

Tuning Telco Cloud Platform 5G Edition for Data Plane Intensive Workloads

VMware Telco Cloud Platform - 5G Edition 2.0

You can find the most up-to-date technical documentation on the VMware website at:

<https://docs.vmware.com/>

VMware, Inc.
3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

Copyright © 2021 VMware, Inc. All rights reserved. [Copyright and trademark information.](#)

Contents

- 1** About the Telco Cloud Platform 5G Edition Performance Tuning Guide 4
- 2** About Telco Cloud Platform 5G Edition 5
- 3** Introduction to 5G 7
- 4** 5G Data Plane Intensive CNFs 9
- 5** Performance Best Practices for 5G Data Plane Intensive CNFs 10
 - Infrastructure Layer 10
 - Physical Compute 11
 - Physical Network 11
 - Virtual Compute 12
 - CaaS Layer 12
 - Workload Cluster for Data Plane CNFs 14
 - CPU pinning for Data Plane CNFs 14
 - Data Plane Acceleration Using DPDK 16
 - Secondary Network for Data Plane Pod 17
- 6** Conclusion 21
 - Summary of Recommendations 21
 - References 23

About the Telco Cloud Platform 5G Edition Performance Tuning Guide

1

This guide focuses on the performance tuning best practices for cloud-native data plane intensive workloads in VMware Telco Cloud Platform™ — 5G Edition. It covers aspects that are directly related to data plane acceleration for 5G cloud-native workloads. Each data plane intensive workload has at least one data plane network interface. Management and control planes use non-accelerated network interfaces and are out of the scope of this guide.

Note This guide is part of [VMware Telco Cloud Platform 5G Edition Release 2.0](#).

Intended Audience

This guide is intended for customers who handle the Cloud-native Network Functions (CNFs) and VMware Telco Cloud Platform on which the CNFs run. Other users such as solution architects, technical sales engineers, field consultants, test engineers, and advanced services specialists in the telecommunication industry can also use this guide.

About Telco Cloud Platform 5G Edition

2

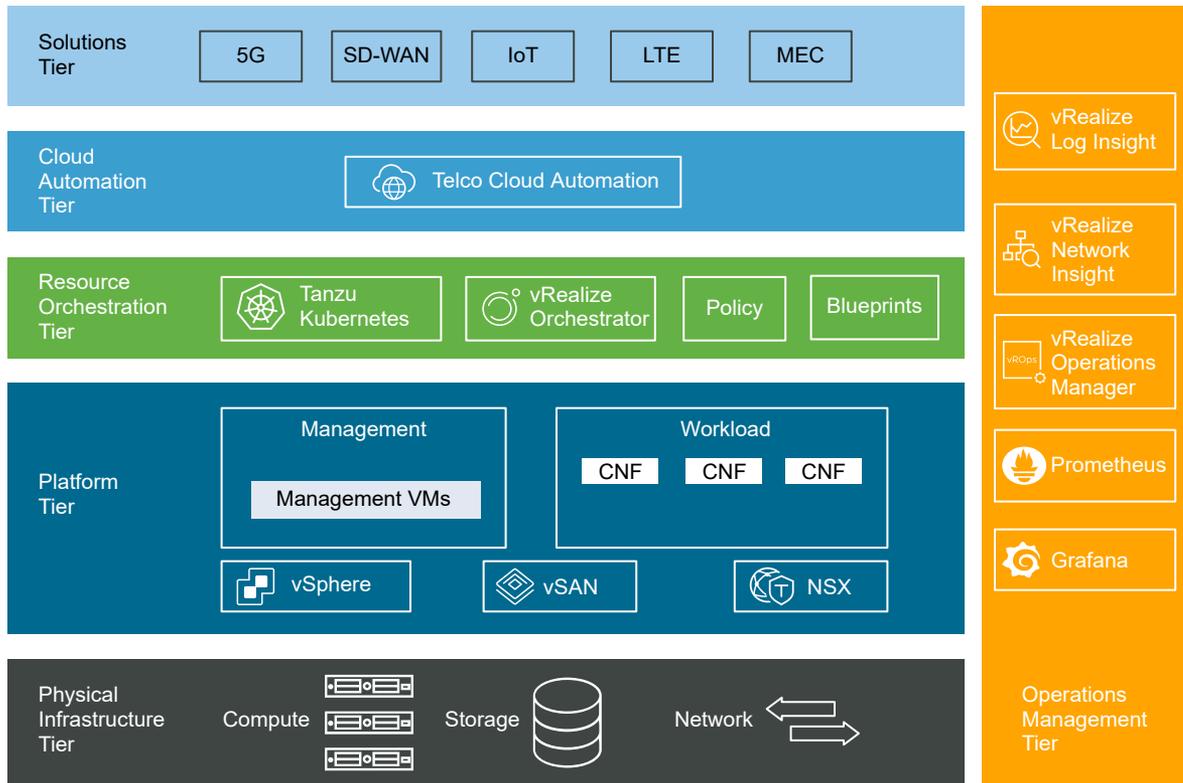
VMware Telco Cloud Platform™ — 5G Edition is a cloud-native platform powered by field-proven VMware Telco Cloud Infrastructure™ coupled with automation. It provides a cloud-first approach that delivers operational agility for virtual, cloud-native 5G, and Edge network functions. Telco Cloud Platform helps Communication Service Providers (CSPs) expedite their innovation cycle to deliver new applications and services. It helps reduce operational complexities, and achieve substantial TCO savings over other approaches, further accelerating CSPs' cloud modernization journey to 5G.

