



# **Yardstick Test Results for OPNFV Brahmaputra Release**

*Release draft (4d239f2)*

**OPNFV**

March 23, 2016



## CONTENTS

<b>1</b>	<b>Yardstick Test Report</b>	<b>1</b>
1.1	Introduction . . . . .	1
1.2	General . . . . .	1
<b>2</b>	<b>Yardstick Test Results</b>	<b>3</b>
2.1	Scenario Results . . . . .	3
2.2	Feature Test Results . . . . .	16



## YARDSTICK TEST REPORT

### 1.1 Introduction

#### 1.1.1 Document Identifier

This document is part of deliverables of the OPNFV release brahmaputra.1.0

#### 1.1.2 Scope

This document provides an overview of the results of test cases developed by the OPNFV Yardstick Project, executed on OPNFV community labs.

OPNFV Continuous Integration provides automated build, deploy and testing for the software developed in OPNFV. Unless stated, the reported tests are automated via Jenkins Jobs.

Test results are visible in the following dashboard:

- **Yardstick Dashboard:** uses influx DB to store test results and Grafana for visualization (user: opnfv/ password: opnfv)

#### 1.1.3 References

- IEEE Std 829-2008. “Standard for Software and System Test Documentation”.
- OPNFV Brahmaputra release note for Yardstick.

### 1.2 General

Yardstick Test Cases have been executed for scenarios and features defined in this OPNFV release.

The test environments were installed by one of the following: Apex, Compass, Fuel or Joid; one single installer per POD.

The results of executed tests are available in [Dashboard](#) and all logs stored in [Jenkins](#).

After one week of measurements, in general, SDN ONOS showed lower latency than SDN ODL, which showed lower latency than an environment installed with pure OpenStack. Additional time and PODs make this a non-conclusive statement, see [Scenarios](#) for a snapshot and [Dashboard](#) for complete results.

It was not possible to execute the entire Yardstick test cases suite on the PODs assigned for release verification over a longer period of time, due to continuous work on the software components and blocking faults either on environment, feature or test framework.

Four consecutive successful runs was defined as criteria for release. It is a recommendation to run Yardstick test cases over a longer period of time in order to better understand the behavior of the system.

### 1.2.1 Document change procedures and history

<b>Project</b>	Yardstick
<b>Repo/tag</b>	yardstick/brahmaputra.1.0
<b>Release designation</b>	Brahmaputra
<b>Release date</b>	Feb 25th, 2016
<b>Purpose of the delivery</b>	OPNFV Brahmaputra release test results.

## YARDSTICK TEST RESULTS

### 2.1 Scenario Results

The following documents contain results of Yardstick test cases executed on OPNFV labs, triggered by OPNFV CI pipeline, documented per scenario.

#### 2.1.1 Ready scenarios

The following scenarios run at least four consecutive times Yardstick test cases suite:

##### Test Results for apex-os-odl\_l2-nofeature-ha

###### Details

**Overview of test results** See [Dashboard](#) for viewing test result metrics for each respective test case.

All of the test case results below are based on scenario test runs on the LF POD1, between February 19 and February 24.

**TC002** The round-trip-time (RTT) between 2 VMs on different blades is measured using ping.

The results for the observed period show minimum 0.37ms, maximum 0.49ms, average 0.45ms. SLA set to 10 ms, only used as a reference; no value has yet been defined by OPNFV.

**TC005** The IO read bandwidth for the observed period show average between 124KB/s and 129 KB/s, with a minimum 372KB/s and maximum 448KB/s.

SLA set to 400KB/s, only used as a reference; no value has yet been defined by OPNFV.

**TC010** The measurements for memory latency for various sizes and strides are shown in [Dashboard](#). For 48MB, the minimum is 22.75 and maximum 30.77 ns.

SLA set to 30 ns, only used as a reference; no value has yet been defined by OPNFV.

**TC011** Packet delay variation between 2 VMs on different blades is measured using Iperf3.

The minimum packet delay variation measured is 2.5us and the maximum 8.6us.

**TC012** See [Dashboard](#) for results.

SLA set to 15 GB/s, only used as a reference, no value has yet been defined by OPNFV.

