Critical NFVI KPIs to Validate

Examples from Jio and VoerEir

Version 7

Abstract

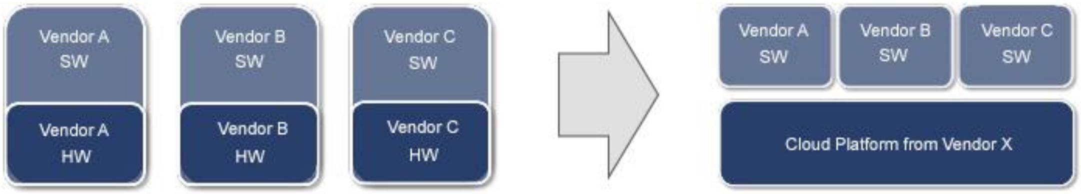
Jio and VoerEir have during the past year cooperated around defining a list of NFVI- KPIs which determine the performance of VNFs deployed on top of that NFVI. There are two prime use cases of these KPIs; First, for a VNF vendor to be able to commit to a performance SLA, the vendor must know the characteristics of the infrastructure on which his VNF shall be deployed. Second, if a VNF does not fulfill his performance SLA, the operator must be able to sort out where the responsibility lays, on the VNF or the NFVI vendor. The tricky part is to find the KPIs which are essential for different types of VNFs' performance. This white paper describes our way of reasoning when creating the list and gives examples of detailed definitions for different kinds of KPIs.

The contributors are listed at the end of the document.

Contents

1 Introduction

Network Function Virtualization (NFV) is the critical technology that is driving the transformation of Telecom Networks. The vision of NFV is to move Telecom Application Software from SW-vendor specific platforms, to run as pure software applications, on top of one or a few instantiations of a shared cloud infrastructure. It is this cloud infrastructure we call "Network Function Virtualization Infrastructure" (NFVI).



Picture 1

NFV promises to deliver:

- **Capex savings** by using general purpose hardware technologies and base infrastructure on Open Source SW
- **Opex savings** by reducing vendor specific complexity during deployment and operation
- **Speed in new service introduction** by applying cloud orchestration SW when deploying Telecom applications

NFV brings significant changes to how Telecom Networks will be deployed, with new challenges in how to secure performance SLAs for the Telecom applications, (now called VNFs). This leads to new requirements for testing. In the era of verticals, a telecom operator would just run tests to verify that the applications fulfilled the contractual agreed SLAs. If they did not, the operator would turn to the vendor and tell him: "*Fix it*." Now in the era of NFV, the operator becomes the middleman between the VNF vendors and the NFVI vendor. Moreover, operators need a strategy to determine responsibilities between his vendors.

The following statement serves as a base for NFV strategy for identifying performance responsibilities between NFV and VNF vendors; "*Verify that your NFV-Infrastructure fulfills its performance KPIs, and it becomes VNF vendors' responsibility to fulfill theirs*." The tricky part here is to find the NFVI performance KPIs that will determine the VNFs' performance. We call these a VNF's essential NFVI KPIs. These will be different for different VNFs, and the definition of these KPIs can be quite complex.

In this white paper, Jio and VoerEir describes the reasoning behind **some** of the KPIs that we propose our customers to use. The total amount of NFVI KPIs we propose is around 200. The purpose is to have a complete set of NFVI performance KPIs, which can serve as the contractual agreement between an NFVI vendor and an operator, as well as between an Operator and his VNF vendors. This set of KPIs shall enable the VNF vendor to predict his VNF's performance given a specific resource in the NFV cloud, and thereby enable him to commit to performance SLAs.

1.1 How this paper relates to The ETSI standardization

The ETSI NFV Test working group has released a set of specification on how an NFVI shall be tested. The specification that is relevant to this white paper is; "ETSI GS NFV-TST 009, Specification of Networking Benchmarks and Measurement Methods for NFVI". TST 009 describes how a benchmark shall be measured. E.g., how a throughput Benchmark is defined and how the Metric is expressed. TST 009 also mention a concept of Use Case, i.e., a description of the circumstances a benchmark is executed. These Use Cases are not defined in TST 009.

In this white paper, a KPI is defined by a Benchmark, following TST 009, and a Use Case. The white paper gives several examples of Use Case definitions and explains why these are essential for understanding how a VNF will perform on top of the NFVI under test.