Using Telco Cloud Platform 5G Edition, you can design and create a Telco cloud as a complete solution for the deployment and operation of VNF and CNF workloads from core to cloud to edge. For more information, see the [VMware Telco Cloud Platform 5G Reference Architecture Guide](#).

The Telco Cloud Platform5G reference architecture guide describes the high-level physical and virtual infrastructure, networking, and storage elements that include infrastructure, automation, and operations layers.

Important Refer to the [Telco Cloud Platform 5G Reference Architecture Guide](#) to understand the Telco Cloud Platform components.

Figure 2-1. Layers of Telco Cloud Platform 5G Edition



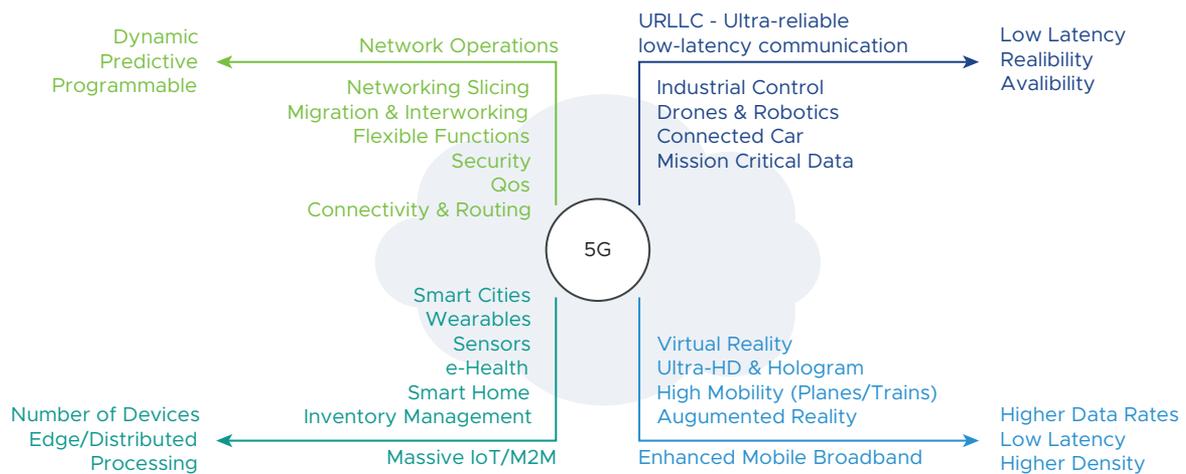
Introduction to 5G

3

The fifth-generation (5G) wireless network technology is designed to meet the challenging system and services requirements of the existing and emerging applications. The main goals of 5G are to improve capacity, reliability, and energy efficiency, while reducing latency and massively increasing connection density.

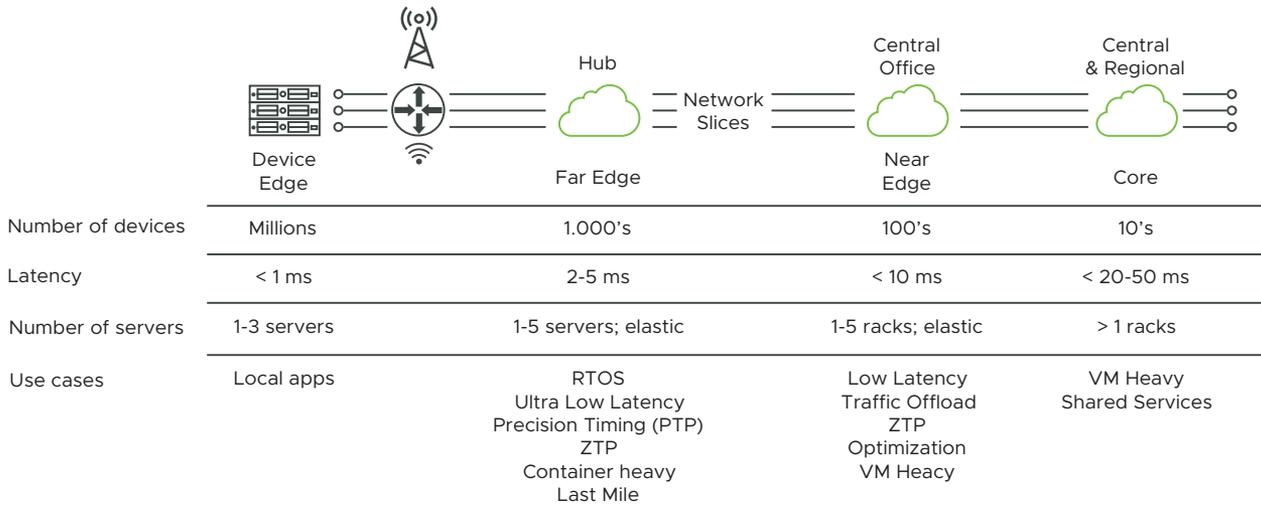
Modern applications such as tele-medicine, smart cities, augmented and virtual reality, and so on, mandate new specifications for throughput, reliability, end-to-end (E2E) latency, network density, and network robustness.

Figure 3-1. 5G Use Cases



With these modern applications and the ever-increasing data traffic in the CSP network, 5G networks are required to support stringent criteria for reliability, throughput, latency, and jitter.