**TC014** The Unixbench processor single and parallel speed scores show scores between 3625 and 3660.

No SLA set.

**TC037** See [Dashboard](#) for results.

**Detailed test results** The scenario was run on LF [POD1](#) with: Apex ODL Beryllium

**Rationale for decisions** Pass

Tests were successfully executed and metrics collected. No SLA was verified. To be decided on in next release of OPNFV.

**Conclusions and recommendations** Execute tests over a longer period of time, with time reference to versions of components, for allowing better understanding of the behavior of the system.

### Test Results for compass-os-nosdn-nofeature-ha

#### Details

**Overview of test results** See [Grafana](#) for viewing test result metrics for each respective test case. It is possible to chose which specific scenarios to look at, and then to zoom in on the details of each run test scenario as well.

All of the test case results below are based on 5 consecutive scenario test runs, each run on the Huawei [SC\\_POD](#) between February 13 and 18 in 2016. The best would be to have more runs to draw better conclusions from, but these are the only runs available at the time of OPNFV R2 release

**TC002** The round-trip-time (RTT) between 2 VMs on different blades is measured using ping. The measurements are on average varying between 1.95 and 2.23 ms with a first 2 - 3.27 ms RTT spike in the beginning of each run (This could be because of normal ARP handling).SLA set to 10 ms. The SLA value is used as a reference, it has not been defined by OPNFV.

**TC005** The IO read bandwidth look similar between different test runs, with an average at approx. 145-162 MB/s. Within each run the results vary much, minimum 2MB/s and maximum 712MB/s on the totality. SLA set to 400KB/s. The SLA value is used as a reference, it has not been defined by OPNFV.

**TC010** The measurements for memory latency are consistent among test runs and results in approx. 1.2 ns. The variations between runs are similar, between 1.215 and 1.278 ns. SLA set to 30 ns. The SLA value is used as a reference, it has not been defined by OPNFV.

**TC011** For this scenario no results are available to report on. Probable reason is an integer/floating point issue regarding how InfluxDB is populated with result data from the test runs.



**TC012** The average measurements for memory bandwidth are consistent among most of the different test runs at 12.98 - 16.73 GB/s. The last test run averages at 16.67 GB/s. Within each run the results vary, with minimal BW of 16.59 GB/s and maximum of 16.71 GB/s of the totality. SLA set to 15 GB/s. The SLA value is used as a reference, it has not been defined by OPNFV.

**TC014** The Unixbench processor single and parallel speed scores show similar results at approx. 3000. The runs vary between scores 2499 and 3105. No SLA set.

**TC027** The round-trip-time (RTT) between VM1 with ipv6 router on different blades is measured using ping6. The measurements are consistent at approx. 4 ms. SLA set to 30 ms. The SLA value is used as a reference, it has not been defined by OPNFV.

**TC037** The amount of packets per second (PPS) and round trip times (RTT) between 2 VMs on different blades are measured when increasing the amount of UDP flows sent between the VMs using pktgen as packet generator tool.

Round trip times and packet throughput between VMs are typically affected by the amount of flows set up and result in higher RTT and less PPS throughput.

When running with less than 10000 flows the results are flat and consistent. RTT is then approx. 30 ms and the number of PPS remains flat at approx. 230000 PPS. Beyond approx. 10000 flows and up to 1000000 (one million) there is an even drop in RTT and PPS performance, eventually ending up at approx. 105-113 ms and 100000 PPS respectively.

**TC040** test purpose is to verify the function of Yang-to-Tosca in Parse, and this test case is a weekly task, so it was triggered by manually, the result whether the output is same with expected outcome is success No SLA set.

**Detailed test results** The scenario was run on Huawei SC\_POD with: Compass 1.0 OpenStack Liberty OVS 2.4.0 No SDN controller installed

**Rationale for decisions** Pass

Tests were successfully executed and metrics collect (apart from *TC011*). No SLA was verified. To be decided on in next release of OPNFV.

