), to Internet of Things (IoT) and Industrial Internet of Things (IIoT), to connected cars. To address the disparate requirements brought by these use cases, 3GPP has rearchitected the 5G network to utilize the best practices of cloud software.

Cloud Native Control Plane and Service-based Architecture

In a continued evolution step from 4G, 5G networks further separate the user plane functions from the control plane. This enables the control plane to evolve into cloud-native, stateless functions that yield benefits in resiliency and elasticity. The control plane has been rearchitected into a service, mesh-like architecture known as the Service Based Architecture (SBA). A benefit of the SBA is reliance on HTTP as the carriage protocol, which makes the 5G control plane far less network sensitive, requiring a simple IP network instead of a specific point-to-point topology.

User Plane Separation

Separation of the User Plane Function (UPF) from the remainder of the 5G functions has benefits not only to the control plane, it also allows the user plane to be distributed closer to the edge, where latency and backhaul bandwidth become less of a concern. This enables 5G architectures to scale better to interactive, high-bandwidth applications such as gaming and augmented reality.

SLA-driven Network

A major effort in 5G networks is the standardization of Slice Templates for describing the type of 5G network to create. The Generic Network Slice Template (GST) specified by the GSMA, is a list of attribute-value pairs that indicate the location, characteristic, behavior, and type of service expected of a 5G slice. When filled out with values, the GST, now known as the NEtwork Slice Type (NEST), describes the SLA requirements of the 5G slice. Standardization of the GST enables service providers to clearly indicate the requirements of the network slice to each entity responsible for the components of that slice even across departmental and enterprise boundaries, ensuring the SLA of the end-to-end slice.

5G Service Management Layers

5G slice management is hierarchical in nature. When coupled with NFV, 5G slice management and ETSI NFV neatly form a series of “as-a-Service” layers with open APIs to automate the life cycle of each layer (Figure 3).

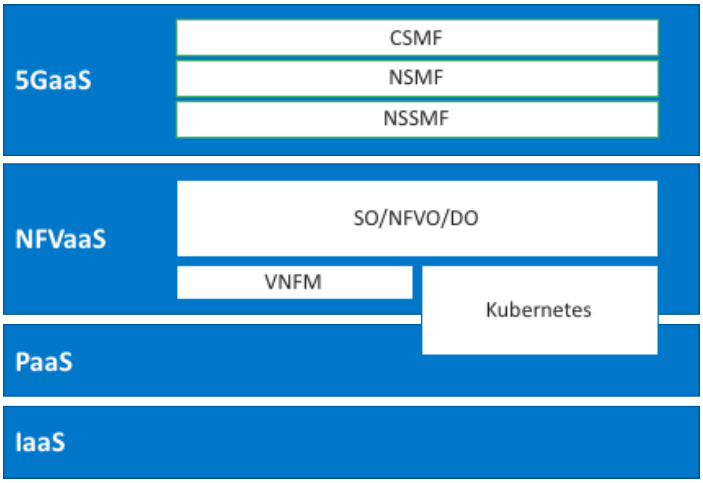

Figure 3 – Layers of 5G slice management

- **Infrastructure as a Service (IaaS)** provides raw resources such as compute, storage, and networks.
- **Platform as a Service (PaaS)** provides base infrastructure services, such as OS and networking, to host cloud native 5G functions. While Kubernetes logically belongs in the PaaS layer, it also provides automation capabilities for managing the life cycle of cloud native network functions, so it spans both the PaaS and Network Functions Virtualization as a Service (NFVaaS) layers.
- **NFV as a Service (NFVaaS)** provides the capability to automate deployment of end-to-end network services. This layer is used by the service providers' network operations teams for day-to-day operations automation. Internally, the NFVaaS layer is further sub-divided:
    - **NFV Orchestrator (NFVO)**, which provides Network Service life cycle management.
    - **VNFM**, responsible for VM-based Virtual Network Function (VNF) life cycle management.
- 5G as a Service (5GaaS) provides APIs for creation and management of end-to-end network slices. The 5G slice management functions, CSMF, NSMF, and NSSMF, are contained within this layer. The role of the 5GaaS layer is to interact with the service providers' OSS/BSS layer and translate the customer facing requirements, which includes the customer SLAs into operational commands that can be realized at the NFVaaS layer.

AWS and RIFT as a 5G Platform

5G enables service providers to deploy new services to their customers, tailored to each customer's individual requirements. The evolution of the 5G architecture addresses all modern use cases, from connectivity for a massive scale Internet of Things to interactive audio/video applications.

Realizing such use cases requires service providers to deal with the new architectural challenges posed by 5G services, including:

- The ability to automatically and rapidly deploy new services
- The ability to place these services in a manner that best serves the use cases in terms of performance, availability, and proximity
- The ability to ensure that the service fulfills the customer's requirements and expectations

To achieve these goals, service providers must be able to automate the processes required to deploy 5G network services from OSS/BSS to the IaaS layer, and must be able to place and connect both VM-based and cloud native network functions on different VIMs to support Control and User Plane distribution. This means that resources must be available in a variety of disparate locations. These network functions and network services must be continuously monitored and automatically adjusted to scale according to network and customer demand, or heal in the case of failures.

While standards bodies such as TM Forum, 3GPP, and ETSI have invested in specifications for deployment automation in the service provider domain, these specifications focus on private cloud deployments, and often overlook the many benefits of the public cloud. AWS Services, such as Amazon Elastic Compute Cloud (Amazon EC2), Amazon ECS, and Amazon EKS, can augment the service provider's private cloud by providing additional resources for placement of 5G services. AWS Outposts offers a consistent, managed, low-latency service using the service provider's on-premises resources.

Properly utilizing AWS Services such as AWS CloudFormation, The AWS Cloud Development Kit (AWS CDK), or Amazon API Gateway requires an automation solution that can bridge the gap between a service provider's standards and the public cloud, enabling 5G services to be automatically deployed from the OSS/BSS and span VM, containers, public, and private cloud.

Deploying 5G Services on AWS

The end-to-end nature of 5G slices demands a system capable of automating the deployment of components across a variety of technology, geographic, and administrative domains. This automation should foster reusability, maintainability, and adaptability. If it does not, service deployment times will suffer, and the deployment itself will be fragile and hard to adapt to new service changes.

