

KPI-validation and SLA monitoring in context of troubleshooting/isolating VNFs' performance issues

Version 1.0

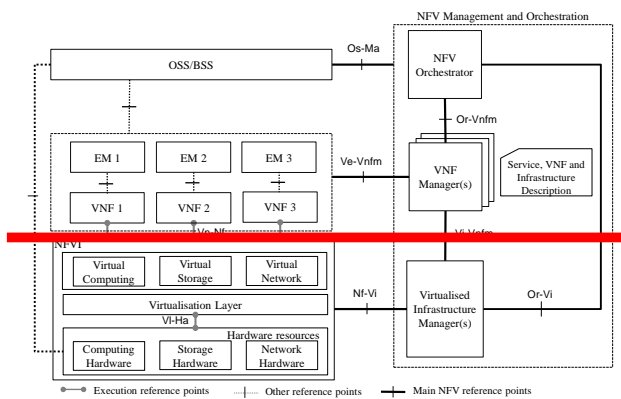
Contents

1	Introduction	2
2	Relationship of terms as used in the document.	3
3	Why NFVI's performance KPI's suddenly changes.....	3
4	Monitoring performance of an OpenStack based NFVI.....	4
5	KPI validation and SLA monitoring.....	6
6	VNF emulation testing for problem isolation	7
7	How touchstone can be used for troubleshooting cloud performance issues	8
7.1	VNF emulation capabilities of Touchstone	8
7.2	Touchstone support for troubleshooting.....	9

1 Introduction

The base for VoerEir’s NFV strategy for identifying performance responsibilities between vendors in the age of NFV is the following statement; “Verify that your NFV-Infrastructure fulfills its performance KPIs, and it becomes VNF vendors’ responsibility to fulfill theirs”. This is a very good starting point and VoerEir’s test system, Touchstone, focuses on this verification.

Our strategy is to focus on the definition of the demarcation point between the NFVI and the VNFs, as showed in picture one. The difficult part here is to find the NFVI performance KPIs that will determine the VNFs’ performance. We call these a VNF’s essential KPIs. As we will discuss later, these will be different for different VNFs and the definition of these KPIs can be quite complex.



Picture 1 The demarcation point in ETSI’s NFV architecture

With Touchstone, the validation of these KPIs can be done during pre-deployment test and by that create the platform for continuous fault isolation work as described in this document.

Unfortunately, sometimes the world is a little more complex than that. A VNF vendor may refuse the responsibility and claim that his VNF is verified to fulfill the performance KPIs given a specified amount of resources (e.g. compute cores). There can be many reasons; Sometimes the NFVI’s performance KPIs change when a specific VNF runs on top, sometimes different VNFs affects each other so one VNF don’t experience the NFVI performances that it has been promised. And sometimes the KPIs might not have been correctly defined.

This paper looks at different reasons for NFVI performance KPIs to change, and what tools and methodology will be necessary to track down the responsibilities for insufficient performance between the NFVI and the different VNFs.

2 Relationship of terms as used in the document.

KPI, Key Performance Indicator: Is a performance capability, measured on a not loaded system. I.e. the definition of the KPI, includes all activities on the system during the measurement.

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.

Monitoring: Collect events and metrics of NFVI resources usage in total and for specific VMs resource usage. (For OpenStack probes and agents normally puts data in Ceilometer or Monasca)

Fault management: Use monitoring data to identify if something is wrong, and initiate the actions needed to fix the issue. (For OpenStack distributions, today typically Zabbix is used, but it will more and more be Monasca going forward)

Performance management: Use monitoring metric to get a view on what is happening in our system. Often graphical, and often used to get early warning that an expansion is necessary.

Billing: Use monitoring metric to calculate how much resources a tenant has been using, and therefore how much we shall charge him.

Troubleshooting: Use whatever data is available, and when necessary, create Test Cases that can reproduce and isolate the issue.

3 Why NFVI's performance KPI's suddenly changes.

We have validated our NFVI's performance KPIs in pre-operation validation but when we load a specific VNF on to it, we will sometimes suddenly see that the NFVI cannot fulfill the SLA, based on that KPI. There can be quite a few reasons for this:

1. **NFVI and VNF compete for the same resources.** Here are some examples;
 - a. Cache memory; most common for a high performance vSwitch and a cache depending VNF.
 - b. Cores; There are some NFVI processes floating and if the VNF is pinned to all available cores, e.g. QEMU, or NUMA balancing might do short interop on VNF fast path, which causes that packages are lost.