Figure 3-2. 5G Network Requirements



5G brings groundbreaking improvements in the radio access technology to support modern applications. 5G also mandates cloud-native principles in the core and edge to provide a flexible and scalable allocation of network resources to different services and applications.

5G New Radio and the agility enabled by cloud-native principles in placing the data plane network functions closer to the user helps achieve exceptional low latency. To support the ever-increasing data throughput, the data plane network functions can be scaled by deploying more cloud resources. However, to reduce the Capital Expenditures (CAPEX), each data plane network function in the 5G network must provide high throughput and low latency with an optimal allocation of cloud resources.

In this guide, you can learn how to use VMware Telco Cloud Platform to support the cloud-native data plane intensive 5G workloads and meet the 5G network requirements.

5G Data Plane Intensive CNFs

4

The Service-Based Architecture (SBA) in 5G uses Control and User Plane Separation (CUPS). CUPS is essential in 5G to allow low latency for modern applications. 5G network architecture has centralized control plane network functions such as AMF and SMF, and distributed user plane network functions such as UPF. In a cloud-native approach, these network functions are realized using container-based microservices and are called Cloud-native Network Functions (CNFs). All the user plane CNFs in the 5G network are referred to as data plane intensive CNFs in this guide.

VMware Telco Cloud Platform 5G Edition reference architecture lays out the foundation for a high performant and efficient cloud platform for all types of CNFs. This performance tuning guide focuses on the performance best practices for data plane CNFs such as UPF in the 5G core network.

Performance Best Practices for 5G Data Plane Intensive CNFs

5

The performance of the data plane intensive CNF in a host depends on the compute and network capacity allocated to the CNF. The compute and network capacity requirements of the data plane intensive CNFs can vary based on the type of CNF.

- **Network-bound CNFs:** Some data plane intensive CNFs do simple packet forwarding. These CNFs are network bound and their throughput is limited by the allocated network capacity in the host.
- **CPU-bound CNFs:** In addition to packet forwarding, other data plane CNFs perform Deep Packet Inspection (DPI), filtering, heuristics, and so on. These data plane CNFs are CPU bound and their throughput is limited by the allocated compute capacity in the host.

The performance best practices for the data plane CNFs focus on the compute and network components in the following logical layers of VMware Telco Cloud Platform 5G Edition.

- **Infrastructure Layer** includes the physical compute, physical network, and the virtual compute provided by VMware vSphere®.
- **CaaS Layer** includes the Container-as-a-Service (CaaS) layer built on top of the Infrastructure layer using VMware Tanzu™ Standard for Telco.

This chapter includes the following topics:

- [Infrastructure Layer](#)
- [CaaS Layer](#)

Infrastructure Layer

The infrastructure layer must be set up correctly to support high data throughput and low latency for the data plane intensive CNFs.

Note VMware Telco Cloud Platform 5G Edition is built with vSphere as its foundation. The infrastructure layer concepts and best practices that are described in the previous [VMware vCloud NFV Performance Tuning Guide](#) remain valid for this Telco Cloud Platform 5G Edition Performance Tuning Guide.

Physical Compute

The physical compute infrastructure that you set up must support a large compute capacity for data plane intensive CNFs. The host BIOS must also have a high-performance CPU configuration to drive maximum performance per CPU core.

When you set up a high performing physical compute infrastructure, consider the following best practices:

- **Choice of CPU:** Use CPUs with more cores and high CPU speeds to achieve improved overall performance. Use CPUs with large L3 cache size. L3 cache is shared among all the cores of a CPU.
- **NUMA Node Interleaving:** Ensure that this setting is disabled in the BIOS of a multi CPU socket server. When enabled, the hypervisor sees the available memory from multiple sockets of a host as one contiguous area, and may store the memory pages in the remote socket memory.
- **Hyperthreading:** Enable Hyperthreading for CPU in the BIOS, if it is supported. Hyperthreading allows a single physical CPU core to execute two independent logical threads simultaneously, and hence increases the number of CPU cores available to the CaaS layer. The two logical threads of a physical core share resources such as memory caches and execution units. BIOS providers might refer to the logical threads as Logical Processors. For more details, see the [VMware vSphere 7.0 documentation](#).
- **High Performance Profile:** Set the Performance profile in BIOS to High or Maximum Performance. This setting ensures that the CPU does not throttle down to preserve power. Hence, the CPU cycles are always available to workloads.
- **C-States and P-States:** Disable these CPU states manually in BIOS, if the High Performance Profile is not available in BIOS or the CPU states are not turned off automatically. These power-saving CPU features can impact performance.
- **Turbo Boost:** Enable Turbo Boost in BIOS to operate the CPU faster than the rated frequency for peak loads. For more information about Turbo Boost, see the [Frequently Asked Questions](#) on the Intel Turbo Boost Technology page.

Note Some BIOS implementations may not allow control or modification of these settings. For information about specific BIOS implementation, see the corresponding BIOS documentation.

Physical Network

The underlying physical network infrastructure must support high network throughput and low network latencies on the data path of the data plane intensive CNFs.

Consider the following best practices when you set up a high performing physical network infrastructure:

- **Choice of NIC:** Before selecting a NIC type, you must have a thorough knowledge of the capabilities and performance of that NIC. NIC throughput, host PCIe speeds, and CPU offload impact the performance of the data plane intensive CNFs. For information about NICs that support the SR-IOV mode, see the VMware Compatibility Guide.

