Solution Blueprint

Multi-System Operators
Virtualized Cable Modem Termination System

Automated Deployment of vCMTS on Red Hat OpenShift

Table of Contents

Cloud-native vCMTS using Intel® Xeon® processors and Red Hat OpenShift offers agility and cost-effective operation for multi-system operators (MSOs).

1 The Rise of MSO Cloud-Native Architectures

Unprecedented consumer demand for internet bandwidth continues to grow, with higher-resolution video as a chief contributor. Momentum is gaining among mainstream users toward ultra-high definition (UHD) and beyond, with 4K-capable television sets in about 44 percent of total television households in 2021, compared to 31 percent in 2019.[1] That trend is not limited to UHD adoption, as 8K sets are expected to start slowly gaining traction, growing from about one million units in 2022 to 72 million in 2025.[2] The traffic volume associated with a specific video resolution varies widely according to the encoder used, but data requirements for an 8K set may exceed 40 GB/hour, roughly four times the requirement for HD 1080p.

Multi-system operators (MSOs) have responded to massive current and future traffic increases with a wave of innovation. Targeting greater agility and cost-efficiency, operators are adopting cloud-native technologies to achieve more dynamic network topologies and elastic capacity. Using standards-based, general-purpose servers, they have decoupled workloads from specific hardware, implementing software-defined infrastructures that dynamically and automatically reconfigure themselves to accommodate fluctuating traffic levels. This approach eliminates the need to maintain dedicated headroom on each host for traffic peaks, reducing infrastructure requirements.

This cloud-native architecture is instantiated by virtualizing network services and encapsulating them in containers to create cloud-native network functions (CNFs), as illustrated in Figure 1. The topology consists of CNFs running on a containers-as-a-service (CaaS) layer based on Red Hat® OpenShift® that runs on bare metal and/or infrastructure-as-a-service.

All of a network function's dependencies are integrated into its container, making it portable across multi-cloud environments. Centralized orchestration by Red Hat OpenShift provides excellent agility and dynamic elasticity. Optimizations

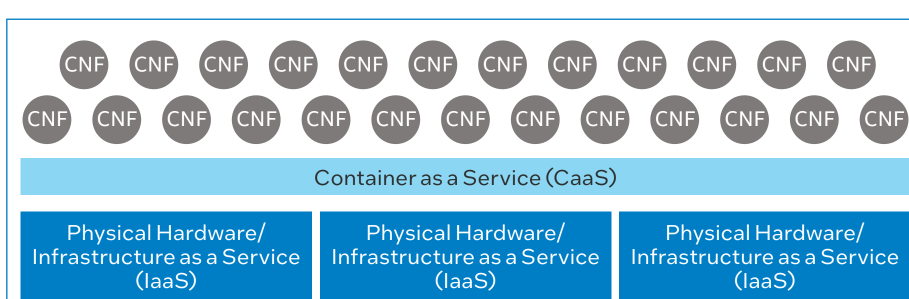

Figure 1. CNFs running on CaaS infrastructure.

for Intel® Xeon® Scalable processors help accommodate large numbers of subscribers per service group and deliver energy efficiency. Red Hat OpenShift also provides a robust automation layer that increases the efficiency of generating and deploying CNFs.

This solution blueprint provides MSOs with future-ready platform guidance to deploy vCMTS using the Intel vCMTS Reference Dataplane as part of their initiatives to develop robust and flexible cloud infrastructures. Intel and Red Hat building blocks form the foundation for deploying cloud native network functions. These methods contribute to the development of open architectures for a common approach to cloud-native support for multiple access technologies.

The performance gains from Intel Xeon Scalable processors help MSOs scale subscriber services on fewer hosts. In addition, these gains can be realized across a range of network workloads, including virtualized broadband network gateways (vBNGs).

2 Reaching Beyond the Limits of Hardware CMTS

Traditional approaches to CMTS distribute broadband data services using dedicated, single-purpose hardware located in the MSO headend facility. Traditional architecture was developed for relatively low levels of end-customer bandwidth, where a single pair of upstream and downstream CMTS ports and a fiber node could support a service group of approximately 500 homes sharing a strand of fiber. In the past several years, MSOs have revised those capacity assumptions to support higher bandwidth services and added upstream and downstream ports, resulting in higher costs.

For example, in order to support a growing number of users, the MSO may be forced to dramatically reduce the size of each service group, add another fiber node, and activate another pair of upstream and downstream CMTS ports. Deploying additional CMTS capacity carries a broad range of costs, including data center facilities capital expense (CapEx) and operating expense (OpEx), expenditures for fiber infrastructure, and the cost of the nodes themselves and the software licenses to support them. As bandwidth demands continue to escalate, continuously adding more hardware is unsustainable.