**Conclusions and recommendations** The pktgen test configuration has a relatively large base effect on RTT in TC037 compared to TC002, where there is no background load at all (30 ms compared to 1 ms or less, which is more than a 3000 percentage different in RTT results). The larger amounts of flows in TC037 generate worse RTT results, in the magnitude of several hundreds of milliseconds. It would be interesting to also make and compare all these measurements to completely (optimized) bare metal machines running native Linux with all other relevant tools available, e.g. lmbench, pktgen etc.

## Test Results for compass-os-odl\_l2-nofeature-ha

### Details

**Overview of test results** See [Dashboard](#) for viewing test result metrics for each respective test case.

All of the test case results below are based on scenario test runs on the Huawei Sclara.

**TC002** See [Dashboard](#) for results. SLA set to 10 ms, only used as a reference; no value has yet been defined by OPNFV.

**TC005** See [Dashboard](#) for results. SLA set to 400KB/s, only used as a reference; no value has yet been defined by OPNFV.

**TC010** See [Dashboard](#) for results. SLA set to 30ns, only used as a reference; no value has yet been defined by OPNFV.

**TC011** See [Dashboard](#) for results.

**TC012** See [Dashboard](#) for results. SLA set to 15 GB/s, only used as a reference; no value has yet been defined by OPNFV.

**TC014** See [Dashboard](#) for results. No SLA set.

**TC037** See [Dashboard](#) for results.

**Detailed test results** The scenario was run on Huawei [Sclara](#) POD with: Compass ODL Beryllium

**Rationale for decisions** Pass

Tests were successfully executed and metrics collected. No SLA was verified. To be decided on in next release of OPNFV.

**Conclusions and recommendations** Execute tests over a longer period of time, with time reference to versions of components, for allowing better understanding of the behavior of the system.

### Test Results for compass-os-onos-nofeature-ha

#### Details

**view of test results** See [Dashboard](#) for viewing test result metrics for each respective test case.

All of the test case results below are based on scenario test runs on the Huawei [Sclara](#).

**TC002** See [Dashboard](#) for results. SLA set to 10 ms, only used as a reference; no value has yet been defined by OPNFV.

**TC005** See [Dashboard](#) for results. SLA set to 400KB/s, only used as a reference; no value has yet been defined by OPNFV.

**TC010** See [Dashboard](#) for results. SLA set to 30ns, only used as a reference; no value has yet been defined by OPNFV.

**TC011** See [Dashboard](#) for results.

**TC012** See [Dashboard](#) for results. SLA set to 15 GB/s, only used as a reference; no value has yet been defined by OPNFV.

**TC014** See [Dashboard](#) for results. No SLA set.

**TC037** See [Dashboard](#) for results.

**Detailed test results** The scenario was run on Huawei [Sclara](#) POD with: Compass ONOS

**Rationale for decisions** Pass

Tests were successfully executed and metrics collected. No SLA was verified. To be decided on in next release of OPNFV.

**Conclusions and recommendations** Execute tests over a longer period of time, with time reference to versions of components, for allowing better understanding of the behavior of the system.

## Test Results for fuel-os-nosdn-nofeature-ha

### Details

**Overview of test results** See [Grafana](#) for viewing test result metrics for each respective test case. It is possible to chose which specific scenarios to look at, and then to zoom in on the details of each run test scenario as well.

All of the test case results below are based on 5 consecutive scenario test runs, each run on the Ericsson [POD2](#) between February 13 and 18 in 2016. The best would be to have more runs to draw better conclusions from, but these are the only runs available at the time of OPNFV R2 release.

**TC002** The round-trip-time (RTT) between 2 VMs on different blades is measured using ping. The measurements are on average varying between 0.5 and 1.1 ms with a first 2 - 2.5 ms RTT spike in the beginning of each run (This could be because of normal ARP handling). The 2 last runs are very similar in their results. But, to be able to draw any further conclusions more runs should be made. There is one measurement taken on February 16 that does not have the first RTT spike, and less variations to the RTT. The reason for this is unknown. There is a discussion on another test measurement made Feb. 16 in [TC037](#). SLA set to 10 ms. The SLA value is used as a reference, it has not been defined by OPNFV.

