Open Source Networking

LFN certification program Evolution proposal

Morgan Richomme, Jamil Chawki
October 2018

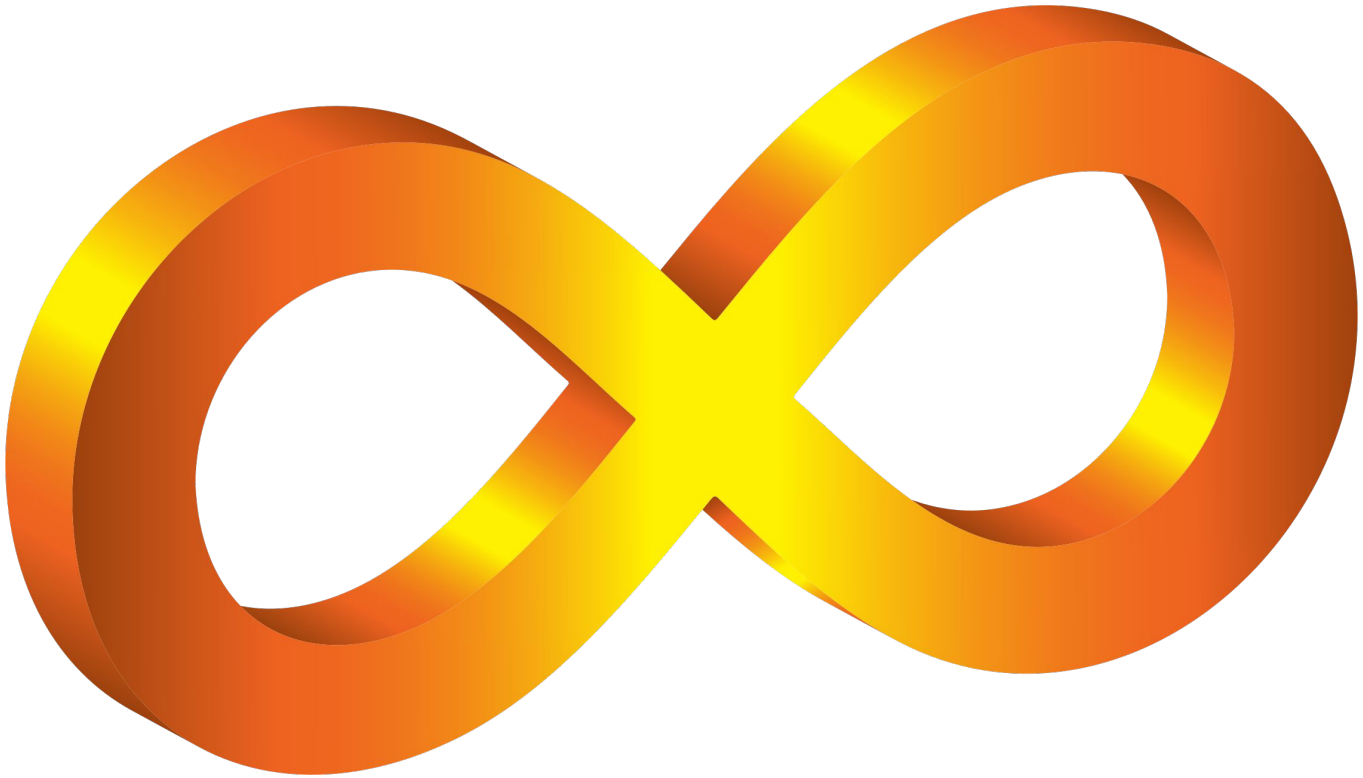

Important

Verified Program is important for the communities

- For the vendors
  - Justify resources allocation/community commitments
  - Integrate community assets in strategy
- For the Telcos
  - Provide Trust/Maturity indicators
  - Reduce interoperability Infra/VNFs testing campaign
- For the Community
  - Differentiate from other Open Source initiatives