In addition, a dedicated CMTS device cannot apply its unused resources to other functions, while other single-function devices are likewise incapable of delivering additional CMTS capacity. An architecture based on fixed-function devices is fundamentally incompatible with forward-looking MSO strategies to create more agile and efficient distributed networks.

One step in modernizing hardware-based CMTS involves transitioning to MAC and PHY functionality split into separate network components: a remote physical (R-PHY) device and a vCMTS, as illustrated in Figure 2. Virtualizing the CMTS function enables it to run cost-effectively using general-purpose servers in place of specialized hardware in a cloud-native environment. Red Hat OpenShift provides a highly automated, scalable environment for deploying vCMTS, optimized for servers based on Intel Xeon Scalable processors.

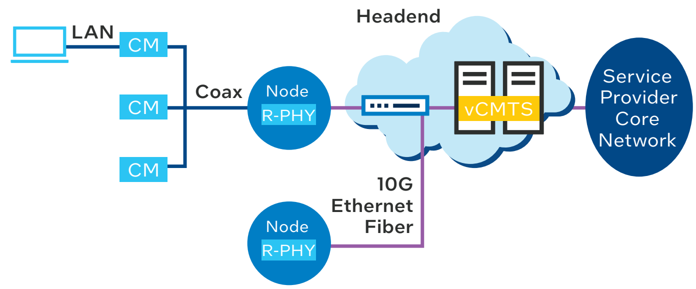

Figure 2. MAC dataplane deployed as vCMTS at cable headend.

By integrating everything but the physical access layer in software, vCMTS provides flexibility and integration with the broader cloud-native architecture, enabling dynamic, software-defined infrastructure. The modular design of CNFs in this disaggregated architecture allows them to scale and be upgraded independently of each other. MSOs can deploy new services quickly, with minimal operational disruption and without requiring hardware upgrades. That future-readiness is critical to achieving and sustaining lower TCO.

3 Design Considerations for Future-Ready Networks

The design and construction of future-focused MSO networks involve a range of transformative technology approaches. Many of these trends have emerged directly in response to changing needs. Navigating the roles of these trends within future networks is a key strategic requirement.

3.1 Cloud-Native Architecture

Taking full advantage of cloud computing models requires software designed explicitly for the cloud, rather than just deploying existing software there. Cloud-native architecture and deployment models are designed from the ground up to be lightweight. This allows applications to maximize agility, performance, and scalability, getting the most value possible from infrastructure investments. At the same time, cloud-native principles can improve the ability of development organizations to serve the larger business, with higher solution quality and faster time to production for new products and services.

- **Microservices** construct applications as sets of small, autonomous, reusable modules on a shared fabric that communicate using standard protocols and APIs to instantiate composite functions.
- **Containers** encapsulate microservices or other software entities along with all their dependencies in lightweight packages that can be deployed anywhere and seamlessly traverse multi-clouds with centralized orchestration.
- **Continuous integration/continuous delivery (CI/CD)** is the concept of making many small, frequent releases rather than infrequent, monolithic ones, increasing the pace of change while reducing its incremental impact.
- **DevOps and GitOps** automate and accelerate modern development, networks, and services operations, to increase efficiency and deterministic repeatability, for consistent services delivery and rapid failure recovery.

3.2 Separation of Control Plane and Dataplane

Separating control-plane functions from dataplane functions allows each to scale independently. For example, more dataplane resources can be deployed to respond to increased traffic without affecting the control plane. That separation allows for either a distributed or centralized deployment. This blueprint calls for the control plane to be centralized, while the dataplane is distributed, as shown in Figure 3.

Distributed compute in the dataplane enhances network performance for latency-sensitive applications by having the control plane intelligently choose network functions—based on capability or proximity, for example. To that end, separating the control plane and dataplane facilitates programmatic control of the dataplane by software-defined networks to increase the efficiency of data delivery.

By abstracting network resources from the underlying hardware, separation of the control plane and dataplane is also instrumental to network slicing, which is the practice of creating multiple virtual segments with defined characteristics and requirements. Resources can be dedicated or prioritized for each network slice, providing differentiated quality of service among specific traffic types or customer service levels. That capability will become increasingly important as MSOs address novel use cases in areas such as IoT, connected vehicles, and connected health. Red Hat OpenShift provides a common container-based architecture across this entire topology, with a unified lifecycle that helps simplify the development and deployment of applications and services.

3.3 Network Disaggregation

Network disaggregation breaks open specialized physical appliances, replacing them with software-based network functions running on general-purpose servers. Communication between open components supplied by multiple hardware and software vendors allows the software-defined network to chain together CNFs to dynamically create applications and services.