2. **VNF depends on performance KPIs not specified and measured/tested** (We call KPIs directly impacting a VNF's performance for **essential KPIs** for that VNF. Sometimes the essential KPIs can have a very complex definition). Over time the situation will improve towards full coverage of KPIs, and VNF knowing their requirements for all these KPIs. But we are not there today. Examples of KPIs we at VoerEir have quite late added to our list are.
 - a. Issues around flow set up capabilities in later generation of vSwitches. A VNF might ask for a KPI called "max throughput with < 2 ppm packet loss, when 200 k new flows/s are established".
 - b. Storage delay time with large IO depth and many parallel Jobs to a centralized storage.
3. **VNF misbehavior:** Could be e.g:
 - a. The VNF is so late to release DPDK buffers, so that these buffers are thrown out of LLC (Last Level Cache)
 - b. Bursty traffic causing overflow in vSwitch's quite small RX buffers.

As we will see later in this document, it is very difficult to catch these kinds of issues only by monitoring the total system in operation, however as long as we are talking about pre-operation testing with single VNF deployment, it is very well suited for measurements with VNF emulation tools and off-operation trouble shooting. We will also discuss this later in this document.

Next step in increased complexity is when VNF works well in pre-operation validation but starts behaving bad as it is put into operation. Also here there can be different kinds of reasons:

1. Other VNFs steals resources
2. Other VNFs misbehave
3. NFVI or VNF have an error which affects performance

In operation it is of course very valuable if root cause can be identified with non-intrusive monitoring, however it might require in-operation troubleshooting or even off-operation measurement and troubleshooting, as we will see below.

4 Monitoring performance of an OpenStack based NFVI

Ceilometer and Monasca are the services within OpenStack available for collecting metering and other monitoring data. Both are flexible and can easily be extended with new agents for data read out and plugins for receiving data from external sources. Monasca also offers the possibility to filter data and generate alarms, e.g. by configuring thresholds for specific data.

(Due to performance and scaling issues, some OpenStack implementations have chosen to use other system for collecting, storing and forwarding metric and event data, an example is Intel's SNAP. That is however not important for the discussion in this document)

With current releases of these services it is possible to get a good overview of system components status, and how VMs consumes resources, with some exceptions. E.g. we cannot get visibility and control over how shared resources such as last-level cache (LLC) and Memory bandwidth are used by VMs. If vSwitch and VMs uses DPDK for network IO it is also not possible to see how NIC bandwidth usage is split between VMs, or if packets are lost in the buffer handling path.

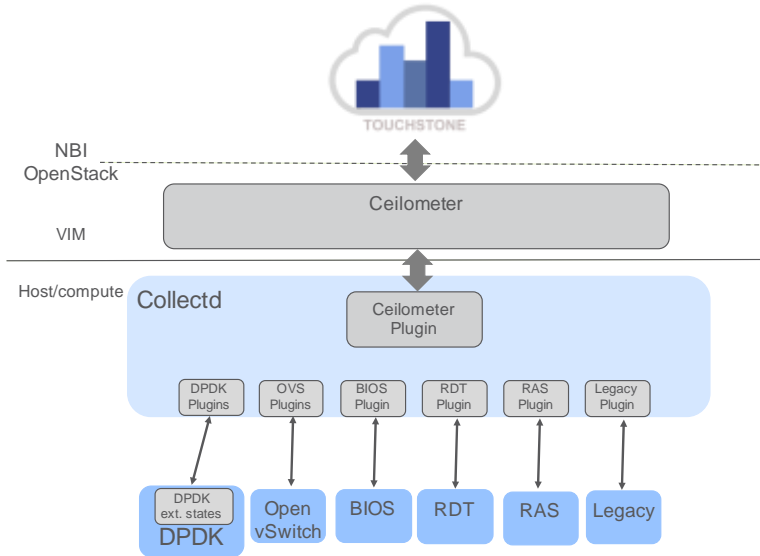
To address these shortcomings Intel is leading an OPNFV project called Barometer, which is targeting:

- Traffic and performance monitoring of the components that provide networking functionality
- Platform monitoring including: CPU, memory, load, cache, thermals, fan speeds, voltages and machine check exceptions.

Current NFVI cloud implementations differ in maturity on which of these monitoring points they already have fixed with for example own Ceilometer plugins and agents. With Barometer we will get a “standard way” to get this data into e.g. Ceilometer or SNAP.

More important is the inclusion of data that up to now has been lacking and Intel already has some interesting tools. With Resource Director Technology (Intel® RDT) we get data of individual VMs or containers usage of LLC and memory bandwidth. By instrumenting Intel’s DPDK libs with timestamp of packets we will be able to read out vSwitch and vNIC data in operation, which gives us jitter and delay measurements. Also packet loss information will be possible to read out of the DPDK layer.

The scope of barometer is limited to collecting/gathering the events and states and relaying them to a relevant end point. Barometer will not enforce or take any actions based on the gathered information.



Picture 2 Barometer architecture, in this case with Ceilometer and Touchstone

For OpenStack the natural end point for Barometer data is Ceilometer and in that way the data is made available over OpenStack’s NBI. With support for search in this data, we will have major improvements in our ability to troubleshoot performance issues for OpenStack based VNFs.