2 Definition of KPI and SLA, and why we need both

This white paper is all about performance SLAs and performance KPIs. Let us, therefore, start with defining SLA and KPI in the context of NFVI performance validation:

KPI, Key Performance Indicator: Is a performance capability, measured on a not used system. I.e., the definition of the KPI, includes all activities on the system during the measurement. A KPI can be measured off service without a specific VNF.

SLA, Service Level Agreement: Is a "contract" on what the NFVI as a minimum shall be able to offer to the VNF in terms of Service characteristics, independent of what is happening on the system. An SLA can only be monitored in Service, under the specific circumstances a VNF is executed.

A VNF wants an SLA from the NFVI, but for the NFVI that is very difficult to give, because the VNF's behavior and deployment will have a significant impact on the characteristics of the NFVI.

Below we give some examples:

a. A large number of flows through one vSwitch will have an impact on routing table access time.
b. A high rate of new flow setup per second will give heavy load on vSwitch
c. Storage load from many VMs with large IO depth and many parallel Jobs to centralized storage will increase storage access delay times.
d. A VNF uses too much cache memory, or hold buffers from vSwitch to long, can lead to that vSwitch cache hit rate decrease.
e. If a VNF generates or receives bursty traffic, this can cause an overflow in vSwitch's quite small RX buffers.

Given that we don't define a VNF's behavior as a fault, the examples above all lead to a need for a more detailed definition of the essential NFVI KPIs for a specific VNF solution.
As described above, it is tough for an NFVI to give guarantee for an SLA, without taking the VNF solution into account. Therefore the "contract" between an NFVI and the VNFs should be based on agreed KPIs, and the sum of these KPIs shall include all typical "correct" VNF behaviors.

For a VNF solution, it is crucial to have its essential NFVI KPIs in the contract, and it is, therefore, crucial for a VNF solution to know its own essential NFVI KPIs and their values. We experience that some VNF vendors don't.

3 Some examples of NFVI KPI definitions

There are three usages of an NFV KPI:

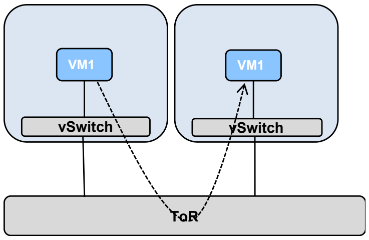
1. Comparing different vendors' NFVIs' capabilities
2. Verify that an installed NFVI implementation fulfills contractual promises
3. For a VNF vendor to be able to make contractual performance commitments based on the KPI values of the NFVI

To fulfill these expectations a KPI definition must be precise, and as different VNFs will have different KPIs essential for their performance, we will need to have a quite large amount of KPIs defined.

In this white paper, we will give a detailed definition of some KPIs, with the motivation of their definition from a VNF perspective. These KPIs are a subset of the 200 KPIs we recommend measuring and serve only as examples of what needs to be supported by the NFVI.

3.1 Plain L2 vSwitch DPDK performance.

This is a very straight forward KPI chosen here as it serves as a base for other more complex network-KPIs. As shown in picture 2, we measure throughput between two VMs placed in two different compute nodes, connected to the same Neutron Network.



Picture 2

This might seem very straight forward, but there are quite a few things that need to be defined before we have a clear definition of the KPI.

VM vSwitch connection technology: In this case, we choose to have a **DPDK** prepared VM with an IP stack in userspace. (Also Kernel-based IP stacks in VM is a relevant case for KPI as many VNFs are designed in that way)

VM – VM transport protocol: We use UDP and see no reason to use anything else as it does not add clarity in NFVI performance or KPI definition. However, from a strictly formal point of view, transport protocol need not be part of the KPI definition, as it is a question for measurement tool implementation.

As this is the very basic KPI, we will have few flows. However, we need some flows for the NFVI's vSwitch to take advantage of if several HW threads are allocated to it. We choose to send packets in **9 flows with equal distribution.** (9, so we have 3 ports in each VM all taking to each other)

It is possible for a VNF to have several queues for sending and receiving packets however that do not increase throughput, so we choose to have **one Queue for sending and one for receiving packets** in the test VM.