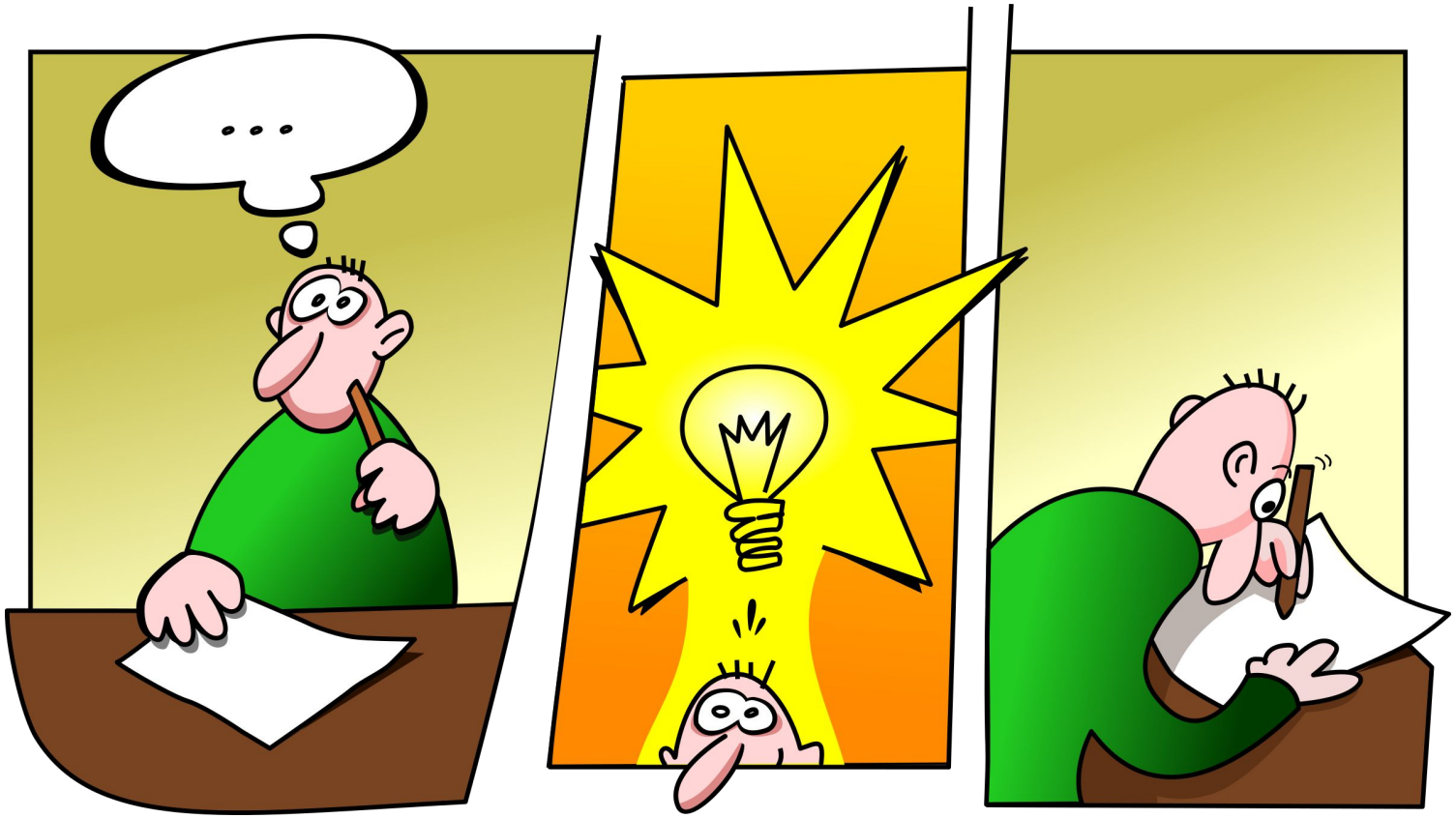
Infrastructure & ONAP VNFs

Infrastructure capabilities?



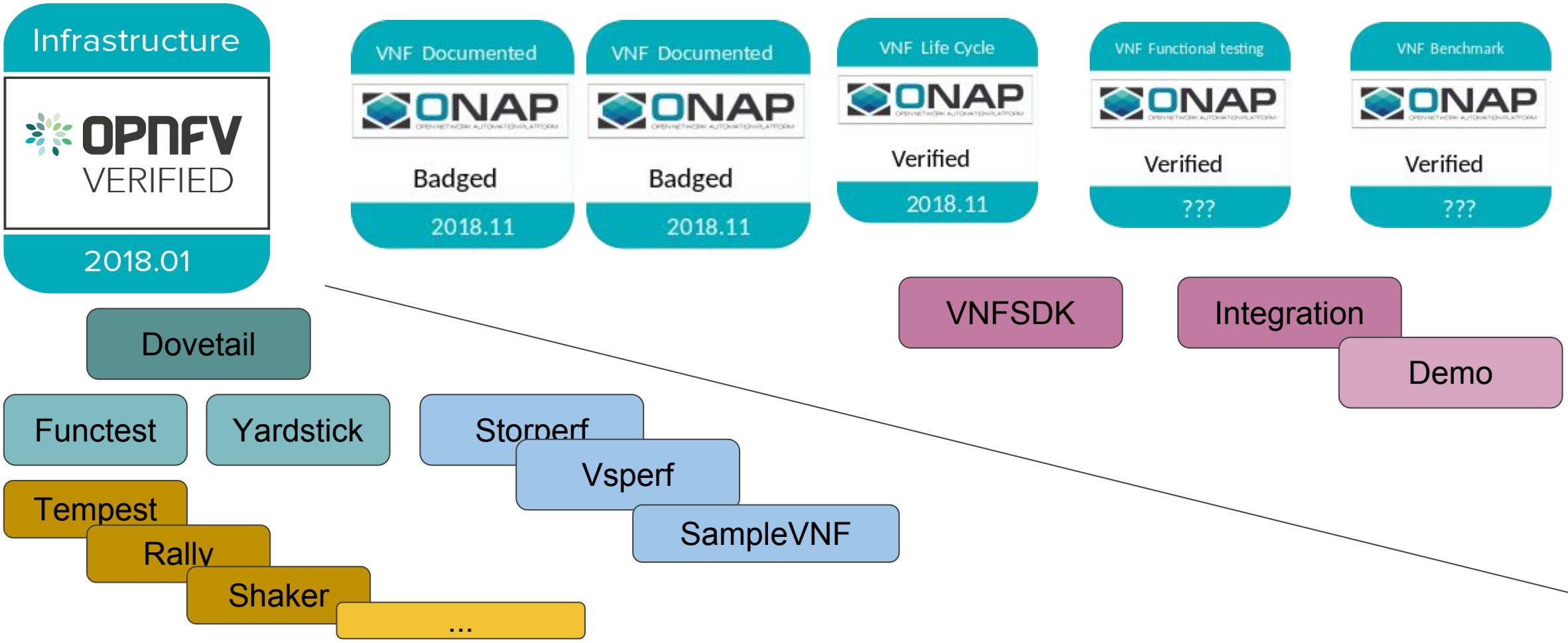
ONAP VNF?