Still, even with Barometer in place, non-intrusive validation of performance KPIs of an NFVI in operation will not be possible. This is quite easy to understand as KPI validation implies max loading of shared resources, it is also per definition done on a not loaded system.

So, going forward we will be able to monitor VMs’ usages of shared resources, and we will be able to spot when a shortage of NFVI resources occurs, hence limiting VNF performance. We will also be able to detect so called Noisy-Neighbors.

What will still be difficult to determine is if a shortage of NFVI resources, under the agreed and validated KPIs, is due to faults or degradation in the performance of NFVI, or due to that the VNF solution itself behaved in a pattern outside of the KPI definition. To address these issues, it is often necessary to do more advanced troubleshooting, including emulation of VNF usage pattern of these specified resources.

5 KPI validation and SLA monitoring

We defined **KPI**, as a performance capability measured on a not loaded system. I.e the definition of the KPI includes all activities on the system during the measurement.

SLA is defined as 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 else is happening on the system.

As a first glance these can be seen as the same thing, but in reality they are quite different. The KPIs definition includes the total load on the NFVI, while the SLA is what a specific VM gets. Therefore a KPI definition can include variation in many parameters, such as e.g. throughput, burst, flow setup/s and available cache to vSwitch. As these combinations is impossible to monitor in a synchronized way, SLA definitions tend to have only one parameter with variation.

As SLAs are what we monitor, it would be an easy way out to state that SLAs are what should be agreed between the NFVI and the VNF. The problem is that it is very often the VNF Solution's VMs that destroy the NFVI ability to fulfill the SLA. The NFVI's performance depends on a large amount of factors that all varies, and it is impossible for the NFVI to contract anything, if these are not under control.

This insight is the background for VoerEir's strong focus on complex KPI measurements.

6 VNF emulation testing for fault/performance issue isolation

NFVI KPI validation must be done in a way that secures that the result is not limited by the tool. Either because the tool has limitation in its performance, or because the tool behaves in a way that affects the NFVI's performance, as described above as VNF misbehavior.

In chapter 3 we also described that the essential KPIs (The KPIs that limit a VNFs performance) is different for different VNFs, as their traffic load characteristics, their storage usage or their dependence of LLC will differ. VNF solutions often consists of many VNFs, each with many VMs. The solution vendor usually has a pre-tested way of distributing and co-locate these different types of VMs, which all can lead to specific rather complex essential NFVI KPIs, looking at what is essential for the VNF solution as a whole. So, if an SLA is not fulfilled, even if KPI is validated, this often depends on that other VNFs was not part of the KPI definition. And in that case a new KPI must be found and agreed upon. Another alternative is that the NFVI in operation is not configured exactly as the NFVI used for KPI validation. This will require troubleshooting as described below, about Touchstone Agent and read out of Host OS static/semi static data.

To be able to validate more complex KPIs, e.g. max throughput for a VM, given a Noisy Neighbor who generates x amount of new flows/s, it is necessary for the test tool to emulate both the Neighbors' traffic load as well as the VNF's. And do so in the way specified in the KPI definition.

Troubleshooting, most operators will claim, is a responsibility for the vendor. This was easy in the days of vertical integrated solutions. As NFV is introduced, the first issue is to be able to point at which vendor has responsibility for troubleshooting, and as we have seen above, that is not always so straight forward. Our experience has showed us that the best way is to start from the bottom. Emulating the critical behavior of the VNFs, with controlled tools, and validating the agreed NFVI KPIs. If the validation fails responsibility goes to the NFVI vendor. If it succeeds the VNF vendor will have to take it. These VNF emulation tests can sometimes be made in an

environment in operation, if considered too intrusive it can be done in a dedicated lab environment.

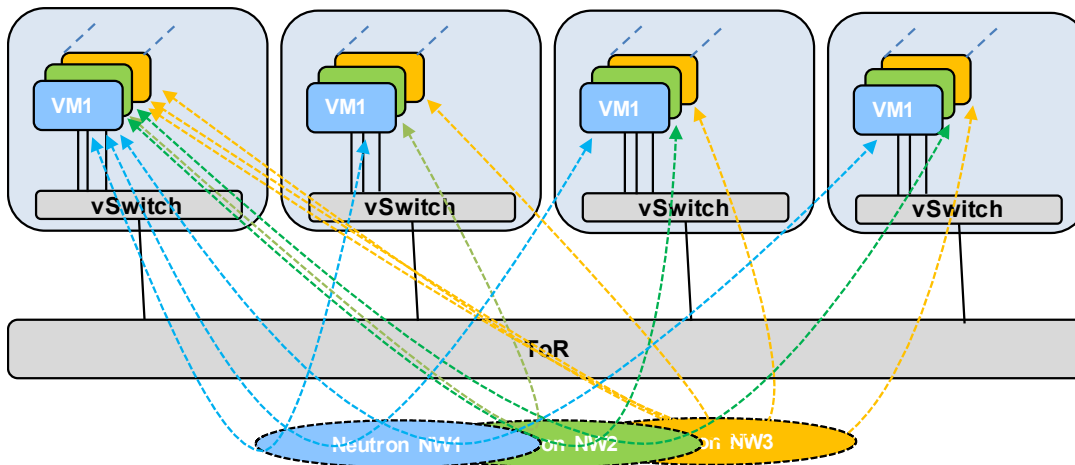
7 How touchstone can be used for troubleshooting cloud performance issues