Network connectivity is vitally important in 5G networks. While its significance is understood in the transport connectivity between RAN and Core, the role of the network is often overlooked in use cases such as geographic resiliency, or in inter-NF or inter-slice subnet connectivity, which may span multiple sites and even cloud types (VM and container).

Finally, it must be possible to integrate 5G slice management with the service provider's existing OSS/BSS, to ensure the service can be ordered, monitored, and managed from a customer perspective.

The 5G Service Lifecycle

The journey of constructing a 5G service starts with the basic building blocks of the service; specifically, the individual Virtual or Containerized Network Functions (CNF) that comprise the 5G network service. With RIFT.ware, any supplier's 5G NF, be it RAN, Core, or Transport, can be onboarded to or constructed using RIFT.ware's Network Function Composer UI and the resultant Network Function Descriptor (NFD) stored in the RIFT.ware NF catalog (Figure 4).

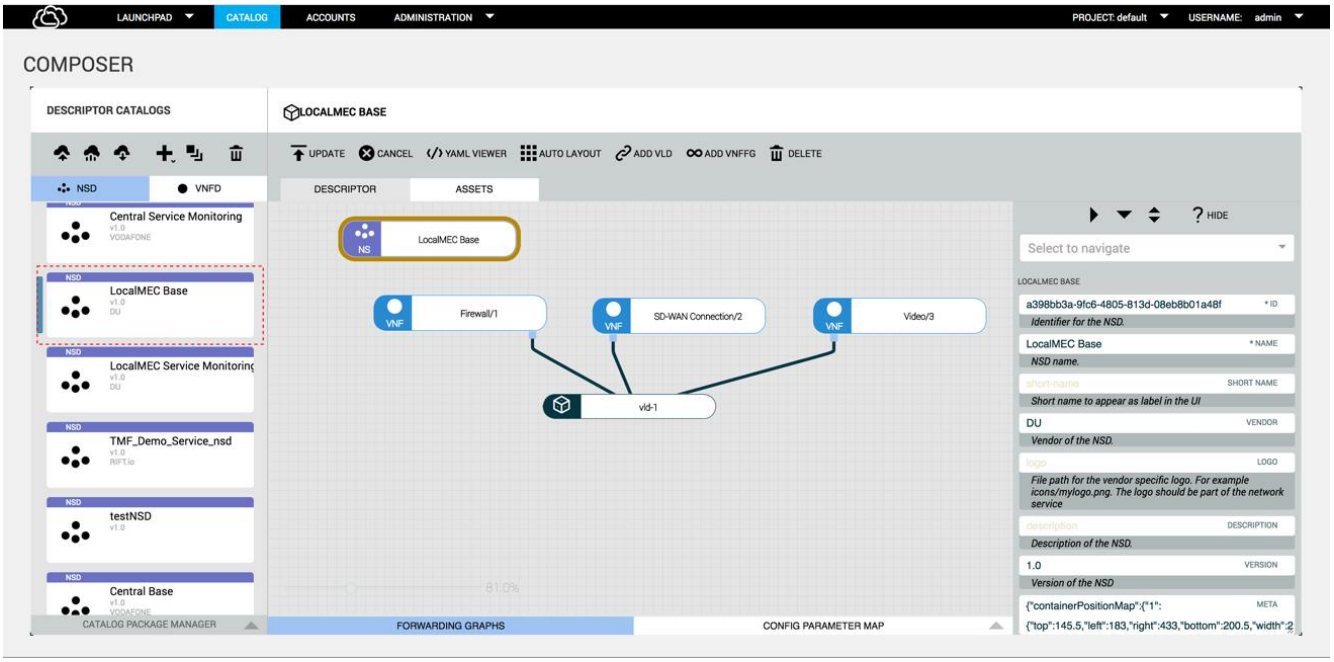

Figure 4 – RIFT.ware network function composition

For cloud native NFs, this process is even simpler, as the Helm chart can be imported via a single click and automatically converted into an ETSI-compliant Containerized Network Function Descriptor (CNFD) (Figure 5).

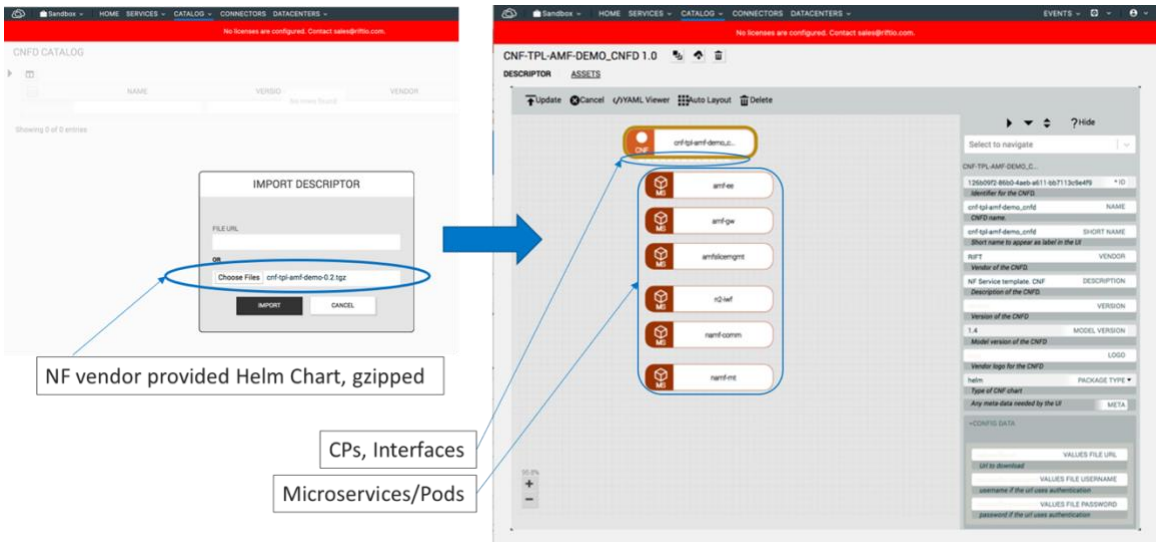

Figure 5 – RIFT.ware Cloud Native Application Onboarding

Creation of the NFD enables operators to easily manipulate the NF, by dragging and dropping any NFD in the RIFT.ware catalog. Operators can create an entire topology of

NFs simply by connecting these NFs into multi-vendor, multi-domain network services (Figure 6).

