WHITE PAPER–SEPTEMBER 2021

vSphere Performance Equivalent to Bare Metal for RAN Workloads

Tests Show Speed on ESXi 7.0U3 Matches Physical Hardware for Radio Access Networks

Table of Contents

THE SYNERGY OF CONTAINERS AND VIRTUAL MACHINES

VMs solve infrastructure-related problems by better utilizing servers, improving infrastructure management, and streamlining IT operations.

Containers solve application-related problems by streamlining DevOps, fostering a microservices architecture, improving portability, and further improving resource utilization.

Running containers on VMs produces a synergy that helps CSPs transition from 4G to 5G networks with ease.

BENEFITS OF HYPERVISORS AND VIRTUAL MACHINES FOR CNFs

- Onboard, deploy, and manage CNFs at scale through automation
- Establish strong security boundaries for containers
- Isolate workloads and apply built-in security measures like micro-segmentation
- Select the best Linux kernel version for your workload
- Optimize the performance of large Kubernetes clusters and mixed workloads on shared infrastructure
- Automate lifecycle management of Kubernetes clusters, RAN functions, and 5G services
- Optimize the placement and performance of CNFs with programmable resource provisioning
- Scale CNFs without the pain of adding, configuring, and managing physical hardware
- Streamline operations and reduce OpEx

RAN Workload Performance Is Equivalent on Bare Metal and vSphere

Is there a performance tax for real-time RAN workloads on VMware vSphere? The answer is no. VMware ran industry-standard real-time micro-benchmarks, namely cyclictest and oslat, to compare the performance of RAN workloads on VMware vSphere and bare metal and found that performance is equivalent.

The tests show that there is no performance penalty or latency tax with VMware vSphere 7.0 Update 3. The performance of radio access network workloads on VMware vSphere 7.0U3 vs. bare metal, as measured by the real-time micro-benchmarks cyclictest and oslat, is equivalent.

Cyclictest, which uses a hardware-based timer to measure platform latency and jitter, demonstrated that the latency on both vSphere 7.0U3 and on bare metal was less than 10 microseconds. A 10-microsecond latency is well within the latency requirements of RAN workloads.

The oslat performance test is an open-source micro-benchmark that measures jitter in a busy loop. Instead of using hardware-based timers, this benchmark uses a CPU bound loop as its measurement—which emulates a virtualized RAN workload in a real-world scenario, such as a polling thread using the Data Plane Development Kit (DPDK).

The following charts show the test results on bare metal and on vSphere 7.0U3 for the two tests. The results are equivalent (in the error range). Photon RT was used for both the bare metal and vSphere tests, with the same configurations, to create a fair comparison.

Equivalent cyclictest results

Here are the results for the cyclictest. The gap measured by cyclictest is the x-axis latency for this graph, and the y-axis is the number of samples at that latency.

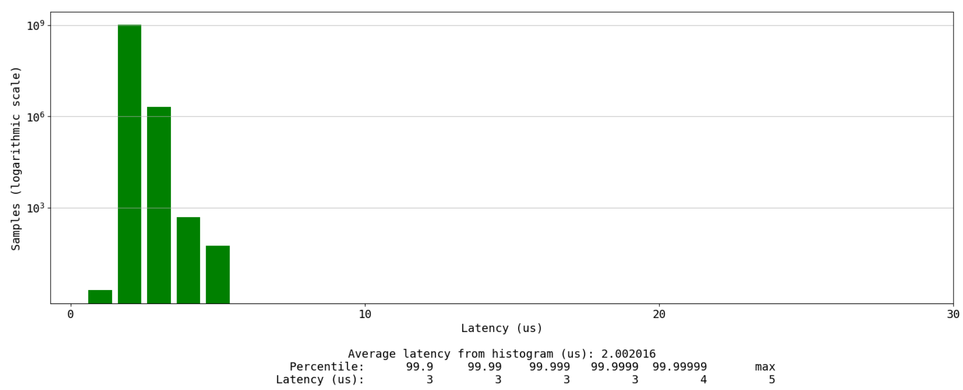

Figure 1: Cyclictest results for bare metal.