Important SR-IOV bypasses the ESXi hypervisor. Hence, you must ensure that the chosen NICs are supported by the 5G data plane CNF vendor.

- **Server Homogeneity:** The server model and configurations used for data plane CNFs must be identical. For example, the same NIC and NIC firmware must be used in all hosts used for data plane CNF.
- **NUMA Node Design:** If multi-socket servers are used for data plane intensive CNFs, design all the Non-Uniform Memory Access (NUMA) nodes identically. For example, if the data plane in a single NUMA uses two NICs, other NUMA nodes must also use two NICs for the data plane applications. This guideline applies to other components that have NUMA locality such as memory and CPU. It does not apply when using single-socket servers.
- **Data Plane Traffic Path:** The speed of any network is determined by its slowest link. To drive the maximum performance out of VMware Telco Cloud Platform that runs the data plane intensive CNFs, the NICs, Top-of-Rack (ToR) switches, and End-of-Row (EoR) devices must support high data plane performance. Ensure that you reserve the NIC ports and ToR switch ports for the data plane traffic and isolate them from the control and management plane traffic.

Virtual Compute

Virtual compute is a virtual machine (VM) or virtual computer that runs an operating system and applications, similar to a physical computer. VMware ESXi, the hypervisor within the vSphere environment, serves as a platform for running VMs and allows the consolidation of computing resources.

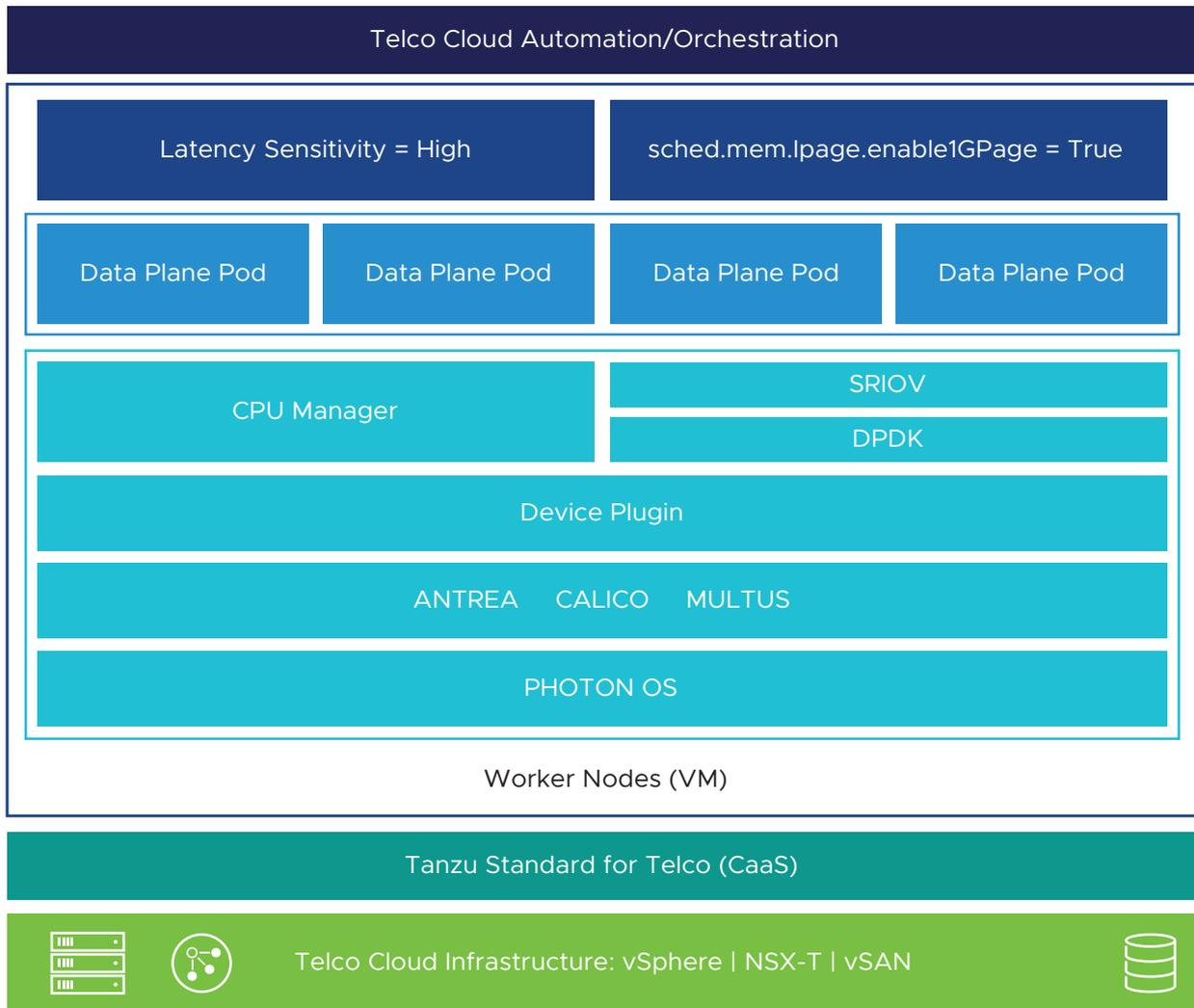
VMs are the building blocks of the CaaS layer within VMware Telco Cloud Platform, so the VMs that host the data plane intensive CNFs must be tuned correctly for high throughput and low latencies. For best practices about VMs within the CaaS layer, see the [CaaS Layer](#) section.

