ETSI GS NFV-TST 009 V3.1.1 (2018-10)

GROUP SPECIFICATION

Network Functions Virtualisation (NFV) Release 3; Testing; Specification of Networking Benchmarks and Measurement Methods for NFVI

Disclaimer

Reference
DGS/NFV-TST009

Keywords
benchmarking, measurement, NFV, NFVI

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Contents

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org/).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

Foreword

This Group Specification (GS) has been produced by ETSI Industry Specification Group (ISG) Network Functions Virtualisation (NFV).

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Executive summary

This unnumbered clause, if present, appears after the "Modal verbs terminology" and before the "Introduction". It is **an optional** informative element and **shall not contain requirements**.

The "Executive summary" is used, if required, to summarize the ETSI deliverable. It contains enough information for the readers to become acquainted with the full document without reading it. It is usually one page or shorter.

Introduction

The widespread adoption of virtualised implementation of functions has brought about many changes and challenges for the testing and benchmarking industries. The subjects of the tests perform their functions within a virtualisation system for additional convenience and flexibility, but virtualised implementations also bring challenges to measure their performance in a reliable and repeatable way, now that the natural boundaries and dedicated connectivity of physical network functions are gone. Even the hardware testing systems have virtualised counterparts, presenting additional factors to consider in the pursuit of accurate results.

The present document draws on learnings from many early benchmarking campaigns and years of benchmarking physical network functions to develop and specify new normative benchmarks and methods of measurement to characterize the performance of networks in the Network Function Virtualisation Infrastructure.

1 Scope

The present document specifies vendor-agnostic definitions of performance metrics and the associated methods of measurement for Benchmarking networks supported in the NFVI. The Benchmarks and Methods will take into account the communication-affecting aspects of the compute/networking/virtualisation environment (such as the transient interrupts that block other processes or the ability to dedicate variable amounts of resources to communication processes). These Benchmarks are intended to serve as a basis for fair comparison of different implementations of NFVI, (composed of various hardware and software components) according to each individual Benchmark and networking configuration evaluated. Note that a Virtual Infrastructure Manager (VIM) may play a supporting role in configuring the network under test. Examples of existing Benchmarks include IETF RFC 2544 [1] Throughput and Latency (developed for physical network functions).

2 References

2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found at https://docbox.etsi.org/Reference.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are necessary for the application of the present document.

[1] IETF RFC 2544 (March 1999): "Benchmarking Methodology for Network Interconnect Devices".

[2] IETF RFC 2285 (February 1998): "Benchmarking Terminology for LAN Switching Devices".

[3] IETF RFC 2889 (August 2000): "Benchmarking Methodology for LAN Switching Devices".

[4] IETF RFC 6985 (July 2013): "IMIX Genome: Specification of Variable Packet Sizes for Additional Testing".

[5] ETSI GS NFV-TST 008 (V3.1.1): "Network Functions Virtualisation (NFV) Release 2; Testing; NFVI Compute and Network Metrics Specification".

[6] ETSI GS NFV 003 (V1.3.1): "Network Functions Virtualisation (NFV); Terminology for Main Concepts in NFV".

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] ETSI GS NFV-INF 003 (V.1.1.1): "Network Functions Virtualisation (NFV); Infrastructure; Compute Domain".

[i.2] Bregni, Stefano, (October 1996): "Measurement of Maximum Time Interval Error for Telecommunications Clock Stability Characterization", IEEE Transactions on Instrumentation and Measurement.

NOTE: Available at http://home.deib.polimi.it/bregni/papers/mtiemeas.pdf.

[i.3] ETSI GS NFV-TST 001 (V1.1.1): "Network Functions Virtualisation (NFV); Pre-deployment Testing; Report on Validation of NFV Environments and Services".

[i.4] FD.io VPP Developer Documentation: "Shared Memory Packet Interface (memif) Library".

NOTE: Available at https://docs.fd.io/vpp/17.10/libmemif_doc.html.

[i.5] FD.io VPP pma-tools: "Software tools for performance and efficiency measurements".

NOTE: Available at https://git.fd.io/pma_tools/tree/jitter.