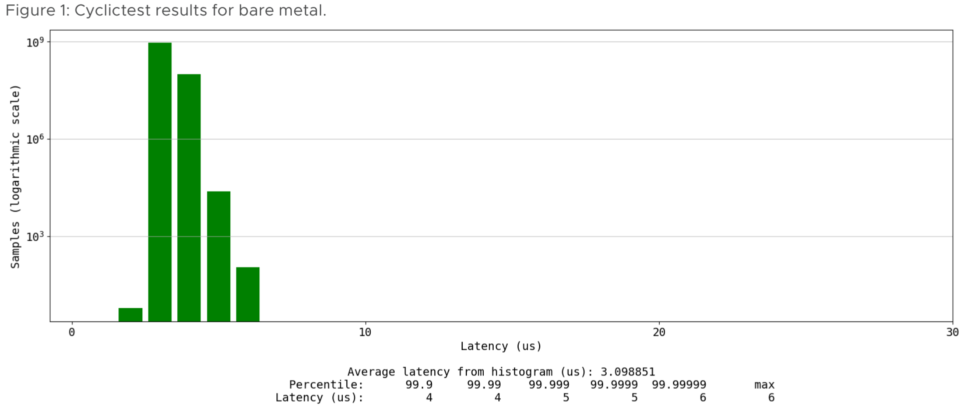

Figure 2: Cyclictest results for vSphere 7.0U3.

Since cyclictest is merely measuring the virtualization overhead of a timer at this point, we wanted to use a tool that would more accurately measure the amount of time that guest execution is interrupted.

Equivalent oslat test results

Short for operating system latency, oslat comes with cyclictest as part of the Linux real-time tests package. The oslat micro-benchmark runs a busy loop that continuously reads the real time. Anything that interrupts guest execution pauses the busy loop and shows up as a gap in real time. Since the oslat test doesn't use timers, there is no timer virtualization overhead.

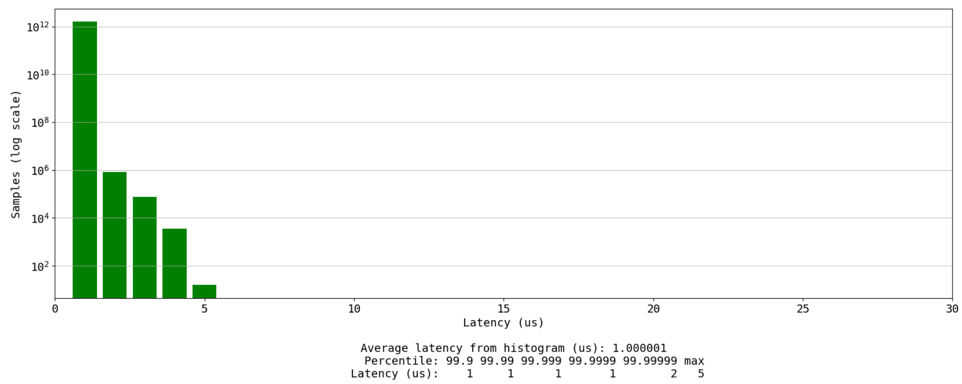

Figure 3: Oslat micro-benchmark results for bare metal.

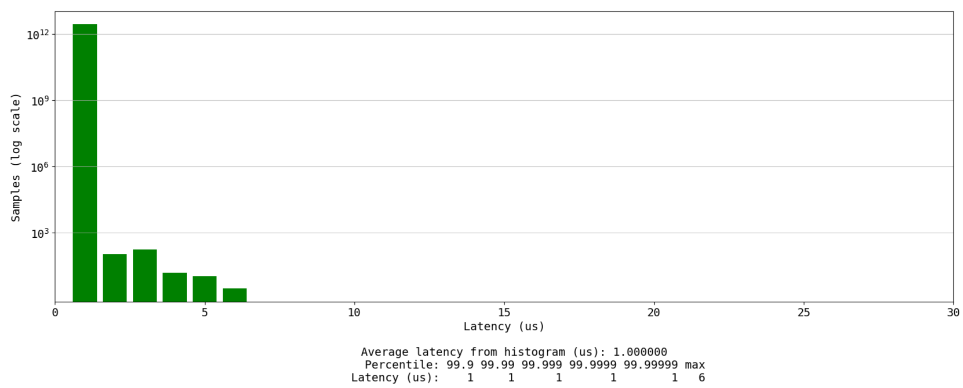

Figure 4: Oslat micro-benchmark results for vSphere 7.0U3.

In the chart, the real-time gap measured by oslat is the x-axis latency for this graph, and the y-axis is the number of samples at that latency. The graph shows the latency distribution of the oslat test run on bare metal and on vSphere 7.0U3, both with Photon 3.0 RT.