Packet size: We choose to use **64-byte frames**. I.e., very small packets. This enables us to measure vSwitch bottleneck for packets per second with a 10 G physical connection. For 40 and 100 G connections also frame size of 1512 bytes is relevant for KPIs. (Indeed, also Jumbo frames of, e.g. 2000 bytes are relevant as some VNFs work with tunnels in tunnels, which makes efficient handling of Jumbo frames relevant to capture in KPI definitions)

The accepted ratio of packets lost: We choose **2 Packets per million.** This means that the KPI measurement shall determine the best possible throughput from one VM to another when no more than 2 packets per million are lost. High-performance vSwitches are very sensitive for disturbance in execution, and therefore the throughput will often be significantly higher if we can accept a higher ratio of lost packets. There are two reasons why a high packet loss ratio cannot be accepted:

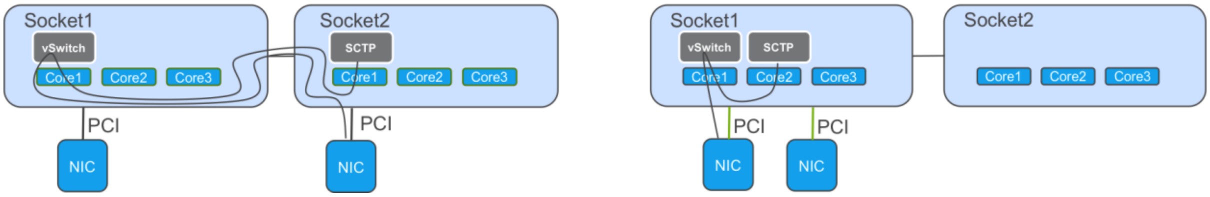
1. TCP connections end to end, where resending of packet windows will start lowering end-user throughput
2. Quality of, e.g. a UDP streaming connection is affected.

For both these cases, 2 ppm is very low — however, this KPI measure just one VM to VM connection. For an end user flow implemented with all nodes as VNFs, there will be many. Take, e.g. a video stream in an IMS session. It will first pass at least 3 different VNFs in the EPC solution of the sending mobile NW, and then it will pass at least 2 Nodes in the sending IMS NW. The Receiving NW will double that. However, that is not enough, and all these VNFs will have their internal load balancing structures making a packet pass a vSwitch up to 4 times per VNF. (To this we can, of course, add quite a few physical FW, Routers and switches in the transport NW and DC infrastructure). Taking all this into account 2 ppm packet loss might even be considered rather high.

Definition of stable throughput level: We choose that when running a specific fixed rate of sent packets, for 10 iterations of 60 seconds, 9 shall have packet loss ration < 2ppm. This is a necessary parameter in a definition as said before high-speed vSwitches are sensitive for disturbances. We have learned that this leads to significant variations in possible throughput. Sometimes a throughput of 4 Million packets per sec gives 0 ppm lost packets, while in other iteration even 3.5 Mpps will give 25 ppm lost. A KPI which defines best-achieved throughput is of little value for a VNF.

VM placement: We choose to **place both sending and receiving VM in the “best” NUMA node**. (The opposite, to place both in worse NUMA node is also relevant, as it determines what co-location strategy that is possible to have.)

Let’s explain this NUMA issue a bit further. With high-speed DPDK based VNFs, placement of VNF processes for communication on NUMA nodes has an impact related both to if there is a distributed vSwitch or not, and towards which Numa node the NIC card is connected. As illustrated in the pictures below.



Picture 3

In std OpenStack NUMA placement cannot be controlled. Different vendors have chosen different strategies to handle this. Some will have a proprietary parameter to state if best or worse NUMA node shall be used, and some VNF vendors will place their VMs over two NUMA nodes and find the right place for the IP stack process. However, if the NIC card sits as in the picture to the left, The KPI must be measured both before and after a failover, as the performance of best NUMA might differ. (To make things even more complicated, there are CPUs on the market that have such a large number of cores that an internal bus architecture within the CPU gives a different performance for different cores also inside one CPU, thereby a two-socket compute blade will have more than two NUMA nodes. In the example KPI below we have chosen not to include this impact.