[i.6] John D. McCalpin: "STREAM: Sustainable Memory Bandwidth in High Performance Computers".

NOTE: Available at https://www.cs.virginia.edu/stream/.

[i.7] Intel Software Developer Zone: "Intel® Memory Latency Checker v3.5".

NOTE: Available at https://software.intel.com/en-us/articles/intelr-memory-latency-checker.

[i.8] IETF RFC 8172 (July 2017): "Considerations for Benchmarking Virtual Network Functions and Their Infrastructure".

[i.9] IETF RFC 8204 (July 2017): "Benchmarking Virtual Switches in the Open Platform for NFV (OPNFV)".

[i.10] Cooper, T., Morton, A., Rao, S., OPNFV Summit Presentation: "Dataplane Performance, Capacity, and Benchmarking in OPNFV", June 2017.

NOTE: Available at https://wiki.opnfv.org/download/attachments/10293193/VSPERF-Dataplane-Perf-Cap-Bench.pptx?api=v2.

[i.11] IETF RFC 5481 (March 2009): "Packet Delay Variation Applicability Statement".

[i.12] Wikipedia: "Binary Search".

NOTE: Available at https://en.wikipedia.org/wiki/Binary_search_algorithm.

[i.13] GeeksforGeeks Computer Science Portal: "Searching Algorithms, Binary Search".

NOTE: Available at https://www.geeksforgeeks.org/binary-search/ (including code implementations).

[i.14] Andrzej Pelc: "Searching games with errors - fifty years of coping with liars", Theoretical Computer Science 270 (2002) pp 71-109.

NOTE: Available at https://www.gwern.net/docs/statistics/comparison/2002-pelc.pdf.

[i.15] Presentation to IETF-102 BMWG Session: "Evolution of Repeatability in Benchmarking: Fraser Plugfest (Summary for IETF BMWG)", Sridhar Rao, Al Morton, and OPNFV VSPERF Project Team.

NOTE: Available at https://datatracker.ietf.org/meeting/102/materials/slides-102-bmwg-evolution-of-repeatability-in-benchmarking-fraser-plugfest-summary-for-ietf-bmwg-00.

[i.16] LightReading/EANTC NFV Tests and Trials: "Validating Cisco's NFV Infrastructure", October 2015.

NOTE: Available at http://www.lightreading.com/nfv/nfv-tests-and-trials/validating-ciscos-nfv-infrastructure-pt-1/d/d-id/718684.

[i.17] Wikipedia: "OPNFV NFVbench Project Description".

NOTE: Available at https://wiki.opnfv.org/display/nfvbench.

[i.18] Python Package Index MLResearch 0.1.1.0: "Library for speeding up binary search using shorter measurements".

NOTE: Available at https://pypi.org/project/MLRsearch/ and https://docs.fd.io/csit/master/report/introduction/methodology.html#mlrsearch-algorithm.

[i.19] Recommendation ITU-T Y.1541 (December 2011): "Network performance objectives for IP-based services".

[i.20] IETF RFC 8337 (March 2018): "Model-based Metrics".

[i.21] IETF RFC 7348: "Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualised Layer 2 Networks over Layer 3 Networks".

3 Definitions of terms and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in ETSI GS NFV 003 [6] and the following apply:

NOTE: A term defined in the present document takes precedence over the definition of the same term, if any, in ETSI GS NFV 003 [6].

burst: stream of two or more frames transmitted with the minimum allowable inter-frame gap, as in a burst of frames

bursty traffic rate: stream consisting of repeated bursts that maintains a specified frequency of transmitted frames per second, such that the frequency equals the reciprocal of the sum of the constant inter-burst gap and the burst serialization time (for a constant frame size and minimum allowable inter-frame gaps between frames in the burst)

NOTE: See section 21 of IETF RFC 2544 [1].

constant frame rate stream: stream that maintains a specified frequency of transmitted frames per second, such that the frequency equals the reciprocal of the sum of the constant inter-frame gap and the frame serialization time (for a constant frame size)

flow: set of frames or packets with the same n-tuple of designated header fields that (when held constant) result in identical treatment in a multi-path decision (such as the decision taken in load balancing)

frame size: fixed length of a frame in octets (or 8-bit bytes), all headers included