Infrastructure certification



ONAP VNF certification

Today in LFN projects



Infrastructure certification evolution proposal

- leverage existing mature testing projects
- higher bar
- bottom-up approach

OVP

- OVP Infrastructure test scope mainly limited due to the OPNFV installer centric paradigm
- Today gating is not good enough to base OVP on installers
- Testing projects are the most active OPNFV projects and bring added-value compared to upstream
- OVP Evolution needed to an OPNFV testing centric view

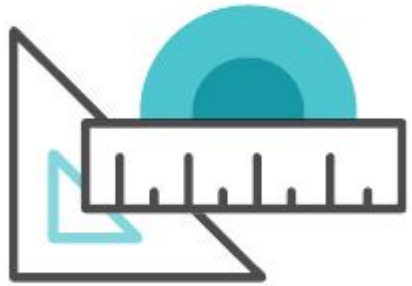
OVP capabilities

Upstream

Functest (Functional tests)

- Healthchek
- Smoke
- Benchmarking
- VNF

1000+ tests including upstream suites for OpenStack and Kubernetes



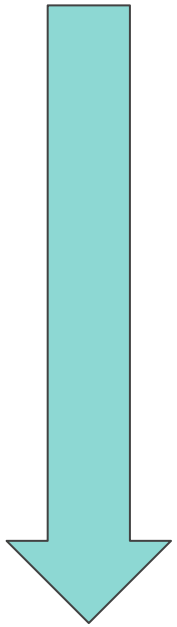
Yardstick + NSB

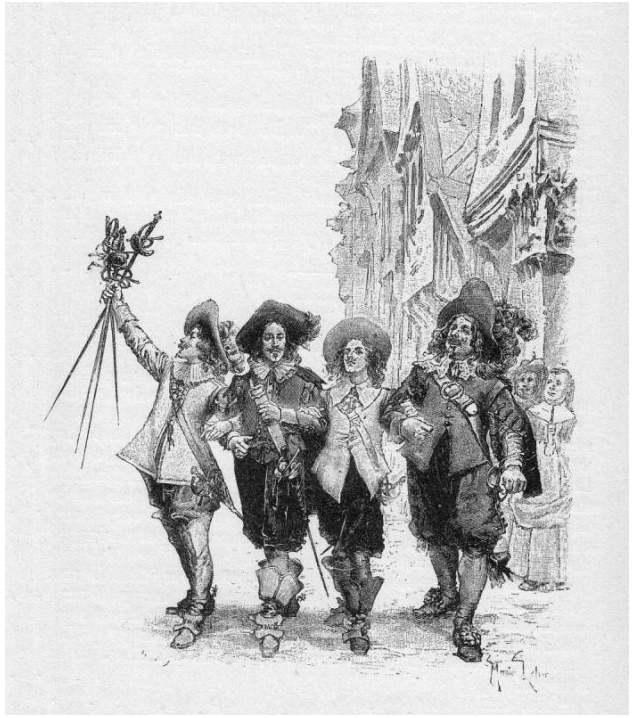
- Compute
- Storage
- Network
- Sample VNFs
- + VSPERF, Storperf,...

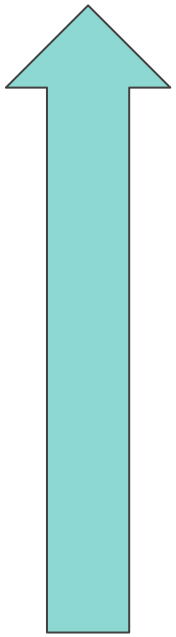
Both projects used independently from the installer, including by the industry and CSP

One for all…. or all for one...