**TC005** The IO read bandwidth look similar between different test runs, with an average at approx. 160-170 MB/s. Within each run the results vary much, minimum 2 MB/s and maximum 630 MB/s on the totality. Most runs have a minimum of 3 MB/s (one run at 2 MB/s). The maximum BW varies much more in absolute numbers, between 566 and 630 MB/s. SLA set to 400 MB/s. The SLA value is used as a reference, it has not been defined by OPNFV.

**TC010** The measurements for memory latency are consistent among test runs and results in approx. 1.2 ns. The variations between runs are similar, between 1.215 and 1.219 ns. One exception is February 16, where the variation is greater, between 1.22 and 1.28 ns. SLA set to 30 ns. The SLA value is used as a reference, it has not been defined by OPNFV.

**TC011** For this scenario no results are available to report on. Probable reason is an integer/floating point issue regarding how InfluxDB is populated with result data from the test runs.

**TC012** The average measurements for memory bandwidth are consistent among most of the different test runs at 17.2 - 17.3 GB/s. The very first test run averages at 17.7 GB/s. Within each run the results vary, with a minimal BW of 15.4 GB/s and maximum of 18.2 GB/s of the totality. SLA set to 15 GB/s. The SLA value is used as a reference, it has not been defined by OPNFV.

**TC014** The Unixbench processor single and parallel speed scores show similar results at approx. 3200. The runs vary between scores 3160 and 3240. No SLA set.

**TC037** The amount of packets per second (PPS) and round trip times (RTT) between 2 VMs on different blades are measured when increasing the amount of UDP flows sent between the VMs using pktgen as packet generator tool.

Round trip times and packet throughput between VMs are typically affected by the amount of flows set up and result in higher RTT and less PPS throughput.

When running with less than 10000 flows the results are flat and consistent. RTT is then approx. 30 ms and the number of PPS remains flat at approx. 250000 PPS. Beyond approx. 10000 flows and up to 1000000 (one million) there is an even drop in RTT and PPS performance, eventually ending up at approx. 150-250 ms and 40000 PPS respectively.

There is one measurement made February 16 that has slightly worse results compared to the other 4 measurements. The reason for this is unknown. For instance anyone being logged onto the POD can be of relevance for such a disturbance.

**Detailed test results** The scenario was run on Ericsson [POD2](#) with: Fuel 8.0 OpenStack Liberty OVS 2.3.1

No SDN controller installed

**Rationale for decisions** Pass

Tests were successfully executed and metrics collects (apart from [TC011](#)). No SLA was verified. To be decided on in next release of OPNFV.

**Conclusions and recommendations** The pktgen test configuration has a relatively large base effect on RTT in TC037 compared to TC002, where there is no background load at all (30 ms compared to 1 ms or less, which is more than a 3000 percentage different in RTT results). The larger amounts of flows in TC037 generate worse RTT results, in the magnitude of several hundreds of milliseconds. It would be interesting to also make and compare all these measurements to completely (optimized) bare metal machines running native Linux with all other relevant tools available, e.g. lmbench, pktgen etc.

## Test Results for fuel-os-odl\_I2-nofeature-ha

### Details

**Overview of test results** See [Grafana](#) for viewing test result metrics for each respective test case. It is possible to chose which specific scenarios to look at, and then to zoom in on the details of each run test scenario as well.

All of the test case results below are based on 6 scenario test runs, each run on the Ericsson [POD2](#) between February 13 and 24 in 2016. Test case [TC011](#) is the greater exception for which there are only 2 test runs available, due to earlier problems with InfluxDB test result population. The best would be to have more runs to draw better conclusions from, but these are the only runs available at the time of OPNFV R2 release.

**TC002** The round-trip-time (RTT) between 2 VMs on different blades is measured using ping. Most test run measurements result on average between 0.3 and 0.5 ms, but one date (Feb. 23) sticks out with an RTT average of 1 ms. A few runs start with a 1 - 2 ms RTT spike (This could be because of normal ARP handling). One test run has a greater RTT spike of 3.9 ms, which is the same one with the 0.9 ms average. The other runs have no similar spike at all. To be able to draw conclusions more runs should be made. SLA set to 10 ms. The SLA value is used as a reference, it has not been defined by OPNFV.