In particular, distributed access architecture (DAA) shifts functions that have traditionally resided in the headend to the distributed network as cloud-native network functions (CNFs). This decentralization reduces the amount of data that needs to be passed over long-haul connections, decreasing bandwidth costs, and improving latency, reliability, and security.

Network disaggregation supports an accelerated rate of change, because bringing a new product or service to market does not require deployment of new infrastructure. Networks are forward-looking, because updates and changes can be made using the CI/CD paradigm, without taking services offline. Particularly as the rate of change at the network edge continues to accelerate, disaggregation helps protect MSOs from the cost and business impacts of evolving dataplane requirements.

4 Automated vCMTS Stack Elements

The past decade or so has seen a transition in the modes of delivering network functions from rigid, single-purpose architectures to flexible ones designed to integrate seamlessly with a dynamically changing environment. This journey began with network functions deployed as monolithic applications on custom, fixed-function hardware. The equipment was expensive and inflexible, and often siloed. Having a single source vendor providing hardware, software, and support tended to limit options for new development.

Today, network functions have become more agile and less tied to specific equipment or topologies. Cloud-native infrastructure deploys CNFs as container-resident microservices, dynamically as needed to provide services to the network. Red Hat OpenShift delivers a consistent foundation for development and deployment of network functions across any combination of private and public cloud environments.

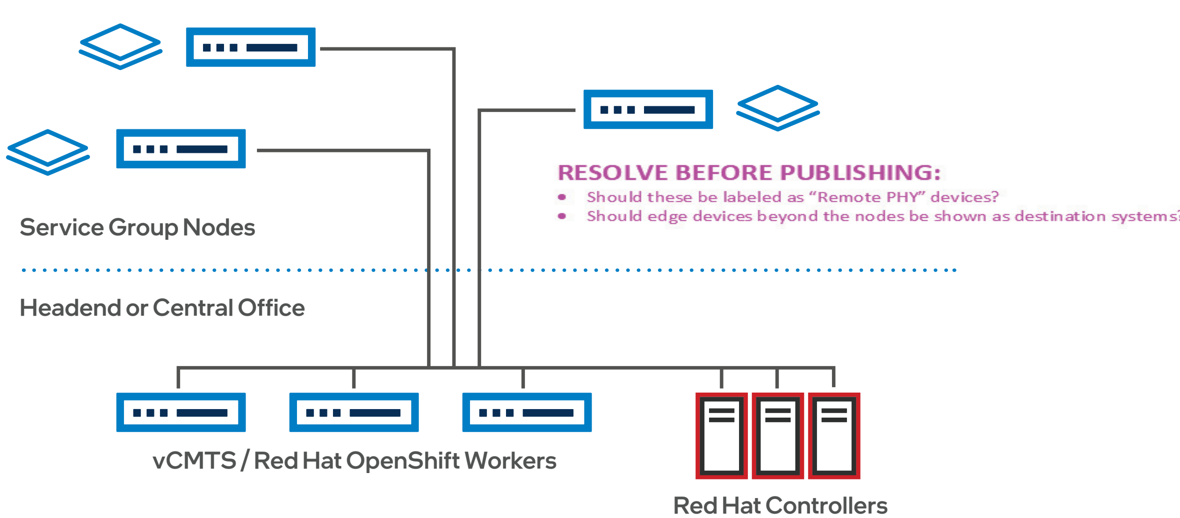

Figure 3. vCMTS and nodes in a service group with the Red Hat OpenShift controllers.