1 project picks up test cases of upstream test suites
and excludes some or specify specific config







XTesting

1 project collects the results as they are

Official Upstream projects defines test suites

Evolution proposal for a bottom up approach:

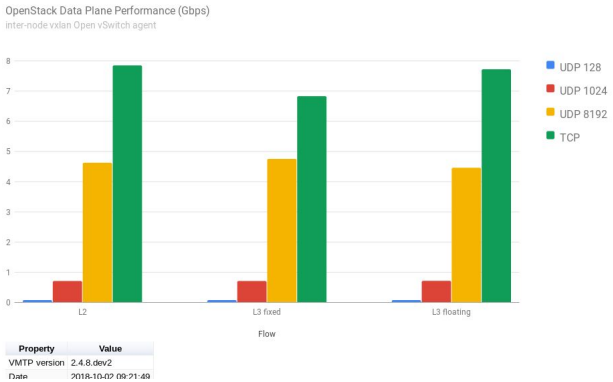
- Dovetail remains the entry point for certification including ONAP VNF, attribute or not the certification
- Test projects (Functest, Yardstick, VNFSDK) are responsible for the test suites, not Dovetail
- Xtesting is used to harmonize the way to launch tests and collect results

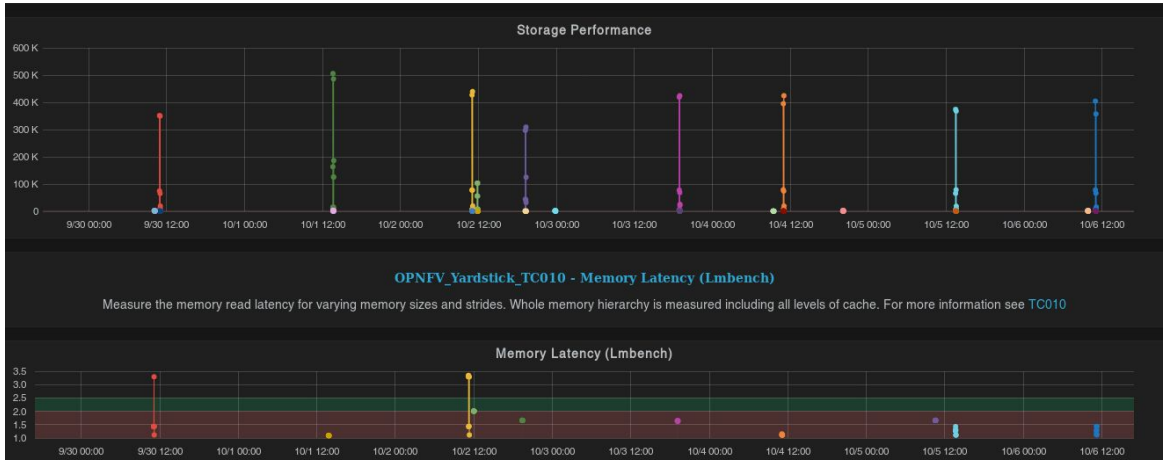
From results to KPIs for certification

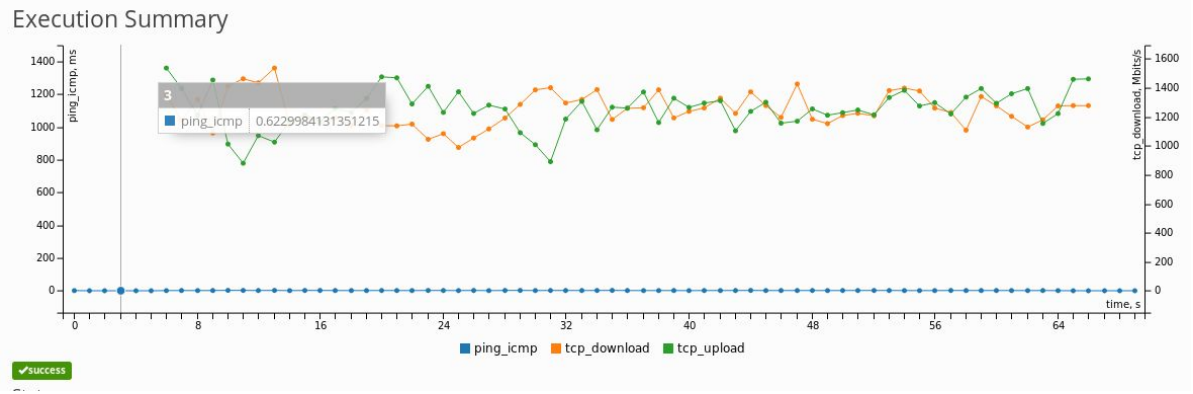
"There are three kinds of lies: lies, damned lies and statistics." M.Twain