The oslat benchmark is a better representation of a real-world virtualized RAN workload because the tests use a poll mode driver to read and write packets to the NIC that connects to the cell tower.

Test configuration

Here are the configurations that were used to run the cyclic and oslat tests at VMware.

The tests used the following server models:

- Dell R640/R740
- Dell VEP4600

The PTP NICs for virtualized RAN are Intel 710 and 810.

Here are the configurations for the firmware and BIOS:

- CPU Power and Performance: Performance
- Configurable TDP: Nominal
- Turbo Boost: Enabled

VMware ESXi 7.0U3 is required. This version of the VMware hypervisor is optimized for virtualized RAN workloads. It includes an enhanced tail latency profile for cyclictest and support for PTP SR-IOV.

Virtual machine configuration

Optimizations in the vSphere CPU scheduler for NUMA architectures ensure that RAN workloads run on VMs with high performance. To optimize vSphere for RAN workloads, we configured some advanced options for the virtual machines. (Information on *setting advanced options for virtual machines* is in the vSphere 7 documentation).

Some of the following values and settings depend on the virtual hardware version in use, and these values and settings are shown with variables.

First, we configured manual VM exclusive pinning:

```
sched.vcpu0.affinity=x # where x is the pcpu number
sched.vcpu1.affinity=y # where y is the pcpu number
sched.vcpu2.affinity=z # where y is the pcpu number
...
# repeat for all vCPUs of the VMs
sched.cpu.affinity.exclusive=TRUE
# mark the above CPU affinity as exclusive
sched.cpu.affinity.exclusiveType=loose
# set the above line if you are pinning two vcpus to a pair of
physical HT twins
```

Second, we configured manual VM memory optimization:

```
sched.mem.min=MEM_SIZE # where MEM_SIZE is the configured memory
size of the VM, or via GUI
sched.mem.pin=TRUE
sched.mem.prealloc=TRUE
sched.mem.prealloc.pinnedMainMem=TRUE
...
sched.mem.lpage.enable1GPage = TRUE # this enables huge page
```

Third, we applied the following advanced settings for an enhanced tail latency profile. Note that these settings are not virtual-hardware-version dependent.

```
sched.cpu.affinity.exclusiveNoStats = "TRUE"
monitor.forceEnableMPTI = "TRUE"
timeTracker.lowLatency = "TRUE"
```

These settings ensure that among other things, vSphere optimizes high-performance workloads by scheduling processes into the same NUMA domain. To ensure high performance, there is no cross-domain context switching or memory access.

Important: With ESXi 7.0U3, the setting `sched.cpu.affinity.exclusiveNoStats = "TRUE"` makes a VM's usage appear to be fully busy at 100 percent per vCPU. It is therefore important to use in-guest stats for CPU utilization and usage evaluation.

Photon RT configuration for both vSphere and bare metal

Photon RT 3.0 was used on the bare metal server and on ESXi for both the cyclictest and oslat micro-benchmarks. Photon RT was configured the same on the bare metal and the ESXi hosts.

Photon OS provides a secure Linux runtime environment for running containers, including CNFs, and a real-time kernel flavor called 'linux-rt' to support low-latency RAN workloads.

linux-rt is based on the Linux kernel PREEMPT_RT patch set that turns Linux into a real-time operating system. In addition to the real-time kernel, Photon OS 3.0 supports several userspace packages such as tuned, tuna, and stalld. These userspace packages are useful to configure the operating system for real-time workloads. The linux-rt kernel and the associated userspace packages together are referred to as Photon Real Time (RT).

Install Photon RT by downloading the latest version of the real-time ISO at, for example, the following URL:

https://packages.vmware.com/photon/3.0/Rev3/iso/photon-rt-3.0-a383732.iso

The *landing page for downloading Photon OS*, including Photon RT, is on GitHub.

We tuned Photon RT on both the bare metal machines and the virtual machines to isolate the CPUs that are running the cyclictest and oslat measurement threads. The following is what is minimally needed to obtain the cyclictest and oslat benchmark numbers. More information on setting the following configuration changes is in the *Photon OS documentation*.