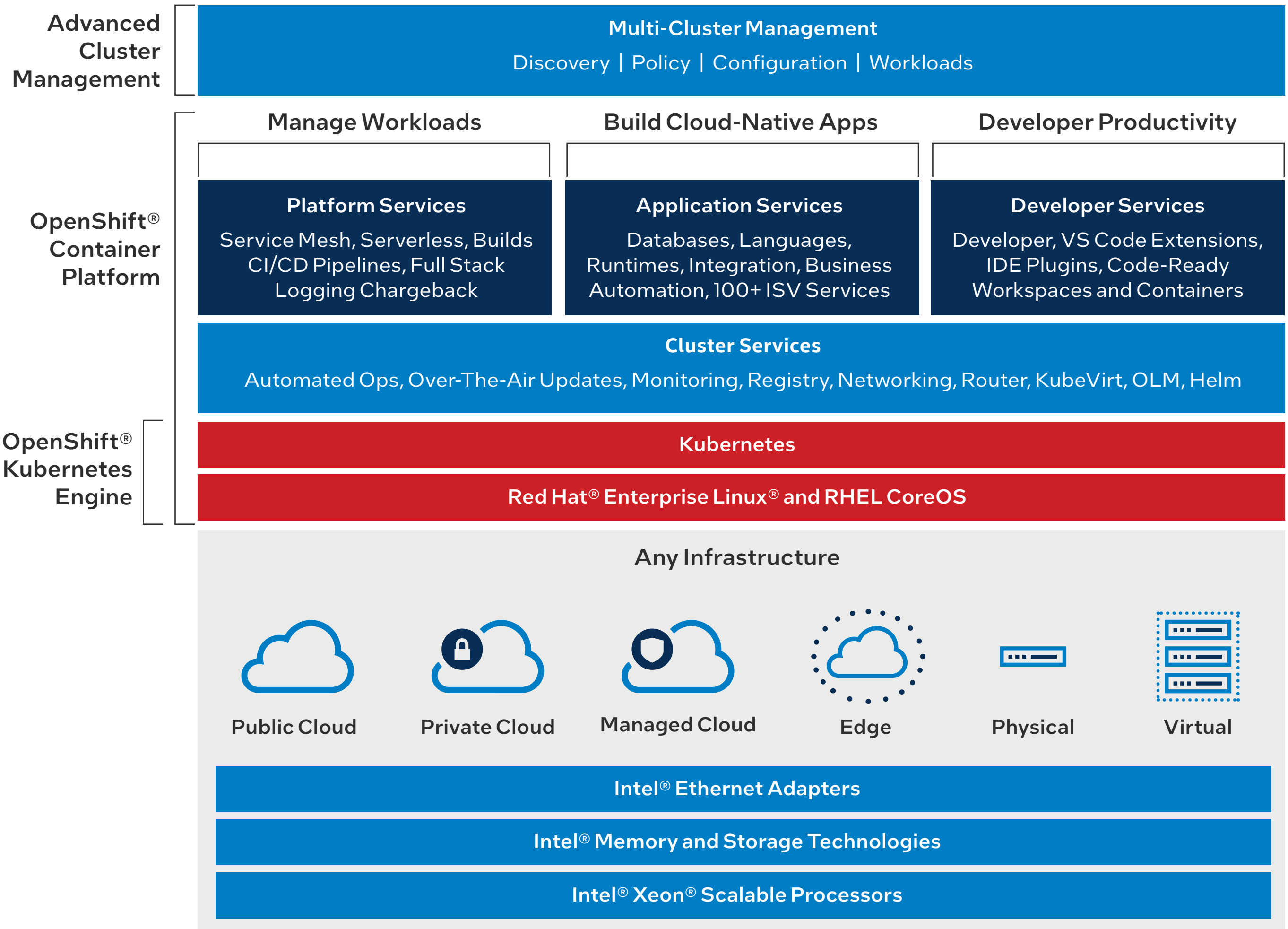

Figure 4. Red Hat OpenShift automates deployment and management of CNFs, including vCMTS.

4.1 Cloud-Native Automation: Red Hat OpenShift

Red Hat OpenShift, illustrated in Figure 4, enhances Kubernetes for MSOs with CI/CD pipeline tools, automation, and security capabilities. The platform therefore benefits from testing, hardening, integration with other Red Hat open source offerings, and certification initiatives by Red Hat. The components are developed and distributed using an open-source model to foster innovation. Red Hat OpenShift is multicloud-ready, enabling clusters to be deployed within the data center, at the edge, and to a variety of public clouds, giving MSOs more flexibility.

Red Hat Enterprise Linux CoreOS is the operating system designed specifically for running containerized applications using Red Hat OpenShift. Deployment and maintenance of Red Hat OpenShift clusters are highly automated, enhancing the efficiency of IT operations. The Red Hat OpenShift environment simplifies the management of multiple cloud-native network elements and workloads, as illustrated in Figure 5.

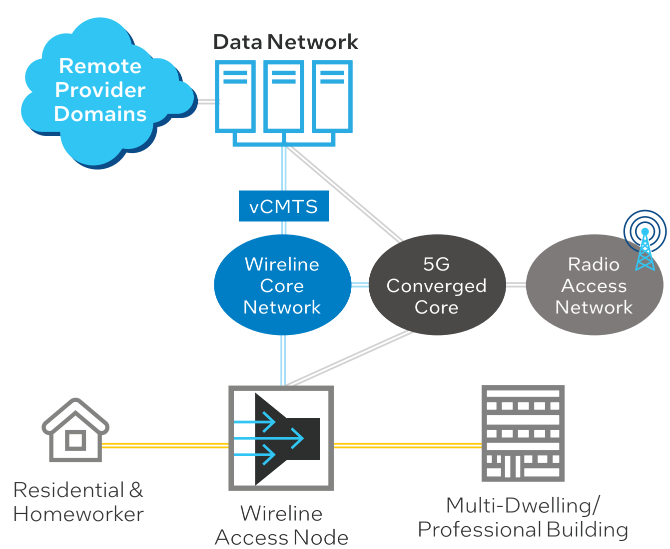

Figure 5. Example MSO Network.