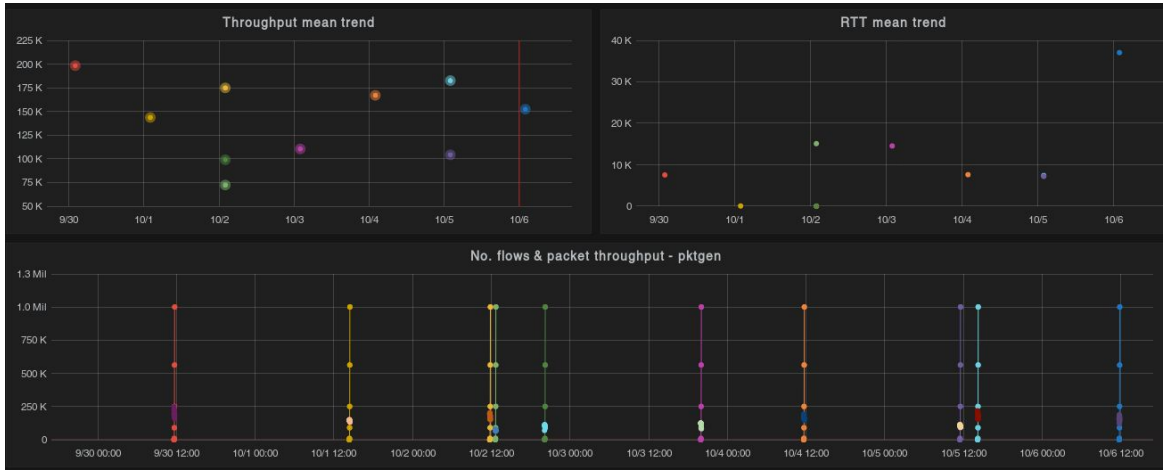
OVP evolution: focus on Performance KPIs

- Functest / benchmarking (upstream vmtp and shaker projects)
- Yardstick

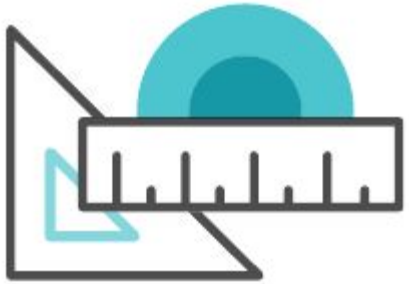








src: https://docs.opnfv.org/en/stable-fraser/submodules/yardstick/docs/testing/user/userguide/02-methodology.html



Recommendation: adopt Xtesting to launch Yardstick (probably only some days needed)

Table 4 - Yardstick Generic Test Cases

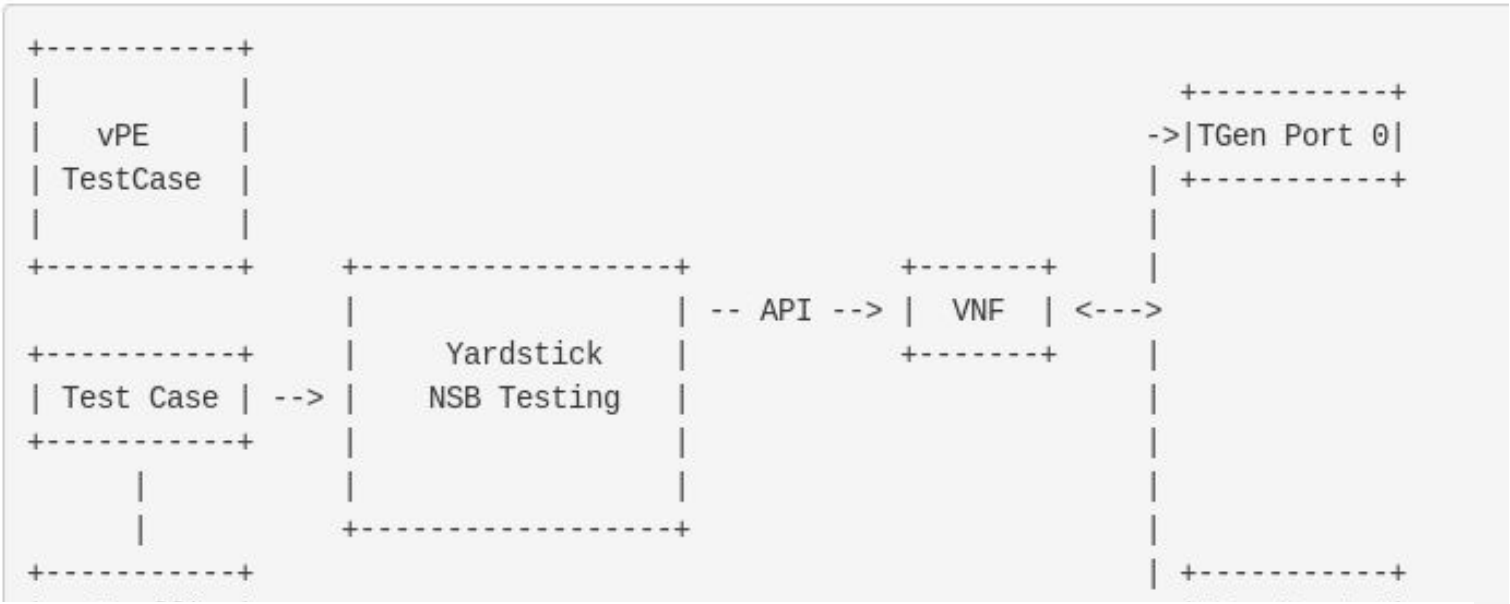
Category	Performance/Speed	Capacity/Scale	Availability/Reliability
Compute	TC003 [1] TC004 TC010 TC012 TC014 TC069	TC003 [1] TC004 TC024 TC055	TC013 [1] TC015 [1]
Network	TC001 TC002 TC009 TC011 TC042 TC043	TC044 TC073 TC075	TC016 [1] TC018 [1]
Storage	TC005	TC063	TC017 [1]

Note

The description in this OPNFV document is intended as a reference for users to understand the scope of the Yardstick Project and the deliverables of the Yardstick framework. For complete description of the methodology, please refer to the ETSI document.