1. We ran the following command to update Photon RT to the latest kernel:

```
$ tdnf update
```

2. We set up the tuned realtime profile.

We edited /etc/tuned/realtime-variables.conf to set the range of the isolated CPUs:

Example: `isolated_cores=2-29`

`After setting the isolated cores, we turned on the tuned realtime profile:`

```
$ tuned-adm profile realtime
```

After these settings were applied, we needed to reboot twice for the isolcpus=2-29 to appear on the Linux kernel command line. It can be verified by running the following command:

```
cat /proc/cmdline
```

3. We set up additional Linux kernel boot parameters:

We edited /boot/grub2/grub.cfg and added "nohz=on nohz_full=2-29" to the end of the Linux boot command line.

Then we rebooted again, and then we ran the tests using the parameters shown below for cyclictest and oslat.

Keep in mind that the number of CPUs might be different, but the same number of test threads (one per CPU) on the same physical CPUs on the same physical socket were used

VMWARE TELCO CLOUD PLATFORM RAN AT A GLANCE

VMware Telco Cloud Platform RAN is a RAN-optimized version of the platform tailored to support the radio access network (RAN). The platform runs virtualized baseband functions, virtualized distributed units (vDUs), and virtualized central units (vCUs) in accordance with RAN performance and latency requirements. VMware Telco Cloud Platform RAN paves a clear path to RAN modernization by enabling CSPs to evolve from their traditional RAN to vRAN and, eventually, open RAN.

KEY BENEFITS AND CAPABILITIES

- Virtualize RAN functions on a horizontal platform optimized for the RAN using the Intel FlexRAN software reference design
- Optimize the placement of DUs and CUs through programmable resource provisioning
- Use the same common platform to virtualize the RAN now and migrate to open RAN in the future
- Deploy and operate both RAN and non-RAN workloads on a horizontal platform
- Transform the RAN into a 5G multi-services hub
- Use a security-hardened Linux host called Photon OS that is optimized for running containers on VMware vSphere®
- Automate lifecycle management of infrastructure, Kubernetes clusters, vRAN functions, and 5G services

on both bare metal and in the VM to make the comparison as equivalent as possible.

More information about *configuring the Photon real-time operating system* for real-time, low-latency workloads is in the Photon documentation.

Parameters for running the cyclictest and oslat micro-benchmarks

Here are the cyclictest and oslat parameters that VMware used to run the tests:

```
$ taskset -c 1-29 rt-tests-2.1/cyclictest -m -p99 -D 1h -i 100 -h 100
-a 2-29 -t 28 --mainaffinity=1
$ taskset -c 2-29 rt-tests-2.1/oslat -c 2-29 -f 99 -D 3600
```

Virtual Machines and Performance Management for CNFs on VMware Telco Cloud Platform RAN

VMware Telco Cloud Platform RAN is a RAN-optimized platform that runs virtualized baseband functions, virtualized distributed units (vDUs), and virtualized centralized units (vCUs) in accordance with RAN performance and latency requirements. The platform uses a telco-grade Kubernetes distribution to orchestrate containers on virtual machines in a telco cloud. Simply put, running containers on virtual machines helps CSPs speed up the transition from 4G to 5G and ease the management of CNFs and 5G services.

With VMware Telco Cloud Platform RAN, the Topology Manager optimally allocates CPU, memory, and device resources on the same NUMA node to support performance-sensitive workloads. VMware Telco Cloud Platform RAN also optimizes the performance of large Kubernetes clusters and mixed workloads.