NOTE: For example, Ethernet frame size includes the frame CRC, but exclude the transmission overhead per frame of 20 octets (the preamble and the inter frame gap).

measurement goal: specific criteria that a measurement result is expected to meet to satisfy the requirements of a benchmark definition

method: series of one or more Sets of Tests conducted to achieve a measurement goal

offered load: both the count (in frames) and transmission rate (in frames per second) generated by the measurement system during a trial, including both directions of transmission with bi-directional streams

pod: partition of a compute node that provides an isolated virtualised computation environment, for one or more virtualisation containers in the context of an Operating System Container virtualisation layer

set: series of one or more tests conducted to achieve a measurement goal

stream: population of frames or packets with various header attributes, that contain one or more flows and comprise the Offered Load

trial: single iteration of a measurement producing one or more results that can be compared with search termination criteria

test: series of one or more trials conducted to achieve a measurement goal

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

AV	Array with loss Verification
BIOS	Basic Input Output System
BLV	Binasry search with Loss Verification
CPU	Central Processing Unit
CRC	Cyclic Redundancy Check
DCI	Data Center Interconnect
DUT	Device Under Test
DV	Delay Variation
FDV	Frame Delay Variation
GRE	Generic Routing Encapsulation
IFDV	Inter-Frame Delay Variation
IMIX	Internet Mix (of frame or packet sizes)
IP	Internet Protocol
L2	Layer 2
MTIE	Maximum Time Interval Error
NA	Not Available
NDR	No Drop Rate
NFV	Network Function Virtualisation
NFVI	Network Function Virtualisation Infrastructure
NSH	Network Service Header
NUMA	Non-Uniform Memory Access
OPNFV	Open Platform for NFV
OS	Operating System
PCIe	Peripheral Component Interconnect express
PDR	Partial Drop Rate
PVP	Physical Virtual Physical (Single VNF)
PVVP	Physical Virtual Virtual Physical (Multiple VNF)
SDN	Software Defined Network
SFC	Service Function Chains
SR-IOV	Single Root - Input Output Virtualisation
SRv6	Segment Routing over IPv6 dataplane
SUT	System Under Test
TCP	Transmission Control Protocol
Thpt	Throughput
VethX	Virtual Ethernet Interface
VIM	Virtual Infrastructure Manager
VLAN	Virtual Local Area Network
VM	Virtual Machine
VNF	Virtual Network Function
VNFC	VNF Component
vSw	virtual Switch
VSPERF	vSwitch Performance Project in OPNFV
VTEP	Virtual extensible local area network - Tunnel End Point
VXLAN	Virtual eXtensible Local Area Network

NOTE: See IETF RFC 7348 [i.21].

VXLAN-GPE	Virtual eXtensible Local Area Network - Generic Protocol Extension

4 Time and Time Intervals for Metrics and Benchmarks

In the present document, coherent compute domains comprise the System Under Test (SUT) [i.3] which contains one or more Devices Under Test (DUT), the platform for software-based measurement systems, and may also provide the automation platform for overall SUT, DUT, and test measurement system installation and configuration. The requirements of this clause are particularly relevant to test devices or test functions (such as the Traffic Generator or Receiver), and are generally referred to as measurement systems.

Coherent compute domains [i.1] usually need access to a clock with accurate time-of-day (or simply date-time) and sources of periodic interrupts. Time sources are accessed to provide timestamps for events and log entries that document the recent history of the compute environment. Periodic interrupts provide a trigger to increment counters and read current conditions in the compute and networking environments. The compute domain may contain a very large number of NFV compute nodes [i.1], and each node needs to execute a process to synchronize its hardware and system clocks to a source of accurate time-of -day, preferably traceable to an international time standard.

With the foundation of time, date, and periodic interrupts, a measurement system can determine the beginning and end of time intervals, which is a fundamental aspect of metrics that involve counting and collecting events (such as frame or packet transmission and reception), and a SUT or DUT can provide accurate event logging and other functions.