NSB

NSB Testing with yardstick framework facilitate performance testing of various VNFs provided.



1. CGNAPT - Carrier Grade Network Address and port Translation
2. vFW - Virtual Firewall
3. vACL - Access Control List
4. **Prox - Packet pROcessing eXecution engine:**
   - VNF can act as Drop, Basic Forwarding (no touch), L2 Forwarding (change MAC), GRE encap/decap, Load balance based on packet fields, Symmetric load balancing
   - QinQ encap/decap IPv4/IPv6, ARP, QoS, Routing, Unmpls, Policing, ACL
5. UDP_Replay

A challenging question

- Very difficult to set threshold per performance tests
  - depends of the hardware... that evolves fastly
  - depends of the configuration...that evolves even faster
- Leverage great OPNFV asset: Pharos federation & its CI/CD results
  - results have been collected for years
  - results stored in a consistent way

- Set thresholds dynamically based on results collected in Pharos labs
- Study how external lab could provide their results and be Pharos partner
- Set a min of occurence to validate the KPI (at least 500 runs/results)

How to define this thresholds mathematically?

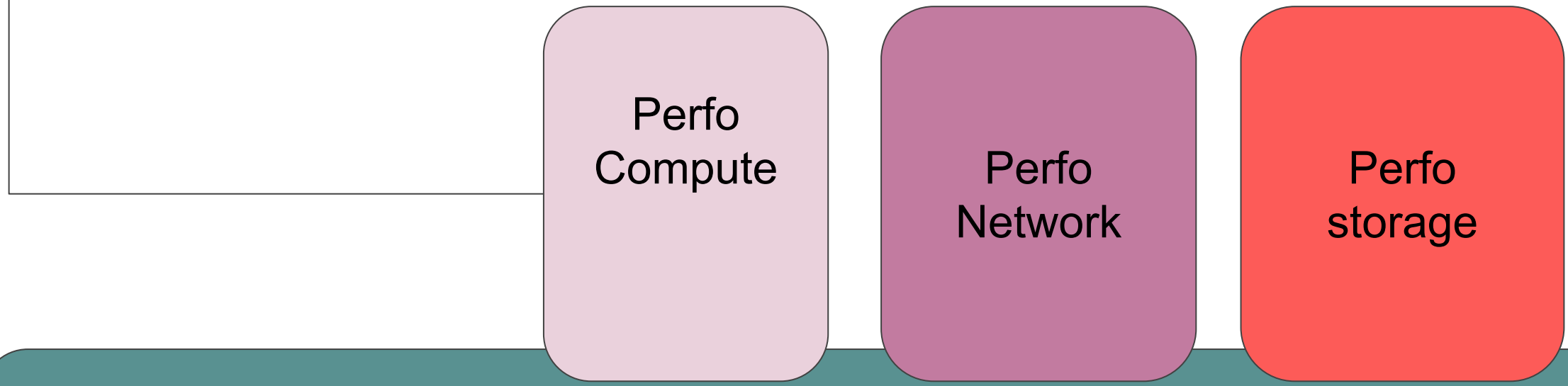
- Leverage existing Open Source TOM tool
  - to postprocess the results
  - set dynamically the thresholds of the KPIS
    - standard threshold: 50%
    - performance threshold: 95%

NB:

- tool introduced in Danube (with Demo during Beijing Summit) https://wiki.opnfv.org/display/testing/R+post-processing+of+the+Yardstick+results
- Try it: https://tomyardstick.sigmant.net/

Certification Profile

$\forall$ test $\in T_{Compute}${TC003, ...TC69 }, result(test) > KPI_{perfo}(test)



Standard profile:
All functional tests PASS
All performance:
$\forall$test $\in$ T(compute, storage, network), result(test) > $KPI_{standard}$ (test)

Example

TC012: The purpose of TC012 is to evaluate the IaaS compute performance with regards to memory throughput. It measures the rate at which data can be read from and written to the memory (this includes all levels of memory).

Metric: Memory read/write bandwidth (MBps)

Domain: Compute

Data collected 1/1/2018 - 6/10/2018: 11182 >> 500

Collect the results

TOM Yardstick | Data | Benchmarking | Analysis

Data from

- (•) Database query
- ( ) CSV file upload

Database query

Measurement

opnfv_yardstick_tc012

Date range:

2018-01-01 - 2018-10-06

Import Data

```
Data from query: SELECT * FROM opnfv_yardstick_tc012 WHERE time>='2018-01-01' AND time<='2018-10-06'
```