A last important factor is if to measure with or without security groups activated, as security groups are implemented in the vSwitch when DPDK is used for communication between NIC and vSwitch, and we have seen that performance for these implementations differs a lot between different vendors. We choose here to have one **security group per VM in node 1.**

With this we can conclude the definition of the first NFVI networking KPI example to be: **Possible throughput between one sending and one receiving VM, placed in different compute nodes, VMs connected to same Neutron NW, with one security group per sending VM, both using DPDK to connect to vSwitch, both VMs placed in “best” NUMA node of their compute node, 9 flows connection, with one sending and one Receiving queue, with packet frames of 64 bytes. Packet lost ration < 2 packets per million, obtained in 9, 60-sec iterations, out of 10.**

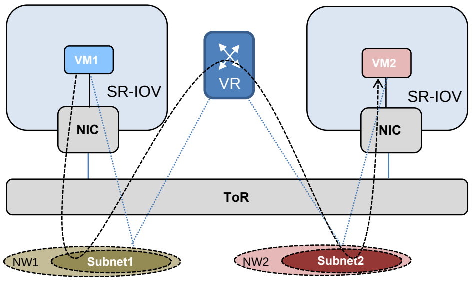
The rest is just measurement implementation…. Here the big challenge is to secure that the bottlenecks in the infrastructure are measured, not tool’s bottlenecks, i.e., securing that packets are not dropped in the sending or receiving VM’s DPDK code. As anyone who has tried to optimize a DPDK based VNF knows, that is not so easy. However, that will be a white paper of its own.

This might be seen as a very cumbersome definition of just a straightforward network KPI. However, as we will see below, most of the definition will be reused in the more complex network KPIs.

**Note, according to ETSI GS NFV-TST 009 this is not a throughput Benchmark, but “Capacity with X% loss rate" Benchmark. In this case X is 0.0002% =2 part per million. TST 009 typically regards a test which do not result in a stable value, as a failed measurement. However, the big challenge with NFV is that a virtualized environment is challenging to get 100% stable, there will always be small disturbances affecting data flows at the limit of the capacity. So as described above we find two ppm as the most relevant value.*

3.2 Plain L3 SRIOV DPDK performance

As shown in picture 3 we measure throughput between two VMs placed in two different compute nodes, connected to different Neutron Network.



Picture 4

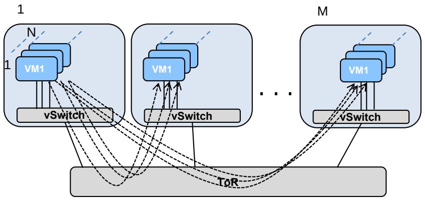
The interesting point here is that a vRouter is used when VMs are in different neutron networks. vRouter implementation is what differs most between different NFVI implementations:

1. In std OpenStack vRouter is implemented in a specific compute node called networking node. This is very seldom used in an NFVI due to performance and scalability issues in such a solution.
2. vRouters in physical NW nodes are quite common. Typically using routing SW in a physical switch. Alternatively, configuring vRouters in a DC GW. For this setup, SRIOV connection from VM is most relevant, as previous KPI covers the bottleneck in vSwitch.
3. vRouter’s forwarding plane distributed to vSwitch is used by, e.g. contrail, and sometimes with other SDN based neutron implementations. To use SRIOV here require that NFVI has implemented a so-called HWvTEP in the physical switches, which is seldom the case. So here a DPDK connection to vSwitch/vRouter must also be defined.

That said, all other considerations described in chapter 3.1 is still valid, that gives another example KPI definition as: **Possible throughput between one sending and one receiving VM, placed in different compute nodes, VMs connected to different Neutron NW, with one security group per sending VM, both using SRIOV with DPDK to connect to physical NIC, both VMs placed in "best" NUMA node of their compute node, 9 flows connection, with one sending and one Receiving queue, with packet frames of 64 bytes. Packet lost ration < 2 packets per million, obtained in 9, 60-sec iterations, out of 10.**

3.3 vSwitch performance with large routing tables.

VNFs are often consisting of a large amount of VMs communicating with each other. In the picture below we see that there are N VMs in node 1, and node 1's vSwitch have level 2 forwarding to in total Nx(M-1) VMs in M-1 nodes. (E.g. a large CSCF N can be 2 and M = 20). Throughput in node 1's vSwitch is measured for the KPI. It is not necessary to have a KPI with M=20. It is quite enough to measure with **N=3 and M=4**, and compare with N=M=1 and extrapolate the trend.