Table 4-1 specifies requirements applicable to time, date, and periodic interrupts for all systems, and includes an additional Requirement for Test Devices/Functions (General-Time-03).

Table 4-1: Requirements applicable to time, date and periodic interrupts

General-Time-01	Each node in the compute domain shall be able to take readings from (or access) a clock with accurate time-of-day and calendar date, having less than ± one millisecond error from an international time standard over a 900 second measurement interval.
General-Time-02	Each node in the compute domain shall have a source of periodic interrupts available which are derived from the time-of-day clock, with configurable period (a parameter of metrics that use this feature).
General-Time-03	The Maximum Time Interval Error (MTIE) between the local clock of the Test Device/Function and a reference time source shall be specified in terms of S, the observation interval, and TE, the maximum peak-to-peak deviation of time error. S is provisionally set at 120 seconds [i.2].

5 Framework for Metric and Benchmark Definitions

The metric and Benchmark definitions in the present document are primarily derived from the industry precedents established when IETF RFC 2544 [1] was first published.

For each metric and Benchmark it specifies, the present document provides the following template of elements in separate sub-clauses, many of which are required for reporting:

- Background
- Name
- Parameters (input factors)
- Scope of coverage (possibly a subset of resources tested)
- Unit(s) of measure
- Definition
- Method of Measurement (unique aspects, in addition to clause 11)
- Sources of Error
- Discussion
- Reporting Format

NOTE: The present document specifies Benchmarks and metrics, some of which are well-known by name, but whose definitions and other details are presented as normative requirements in the present document, according to the framework of elements above. See clauses 8 and above for the current implementation of this framework and template elements.

Where a clause specifies both a Benchmark and one or more metric-variants, the tester shall measure and report the Benchmark along with optional metric-variants in all testing campaigns claiming compliance with the present document.

6 Test Set-ups and Configuration

6.1 Goals of Benchmarking and Use Cases

The use cases supported by the testing effort shall be clearly identified, in order to select the applicable set of test setups. For example (referring to setups in clause 6.2, figure 6.2-1), testing the firewall-related functions of a vSwitch (using match-action rules) can be accomplished using the Phy2Phy setup, while a virtualised web host would typically require the PVP setup, and a chain of payload-processing VNFs (transcoding and encryption) would require the PVVP setup. The following list contains important considerations for Test Setups:

1) The use case(s) addressed by the goals of the test will help to determine whether tests with streams composed of multiple flows are necessary, and how the streams should be constructed in terms of the header fields that vary (such as address fields).

2) The use case(s) addressed by the goals of the test will help to determine what protocol combinations should be tested, and how the streams should be constructed in terms of the packet header composition and mix of packet sizes [4]. There may be one or more unique frame size associated with the particular use case of interest, and these may be revealed using traffic monitoring as described in IETF RFC 6985 [4].

3) At least two mixes of packet size should be specified for testing, sometimes called "IMIX" for Internet mix of sizes. IETF RFC 6985 [4] provides several ways to encode and specify mixes of packet sizes, and the RFC encourages repeatable packet size sequences by providing specifications to support reporting and reproducing a given sequence (called the IMIX Genome).

4) Bidirectional traffic should possess complimentary stream characteristics (some use cases involve different mixes of sizes in each direction).

5) Protocols like VLAN, VXLAN, GRE, VXLAN-GPE, SRv6 and SFC NSH are needed in NFVI deployments and some will require multiple encapsulations (VLAN and VXLAN).

6.2 Test Setups

The following general topologies for test setups should be used when called for by the target use cases. The test designers shall prepare a clear diagram of their test setup, including the designated addresses (Layer 2 and/or IP subnet assignments, if used). This diagram will help trouble resolution in a setup with connectivity issues.

Figure 6.2-1 illustrates the connectivity required for test device/functions, NFVI components, and testing-specific VNFs. Note that arrows indicate one direction of transmission (one-way), but bi-directional testing shall be performed as required. Also, the data plane switching/forwarding/routing function is illustrated in red and labelled "vSw". The number of Physical port pairs in use may also be expanded as required (only one pair is shown).