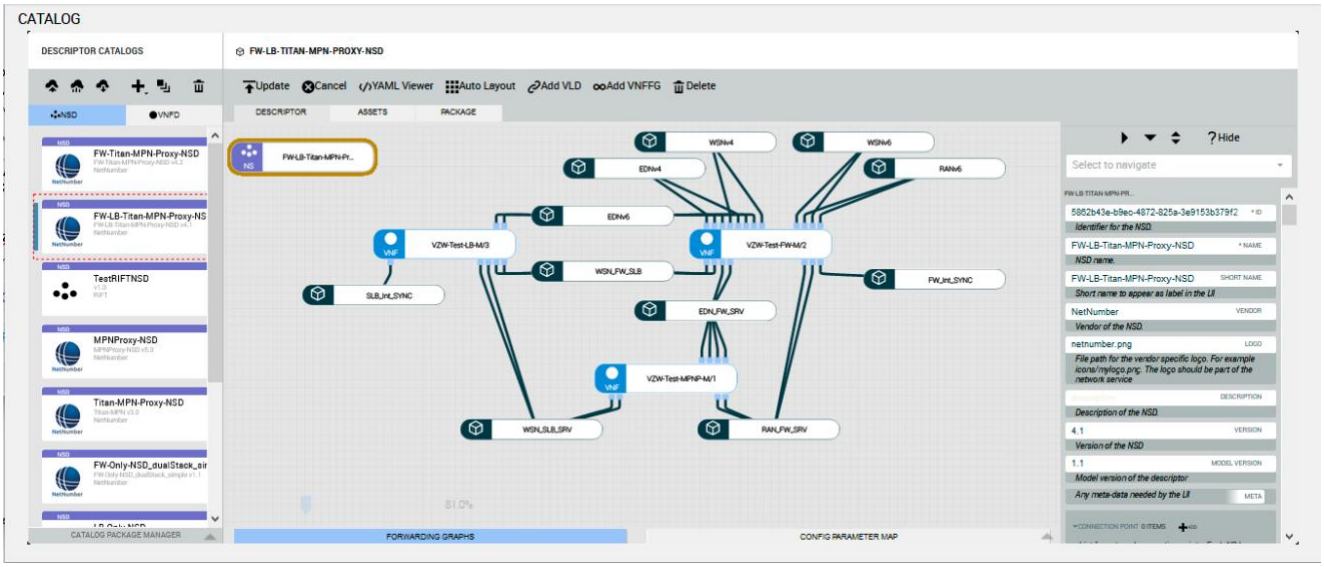

Figure 6 – RIFT.ware network service composition

The ability to create service chains via a drag-and-drop UI is especially important for 5G network functions and service provider NFs in particular, because many NFs require multi-homed Pods and optimized input/output (I/O) for user plane packet forwarding and/or redundancy purposes in an AWS Region (Figure 7)

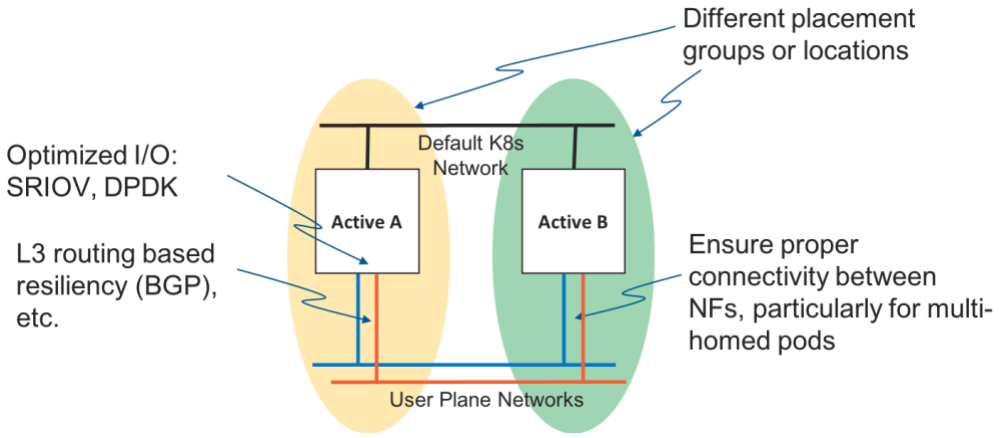

Figure 7 – Data Plane Network Function placement

The resultant network function and network service templates created through the design time process are cloud-agnostic templates that can be placed on any cloud type (private or public) or location (Figure 8). Each template can be customized at instantiation time through a simple input file containing parameters such as names of placement groups, IP address pools, and Domain Name System / Dynamic Host Configuration Protocol (DNS/DHCP) servers, to tailor the deployment for site or use case specific parameters. This enables the NF and NS templates to be reused across service provider use cases, from enterprise 5G to massive Internet of Things (mIoT) deployments.