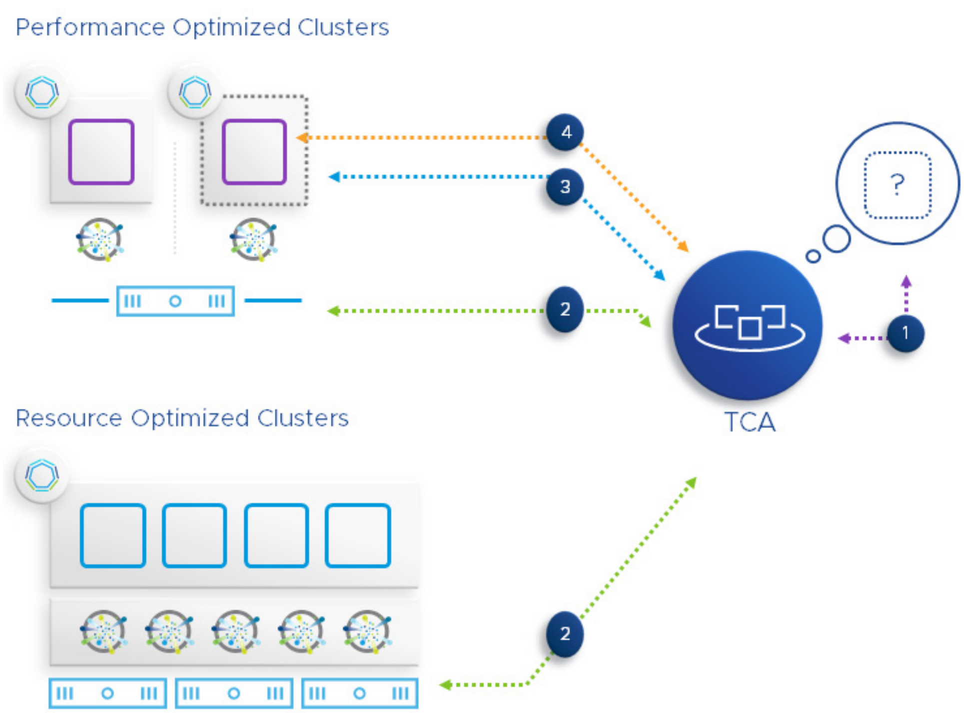

Figure 5: Programmable resource provisioning optimizes the placement of 5G services and CNFs to maximize resources and RAN performance through the following steps, which are illustrated by the numbered steps in the figure:
1. Assess a service's requirements.
2. Gauge the resources of Kubernetes and the hardware and infrastructure.
3. Deploy a performance-optimized Kubernetes cluster.
4. Place service on the cluster.

"Although containers are sometimes thought of as the next phase of virtualization, surpassing hardware virtualization, the reality for most organizations is less about revolution than evolution. Containers and hardware virtualization not only can, but very frequently do, coexist well and actually enhance each other's capabilities. VMs provide many benefits, such as strong isolation, OS automation, and a wide and deep ecosystem of solutions. Organizations do not need to make a choice between containers and VMs. Instead, organizations can continue to use VMs to deploy, partition, and manage their hardware, while using containers to package their apps and utilize each VM more efficiently."

APPLICATION CONTAINER SECURITY GUIDE, NIST SPECIAL PUBLICATION 800-190

Elasticity for balancing cost and performance

With virtualization, you can use vSphere's built-in elasticity of scale to balance cost and performance when you need to.

- Workloads can be migrated to maximize performance.
- Workloads can be consolidated to minimize resource costs.

The outcome is cost-effective performance: Workloads that need high performance get it, and workloads that don't need high performance can be run at a lower cost.

Centralized performance management with automation

Through dynamic resource allocation and *late binding*, VMware Telco Cloud Platform RAN optimizes workload placements. When the system instantiates a workload, it optimizes a Kubernetes cluster or creates a new one to match the CNF requirements through late binding. Here are some ways late binding optimizes worker nodes and their resources:

- Enable Huge-Pages
- Install the Photon OS real-time kernel
- Isolate Cores
- Configure TuneD
- Allocate vGPU

As for the components of a virtualized RAN, programmable resource provisioning optimizes where to locate DUs and CUs. When you onboard a virtualized RAN function, you can programmatically adjust the underpinning availability and resource configuration based on the function's requirements.

To meet high-performance, low-latency requirements, DUs can be placed at the far edge near users. CUs, meanwhile, can be automatically placed or dynamically moved closer to the core to maximize resource utilization. These late-binding capabilities let you dynamically move DU and CU resources on demand to improve resource utilization or to add more resources when necessary.

And you can do it at scale, from a centralized location, with automation. VMware Telco Cloud Platform RAN automates performance configurations and management for your RAN workloads.

Kubernetes and containers on bare metal or virtual machines?

For CSPs, performance, security, and management are key factors.

- *Studies show that optimizations* in the vSphere CPU scheduler for NUMA architectures quash the belief that running Docker containers on VMs comes with a performance tax. For more information, see the *performance study*.
- Noisy neighbor situations can cause interference for co-located containers on physical hardware, and cross-container interference can result from containers sharing the same kernel resources or components.
- Kubernetes on bare metal is unlikely to outperform Kubernetes on VMware vSphere, which uses advanced scheduling algorithms to optimize all workloads. A recent test of vSphere 7 with Kubernetes shows better performance compared with a bare-metal Kubernetes node because the VMware hypervisor does a better job at scheduling pods on the right CPUs, thereby reducing random memory accesses.