CaaS Layer

Within the Container-as-a-Service (CaaS) layer, VMs with VMware Photon OS as the guest OS are used as container hosts to provision Tanzu Kubernetes clusters. The container hosts that host the CNFs are commonly called Worker Node VMs or Worker Nodes. You must apply several compute and network best practices directly to the Worker Nodes to meet the performance requirements of the data plane CNFs.

Most of the VM best practices described in the previous [vCloud NFV Performance Tuning Guide](#) remain valid for the Worker Nodes, but they must be revisited in terms of the CaaS layer provisioning. In Telco Cloud Platform 5G Edition, VMware Telco Cloud Automation™ is used to configure the performance requirements of the data plane CNFs during CNF onboarding. These performance requirements include the Worker Node (VM) specification, Photon OS customization, networking configuration, CPU affinity, and NUMA considerations.

Figure 5-1. Data Plane Reference Stack of Telco Cloud Platform



A CNF vendor can use the Cloud Service Archive (CSAR) package of the CNF to specify performance requirements on the CaaS layer. VMware Telco Cloud Automation customizes the Worker Node configuration based on those requirements. For more details, see [Customizing Network Function Infrastructure Requirements](#) in the Telco Cloud Automation documentation.

Workload Cluster for Data Plane CNFs

CNFs are installed in the Tanzu Kubernetes Workload Cluster within the CaaS layer. When you deploy the Workload Cluster, consider the performance aspects of data plane intensive CNFs and assign maximum compute resources to the Worker Nodes.

Apply the following best practices when you create Workload Cluster templates using Telco Cloud Automation. For more information about how to create a Workload Cluster template, see the [VMware Telco Cloud Automation documentation](#).

Dedicated node pool for data plane intensive CNFs: Create a dedicated Node Pool for hosting data plane CNFs. Data plane CNFs have a different set of infrastructure requirements such as CPU pinning for data plane containers, SR-IOV, DPDK, and so on. A Node Pool groups the worker nodes that can support these requirements. For more information, see [Add a Node Pool](#) in the Telco Cloud Automation documentation.

Worker Node Size: The dedicated node pool for the data plane CNFs must use maximum compute resources available in the vSphere. Assign the number of CPU cores to a Worker Node such that the Worker Node can be deployed in a single NUMA node. This helps avoid the NUMA misalignment for the Worker Node and the associated performance impact.

Note For information about NUMA, see [Optimizing Applications for NUMA](#) on the Intel page.

The size of the Worker Node is chosen based on the requirements of the data plane CNF. Some of the factors that you must consider to determine the size of the Worker Node are as follows:

- The size of a single data plane Pod
- The number of interfaces on the data plane Pod.
- The number of data plane Pods that can run in a single Worker Node.

The Worker Nodes consume CPU resources, and the usage of many small Worker Nodes increases the CPU cost. The Network Equipment Providers (NEPs) are commonly adopting the single large UPF deployment design in a NUMA node. This design is implemented by deploying a single large Worker Node in a NUMA node.

The number of CPUs assigned to the Worker Node also depends on the CPU pinning requirements for the data plane CNF. For more information, see the CPU Pinning for Data Plane CNFs section.

Node Pool CPU Manager policy: To enable static CPU reservation on the Worker Nodes, use 'Static' as the CPU Manager policy. This setting is required to configure exclusive CPU affinity for the Worker Node CPUs.

CPU pinning for Data Plane CNFs

Data plane intensive CNFs require CPU pinning to avoid CPU scheduling delays. CPU pinning also ensures that a deterministic CPU capacity is available consistently to the data plane CNF. The availability of CPU capacity leads to deterministic network performance, which is the desired outcome for data plane intensive workloads.

You must apply CPU pinning for data plane CNFs through two distinct levels in the CaaS layer:

- The Worker Node CPUs are pinned (with exclusive affinity) to the CPU cores (Physical or Hyperthreaded) in the ESXi Hypervisor.
- The data plane containers are pinned to the Worker Node CPUs.

This ensures that the data plane containers have exclusive access to the CPU cores and meet the low latency and high throughput requirements.

Note Depending on the characteristics of the data plane intensive CNF, you can pin the Worker Node CPUs to either Physical or Hyperthreaded CPU cores in the ESXi Hypervisor. Ensure that you do not assign all cores in a NUMA node, particularly NUMA node 0, to the Worker Node. Some compute must be available for ESXi and other system services.

Pin all Worker Node CPUs to Physical CPU cores

For the data plane intensive CNFs that are extremely sensitive to latency, set `LatencySensitivity=High` on the Worker Nodes. This setting enables CPU pinning for all Worker Node CPUs by applying exclusive affinity. It also enables complete core isolation for the pinned cores by leaving the hyperthread siblings idle in ESXi. CPU pinning and core isolation reduce the interference and improve the performance of the workloads running in the Worker Node.

When the `LatencySensitivity=High` setting is applied on the Worker Node, its CPU and memory resources are NUMA aligned. Because the hyperthread siblings remain idle in ESXi, the Worker Nodes in a single NUMA can only use half of the maximum number of the Logical cores available in a CPU.

Note The following node customization is required for applying `LatencySensitivity=High` configuration on the Worker Nodes:

- Add the node component `latency_sensitivity` with the value `high`

For more details, see [Node Customization](#) in the Telco Cloud Automation documentation.