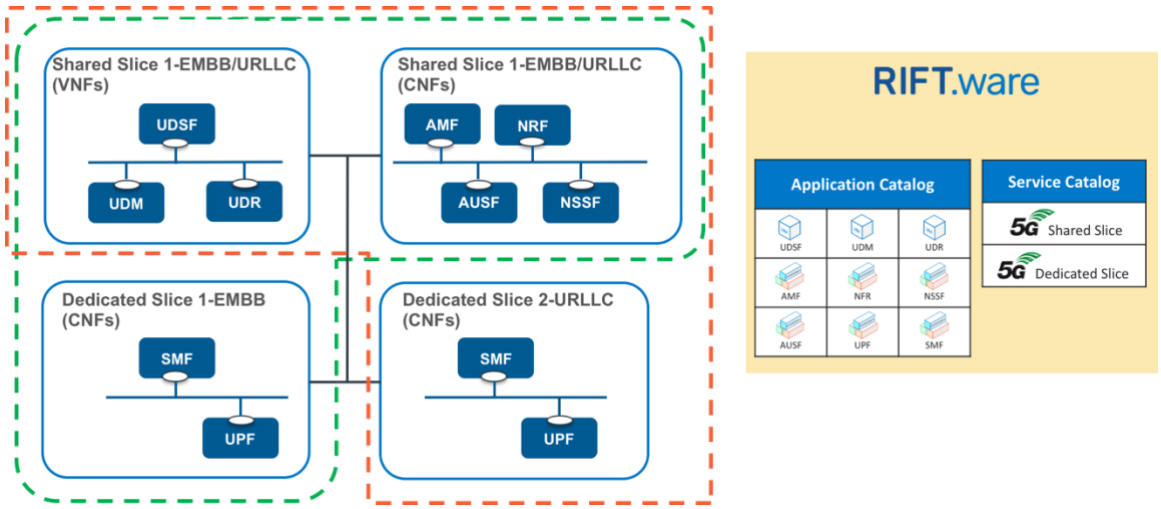

Figure 8 – Multi-site, hybrid cloud 5G slice deployment with RIFT.ware

Prior to deployment, the NS template can be put through a CI/CD pipeline to test the functionality and determine its operational characteristics. This stage is crucial to 5G service deployments, as the 5G function and the resultant service is tested against the desired 5G NESTs supported by the service provider. Characteristics examined may include:

- Performance curves, such as throughput or sessions per second with varying sizes and numbers of CPU, memory, and storage
- Latency characteristics
- Resiliency characteristics, including recovery times and placement requirements for local and geographic redundancy
- Optimization parameters, such as use of Single-Root I/O Virtualization (SR-IOV) and/or Data Plane Development Kit (DPDK) and effect on latency and performance

After the behavior of the NF is understood, the service provider may choose to fine-tune the CNFD/Virtual Network Function Descriptors (VNFD) for deployment by modifying the model to include:

- Key Performance Indicator (KPIs) information, to enable RIFT.ware to monitor performance and health of the NF
- Scaling or healing policies triggered by the KPIs
- Initial scaling size for the NF
- Placement rules, such as EC2 placement groups, host aggregates, affinity/anti-affinity, and security groups
- Optimization rules, such as need for DPDK, SR-IOV, security and encryption assist, which are of particular importance to User Plane functions

These attributes are encoded in the models to enable repeatable deployments. As many attributes can be parameterized for run-time customization, these models may be used even if deployed in different locations (such as AWS Outposts vs. a Region where user plane function is deployed on Outposts for local breakout, and control plane functions are in a Region to enable centralized management and operation), cloud form factors (such as MEC platform / Universal customer premises equipment (uCPE) vs. data center), and cloud capabilities (such as availability or absence of SR-IOV). Use of an abstract model allows a “low code” approach to cloud deployments, which reduces the time needed to put the NFs into service, and reduces risk.

A similar approach is used for construction of the service. Using RIFT.ware, service providers can drag and drop NFs into the RIFT.ware Service Composer pane to create

Network Service Descriptors (NSD) containing both CNFs and VNFs. The CNFs and VNFs are connected using Virtual Links (VLs) to form service chains. At instantiation time, RIFT.ware interacts with the VIM and networking layers to enable placement of the NFs on any cloud and any site, and ensures the connection between NFs is made regardless of whether connectivity is between CNFs and VNFs or across clouds. Due to the portability of the NSD, the entire NSD may be inserted to a CI/CD pipeline for testing and fine-tuned afterwards, in the same manner as the NFD.

The NSD model is particularly useful in the 5G orchestration to support control plane/user plane (CP/UP) separation. As previously discussed, a key goal of 5G is to support interactive, high-bandwidth applications such as gaming and augmented reality. By distributing the UP NFs closer to the edge, latency can be mitigated and traffic can be broken out locally instead of being backhauled, allowing bandwidth to be used more efficiently.

This type of deployment requires coordination between different technology domains including:

- Ensuring the UP NFs are placed on capable hosts that support high bandwidth, low latency applications through use of DPDK, SR-IOV, and similar technologies.
- Ensuring the container cluster supports multi-homing, an essential requirement of UP NFs.
- Ensuring the NFs are chained together to form the correct service topology. This is particularly important for UP NFs which have multiple interfaces.
- Ensuring the network connectivity supports the necessary latency and bandwidth required for a distributed, multi-site deployment.

To support these requirements, the candidate AWS placement locations in the form of EC2, ECS, or EKS accounts that meet these criteria are added into RIFT.ware as VIM accounts or Container-as-a-Service (CaaS) accounts. Once added, all VIM and CaaS accounts are available as virtual data center resources which can be selected during NSD instantiation, to be used by that NSD for placement of NF workloads (Figure 9).

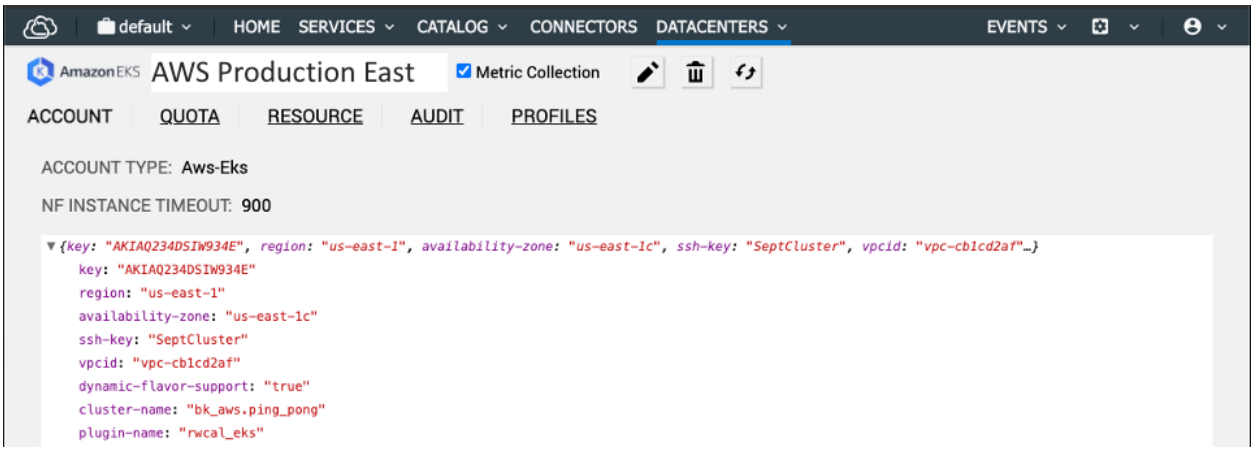

Figure 9 – AWS EKS VIM provisioned within RIFT.ware

Once the NF descriptors, NS descriptors, and VIM/CaaS accounts have been created, the entire instantiation process can be automated through a combination of RIFT.ware and AWS capabilities, such as the interworking between ETSI NSD, CNFD, and AWS CloudFormation Templates and AWS APIs:

1. As part of the NS instantiation process, a VIM or CaaS account is selected for each constituent VNF and CNF in the NS. These locations are selected based on serving area, latency requirements, and other attributes. The selection may be completed manually or programmatically via the RIFT.ware northbound SOL 005 API.
2. As part of the NS instantiation process, the NS deployment requirements are gathered from the NSD model. For example, in the case of a distributed deployment where the UP may be located on an AWS Outposts instance and the CP is located at the service provider data center, RIFT.ware will:
    a. Create and provision the necessary L3 access points for the NFs each location, as directed by the Virtual Link Descriptor (VLD) model within the NSD.
    b. Create and provision the L3 WAN connectivity between sites through interfacing with the network controllers at each site.
3. RIFT.ware then creates the inter-NF virtual links (networks), attaching these to the access points created in step 2.
4. As part of the instantiation process of each constituent NF, the optimization and placement policies such as multi-homing, EC2 placement groups, SR-IOV, DPDK, and affinity / anti-affinity are determined from the NF descriptor models.

5. Using the requirements in step 4, a suitable cluster is located for placement of the NF. If a suitable cluster is not located, RIFT.ware constructs a CloudFormation Template tailored to the NF's requirements, and creates a cluster to host the optimized NFs.
6. RIFT.ware places the NF into the selected VIM or CaaS account and creates the intra-NF networks. For cloud-native NFs, this is achieved via the Amazon ECS or EKS APIs, which are used to fully instantiate and manage the CNF. Day 0 configuration is also applied at this stage.
7. Following instantiation, RIFT.ware applies Day 1 configuration to the NF.
8. RIFT.ware attaches the NFs to the inter-NF VLs created in step 3.
9. RIFT.ware then configures the NFs with service level configuration, such as ensuring each NF knows the IP address of other NFs in the service, or knows the identity of the geo-redundant pair.

Like the NF models, a model-driven approach to network service deployments enables a low-code, low-maintenance, and repeatable deployment across cloud locations and types, enabling the distributed, high-performance deployments required by 5G services.

End-to-end Architecture for 5G Service Deployment on AWS

A primary goal of 5G networks is the automated deployment of 5G slices based on direct customer request, through a marketplace or similar mechanism. Unlike application marketplaces, which typically involve the deployment of a single application in one site, a distinguishing characteristic of 5G slices is that they require a coordinated launch of multi-vendor applications across several sites, they require the networking of these applications in a specific topology.

Another attribute of 5G slices is the level of guarantees expected of a carrier-grade service. Items like latency and availability are vital in 5G, so placement of NFs in specific locations and zones to ensure proximity, accessibility to networking resources, and fault isolation is key to meeting the service level requirements of 5G.

The ETSI NFV specifications are mainly focused on automating tasks related to network operations, such as the creation of resources to support applications, and the chaining of applications to support Network Services. The APIs presented by ETSI are inherently resource and deployment focused, and speak in terms of compute, memory, storage, and IP addresses. While these APIs are ideal for specifying detailed placement and

connectivity, they lack information regarding customer intent, such as gold, silver, and bronze level, use case, and service area. To achieve the goal of automated deployment through a marketplace-like mechanism, it is necessary to receive customer requests that specify the intent, and translate these into instructions suitable for deployment.

TM Forum has invested heavily in creating a set of APIs specifically to address the problem of customer intent. The TM Forum Open APIs allow service providers to specify attributes that are more meaningful to the end customer, such as service tier described in a higher level, abstract terms such as gold, silver, or bronze, or SLA terms such as availability characteristics.

While TM Forum Open APIs provide the missing piece for the customer APIs, these APIs are very generic and contain no 5G slice semantics, which can lead to proprietary behavior that complicates the integration between OSS/BSS and other customer systems to the 5G complex. To ensure openness, 3GPP has defined a set of functions complete with models and APIs, to standardize the handover from the customer layer (OSS/BSS) to the resource layer (ETSI NFV).

As shown in Figure 3, 5G slice management functions act as the intermediary layer between the OSS/BSS and the ETSI NFV layer. Figure 10 shows the slice management architecture as defined in ETSI GR NFV-EVE 012.

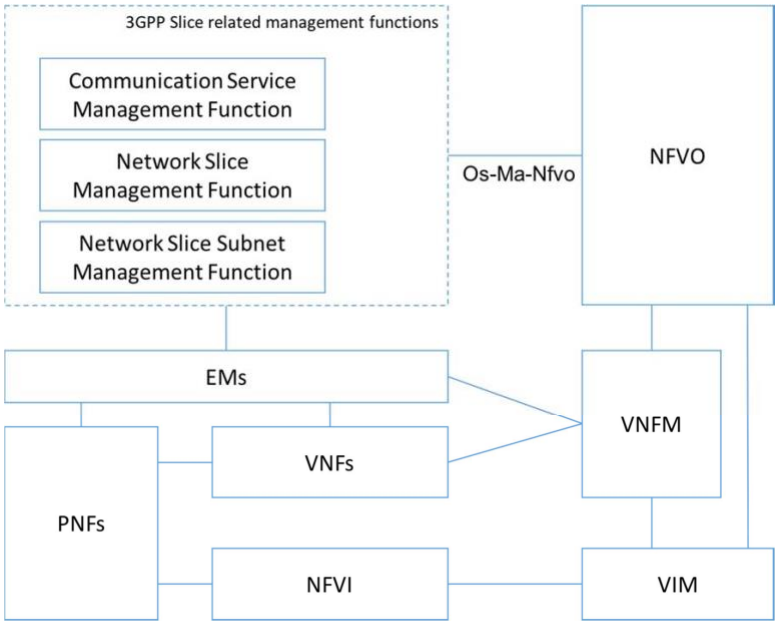

Figure 10 – 5G Slice management architecture (ETSI GR NFV-EVE 012)

3GPP slice management functions provide a key translation step from TM Forum APIs to ETSI APIs. In conjunction with the GSMA defined NEST template, the 3GPP functions enable deployments of SLA-driven 5G networks automatically from customer to resource.