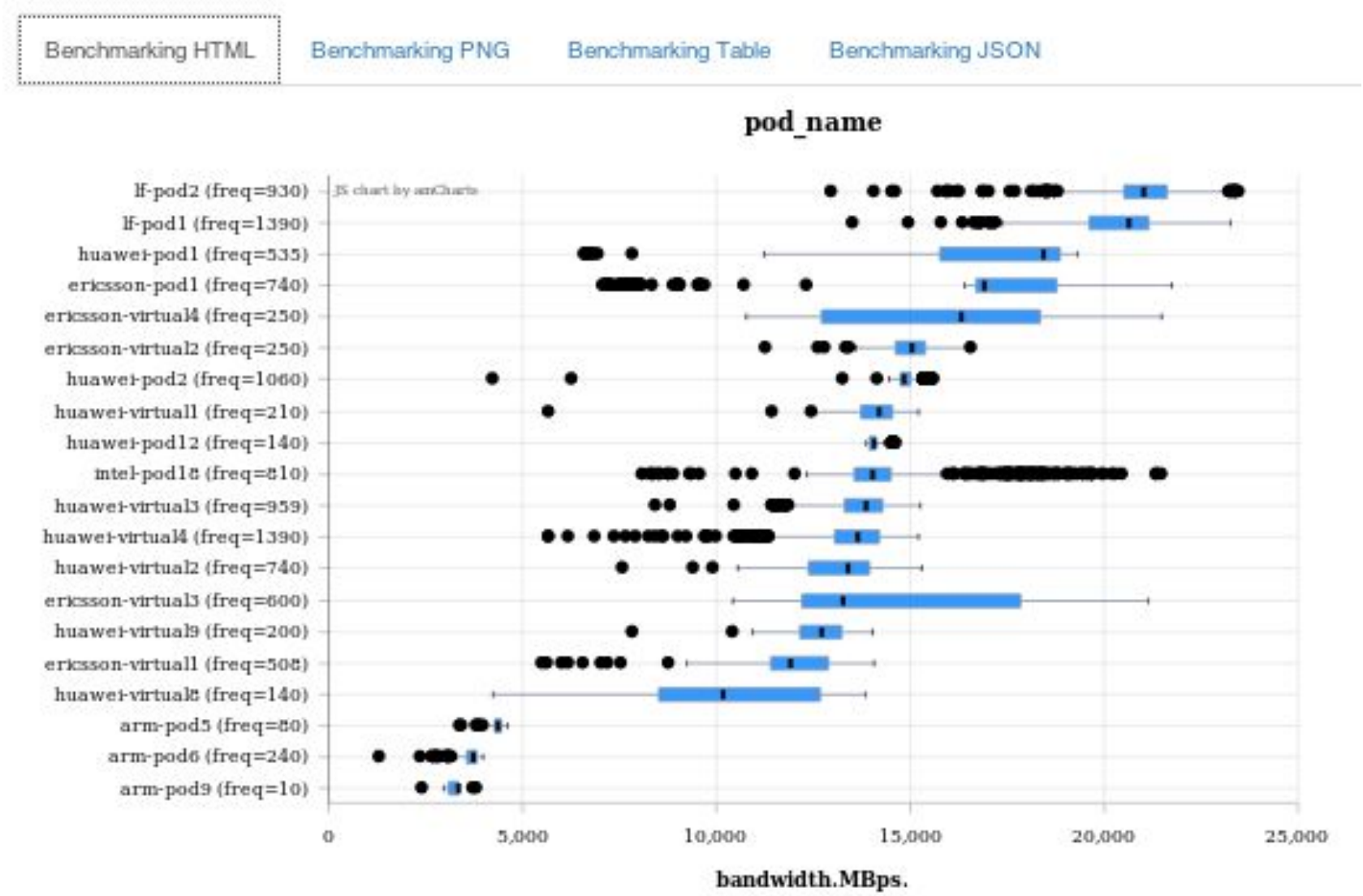
Show 5 entries Search:

	bandwidth.MBps.	deploy_scenario	host	installer	pod_name	runner_id	scenarios	size.MB.	task_id	version
1	13697.35	os-nosdn-nofeature-ha		compass	huawei-virtual2			10.49	d5985cf9-a3fb-4a16-b4d0-9565a05bf4c7	master
2	13865.47	os-nosdn-nofeature-ha		compass	huawei-virtual2			10.49	d5985cf9-a3fb-4a16-b4d0-9565a05bf4c7	master
3	13973.56	os-nosdn-nofeature-ha		compass	huawei-virtual2			10.49	d5985cf9-a3fb-4a16-b4d0-9565a05bf4c7	master
4	13932.25	os-nosdn-nofeature-ha		compass	huawei-virtual2			10.49	d5985cf9-a3fb-4a16-b4d0-9565a05bf4c7	master
5	13851.73	os-nosdn-nofeature-ha		compass	huawei-virtual2			10.49	d5985cf9-a3fb-4a16-b4d0-9565a05bf4c7	master