Pin Worker Node CPUs to Hyperthreaded CPU cores

For CPU-bound data plane intensive CNFs that are not extremely sensitive to latency, pin the Worker Node CPUs to the Hyperthreaded CPU cores in ESXi. This ensures that the Worker Nodes in a single NUMA can use the maximum number of the Logical cores available in a CPU.

`LatencySensitivity=High` must not be set on the Worker Node when pinning its CPUs on the Hyperthreaded cores in ESXi. CPU pinning is enabled by applying exclusive affinity in ESXi. Core isolation is not completely applied because Hyperthread siblings must share hardware resources.

Note It is not mandatory and also not recommended to pin all the CPUs of the Worker Node to the Hyperthreaded CPU cores in ESXi. Leave some CPUs in a Worker node for its OS and Kubernetes tasks. These unpinned CPUs will use shared resources allocated by ESXi.

Node Customization in Telco Cloud Automation ensures that the CPU and memory resources of the Worker Node are fully reserved to apply exclusive affinity and they are also NUMA aligned in a multi-socket server.

Note The following node customizations are required for pinning Worker Node CPUs to the Hyperthreaded CPU cores in ESXi:

- Define the kernel argument `isoLcpus` under `kernel: kernel_args`
- Add the node component `isNumaConfigNeeded` with the value `true`

For more details, see [Node Customization](#) in the Telco Cloud Automation documentation.

Pin Data Plane Container to Worker Node CPUs

When you deploy the Workload Cluster with a dedicated Node Pool for the data plane intensive CNF, use the CPU Manager Policy as 'Static'. This setting enables 'Static' CPU management policy in the Kubernetes CPU Manager. Static policy is required to allow CPU affinity for the data plane containers.

For deterministic performance, the data plane containers must be exclusively pinned to the Worker Node CPUs. To apply this setting, you must define the container as part of a Guaranteed Pod and with integer CPU requests.

Data Plane Acceleration Using DPDK

Data Plane Development Kit (DPDK) is the preferred data path framework for most of the data plane intensive CNFs. DPDK allows significantly fast packet processing by polling the network devices in the user space.

The Worker Nodes in the Node Pool, which is dedicated to the data plane CNF, require customizations to support data plane acceleration with DPDK. These customizations are applied during the Node Customization in Telco Cloud Automation.

Huge Pages

Huge pages are mandatory for DPDK. Huge pages are blocks of memory whose size is larger than the regular memory page. Data plane intensive CNFs have a large memory footprint. Huge pages significantly reduce the number of page table entries that are managed by the Memory Management Unit (MMU) for the data plane CNFs. Hence, the amount of page faults and Translation Lookaside Buffer (TLB) misses are reduced significantly, and the performance of the data plane CNFs are improved.

Most CPUs support 2 MB and 1 GB huge pages. The size and the number of huge pages in the Worker Node must be selected based on the memory footprint of the data plane CNF. Use 1 GB huge pages in the Worker Nodes, whenever possible. When selecting the number of huge pages, you must also consider the memory reserved on the Worker Node.

When configuring 1 GB huge pages through Telco Cloud Automation, ESXi 1 GB large page configuration is automatically enabled on the Worker Node. ESXi large page is required to support huge pages within Worker Nodes. For more information about ESXi 1 GB large page, see [Backing Guest vRAM with 1GB Pages](#) in the vSphere documentation.

Note The following node customization is required for configuring huge pages in the Worker Nodes:

- Define the kernel arguments `default_hugepagesz`, `hugepagesz`, and `hugepages` with the recommended values under `kernel: kernel_args`

For more information, see [Node Customization](#) in the Telco Cloud Automation documentation.

DPDK Kernel Module

You can use the following two DPDK kernel modules for binding a network device with DPDK: `vfio-pci` and `igb_uio`. Unless the data plane CNF has a mandatory requirement for `vfio-pci`, use `igb_uio` for accelerating the data plane.

`vfio-pci` uses Input-Output Memory Management Unit (IOMMU) to provide a more secure userspace driver environment, but it causes notable performance impact to medium and large-sized data packets. When you customize the Worker Nodes to use `vfio-pci`, `viOMMU` is also enabled on those Worker Nodes. `viOMMU` is an IOMMU emulation and it impacts the performance. Hence, if the safety features offered by `vfio-pci` through `viOMMU` are not critical, use `igb_uio` to achieve the best data plane performance.

Note The following node customizations are required for configuring the DPDK kernel module in the Worker Nodes:

- Define the kernel module `dpdk` and its version under `kernel: kernel_modules`.
- Define the package `pciutils` under `custom_packages`.
- For the network devices that need to be bound to DPDK, define `dpdkBinding` with the value `igb_uio` under `network: devices`.
- If you are using `vfio-pci`, define the following additional kernel arguments under `kernel: kernel_args`:
 - `intel_iommu` with the value `on`
 - `iommu` with the value `pt`

For more information, see [Supported DPDK and Kernel Versions](#) and [Node Customization](#) in the Telco Cloud Automation documentation.

Secondary Network for Data Plane Pod

Each CNF consists of multiple Pods. The default network interface on the Pods is called Primary Network. The Primary Network can serve all types of communication including management,

Pod-to-Pod, Pod-to-Service, and External-to-Service communications. But, the Primary network is not suitable for creating a fast data path due to the following reasons.

- The Primary Network on the Pod must use a supported CNI plugin such as Calico or Antrea.
- The special data plane requirements such as consistent high throughput, low latency, and SR-IOV with DPDK cannot be met using Primary Network.
- The Primary Network traffic requires isolation from the heavy data plane traffic.