The NEST Template

NEST templates define how a network slice is to be created, by describing the SLAs the slice is intended to fulfil, plus high-level instructions that determine the placement of the slice. Attributes in the NEST enable the customer to communicate the use case (enhanced Mobile Broadband (eMBB), mIoT, Ultra-Reliable Low-Latency Communication (URLLC)), performance, scalability, and location in mobile network terminology, which in turn enables the service provider to determine the resources required to fulfil that service.

For example, consider the “Area of service” attribute in the NEST. Based on the service provider’s knowledge of the geographic locations corresponding to the public land mobile network (PLMN) ID and Tracking Areas being requested, this attribute can be used to locate a data center, an AWS Region, or an AWS Outposts instance on which to place certain NFs. This information can then be conveyed to the RIFT.ware NFVO via the SOL 005 reference point for instantiating the network service (NS). Similar attributes in the NEST can also be used to determine NS and NF sizing, optimization parameters, and network connectivity.

NEST and the Slice Management Functions

The role of the NEST is key for ensuring correct instantiation of the 5G service. While it is possible to instantiate any 5G NF or even NS (5G or otherwise) directly through the TM Forum APIs alone, the NEST specifies the SLAs expected by the customer. The role of the 5G Slice Management Functions is to correctly map, decompose, and transform between TM Forum APIs and data models to ETSI APIs and data models (Figure 11).

Figure 11 – Model Transformations in 5G Slice Management

Following this sequence, you can see that:

- The CSMF selects a NEST based on the customer service request received from OSS, using TM Forum Open APIs. The APIs used, for example, TMF641 “Service Order”, and fields within the API such as the Service Specification Relationship are used to select the NEST template to use. The NEST is then communicated to the NSMF as a Service Profile using 3GPP APIs.
- The NSMF receives the Service Profile. Based on the Service Profile (NEST), the NSMF further decomposes the slice into slice subnets, and requests allocation of each subnet with the selected NSSMF using the 3GPP Slice Profile.
- The NSSMF’s role is to transform the Slice Profile into ETSI APIs, by mapping the NEST fields into SOL 005 instructions. This mapping is performed by RIFT.ware using a transformation engine that allows the Service Profile to select AWS Region, VIM (Outpost or AWS instance), placement parameters, and Cloud Formation Templates, based on the service provider’s CI/CD results and other considerations such as location and networking availability and capability. The NSSMF then requests deployment of the slice from the RIFT.ware NFVO using SOL 005 APIs.

As each service provider has unique deployment conditions such as type and location of data centers, services, suppliers, and capabilities, RIFT provides a simple yet flexible mechanism for supporting model transformations and selection of VIMs, suppliers, and services in the RIFT.ware 5G Slice Management functions. This mechanism supports slices and slice subnets across clouds, chaining together the NFs placed on Service Provider data centers, the AWS Region, and AWS Outposts.

Closed Loop SLA Control

Because the NEST template and 5G Slice Management as a whole is highly tailored towards SLA management, it is vitally important that the SLA of the end-to-end slice be continuously monitored and adjusted around changing demands and/or outages.

For this reason, AWS and RIFT have partnered to provide a Closed Loop SLA control mechanism, to enable the 5G slices to autonomously and automatically scale, heal, and adapt to changing network conditions.

Utilizing RIFT.ware’s auto-scaling and auto-healing framework plus Amazon CloudWatch analytics functionality, AWS and RIFT provides an end-to-end service assured slice management solution (Figure 12).

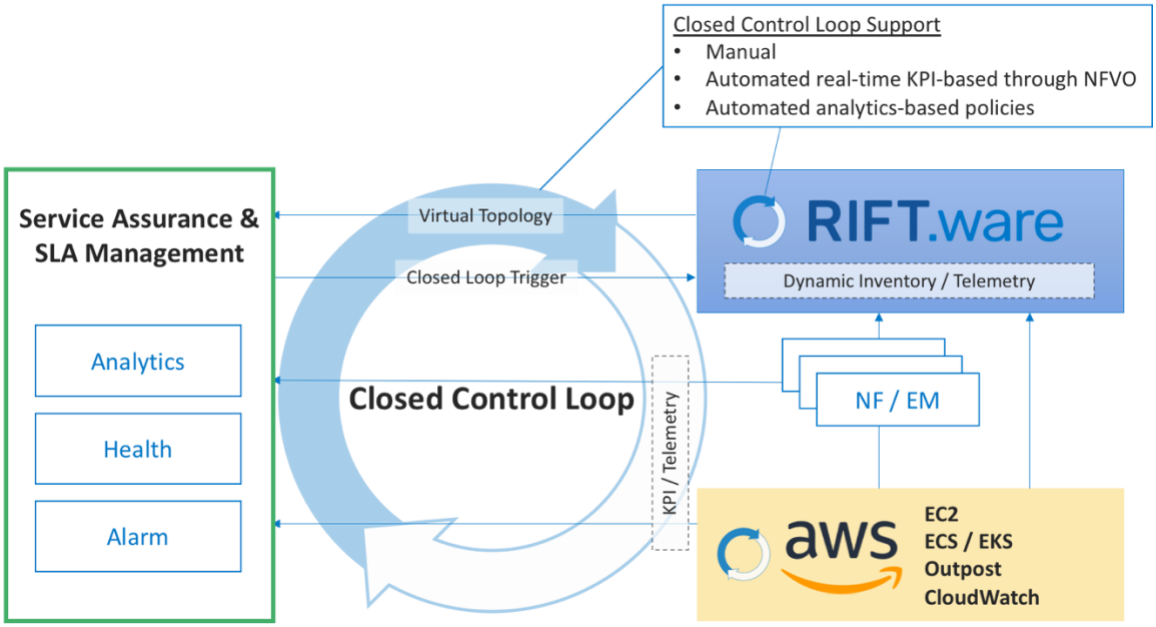

Figure 12 – Closed Control Loop reference architecture

RIFT is a leading innovator in Closed Control Loop Automation, providing a suite of advanced capabilities that allow service providers to automate Day 2 operations using intelligent, policy-based triggers and operator-designed, model-driven responses.