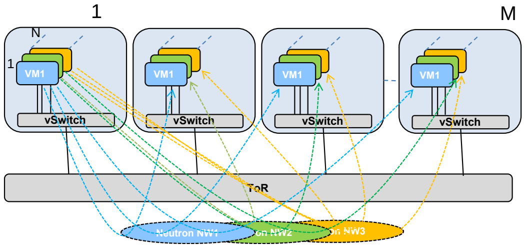
Picture 5

All other factors are as in chapter 3.1.

That gives another example KPI definition as: **Possible throughput through one vSwitch in a compute node where 3 VMs each are sending packets to 3 VMs placed in 3 different compute nodes, i.e., to 9 receiving VMs, all VMs connected to same Neutron Network, with one security group per sending VM, all VMs connected to vSwitch with DPDK, all VMs placed in "best" NUMA node of their compute node, 9 flows connection, with one sending and one Receiving queue, with packet frames of 64 bytes. Packet lost ration < two packets per million, obtained in 9, 60-sec iterations, out of 10.**

3.4 vSwitch performance with a growing amount of vXLANs

For VNF solutions, e.g., IMS or EPC, the solution vendor often place just one VM for a VNF in a compute node, to handle VNF high Availability. Instead, other VNFs from the solution uses the same compute nodes. Usually, different nodes will be using different Neutron Networks. This increases the total amount of vXLANs that a specific vSwitch has to handle. The KPI is **total throughput through vSwitch in node one**, which will be handling NxM vXLANs.



Picture 5

A typical large VNF solution could be N = 3 and M = 30. It is not necessary to have a KPI with M=30. It is quite enough to measure with **N=3 and M=4**, and compare with N=M=1 and extrapolate the trend. We choose to have **one security group per Neutron NW**.

All other factors are as in chapter 3.1.

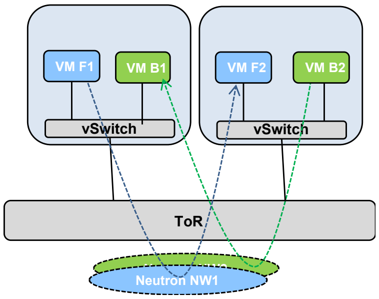
That gives another example KPI definition as: **Possible throughput through one vSwitch in a compute node where 3 VMs each are sending packets to 3 VMs placed in different compute nodes, i.e., to 9 receiving VMs, 3 Neutron Networks where every compute node have 1 VM connected to each one of the NWs. All connections between VMs are in the same NW, with one security group per NW, all VMs connected to vSwitch with DPDK, all VMs placed in "best" NUMA node of their compute node, 9 flows connection, with one sending and one Receiving queue, with packet frames of 64 bytes. Packet lost ration < 2 packets per million, obtained in 9, 60-sec iterations, out of 10.**

3.5 vSwitch performance with a large number of flows and new flows per second.

Newer versions of vSwitches are basing its routing tables on L4 flows. If the amount of flows becomes very large, lookup time in the routing table will affect throughput. Also creating new flows will use vSwitch capacity and affect throughput.

Some VNFs, typically a Value Added Service node in an EPC solution, will receive and send a massive amount of flows, of which a significant amount will be new every second. (As it handles end users' UDP and TCP sessions)

New flows take some time to set up, if packets start flowing before this is done, packets are dropped. It is in a KPI measurement necessary to distinguish between packets dropped in new flows and packets dropped in steady flows. Therefore, the KPI definition we choose is to generate new flows in “background” and a steady flow in “foreground” where the max speed for a given acceptable packet loss rate is measured, in the foreground. As described in picture 6.



Picture 6

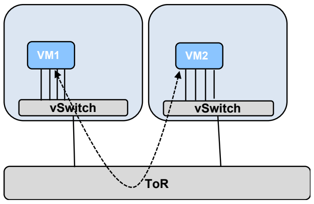
We choose to always **have 50k flows in the background generating 10 % of these new every sec. In total, we choose to have a background load of 500k pps. The KPI measurement will be possible throughput in the foreground connection defined as in chapter 3.1.**

That gives another example KPI definition as: **Possible throughput between one sending and one receiving VM, placed in different compute nodes, VMs connected to same Neutron NW, with one security group per sending VM, both using DPDK to connect to vSwitch, both VMs placed in “best” NUMA node of their compute node, 9 flows connection, with one sending and one Receiving queue, with packet frames of 64 bytes. Packet lost ration < 2 packets per million, obtained in 9, 60-sec iterations, out of 10. Given a background load of NW traffic between two VMs om same two compute nodes, connected to a different NW having a throughput of 500k pps with 50 k flows of which 10% is generated new every second.**