Red Hat OpenShift Operators are software entities that represent fundamental sets of capabilities within Red Hat OpenShift, as shown in Figure 6. They capture human operational knowledge in code, to encapsulate the processes for packaging, deploying, and managing Kubernetes applications. Operators foster repeatability in IT processes, perform ongoing health checks of system components, and handle over-the-air updates to Red Hat OpenShift and third-party software. Some key Operators are described in more detail in the remainder of this section.

4.1.1 Performance Add-On Operator

The Performance Add-on Operator (PAO) enables advanced node performance tunings on sets of nodes. The PAO controls the host allocation and configuration of performance-enhancing features such as CPU isolation/reservation, memory hugepages, and IRQ management. The PAO can also enable the use of a real-time kernel for certain low-latency environments. For vCMTS workloads, the PAO is used to enable CPU tuning and hugepage memory allocation.

4.1.2 SR-IOV Operator

The SR-IOV Network Operator creates and manages the components of the SR-IOV stack within the Red Hat OpenShift. It performs the following functions:

- Orchestrates the discovery and management of SR-IOV network devices
- Generates **NetworkAttachmentDefinition** custom resources for the SR-IOV Container Network Interface (CNI)
- Creates and updates the configuration of the SR-IOV network device plug-in
- Creates node-specific **SriovNetworkNodeState** custom resources

The SR-IOV Network Operator is used to allocate SR-IOV VFs for use by CNFs. After creation of SR-IOV networks, pods and instances only need annotations to allow network connections.

4.1.3 Intel Ethernet Operator

The Intel Ethernet Operator automates the use of hardware PPPoE packet processing to enhance session management and flow steering in dataplane applications. CNF-specific flow definitions that have been created in the Intel Ethernet Operator can be referenced as annotations in a pod specification in conjunction with a Custom Resource Definition. After an SR-IOV interface has been rendered by the SR-IOV Network Operator, a component of the Intel Ethernet Operator is called to enable the required flow definition.

4.1.4 Intel QuickAssist Technology (Intel QAT) Operator

The Intel QAT Operator enables the use of Intel QuickAssist Adapters to accelerate compression and encryption operations in Red Hat OpenShift workloads. Performance-optimized operations across applications and platforms include symmetric encryption and authentication, asymmetric encryption, digital signatures, RSA, DH, and ECC, and lossless data compression. Encryption capabilities also include inline IPSec, which enables customers to free up precious compute cores for other applications.