RIFT.ware's ETSI-compliant data models contain built-in attributes for application layer KPI from VNFs and CNFs. For cloud-native applications, in which large numbers of worker tasks may be placed on widely dispersed VMs, application layer KPIs such as sessions per second, processing latency and the like are far more indicative of NF performance and congestion over infrastructure-level KPIs such as CPU or memory utilization percentage.

These application layer KPIs, or *monitoring parameters*, can then be aggregated to form policies that trigger life cycle management actions when a threshold is crossed. Such actions may include healing actions such as restarting tasks or VMs, or scaling a NS/NF via RIFT.ware's Autoscaling Framework.

RIFT.ware's Autoscaling Framework works with the NS/NF life cycle management workflows to ensure that new capacity (VMs/VNF and Containers/CNF) are added and removed seamlessly from service, with minimal impact to upstream and downstream systems, and to ensure even load distribution across all available capacity (Figure 13). The Autoscaling Framework uses the RIFT.ware's built-in life cycle management workflows to automate all aspects of scaling a multi-NF network service, including:

- Instantiations of *scaling groups* (Figure 13) consisting of one or more NFs based on the policy trigger
- Configuration of all new NFs
- Rebalancing load across all NFs
- Busy-out of user sessions, in the case of scale-in
- Reprogramming any load balancers in the network service

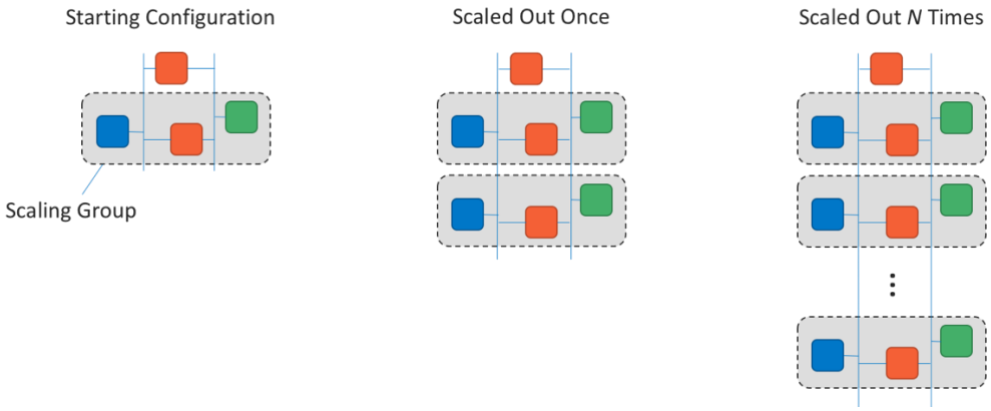

Figure 13 – Model-driven scaling groups in RIFT.ware

RIFT.ware's support of open standards-based APIs and data models also enables simplified integration with the service provider's service assurance system, and Amazon CloudWatch for bridging anomaly event detection, which can provide more complex analytics and artificial intelligence / machine learning (AI/ML)-based triggers in a non-real time fashion. To support this scenario, RIFT.ware exports the entire virtual topology, including all cross-layer correlation data to the analytics system, which can then be combined with telemetry obtained from CloudWatch to trigger intelligent policy-based actions.

Using Amazon CloudWatch and RIFT.ware, service providers can create complex use cases such as:

- Automated instantiation of new Slice Subnet in response to degradation of user experience
- Automated recovery during loss of Slice Subnet
- Proactive scaling based on capacity planning trends

- Reactive scaling due to outages based on pre-determined disaster recovery plans

Through Amazon CloudWatch, advanced ML-based analytics can be used to drive Closed Loop automation for 5G networks in a rapid, repeatable, efficient manner.

5G Use Cases

5G Slice Deployment Automation

Automation of 5G deployments requires automation across all layers of the 5G orchestration stack. AWS Services and tools, coupled with RIFT's RIFT.ware Orchestration and Automation suite, provide top-to-bottom automation of 5G services.

AWS programmable services related to IaaS and PaaS orchestration and management, such as AWS Lambda, AWS Config, CloudFormation, Step Function, and CDK, can help design and realize network slicing at the optimal resource level, while RIFT's RIFT.ware service orchestration and automation solution can be used to design, deploy, and manage slices, slice subnets, and other network services by providing standards-based APIs and functions to the service provider.

RIFT's RIFT.ware Orchestration and Automation suite provides ETSI NFV service (NFVO) and VNF (VNFM) level components. As a use-case, agnostic, standards-based ETSI NFV orchestration suite, RIFT has demonstrated the onboarding and life cycle management of nearly 100 VM-based and containerized cloud native Network Functions from over 40 different vendors. Using the RIFT.ware NF and Service Composer components, service providers can rapidly onboard new NFs and design carrier-scale end-to-end services using a drag-and-drop UI that facilitates multi-site hybrid cloud deployments and geographically redundant, high-performance, optimized network services. The RIFT.ware automation suite also makes use of AWS Services and tools, such as CloudFormation Templates, to automate the creation of clusters in the AWS Cloud to support the deployment.

To enable 5G-specific automation for the creation and management of 5G slices, RIFT has introduced the RIFT.ware Slice Management Automation suite to automate the reception of customer service orders, fulfilment of these orders into a deployed end-to-end network service on AWS infrastructure, and continuous monitoring and Closed Loop life cycle management for automated SLA management.

CI/CD Pipeline

The automation capabilities described in the previous section can also be used to drive CI/CD pipelines in order to create predictable, carrier-grade services. A well-established CI/CD process enables service providers to characterize, and in some cases, predict NF and NS behavior, which is essential to ensuring that the service, once released, is able to fulfil the SLAs in the customer's requested NEST.

The best CI/CD processes make use of automation tools to ensure repeatability of the process, and closely mimic the conditions under which the service is to be deployed. Using a combination of AWS and RIFT automation, service providers can accurately reproduce deployment environments in a sandbox (Figure 14).

