



# Yardstick Release Note

*Release draft (af74d0a)*

**OPNFV**

March 23, 2016



CONTENTS

<b>1</b>	<b>OPNFV Brahmaputra Release Note for Yardstick</b>	<b>1</b>
1.1	Abstract . . . . .	1
1.2	License . . . . .	1
1.3	Version History . . . . .	1
1.4	Important Notes . . . . .	1
1.5	Summary . . . . .	1
1.6	Release Data . . . . .	2



# OPNFV BRAHMAPUTRA RELEASE NOTE FOR YARDSTICK

## 1.1 Abstract

This document compiles the release notes for the OPNFV Brahmaputra release for Yardstick framework as well as Yardstick Project deliverables.

## 1.2 License

The *Yardstick framework*, the *Yardstick test cases* and the *ApexLake* experimental framework are opensource software, licensed under the terms of the Apache License, Version 2.0.

## 1.3 Version History

<i>Date</i>	<i>Version</i>	<i>Comment</i>
Feb 25th,2016	1.0	Brahmaputra release

## 1.4 Important Notes

The software delivered in the OPNFV *Yardstick* Project, comprising the *Yardstick framework*, the *Yardstick test cases* and the experimental framework *Apex Lake* is a realization of the methodology in ETSI-ISG *NFV-TST001*.

The *Yardstick* framework is *installer*, *infrastructure* and *application* independent.

## 1.5 Summary

This Brahmaputra release provides *Yardstick* as a framework for NFVI testing and OPNFV feature testing, automated in the OPNFV CI pipeline, including:

- Documentation generated with Sphinx
  - User Guide
  - Code Documentation
  - Release notes (this document)
  - Results

- Automated Yardstick test suite (daily, weekly)
  - Jenkins Jobs for OPNFV community labs
- Automated Yardstick test results visualization
  - [Dashboard](#) using Grafana (user:opnfv/password: opnfv), influxDB used as backend
- Yardstick framework source code
- Yardstick test cases yaml files

For Brahma Putra release, the *Yardstick framework* is used for the following testing:

- OPNFV platform testing - generic test cases to measure the categories:
  - Compute
  - Network
  - Storage
- Test cases for the following OPNFV Projects:
  - High Availability
  - IPv6
  - KVM
  - Parser

The *Yardstick framework* is developed in the OPNFV community, by the [Yardstick](#) team.

---

**Note:** The test case description template used for the Yardstick test cases is based on the document ETSI-ISG NFV-TST001; the results report template used for the Yardstick results is based on the IEEE Std 829-2008.

---

## 1.6 Release Data

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

### 1.6.1 Version Change

#### Module Version Changes

This is the first tracked release of Yardstick. It is based on following upstream versions:

- OpenStack Liberty
- OpenDaylight Beryllium

## Document Version Changes

This is the first tracked version of the Yardstick framework in OPNFV. It includes the following documentation:

- Yardstick User Guide
- Yardstick Code Documentation
- Yardstick Release Notes for Yardstick
- Test Results report for Brahmaputra testing with Yardstick

## 1.6.2 Reason for Version

### Feature additions

This is the first tracked version of OPNFV Yardstick.

### Corrected Faults

This is the first tracked version of OPNFV Yardstick.

### Known Faults

JIRA REFERENCE	SLOGAN
JIRA: YARDSTICK-175	Running test suite, if a test cases running failed, the test is stopped.
JIRA: YARDSTICK-176	Fix plotter bug since Output format has been changed.
JIRA: YARDSTICK-216	ArgsAlreadyParsedError: arguments already parsed: cannot register CLI option.

**Note:** The faults not related to *Yardstick* framework, addressing scenarios which were not fully verified, are listed in the OPNFV installer's release notes.

## 1.6.3 Deliverables

### Software Deliverables

Yardstick framework source code <brahmaputra.1.0>

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

### Contexts

Context	Description
<i>Heat</i>	Models orchestration using OpenStack Heat
<i>Node</i>	Models Baremetal, Controller, Compute

### Runners

Runner	Description
<i>Arithmetic</i>	Steps every run arithmetically according to specified input value
<i>Duration</i>	Runs for a specified period of time
<i>Iteration</i>	Runs for a specified number of iterations
<i>Sequence</i>	Selects input value to a scenario from an input file and runs all entries sequentially

### Scenarios

Category	Delivered
<i>Availability</i>	Attacker: <ul style="list-style-type: none"> <li>• baremetal, process</li> </ul> HA tools: <ul style="list-style-type: none"> <li>• check host, openstack, process, service</li> <li>• kill process</li> <li>• start/stop service</li> </ul> Monitor: <ul style="list-style-type: none"> <li>• command, process</li> </ul>
<i>Compute</i>	<ul style="list-style-type: none"> <li>• cpuload</li> <li>• cyclictst</li> <li>• lmbench</li> <li>• perf</li> <li>• unixbench</li> </ul>
<i>Networking</i>	<ul style="list-style-type: none"> <li>• iperf3</li> <li>• netperf</li> <li>• ping</li> <li>• ping6</li> <li>• pktgen</li> <li>• sfc</li> <li>• sfc with tacker</li> <li>• vtc instantiation validation</li> <li>• vtc instantiation validation with noisy neighbors</li> <li>• vtc throughput</li> <li>• vtc throughput in the presence of noisy neighbors</li> </ul>
<i>Parser</i>	Tosca2Heat
<i>Storage</i>	fio

### API to Other Frameworks

Frame-work	Description
<i>ApexLake</i>	Experimental framework that enables the user to validate NFVI from the perspective of a VNF. A virtual Traffic Classifier is utilized as VNF. Enables experiments with SR-IOV on Compute Node.

### Test Results Output

Dispatcher	Description
file	Log to a file.
http	Post data to html.
influxdb	Post data to influxdB.



## Delivered Test cases

- Generic NFVI test cases
  - OPNFV\_YARDSTICK\_TCOO1 - NW Performance
  - OPNFV\_YARDSTICK\_TCOO2 - NW Latency
  - OPNFV\_YARDSTICK\_TCOO5 - Storage Performance
  - OPNFV\_YARDSTICK\_TCOO8 - Packet Loss Extended Test
  - OPNFV\_YARDSTICK\_TCOO9 - Packet Loss
  - OPNFV\_YARDSTICK\_TCO10 - Memory Latency
  - OPNFV\_YARDSTICK\_TCO11 - Packet Delay Variation Between VMs
  - OPNFV\_YARDSTICK\_TCO12 - Memory Bandwidth
  - OPNFV\_YARDSTICK\_TCO14 - Processing Speed
  - OPNFV\_YARDSTICK\_TCO24 - CPU Load
  - OPNFV\_YARDSTICK\_TCO37 - Latency, CPU Load, Throughput, Packet Loss
  - OPNFV\_YARDSTICK\_TCO38 - Latency, CPU Load, Throughput, Packet Loss Extended Test
- Test Cases for OPNFV HA Project:
  - OPNFV\_YARDSTICK\_TCO19 - HA: Control node Openstack service down
  - OPNFV\_YARDSTICK\_TC025 - HA: OpenStack Controller Node abnormally down
- Test Case for OPNFV IPv6 Project:
  - OPNFV\_YARDSTICK\_TCO27 - IPv6 connectivity
- Test Case for OPNFV KVM Project:
  - OPNFV\_YARDSTICK\_TCO28 - KVM Latency measurements
- Test Case for OPNFV Parser Project:
  - OPNFV\_YARDSTICK\_TCO40 - Verify Parser Yang-to-Tosca