**TC005** The IO read bandwidth looks similar between different dates, with an average between approx. 165 and 185 MB/s. Within each test run the results vary, with a minimum 2 MB/s and maximum 617 MB/s on the totality. Most runs have a minimum BW of 3 MB/s (two runs at 2 MB/s). The maximum BW varies more in absolute numbers between the dates, between 566 and 617 MB/s. SLA set to 400 MB/s. The SLA value is used as a reference, it has not been defined by OPNFV.

**TC010** The measurements for memory latency are similar between test dates and result in approx. 1.2 ns. The variations within each test run are similar, between 1.215 and 1.219 ns. One exception is February 16, where the average is 1.222 and varies between 1.22 and 1.28 ns. SLA set to 30 ns. The SLA value is used as a reference, it has not been defined by OPNFV.

**TC011** Only 2 test runs are available to report results on.

Packet delay variation between 2 VMs on different blades is measured using Iperf3. On the first date the reported packet delay variation varies between 0.0025 and 0.011 ms, with an average delay variation of 0.0067 ms. On the second date the delay variation varies between 0.002 and 0.006 ms, with an average delay variation of 0.004 ms.

**TC012** Results are reported for 5 test runs. It is not known why the 6:th test run is missing. Between test dates the average measurements for memory bandwidth vary between 17.4 and 17.9 GB/s. Within each test run the results vary more, with a minimal BW of 16.4 GB/s and maximum of 18.2 GB/s on the totality. SLA set to 15 GB/s. The SLA value is used as a reference, it has not been defined by OPNFV.

**TC014** Results are reported for 5 test runs. It is not known why the 6:th test run is missing. The Unixbench processor test run results vary between scores 3080 and 3240, one result each date. The average score on the total is 3150. No SLA set.

**TC037** Results are reported for 5 test runs. It is not currently known why the 6:th test run is missing. The amount of packets per second (PPS) and round trip times (RTT) between 2 VMs on different blades are measured when increasing the amount of UDP flows sent between the VMs using pktgen as packet generator tool.

Round trip times and packet throughput between VMs can typically be affected by the amount of flows set up and result in higher RTT and less PPS throughput.

The RTT results are similar throughout the different test dates and runs at approx. 15 ms. Some test runs show an increase with many flows, in the range towards 16 to 17 ms. One exception standing out is Feb. 15 where the average RTT is stable at approx. 13 ms. The PPS results are not as consistent as the RTT results. In some test runs when running with less than approx. 10000 flows the PPS throughput is normally flatter compared to when running with more flows, after which the PPS throughput decreases. Around 20 percent decrease in the worst case. For the other test runs there is however no significant change to the PPS throughput when the number of flows are increased. In some test runs the PPS is also greater with 1000000 flows compared to other test runs where the PPS result is less with only 2 flows.

The average PPS throughput in the different runs varies between 414000 and 452000 PPS. The total amount of packets in each test run is approx. 7500000 to 8200000 packets. One test run Feb. 15 sticks out with a PPS average of 558000 and approx. 1100000 packets in total (same as the on mentioned earlier for RTT results).

There are lost packets reported in most of the test runs. There is no observed correlation between the amount of flows and the amount of lost packets. The lost amount of packets normally range between 100 and 1000 per test run, but there are spikes in the range of 10000 lost packets as well, and even more in a rare cases.

**Detailed test results** The scenario was run on Ericsson **POD2** with: Fuel 8.0 OpenStack Liberty OpenVirtualSwitch 2.3.1 OpenDayLight Beryllium

**Rationale for decisions** Pass

Tests were successfully executed and metrics collected. No SLA was verified. To be decided on in next release of OPNFV.

**Conclusions and recommendations** The pktgen test configuration has a relatively large base effect on RTT in TC037 compared to TC002, where there is no background load at all. Approx. 15 ms compared to approx. 0.5 ms, which is more than a 3000 percentage difference in RTT results. Especially RTT and throughput come out with better results than for instance the *fuel-os-nosdn-nofeature-ha* scenario does. The reason for this should probably be further analyzed and understood. Also of interest could be to make further analyzes to find patterns and reasons for lost traffic. Also of interest could be to see if there are continuous variations where some test cases stand out with better or worse results than the general test case.