So, the data plane CNF Pod must have one or more Secondary Network interfaces for creating a dedicated and fast data path.

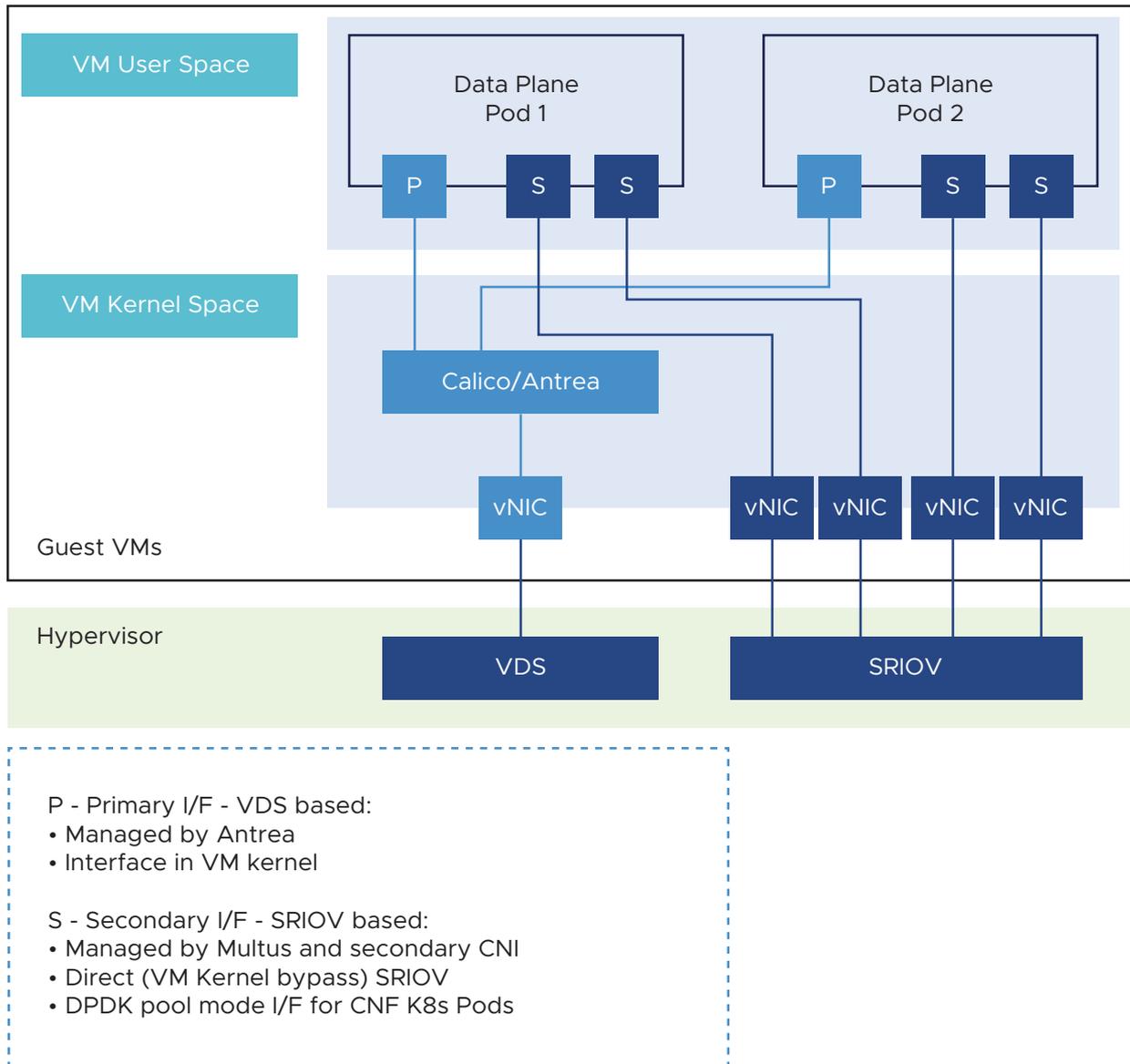
Telco Cloud Automation configures Multus CNI in the CaaS layer to enable Secondary Networks on the Pods. Ensure that the Primary and Secondary Networks on the Pod use different network interfaces on the Worker Node.

Secondary Network with SR-IOV

With VMware Telco Cloud Platform 5G Edition, you can use DPDK binding only with SR-IOV passthrough network device in the Worker Nodes. The SR-IOV passthrough network device in a Worker Node is backed by an SR-IOV Virtual Function (VF) in ESXi. Hence, the Secondary Network interface on the data plane Pods must support network attachment on SR-IOV VF.

With SR-IOV, the Worker Node and the physical NIC adapter bypass the VMKernel in the ESXi and exchange data directly. Bypassing the VMKernel for networking reduces the latency and improves the CPU efficiency for high data transfer performance.

Figure 5-2. Pod Networking Using SR-IOV



With SR-IOV enabled in the Infrastructure layer (BIOS and NIC), the SR-IOV usage model in the CaaS layer is CNF driven. Ensure that the data plane CNF has SR-IOV and DPDK requirements defined in its CSAR package. Telco Cloud Automation uses SR-IOV and DPDK requirements from the CSAR of the CNF and applies the following node customizations during CNF onboarding:

- Add the SR-IOV VF interfaces to the Worker Nodes.
- Bind the SR-IOV VF interfaces to DPDK in the Worker Nodes.
- Publish the DPDK bound SR-IOV VF interfaces as named allocatable network resources in the Kubelet by using the SR-IOV network device plugin.