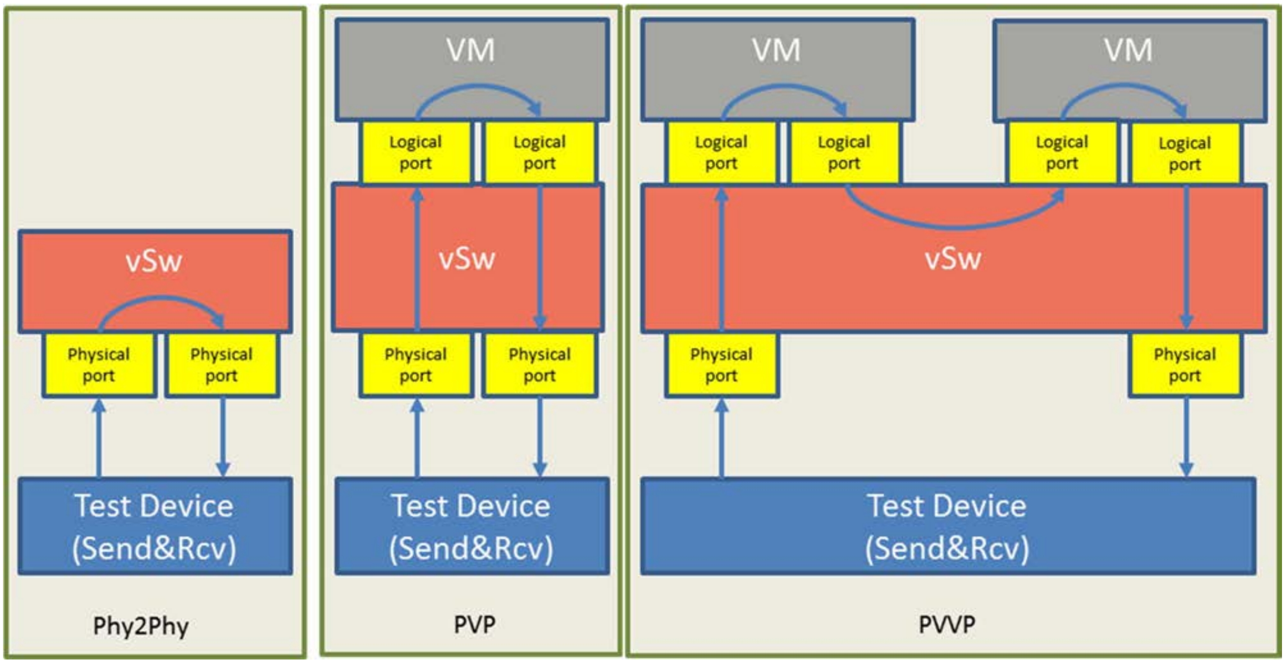

Figure 6.2-1: General Dataplane Test Setups in a single host: Baseline(Phy2Phy), Single VNF (PVP), Multiple VNF (PVVP)

A control-protocol may be used between the "vSw" and an SDN controller, but this control plane connectivity is not illustrated in figure 6.2-1.