3.6 vSwitch performance for VNF with Kernel IP stack and multi-queue

This is the same base configuration as in the first KPI. However, this time the KPI is valid for VNFs which uses the guest OS kernel-based IP stack for communication. I.e., the communication between the VM and the vSwitch is not using DPDK. (If DPDK is used between the vSwitch and the NIC, is viewed as an internal development choice for the NFVI)

VNFs using the kernel IP stack, but still looking for rather high communication bandwidth, will use Multiple queues towards the vSwitch, i.e., there will be multiple instances of the IP stack running on different cores. We have chosen to have 4 queues and 16 flows (to have a balanced load on queues)



Picture 7

All other factors are as in chapter 3.1.

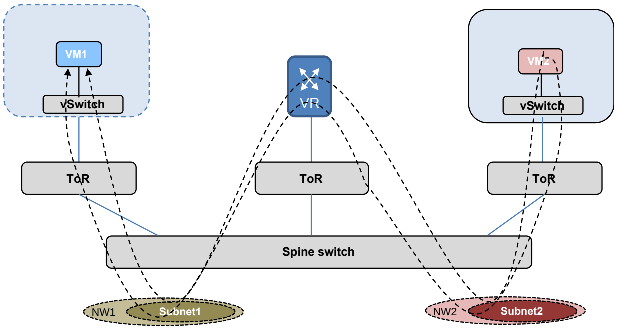
That gives another example KPI definition as: **Possible throughput between one sending and one receiving VM, placed in different compute nodes, VMs connected to same Neutron NW, with one security group per sending VM, both using kernel IP stack and 4 queues, both VMs placed in "best" NUMA node of their compute node, 16 flow connection, with packet frames of 64 bytes. Packet lost ration < 2 packets per million, obtained in 9, 60-sec iterations, out of 10.**

3.7 Network Latency KPIs

Network Latency KPIs is usually measured as a round trip delay for a packet sent between to VMs. Because implementation of the OpenStack Neutron functionality can be very different in different NFVIs, weaknesses in an NFVI's latency KPIs can be found for different kind of latency measurements. Typically for a system based on a modern vSwitch, and an SDN solution, we need to look at different KPIs;

1. Packets using "slow path" through the vSwitch data plan. This is, e.g. the case for a "ping" packet used for VNF internal supervision of VMs. If this is too long, the High Availability solution of the VNF might become unstable.
2. The first packet in a flow; As the SDN controller will be involved in establishing a path through the vSwitch's "fast path." This can be quite long and might even lead to time-out in the end to end session establishments, as a large amount of delayed is added.
3. Fast path for a UDP packet. This is critical for good quality for RTP traffic. Here also Jitter is critical to look at.

Typically, here we should look after the worst-case scenarios. E.g., traffic passing over vRouter and spine switches. As shown in the picture below.



Picture 8

That gives another example KPI definition as: **Round trip delay for an ECMP packet (slow path) sent between two VMs, placed in different compute nodes, placed in racks connected to different ToR switches, VMs connected to different Neutron NW, with one security group per sending VM, both using DPDK to connect to vSwitch, both VMs placed in "best" NUMA node of their compute node.**

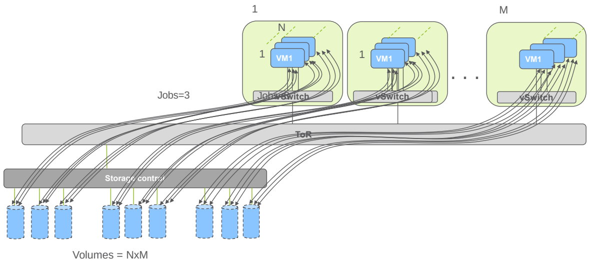
3.8 Storage system KPIs

Storage system performance is essential for many VNFs. There are many different parameters that can also vary when measuring performance for a storage system; Amount of storage user VMs, amount of Job threads per VM, IOdepth Defines the number of I/O units to be used in parallel in the test. A value higher than 1 is only valid for asynchronous I/O engines, block size to write and read. What operations to use; write, read, or a combination of reading/writing is done with serial or Random access.