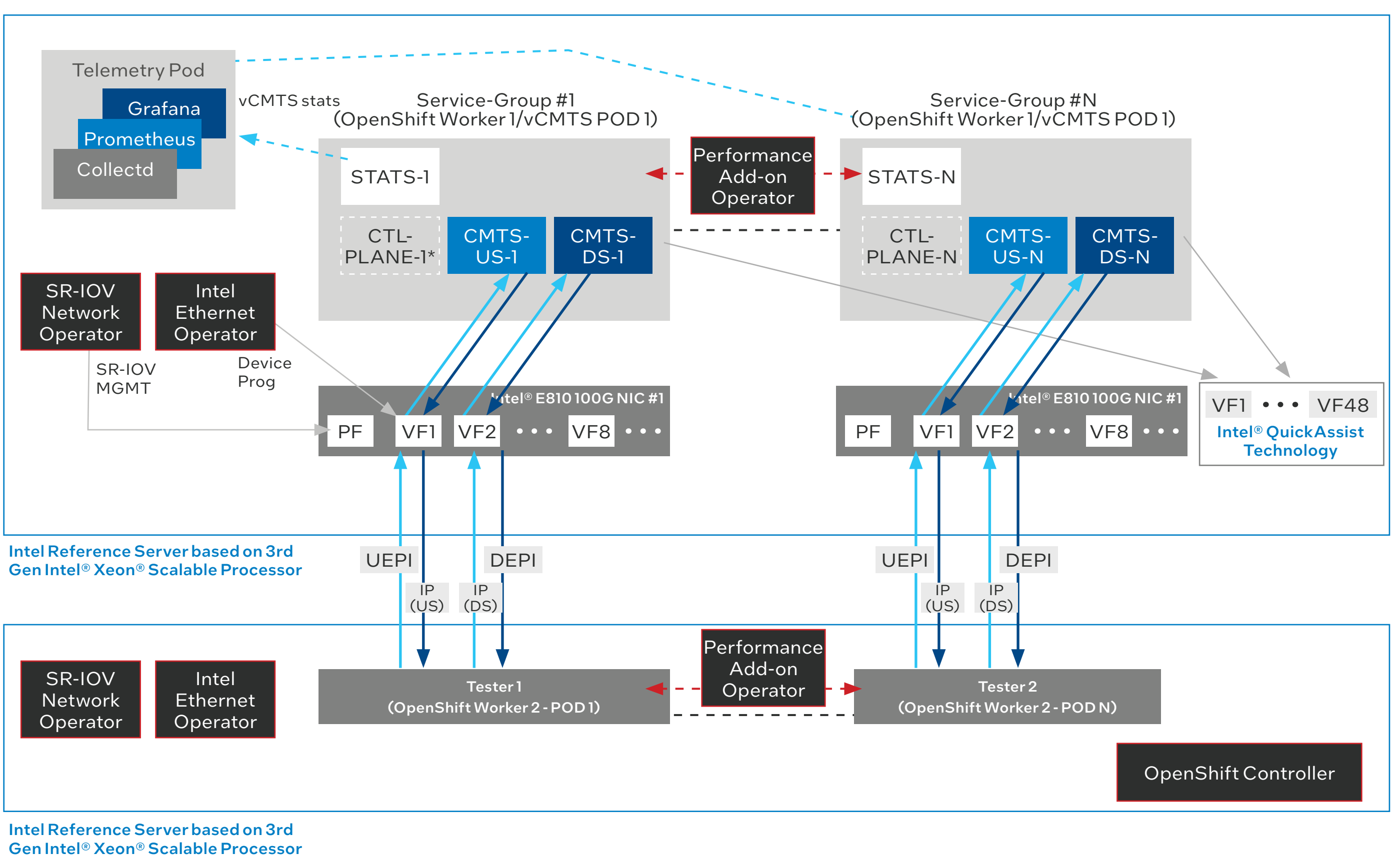

Figure 6. Red Hat OpenShift operator environment.

4.2 Sample Implementation: Intel vCMTS Reference Dataplane

The Intel vCMTS Reference Dataplane provides a tool for characterizing vCMTS dataplane packet-processing performance and power consumption on platforms based on both Intel Xeon Scalable processors and Intel Xeon D processors. The architecture is based primarily on DPDK libraries throughout the packet-processing pipeline, offering an open, flexible basis for vCMTS deployments by MSOs.

This capability enables MSOs to assess the potential throughput and energy efficiency benefits of deploying cloud-native vCMTS on Intel architecture. The core component of this package is a MAC dataplane based on the DPDK packet-processing framework that complies with Data-Over-Cable Service Interface Specification (DOCSIS) elements, including the following:

- Media Access Control (MAC) and Upper Layer Protocols Interface (MULPI) specification
- Downstream External PHY Interface (DEPI) specification
- Upstream External PHY Interface (UEPI) specification
- Security (SEC) specification

The environment includes traffic-generation and vCMTS dataplane nodes based on Intel Xeon processors that can be scaled up to include multiple instances of each on multiple physical servers. A JSON configuration file contains subscriber cable modem information. Each vCMTS dataplane instance represents a cable service group and is deployed as a container pod. While the reference environment supports both discrete Intel QAT cards and Intel QAT integrated into Intel Xeon D and Intel Xeon Scalable processors, this solution blueprint uses discrete cards.

The Intel vCMTS Reference Dataplane runs on the Intel Container Bare Metal Reference Architecture (BMRA) cloud-native software stack. This Kubernetes-based reference software environment is designed to facilitate early adoption of Intel platforms and open source platform software capabilities. It supports Kubernetes clusters based on multiple worker nodes that are managed by one or more control nodes. The Kubernetes topology from the BMRA is easily transferable to Red Hat OpenShift, offering enhanced automation and network hardening for vCMTS implementations.

Intel® Select Solutions for NFVI Forwarding Platform: Standard, Validated Forwarding-Plane Designs

Network implementations based on NFV infrastructure (NFVI) can accelerate dataplane functions for greater speed and agility, improved time to market, and scale for future capacity and new services. To continue their transformations to fully virtualized, cloud-native networks, MSOs must adopt disaggregated solutions.

The Intel Select Solutions for NFVI Forwarding Platform are predefined, workload-optimized designs that help reduce the challenges of infrastructure evaluation and deployment for vCMTS and other dataplane workloads, such as BNG, UPF, and AGF. Solutions are validated by OEMs/ODMs, certified by ISVs, and verified by Intel. These solutions are developed in extensive collaboration between Intel, vendor partners such as Red Hat, and the world's leading data center and service providers. All Intel Select Solutions are tailored combinations of Intel data center compute, memory, storage, and network technologies that deliver predictable, trusted, and compelling performance.