The prime focus for Touchstone is to provide all test cases needed for validating NFVI KPIs. This includes the ones that are rather straight forward, as well as the more complex ones. As VoerEir has been working with NFVI and VNF vendors as well as Cloud operators, on the issue of determine the right KPIs for specific VNFs, Touchstone already has the capability to offer a large amount of Test Cases for KPI validations. This is a new area for everyone, and we are continuously learning, and as we do more KPIs will be added to our recommended list. Often these can be done by configuring existing TCs, sometimes new TCs or configuration capabilities will be added.

7.1 VNF emulation capabilities of Touchstone

In Touchstone, a Test Case is a combination of:

- **Scenario/Benchmark**, which describes exactly what kind of load shall be generated, and what shall be measured.
- **Context**, which describes which VMs shall be created, exactly how these shall be placed, what networks they shall use, and what volumes they shall use.



Picture 3: Touchstone context H (With N = 3)

Both Scenarios and Contexts can be configured within given boundaries. Creating a large flexibility for how a specific TC is instantiated. There are scenarios which can be configured to generate a specific type of “dynamic load”, e.g. bursts or flow creation-termination/s.

Scenario Context combinations can also be used as Background load, with or without generating Test Reports. Such a background load can be started, stopped or triggered to change behavior based on events in a TC. By running a TC with one or several tailored background loads, Touchstone can emulate any VNF solution, from the NFVI's point of view. These test can be done in an NFVI in production or in a validation LAB.

7.2 Touchstone support for troubleshooting

Touchstone subscribes for Ceilometer/Barometer data over the OpenStack NBI, as seen in picture 2. This means that whatever monitoring data the NFVI vendor has chosen to feed into Ceilometer is available for Touchstone's troubleshooting support.

Touchstone does also have an optional Agent used for:

- Reading out host OS configuration incl. BIOS parameters
- White box functional testing of e.g. EPA features
- Restart/stop test VMs for Robustness testing
- Restart/stop Hosts for Robustness testing

As it needs access to host OS, it is designed with a set of security features:

- Temporary deployment using container or Debian packages
- SSH to agent NBI
- Rest API limiting operations, subset of API can be chosen for further limitation

It is VoerEir's intention to OpenSource this Agent, and contribute it to OPNFV, as we see its ability to read out a large amount of static or semi static parameters from Host OSs, as essential for troubleshooting NFVI clouds for performance issues, or to find difference in Host OS configuration, from validation region, and the regions in operation.

Over time we expect more and more clouds to have Barometer installed in their Host Oses. Therefore, we have chosen to base our Troubleshooting support to a large extend on Barometer data through Ceilometer (or other services e.g. SNAP). In lab environments Touchstone Host OS agent can load Barometer on the host OS if it is not provided by the NFVI vendor.

Below are two examples how Touchstone uses Barometer data for Troubleshooting:

1. A vSwitch during high usage depends a lot on LLC. If it cannot get the LLC it needs, packets will be dropped. By reading out data Touchstone adds vSwitch LLC usage and cache hit rate to the test report. This can then be compared to the same data in

operation, with the VNF generating the network load. This Indicates if VNF LLC usage creates problem for the vSwitch.

2. By reading out DPDK layer data for packet loss from Host and compare this data with data read out from the NIC, when dropped packages are observed, it is possible to see where packets are lost.

Touchstone will store Ceilometer Data it subscribes for, in a tool with advanced search and analytic support, together with all logs and test reports generated by touchstone and its test VMs. This makes it flexible and easy to write search scripts for data important for specific troubleshooting issues.

VoerEir works with both vendors and operators of cloud solutions, and by that we are by continuously learning what root causes to issues can be, what TC and background loads to execute, and what data search scripts to use, to verify our hypothesis. In this way our experience and knowledge is made available for our customers.

Other results of this cooperation, and learning from data analytics, is improved understanding of what is a specific VNF's essential KPIs. And improved correctness of VNF emulation. Going forward Touchstone will contain "Complex" Test Cases named after the VNF they emulate.