Here we need to have a rather large amount of KPIs, as different VNFs will use the storage in very different ways. We choose to vary between "extreme" values one parameter at the time so IOdepth 1 and 16, Jobs 1 and 16, block size 4 k and 1 M.

We see two different KPIs to measure for each set of parameters; IO operations per second and latency per operation.

We use a set up as shown in the picture below, meaning there will be NxM VMs (N = 3 and M= 5) all loading the storage as much as they can, given amount of jobs and IOdepth. Typically the results from different VMs can vary quite a lot. We recommend having KPIs for average and for worst performing VM.

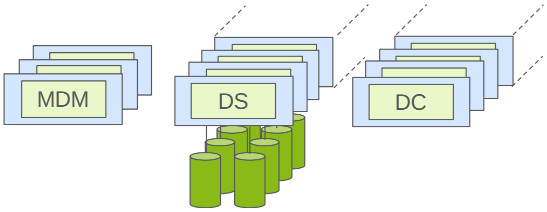


Picture 9

One KPI definition will then be: **Run 5 nodes with 3 VMs in each. Each VM will load system with Job=1, IOdepth=16, block = 4K. Random access max possible load. Measure IO operations per second in all VMs and use the worst result as KPI value.**

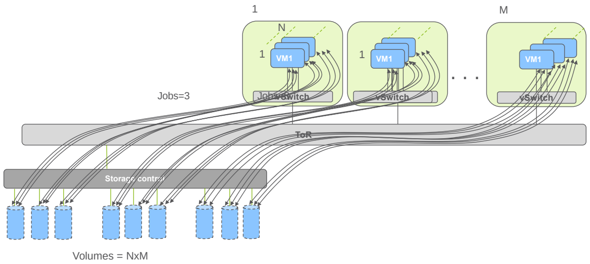
3.9 Non-deterministic response time from storage system:

Most modern Distributed Soft Storage systems, like, e.g. Ceph, have an architecture with Disc servers, Data client and a Metadata handler. As shown in the picture below:



Picture 10

The MetaData handler is working on tables in RAM, occasionally written to the disc. This can give rather long times when the storage system does not respond, and therefore gives long response delays. When this co-occurs with specific actions in a VNF, faults can happen. These faults are complicated to reproduce, and therefore very difficult to solve. This is why we propose to measure the longest delay response in a busy storage system.



Picture 11

We choose the following example KPI definition: **Load storage with 3 VMs per compute nodes and 6 compute nodes. Each VM is generating a load with read/write (70:30), block size of 4K, Sequential access, 2000 IOPS, 16 jobs in parallel with Iodepth of 16 and running for 1 hour. KPI is the longest delay time measured during this hour.**

3.10 Compute performance testing

One could think that CPU performance was entirely dependent on HW used, but it is not quite that simple. There are quite a few configuration choices that will impact. E.g. Is turbo mode enabled, is hyperthreading enabled, is the VM pinned, are the host OS given its cores, is the host OS real-time configured or optimized for high throughput……. All this will impact the VNFs' performance on a VM with a specific flavor.

We believe it is valid to have KPIs both on CPU performance benchmark for one and many cores, and here we use UnixBench instructions per second, as well more complex defined index scores. Also here we have chosen to use UnixBench.

The definition of one of these KPIs would be: Run UnixBench on a pinned VM with 16 CPUs spanning over 2 NUMA nodes. Measure Instructions per second Dhrystone Benchmark for parallel execution on 16 CPUs.

4 Conclusion

Lack of sound knowledge of cloud resources can become a bottleneck for different VNFs. It is essential to define which KPIs to use, and continuously measure them during the life cycle of an NFVI. Only with a clearly defined definition of KPIs, an operator will be able to handle the middleman dilemma and achieve NFV transformation. From a Long- term perspective, it would be ideal if these KPI definitions could be regarded as an industry standard. Further evolution of KPIs can continue based on operational experience.

5 Contributors

The contributors to Jio's and VoerEir's opinion on this topic are;

Aayush Bhatnagar - Sr.Vice President Jio

Thomas Lindquist - Chief Technology Officer VoerEir

Adityakar Jha - Asst. Vice President Jio

Arif Khan - Sr.Vice President Development VoerEir

Munir Sayyad - Asst. Vice President Jio

Sandeep Bisht - Solution Architect Jio

Ashok Kumar - Chief Architect VoerEir