### Test Results for fuel-os-onos-nofeature-ha

#### Details

**Overview of test results** See [Grafana](#) for viewing test result metrics for each respective test case. It is possible to chose which specific scenarios to look at, and then to zoom in on the details of each run test scenario as well.

All of the test case results below are based on 7 scenario test runs, each run on the Ericsson **POD2** between February 13 and 21 in 2016. Test case *TC011* is not reported on due to an InfluxDB issue. The best would be to have more runs to draw better conclusions from, but these are the only runs available at the time of OPNFV R2 release.

**TC002** The round-trip-time (RTT) between 2 VMs on different blades is measured using ping. The majority (5) of the test run measurements result in an average between 0.4 and 0.5 ms. The other 2 dates stick out with an RTT average of 0.9 to 1 ms. The majority of the runs start with a 1 - 1.5 ms RTT spike (This could be because of normal ARP handling). One test run has a greater RTT spike of 4 ms, which is the same one with the 1 ms RTT average. The other runs have no similar spike at all. To be able to draw conclusions more runs should be made. SLA set to 10 ms. The SLA value is used as a reference, it has not been defined by OPNFV.

**TC005** The IO read bandwidth looks similar between different dates, with an average between approx. 170 and 185 MB/s. Within each test run the results vary, with a minimum of 2 MB/s and maximum of 690MB/s on the totality. Most runs have a minimum BW of 3 MB/s (one run at 2 MB/s). The maximum BW varies more in absolute numbers between the dates, between 560 and 690 MB/s. SLA set to 400 MB/s. The SLA value is used as a reference, it has not been defined by OPNFV.

**TC010** The measurements for memory latency are similar between test dates and result in a little less average than 1.22 ns. The variations within each test run are similar, between 1.213 and 1.226 ns. One exception is the first date, where the average is 1.223 and varies between 1.215 and 1.275 ns. SLA set to 30 ns. The SLA value is used as a reference, it has not been defined by OPNFV.

**TC011** For this scenario no results are available to report on. Reason is an integer/floating point issue regarding how InfluxDB is populated with result data from the test runs. The issue was fixed but not in time to produce input for this report.

**TC012** Between test dates the average measurements for memory bandwidth vary between 17.1 and 18.1 GB/s. Within each test run the results vary more, with a minimal BW of 15.5 GB/s and maximum of 18.2 GB/s on the totality. SLA set to 15 GB/s. The SLA value is used as a reference, it has not been defined by OPNFV.

**TC014** The Unixbench processor test run results vary between scores 3100 and 3260, one result each date. The average score on the total is 3170. No SLA set.

**TC037** The amount of packets per second (PPS) and round trip times (RTT) between 2 VMs on different blades are measured when increasing the amount of UDP flows sent between the VMs using pktgen as packet generator tool.

Round trip times and packet throughput between VMs can typically be affected by the amount of flows set up and result in higher RTT and less PPS throughput.

There seems to be mainly two result types. One type a high and flatter PPS throughput not very much affected by the number of flows. Here also the average RTT is stable around 13 ms throughout all the test runs.

The second type starts with a slightly lower PPS in the beginning than type one, and decreases even further when passing approx. 10000 flows. Here also the average RTT tends to start at approx. 15 ms ending with an average of 17 to 18 ms with the maximum amount of flows running.

Result type one can with the maximum amount of flows have a greater PPS than the second type with the minimum amount of flows.

For result type one the average PPS throughput in the different runs varies between 399000 and 447000 PPS. The total amount of packets in each test run is between approx. 7000000 and 10200000 packets. The second result type has a PPS average of between 602000 and 621000 PPS and a total packet amount between 10900000 and 13500000 packets.

There are lost packets reported in many of the test runs. There is no observed correlation between the amount of flows and the amount of lost packets. The lost amount of packets normally range between 100 and 1000 per test run, but there are spikes in the range of 10000 lost packets as well, and even more in a rare cases. Some case is in the range of one million lost packets.

**Detailed test results** The scenario was run on Ericsson **POD2** with: Fuel 8.0 OpenStack Liberty OpenVirtualSwitch 2.3.1 OpenNetworkOperatingSystem Drake

**Rationale for decisions** Pass

Tests were successfully executed and metrics collected. No SLA was verified. To be decided on in next release of OPNFV.