Currently in version 2, the Intel Select Solutions for NFVI Forwarding Platform incorporates the 3rd Generation Intel Xeon® Scalable processor and Intel Ethernet 800 Series Network Adapters. Configurations used within the solutions include the following:

<table>
<tr><th>Ingredient</th><th>Cloud Node
Plus Configuration</th><th>Cloud Node
Base Configuration</th><th>Controller Node
Configuration</th></tr>
<tr><td>Processors</td><td>2x Intel® Xeon® Gold 6338N processor</td><td>2x Intel Xeon Gold 5318N processor</td><td>2x Intel Xeon Gold 5318N processor</td></tr>
<tr><td>Memory</td><td>512 GB DDR4</td><td colspan="2">256 GB DDR4</td></tr>
<tr><td>Intel® Optane™ Persistent Memory</td><td colspan="3">Recommended</td></tr>
<tr><td>Discrete Network Adapters</td><td>4x Intel® Ethernet Network Adapter E810 2CQDA2</td><td>2x Intel Ethernet Network Adapter E810 2CQDA2
or
4x Intel® Ethernet Network Adapter E810 CQDA2</td><td>2x Intel Ethernet Network Adapter E810 CQDA2</td></tr>
<tr><td>Intel® QuickAssist Adapter</td><td colspan="2">Optional</td><td>Recommended</td></tr>
<tr><td rowspan="2">LAN on Motherboard</td><td colspan="3">10 Gbps or 25 Gbps port for Pre-boot Execution Environment (PXE) and Operation, Administration and Management (OAM)</td></tr>
<tr><td colspan="3">1/10 Gbps port for management</td></tr>
</table>

Next-Generation Performance for Common MSO Workloads

Compared to predecessor processor platforms, the architecture enhancements designed into the 3rd Generation Intel Xeon Scalable processor deliver results to MSOs such as the following:

- **72 percent better vCMTS platform performance**, as well as the potential for a further 10 percent improvement with additional Intel QAT offload[5]
- **21 percent boost in vBNG performance**, while enabling increased flexibility, manageability, and scalability to expand emerging use cases[5]

4.3 Flexible Performance: 3rd Generation Intel Xeon Scalable Processors

The 3rd Generation Intel Xeon Scalable processors offer improved per-core performance over their predecessors, helping MSOs reduce infrastructure requirements and CapEx. To provide a balanced platform, those execution resource improvements are complemented by enhancements to the memory subsystem and I/O. The processor platform also provides high energy efficiency that helps reduce OpEx for multi-dimensional improvements to TCO.

- **Enhanced execution resources**. Microarchitectural improvements that accelerate the compute pipeline are supported by up to 40 cores per socket and L1 caches up to 60 MB.
- **Updated I/O subsystem**, with support for PCI Express Gen4, which provides faster data movement with 2x the bandwidth of PCI Express Gen3.
- **Improved memory resources**, with up to 1.6x higher memory bandwidth[3] and up to 2.66x higher memory capacity[4] than the prior-generation processors.

The processor platform targets low-latency, high-throughput deterministic performance. Solutions can tailor processor capacity to the needs of a specific implementation by choosing from among a wide range of core counts, clock speeds, and feature sets. New instructions enhance cryptographic acceleration on the CPU to complement the offload capabilities of the Intel QuickAssist Adapter. The 3rd Generation Intel Xeon Scalable processor also incorporates Intel Software Guard Extensions (Intel SGX), which allows protected execution enclaves to be designated as private areas of memory where only trusted software components can access secret data, such as encryption keys and passwords.

The 3rd Generation Intel Xeon Scalable processor enables MSOs to buy server hardware from their vendor of choice. General-purpose servers avoid the need to buy dedicated single-function physical appliances with extra headroom for each function type. Instead, they support software-defined infrastructure where function instances are dynamically generated as needed. These implementations benefit from the unmatched ecosystem of software and solutions for Intel architecture, with a high degree of feature integration across Intel CPUs, GPUs, FPGAs, and other accelerators.

4.4 Intelligent Networking: Intel Ethernet 800 Series Network Adapters

The Intel Ethernet 800 Series Network Adapters provide network I/O that complements the 3rd Generation Intel Xeon Scalable processor's compute and memory advances. The adapters use PCI Express Gen4 for improved bandwidth to the system board with network throughput up to 100 Gbps per adapter port. They deliver high networking performance across NFV and CNF workloads through a combination of sophisticated packet processing, intelligent offloads and accelerators, and high-quality open source drivers for dataplane processing.