The CNF uses an allocatable network resource (DPDK-bound SR-IOV VF interface) to create a Secondary Network attachment on the data plane Pod.

Ensure that you use VMware compatible NICs, firmware, and driver for SR-IOV. For information about compatible versions, see the [VMware Compatibility Guide](#).

Note The following node customizations are required for using Secondary Network with SR-IOV and DPDK on the data plane Pod:

- Add the following attributes for new network devices under `network: devices`:
 - `deviceType` with value `sriov`
 - `dpdkBinding` with value `igb_uio`

Other network device attributes such as `resourceName`, `networkName`, and `count` provide an easy network resource consumption model for the CNF. For more information, see [Node Customization](#) in the TCA documentation.

Important If you use SR-IOV, several virtualization features such as vMotion, High Availability, DRS, and Snapshots become unavailable. For more information, see [SR-IOV support](#) in the vSphere documentation.

Conclusion

6

To support the fast-evolving 5G application ecosystem, the transition from the traditional Network Functions to Cloud-native Network Functions (CNFs) has become mandatory. The data plane performance in the cloud platform continues to be one of the key considerations in the transition. With Telco Cloud Platform 5G Edition, VMware offers numerous capabilities to address the data plane performance considerations.

This performance tuning guide provides a holistic approach to both the physical and virtual domains that help you achieve accelerated performance for the data plane CNFs. VMware collaborates with several Network Equipment Providers (NEPs) to ensure that the data plane CNFs benefit from VMware Telco Cloud Platform.

This chapter includes the following topics:

- [Summary of Recommendations](#)
- [References](#)

Summary of Recommendations

The following tables summarize the data plane performance best practices described in this guide.

Infrastructure Layer

Performance Decision	Performance Justification	Performance Implication
Set the BIOS power profile to Maximum Performance.	The CPUs assigned to the data plane CNFs must not throttle down their speed.	CPU speeds are not throttled. High and consistent data plane performance is achieved.
Enable Hyperthreading.	CPU-bound data plane CNFs need more CPU cores to deliver high throughput.	The number of available logical CPU cores is increased, thereby improving the performance of the data plane CNFs. Note: You can always keep Hyperthreading enabled to allow shared resources among ESXi and system tasks.

Performance Decision	Performance Justification	Performance Implication
Enable Turbo Boost.	The data plane CNFs must be able to use the maximum CPU speeds available.	CPUs are overclocked to provide higher than base CPU speed for accelerating the data plane CNFs. Note: Ensure that the processor is working in the power and temperature specification limits of the thermal design power.
Disable NUMA Node Interleaving.	Remote memory access has a penalty on data plane latency and throughput.	The memory pages are placed local to the CPU, thereby avoiding remote memory access. This improves data plane latency and throughput.

CaaS Layer

Performance Decision	Performance justification	Performance Implication
Assign maximum compute in the host to Worker Nodes.	Data plane CNFs require large compute capacity.	Data plane CNFs use maximum compute available in the host and achieve high throughput.
Pin the Worker Node CPUs to the Physical CPU cores using LatencySensitivity=High.	CPU core pinning and complete CPU core isolation must be available to achieve low data plane latency.	Hyperthread siblings are disabled on the pinned CPU cores. Data plane CNFs use the isolated physical CPU cores exclusively to achieve low latency and high throughput.
Pin the Worker Node CPUs to the Hyperthreaded CPU cores.	CPU-bound data plane CNFs can compromise latency slightly by using Hyperthreading.	The number of available CPU cores for data plane CNFs increases, thereby improving the packet throughput.
Set the CPU Manager policy to Static.	Deterministic performance is required. CPU affinity and exclusivity must be available.	Allows the requested CPU affinity and exclusivity to the data plane CNFs.
Use integer CPU requests for the data plane CNF Pods.	Deterministic performance is required. CPU affinity and exclusivity must be available.	Data plane CNFs get the requested CPU affinity and exclusivity. CPUs are not shared, and deterministic performance is achieved.
Use 1 GB huge pages.	High performance with a large memory footprint.	The number of memory pages required for the data plane CNF is reduced. MMU operates efficiently and performance is improved.
Use the DPDK kernel module <code>igb_uio</code> for DPDK acceleration.	Userspace DPDK framework must be used for fast packet processing.	Unlike <code>vfio-pci</code> , <code>igb_uio</code> can operate without IOMMU emulation overheads. Any performance impact due to IOMMU emulation is avoided.

Performance Decision	Performance justification	Performance Implication
Use the Secondary Network on the data plane Pod.	Data plane interface must support high throughput, low latency, and advanced technologies such as SR-IOV with DPDK.	The Primary Network Pod traffic does not interfere with the data plane traffic on the Secondary Network. The data plane is accelerated using DPDK and SR-IOV.
Use SR-IOV for the Secondary Network on the data plane Pod.	The data plane interface must support high throughput and low latency.	SR-IOV eliminates all emulation layers in the data path. It delivers high throughput and low latency for the data plane traffic.

References

- [Tuning vCloud NFV for Data Plane Intensive Workloads.](#)
- [Performance Best Practices for VMware vSphere 7.0.](#)
- [Run CNFs on Virtual Machines to Optimize your 5G Networks.](#)