**Conclusions and recommendations** The pktgen test configuration has a relatively large base effect on RTT in TC037 compared to TC002, where there is no background load at all. Approx. 15 ms compared to approx. 0.5 ms, which is more than a 3000 percentage difference in RTT results. Especially RTT and throughput come out with better results than for instance the *fuel-os-nosdn-nofeature-ha* scenario does. The reason for this should probably be further analyzed and understood. Also of interest could be to make further analyzes to find patterns and reasons for lost traffic. Also of interest could be to see why there are variations in some test cases, especially visible in TC037.

### Test Results for joid-os-odl\_I2-nofeature-ha

#### Details

**Overview of test results** See [Dashboard](#) for viewing test result metrics for each respective test case.

All of the test case results below are based on scenario test runs on the Orange POD2, between February 23 and February 24.

**TC002** See [Dashboard](#) for results. SLA set to 10 ms, only used as a reference; no value has yet been defined by OPNFV.

**TC005** See [Dashboard](#) for results. SLA set to 400KB/s, only used as a reference; no value has yet been defined by OPNFV.

**TC010** Not executed, missing in the test suite used in the POD during the observed period.

**TC011** Not executed, missing in the test suite used in the POD during the observed period.

**TC012** Not executed, missing in the test suite used in the POD during the observed period.

**TC014** Not executed, missing in the test suite used in the POD during the observed period.

**TC037** See [Dashboard](#) for results.

**Detailed test results** The scenario was run on Orange [POD2](#) with: Joid ODL Beryllium

**Rationale for decisions** Pass

Most tests were successfully executed and metrics collected, the non-execution of above-mentioned tests was due to test cases missing in the Jenkins Job used in the POD, during the observed period. No SLA was verified. To be decided on in next release of OPNFV.

**Conclusions and recommendations** Execute tests over a longer period of time, with time reference to versions of components, for allowing better understanding of the behavior of the system.

### Test Results for yardstick-opnfv-rtc

#### Details

**Overview of test results** The virtual Traffic Classifier (rtc) Scenario supported by Yardstick is used by 4 Test Cases:

- TC006
- TC007
- TC020
- TC021



- TC006

TC006 is the Virtual Traffic Classifier Data Plane Throughput Benchmarking Test. It collects measures about the end-to-end throughput supported by the virtual Traffic Classifier (vTC). Results of the test are shown in the [Dashboard006](#). The throughput is expressed as percentage of the available bandwidth on the NIC.

- TC007

TC007 is the Virtual Traffic Classifier Data Plane Throughput Benchmarking in presence of noisy neighbors Test. It collects measures about the end-to-end throughput supported by the virtual Traffic Classifier when a user-defined number of noisy neighbors is deployed. Results of the test are shown in the [Dashboard007](#). The throughput is expressed as percentage of the available bandwidth on the NIC.

- TC020

TC020 is the Virtual Traffic Classifier Instantiation Test. It verifies that a newly instantiated vTC is alive and functional and its instantiation is correctly supported by the underlying infrastructure. Results of the test are shown in the [Dashboard020](#).

- TC021

TC021 is the Virtual Traffic Classifier Instantiation in presence of noisy neighbors Test. It verifies that a newly instantiated vTC is alive and functional and its instantiation is correctly supported by the underlying infrastructure when noisy neighbors are present. Results of the test are shown in the [Dashboard021](#).

#### Detailed test results

- TC006

The results for TC006 have been obtained using the following test case configuration:

- Context: Dummy
- Scenario: vtc\_throughput
- Network Technology: SR-IOV
- vTC Flavor: m1.large
- TC007

The results for TC007 have been obtained using the following test case configuration:

- Context: Dummy
- Scenario: vtc\_throughput\_noisy
- Network Technology: SR-IOV
- vTC Flavor: m1.large
- Number of noisy neighbors: 2
- Number of cores per neighbor: 2
- Amount of RAM per neighbor: 1G
- TC020

The results for TC020 have been obtained using the following test case configuration:

The results listed in previous section have been obtained using the following test case configuration:

- Context: Dummy
- Scenario: vtc\_instantiation\_validation

- Network Technology: SR-IOV
- vTC Flavor: m1.large
- TC021

The results listed in previous section have been obtained using the following test case configuration:

- Context: Dummy
- Scenario: vtc\_instantiation\_validation
- Network Technology: SR-IOV
- vTC Flavor: m1.large
- Number of noisy neighbors: 2
- Number of cores per neighbor: 2
- Amount of RAM per neighbor: 1G

For all the test cases, the user can specify different values for the parameters.