The enhanced Data Plane Development Kit (DPDK) is an open-source set of libraries and drivers supported by the Intel Ethernet 800 Series Network Adapters that accelerates packet processing in the data path. It also facilitates building packet forwarders designed to operate on general-purpose servers. DPDK technology enables Intel Ethernet 800 Series Network Adapters to be controlled entirely in user space. This approach accelerates operations by allowing network packets to bypass the kernel network stack entirely. With 3rd Generation Intel Xeon Scalable processors, MSOs can enhance DPDK L3 Forwarding performance by up to 88 percent compared to the prior generation.[5]

4.5 Accelerated Encryption and Compression: Intel QuickAssist Adapter 8970

The pervasive encryption that is commonly deployed to protect data for edge workloads such as vCMTS places significant overhead requirements on the compute infrastructure. Data must be encrypted while at rest or in transit and decrypted while in use, so encryption and decryption operations are a constant requirement. Likewise, the platform must carry out frequent compression tasks to shrink data sizes and make more efficient use of bandwidth. Resources dedicated to encryption and decryption cannot be used for primary edge workloads, which creates contention and potential bottlenecks that reduce throughput and raise total cost of ownership.

As part of this solution, Intel QAT provides hardware acceleration to reduce the overhead associated with encryption and compression. The vCMTS dataplane uses Intel QAT to offload symmetric crypto from the CPU to dedicated hardware engines as part of the BPI+ stage of the DOCSIS MAC. Because the offload hardware is purpose-built for this work, it handles the offloaded workloads efficiently, while CPU resources are freed up for other work, increasing overall solution throughput.

The Intel QuickAssist Adapter 8970 is the low-profile PCI Express 3.0 x16 add-in-card (AIC) that provides Intel QAT hardware acceleration for this solution blueprint. It uses industry-standard security algorithms for functions such as SSL/TLS, IPSec, and firewall applications. The adapter supports efficient software development with its software libraries and APIs, with support for standard operating systems. The acceleration stack exposes a common interface for development of applications as well as new acceleration functions.

Conclusion

The application of cloud-native principles to cable and broadband access aggregation sites will allow MSOs to secure needed flexibility and afford them more efficient scaling for network provisioning. Combining modern software methodologies with open source operators as a standard way to expose Intel hardware accelerators enables the CNF provider ecosystem to build and pre-validate performance-optimized solutions. These innovations are vital for MSOs working to replace current fixed access solutions and improve their TCO.

This forward-looking approach combines open-source cloud expertise from Red Hat and Intel with their leadership on high-speed enterprise systems. It lays a future-ready path to cloud-native networks, an emerging transition that will make current services more efficient and enable MSOs to bring new services to market faster and more efficiently.

More Information

Intel Developer Zone vCMTS Reference Dataplane: intel.com/content/www/us/en/developer/topic-technology/open/vcmts-reference-dataplane/overview.html

Intel® Select Solutions for NFVI Forwarding Platform: networkbuilders.intel.com/solutionslibrary/intel-select-solutions-for-nfvi-forwarding-platform-v2

Intel Xeon® Scalable processors: intel.com/xeonscalable

Intel Ethernet: intel.com/go/ethernet

Red Hat and Intel – Cloud-Native Hybrid-Multicloud Platform (NFV): intel.com/content/www/us/en/big-data/partners/redhat/red-hat-openshift-hybrid-cloud-reference-architecture.html

Red Hat® OpenShift®: redhat.com/en/technologies/cloud-computing/openshift

[1] Statista, February 14, 2022. "4K Ultra HDTV household penetration in the United States in 2019 and 2021." https://www.statista.com/statistics/1247334/4k-ultra-hdtv-us-household-penetration/.

[2] Strategy Analytics, April 7, 2021. "8K TV Owning Households to Reach 72 Million Worldwide by 2025." https://www.businesswire.com/news/home/20210407005309/en/Strategy-Analytics-8K-TV-Owning-Households-to-Reach-72-Million-Worldwide-by-2025.

[3] 3rd Gen Intel Xeon Platinum 8380 CPU: 8 channels, 3200 MT/s (2 DPC) vs. 2nd Gen Intel Xeon Platinum 8280 CPU: 6 channels, 2666 MT/S (2 DPC).

[4] 3rd Gen Intel Xeon Platinum 8380 CPU: 8 channels, 2 DPC (256 GB DDR4) vs. 2nd Gen Intel Xeon Platinum 8280 CPU: 6 channels, 2 DPC (128 GB DDR4).

[5] Performance varies by use, configuration, and other factors. See [91,92] at https://www.intel.com/3gen-xeon-config.

0822/RKM/MESH/349359-001US