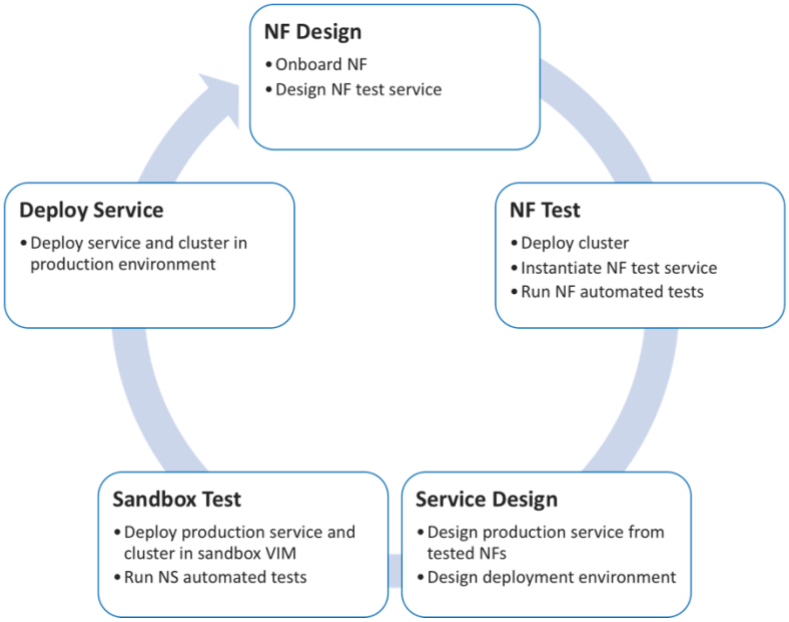

Figure 14 – CI/CD pipeline automation

With its rich, standards-based APIs, RIFT.ware can be incorporated into the service provider's existing automation frameworks such as Jenkins or Robot Framework, to drive automated deployment of network functions or network services into a sandbox environment using AWS developer tools, bookend the NF/NS with test harnesses or traffic generation tools, and fully configure the service as a final step for automation tests or deployment.

Conclusion

AWS and RIFT provide capabilities for you to deploy and leverage 5G infrastructure globally, to attain scalability, elasticity, and high availability. Customers are using AWS, AWS Partner Network (APN) Partners, and open-source solutions to host mobile workloads on AWS. This has resulted in reduced cost, greater agility, and a reduced global footprint. For partner solutions, AWS has the broadest and strongest partners in the ecosystem, available through AWS Marketplace and the APN Partner Central for each part of the stack presented in this paper.

The reference architectures and best practices provided in this whitepaper can help you successfully set up 5G workloads on AWS and optimize the solutions to meet end user requirements, all while optimizing for the cloud. AWS extends its cloud beyond Regions to the distributed edge. This provides CSPs with a choice between AWS Outposts (to implement cloud native user plane) or Outposts and AWS Wavelength to host MEC applications and latency sensitive workloads. Additionally, management and orchestration, as well as network slicing, can be deployed cost effectively, following cloud-native architectures and with an easy path to use AI/ML capabilities to create predictive and self-healing networks.

Contributors

Contributors to this document include:

- René Tio, Vice President Product Management, RIFT, Inc.
- Matt Harper, CTO, RIFT, Inc.
- Noel Charath, Vice President Services Delivery, RIFT, Inc.
- Tetsuya Nakamura, Senior Partner Solutions Architect, WW Telco Partner, Amazon Web Services
- Young Jung, Ph.D., Senior Partner Solutions Architect, WW Telco Partner, Amazon Web Services
- Tipu Qureshi, Principal Engineer, AWS Premium Support, Amazon Web Services

Document Revisions

Date	Description
January 2021	First publication.

Acronyms

- **3GPP** — 3rd Generation Partnership Project
- **5GaaS** — 5G as a Service
- **AI** — Artificial Intelligence
- **API** — Application Programming Interface
- **APN** — AWS Partner Network
- **AR** — Augmented Reality
- **AZ** — Availability Zone
- **CaaS** — Container as a Service
- **CI / CD** — Continuous Integration / Continuous Delivery
- **CNF** — Containerized Network Function (or Cloud-native Network Function)
- **CNFD** — Containerized Network Function Descriptor
- **CP** — Control Plane
- **CSMF** — Communication Service Management Function
- **CSP** — Communications Service Provider
- **DHCP** — Dynamic Host Configuration Protocol
- **DNS** — Domain Name System
- **DPDK** — Data Plane Development Kit
- **eMBB** — enhanced Mobile Broadband
- **ETSI** — European Telecommunication Standards Institute
- **GSMA** — GSM (Global System for Mobile Communications) Association

- **GST** — Generic Network Slice Template
- **IaaS** — Infrastructure as a Service
- **IETF** — Internet Engineering Task Force
- **I/O** — Input / Output
- **IoT** — Internet of Things
- **IIoT** — Industrial Internet of Things
- **ISG** — Industry Specification Group
- **KPI** — Key Performance Indictor
- **MANO** — Management and Orchestration
- **MEC** — Multi-access Edge Computing
- **mIoT** — Massive Internet of Things
- **ML** — Machine Learning
- **NEST** — NEtwork Slice Type
- **NF** — Network Function
- **NFD** — Network Function Descriptor
- **NFV** — Network Functions Virtualization
- **NFVaaS** — Network Functions Virtualization as a Service
- **NFVI** — Network Functions Virtualization Infrastructure
- **NFVO** — Network Functions Virtualization Orchestrator
- **NS** — Network Service
- **NSD** — Network Service Descriptor
- **NSMF** — Network Slice Management Function
- **NSSMF** — Network Slice Subnet Management Functions
- **NFVFaaS** — Network Functions Visualization as a Service
- **ONAP** — Open Network Automation Platform
- **OSM** – Open Source MANO

- **OSS/BSS** — Operation Support System / Business Support System
- **PaaS** — Platform as a Service
- **PLMN** — Public Land Mobile Network
- **PNF** — Physical Network Function
- **RAN** — Radio Access Network
- **SBA** — Service Based Architecture
- **SD-WAN** — Software-Defined Networking in a Wide Area Network
- **SLA** — Service Level Agreement
- **SR-IOV** — Single Root I/O Virtualization
- **SSO** — Single Sign-on
- **uCPE** — Universal Customer Premises Equipment
- **UP** — User Plane
- **UPF** — User Plane Function
- **URLLC** — Ultra-Reliable Low-Latency Communication
- **VIM** — Virtualized Infrastructure Manager
- **VL** — Virtual Link
- **VLD** — Virtual Link Descriptor
- **VM** — Virtual Machine
- **VNF** — Virtual Network Function
- **VNFD** — Virtual Network Function Descriptors
- **VNFM** — Virtual Network Function Manager
- **VR** — Virtual Reality