### **Rationale for decisions**

- TC006

The result of the test is a number between 0 and 100 which represents the percentage of bandwidth available on the NIC that corresponds to the supported throughput by the vTC.

- TC007

The result of the test is a number between 0 and 100 which represents the percentage of bandwidth available on the NIC that corresponds to the supported throughput by the vTC.

- TC020

The execution of the test is done as described in the following:

- The vTC is deployed on the OpenStack testbed;
- Some traffic is sent to the vTC;
- The vTC changes the header of the packets and sends them back to the packet generator;
- The packet generator checks that all the packets are received correctly and have been changed

correctly by the vTC.

The test is declared as PASSED if all the packets are correctly received by the packet generator and they have been modified by the virtual Traffic Classifier as required.

- TC021

The execution of the test is done as described in the following:

- The vTC is deployed on the OpenStack testbed;
- The noisy neighbors are deployed as requested by the user;
- Some traffic is sent to the vTC;
- The vTC change the header of the packets and sends them back to the packet generator;
- The packet generator checks that all the packets are received correctly and have been changed

correctly by the vTC

The test is declared as PASSED if all the packets are correctly received by the packet generator and they have been modified by the virtual Traffic Classifier as required.

### Conclusions and recommendations

- TC006

The obtained results show that the virtual Traffic Classifier can support up to 4 Gbps (40% of the available bandwidth) correspond to the expected behaviour of the virtual Traffic Classifier. Using the configuration with SR-IOV and large flavor, the expected throughput should generally be in the range between 3 and 4 Gbps.

- TC007

These results correspond to the configuration in which the virtual Traffic Classifier uses SR-IOV Virtual Functions and the flavor is set to large for the virtual machine. The throughput is in the range between 2.5 Gbps and 3.7 Gbps. This shows that the effect of 2 noisy neighbors reduces the throughput of the service between 10 and 20%. Increasing number of neighbours would have a higher impact on the performance.

- TC020

The obtained results correspond to the expected behaviour of the virtual Traffic Classifier. Using the configuration with SR-IOV and large flavor, the expected result is that the vTC is correctly instantiated, it is able to receive and send packets using SR-IOV technology and to forward packets back to the packet generator changing the TCP/IP header as required.

- TC021

The obtained results correspond to the expected behaviour of the virtual Traffic Classifier. Using the configuration with SR-IOV and large flavor, the expected result is that the vTC is correctly instantiated, it is able to receive and send packets using SR-IOV technology and to forward packets back to the packet generator changing the TCP/IP header as required, also in presence of noisy neighbors.

## 2.1.2 Limitations

The following scenarios run at least one time Yardstick test cases suite, partially or complete:

- fuel-os-odl\_12-sfc-ha
- fuel-os-odl\_12-bgpvpn-ha
- fuel-os-odl\_13-nofeature-ha
- joid-os-nosdn-nofeature-ha

Test results of executed tests are available in [Dashboard](#) and logs in [Jenkins](#).

## 2.1.3 Not executed

For the following scenarios, Yardstick test cases were not run:

- apex-os-odl\_12-sfc-noha
- apex-os-odl\_13-nofeature-ha
- apex-os-onos-nofeature-ha
- fuel-os-nosdn-kvm-ha
- fuel-os-nosdn-ovs-ha

- fuel-os-odl\_13-nofeature-ha
- joid-os-onos-nofeature-ha

Logs are available in [Jenkins](#).

## 2.2 Feature Test Results

The following features were verified by Yardstick test cases:

- IPv6
- HA (see `yardstick-opnfv-ha`)
- KVM
- Parser

---

**Note:** The test cases for IPv6 and Parser Projects are included in the compass scenario.

---