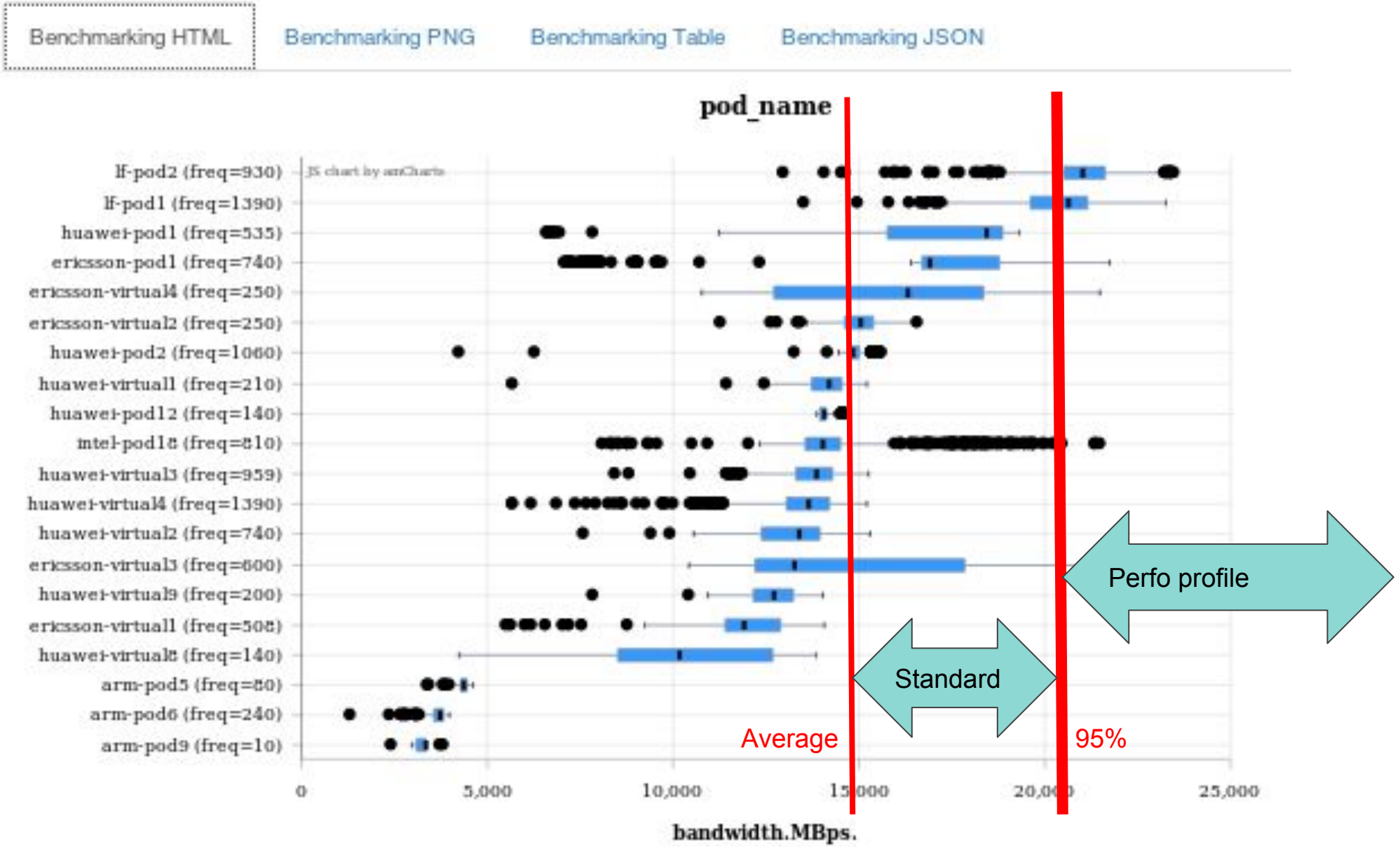
Showing 1 to 5 of 11,182 entries Previous 1 2 3 4 5 … 2237 Next

Visualize the results

Considering the most representative param - usually the pod_name=hardware



Set the thresholds



Dynamic thresholding

- Interesting because reflect Hardware
  - statistical effect (HA/No HA/performant/Standard Hardware)
  - if new performant hardware introduced in Pharos, the threshold will automatically increase
- Possible if enough values (> 500 and avoid hysteresis effect)
  - Study to see how to reference external labs to consolidate the results
-

Open questions

- Is Functest + Yardstick scope sufficient?
  - Both projects involved in OPNFV since the beginning
  - Functest leveraging active upstream projects from OpenStack and Kubernetes
  - if not, call to contribute to these projects…
  - NFVBench to be integrated in Yarsdstick?
- Can the Yardstick TC00X tests be post processed using TOM to get 1 dynamic KPI
- Is NSB mature enough to include first VNF/Infra KPIs (not seen today as TC00X)

Open questions

- Shall we still consider OPNFV scenario?
  - according to TOM: influence of the scenario << influence of the Hardware
    - on TC012: Hardware = 65%, scenario weights for 13% in the results
  - Lots of scenarios in Colorado/Danube but few scenarios maintained over the different versions
    - os-nosdn-nofeature-ha
    - os-odl-nofeature-ha
- We need representative results
  - Gating quality is decreasing
    - less OPNFV installations at the moment => less test results
    - less scenarios
  - Shall we open the data collections to other lab? (not officially in Pharos)

Thanks