A *test of vSphere 7 with Kubernetes* shows better performance compared with a bare-metal Linux Kubernetes node. vSphere native pods, isolated by the hypervisor, can achieve up to *8 percent better performance* than pods on a bare-metal Linux Kubernetes node. For more information, see the *blog post* discussing the test results of the vSphere native pods.

"While network virtualization presents risk, it also allows advanced and flexible network protections. For this reason, a well-built virtualised network can be more secure and resilient than an equivalent network built on dedicated hardware."

SECURITY ANALYSIS FOR THE UK TELECOM SECTOR: SUMMARY OF FINDINGS, NATIONAL CYBER SECURITY CENTRE, JANUARY 2020

What's more: By using virtual machines, you can easily select the container host operating system that works best for the performance demands of your CNF. Photon OS is one option: It provides a secure runtime environment for running containers, including CNFs, and a real-time kernel for RAN applications.

And with VMware Telco Cloud Platform RAN, you can automate performance configurations and performance management for your telco workloads.

Boosting performance by selecting a Linux kernel version

With containers, an important factor is the version of the container host's kernel and its performance characteristics. Another performance advantage of running containers on virtual machines is that you can not only easily select the container host that you want to use but also maximize the CPU resources of the underlying hardware by running multiple virtual machines, each with its own choice of container host.

In other words, each virtual CPU allocated to each virtual machine on vSphere can run a different kernel; a bare metal server, however, can run only one kernel.

By using virtual machines, you can easily select the container host operating system that works best for the performance demands of your CNF. On vSphere, one example is Photon OS, the security-hardened minimalist host operating system that was used for the cyclictest and oslat micro-benchmarks discussed earlier in this paper. Its Linux kernel is optimized for performance on vSphere, and it supports new devices such as ARM64 (Raspberry Pi 3) to, for instance, help enable Internet of things applications at edge sites.

Separating the container host operating system from bare metal by using VMs also eases the lifecycle management of those operating systems.

Accelerating workloads with DPDK

To architect the network for optimum application response times and scalability, VMware Telco Cloud Platform RAN uses the Data Plane Development Kit, or DPDK, an Intel-led packet processing acceleration technology. Capabilities include optimizations through poll mode drivers, CPU affinity and optimization, and buffer management.

Workload acceleration with SR-IOV

SR-IOV is a specification that allows a single Peripheral Component Interconnect Express (PCIe) physical device under a single root port to appear as multiple separate physical devices to the hypervisor or the guest operating system. SR-IOV uses physical functions (PFs) and virtual functions (VFs) to manage global functions for the SR-IOV devices. PFs are full PCIe functions that can configure and manage the SR-IOV functionality. VFs are lightweight PCIe functions that support data flow but have a restricted set of configuration resources. The number of virtual functions provided to the hypervisor or the guest operating system depends on the device. SR-IOV enabled PCIe devices require appropriate BIOS and hardware support, and SR-IOV support in the guest operating system driver or hypervisor instance.

Automation for RAN Workloads

VMware Telco Cloud Platform RAN automates the discovery, registration, and creation of Kubernetes clusters while enabling continuous synchronization between the CaaS layer and VMware Telco Cloud Automation. This synchronization creates constant Kubernetes cluster resource awareness, centralizes fault and performance monitoring, and optimizes workload placements.

Furthermore, during the workload instantiation process, if none of the available Kubernetes cluster profiles is suitable, the system will on-demand or automatically optimize an existing cluster or create a new one to match the cloud-native network

TELCO-GRADE KUBERNETES

The CaaS functionality of VMware Telco Cloud Platform simplifies the operation of Kubernetes for multi-cloud deployments, centralizing management and governance for clusters. The platform provides telco-grade CaaS enhancements, such as the following:

- Multus to attach multiple container networking interfaces to Kubernetes pods through its plugins
- Topology Manager to optimally allocate CPU memory and device resources on the same NUMA node to support performance-sensitive applications
- Kubernetes cluster automation to simplify deployments and management of Kubernetes master and worker nodes.

With these enhancements, CSPs can take advantage of a telco-grade Kubernetes platform to address emerging 5G use cases at the RAN.

MULTI-LAYER LIFECYCLE MANAGEMENT AUTOMATION

- VMware Telco Cloud Platform RAN lets CSPs centrally manage and automate the virtualized architecture, from CaaS to network services.
- Application management (G-xNFM) unifies and standardizes network function management across the VM and container-based infrastructure.
- Domain orchestration (NFVO) simplifies the design and management of centralized or distributed multi-vendor network services. CSPs can onboard VNFs and CNFs using standard-compliant TOSCA templates.
- The multi-cloud infrastructure and CaaS automation ease multi-cloud registration (VIM/Kubernetes), enable centralized CaaS management, and synchronize multi-cloud inventories and resources. Kubernetes clusters can be created and optimized automatically to align with the requirements of network functions and services.

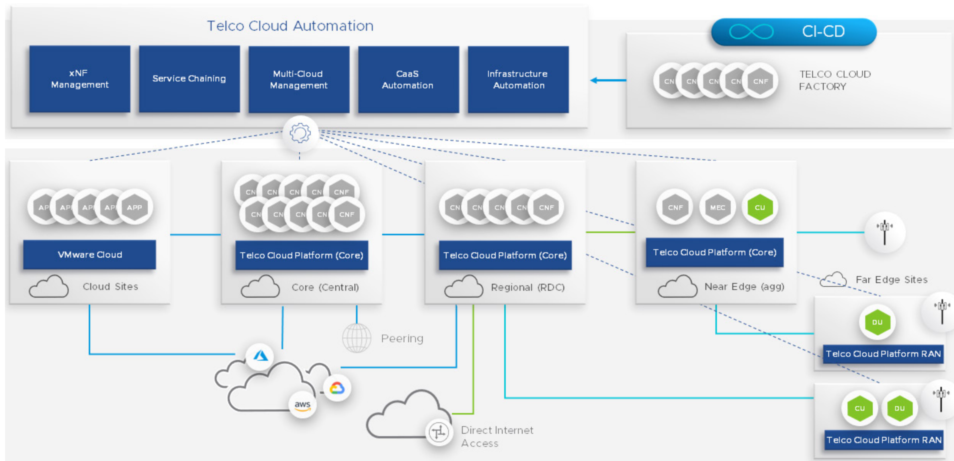

Figure 6: The VMware Telco Cloud for deploying, automating, and managing a 5G network, from the core to the radio access network.

function's requirements through late binding configurations such as the operating system. Achieving this kind of automation and dynamic optimizations using bare metal is likely to be difficult and expensive.

The platform enables CSPs to automate the onboarding and upgrading of network functions and infrastructure components with zero-touch provisioning. Full lifecycle management can apply policies to automate deployments, operations, and performance.

Infrastructure Management, IT Operations, and Lifecycle Management

Running containers on physical hardware would resurrect difficult infrastructure management and operational problems. Kubernetes can manage containerized functions and services, not the underlying infrastructure on which they are running. If you were to choose to use physical hardware as the underlying infrastructure, you must address a number of requirements while avoiding the creation of difficult-to-manage silos:

- Infrastructure deployment and configuration
- Patching, updating, and upgrading
- Backup and disaster recovery
- Logging and monitoring

Multiple infrastructure silos can duplicate teams, tooling, and processes. Instead of driving focused innovation on a single platform, IT ends up repeatedly performing the same tasks.

By running containers on physical hosts, many of the old problems that virtualization solved would come back to plague IT at the same time that IT is under pressure to increase agility, help accelerate time to market for new 5G services, improve security, and minimize costs—all without increasing complexity and risk. These are now core IT requirements, or rapidly becoming new requirements. But as heterogeneous cloud services enter the telecommunications landscape, IT is finding it more and more difficult to fulfill these requirements.

VMware Telco Cloud Platform RAN solves these problems with a comprehensive, flexible solution that uses the power of automation to deploy and manage multiple Kubernetes clusters as well as to patch the container host operating system. Multi-layer lifecycle management automates the provisioning and management of RAN components, including virtual machines, to streamline operations and reduce costs.

Conclusion

There is no performance tax for real-time RAN workloads on VMware vSphere. VMware tests show that the performance of benchmarks that mimic real-world RAN workloads on bare metal and vSphere 7.0U3 is equivalent.

In addition to performance, operating CNFs in production requires resource management, lifecycle management, and automation—all of which are an integral part of VMware vSphere and VMware Telco Cloud Platform RAN.

LEARN MORE

For more information about VMware Telco Cloud Platform RAN, call 1-877-VMWARE (outside North America, dial +1-650-427-5000) or visit *https://telco.vmware.com/*