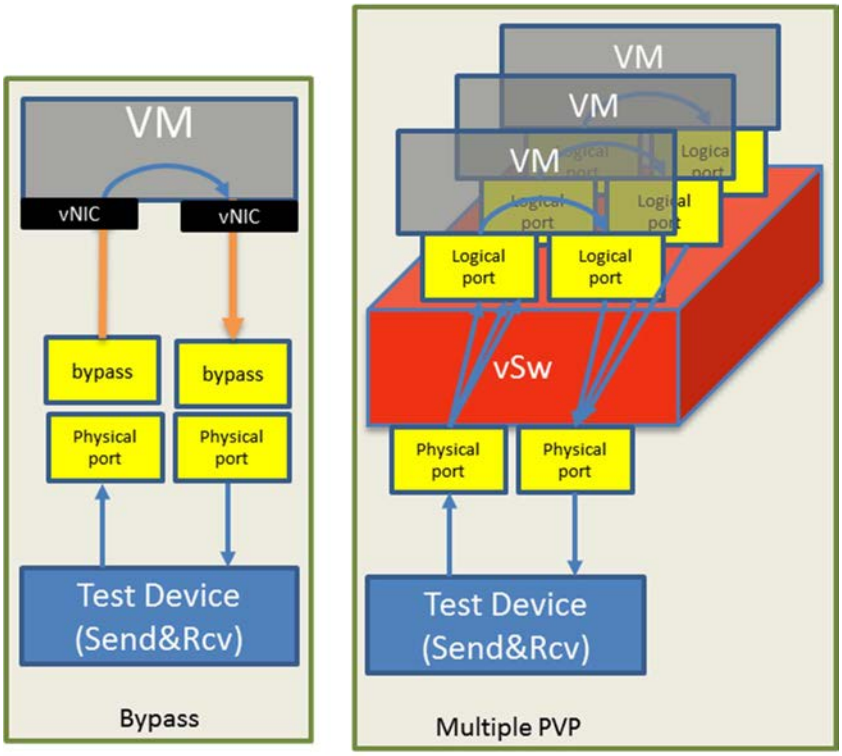

Figure 6.2-2: Additional Data plane Test Setups in a single host: Bypass and Multiple PVP

Figure 6.2-2 illustrates additional test setups, where Bypass permits testing of technologies such as Single-Root Input Output Virtualisation (SR-IOV), and Multiple PVP accommodates testing with many parallel paths through VMs. Of course, the Multiple VNF setup (PVVP) can be expanded with additional VMs in series. Mesh connectivity could also be applied between the vSwitch and multiple physical port interconnections, similar to the one to many connectivity illustrated in the Multiple PVP scenario in figure 6.2-2. In the limit, many test setups could be deployed simultaneously in order to benchmark a "full" SUT, with or without CPU oversubscription.

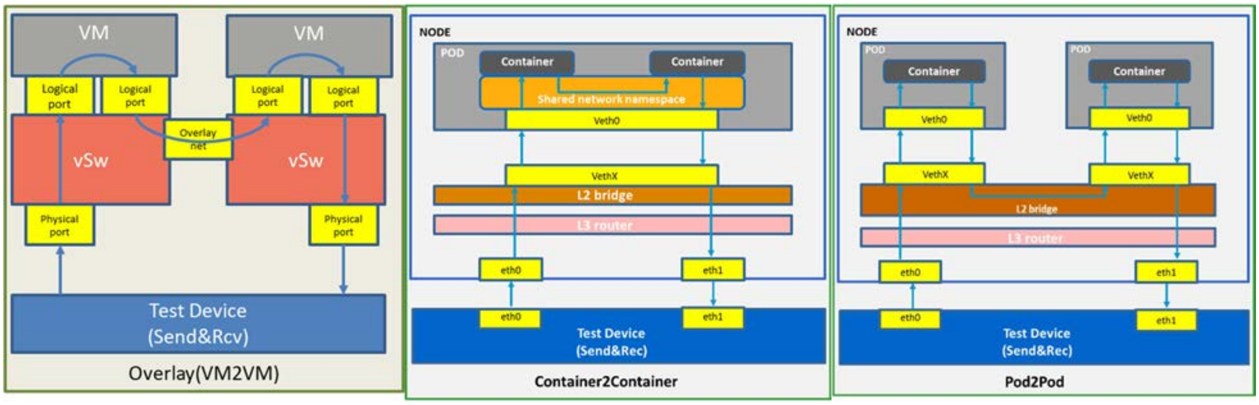

Figure 6.2-3: Additional Data plane Test Setups in a single host: Overlay(VM2VM), Container2Container and Pod2Pod

Figure 6.2-3 illustrates some of the unusual connectivity scenarios that are possible, using overlay networks to connect between VMs, the capability of namespace networking for Container to Container communications in Operating Systems Containers, or the L2 bridge provided by Container Orchestration System for Pod to Pod communication in the same node. Bypass technologies for more direct access are also possible with Containers (SR-IOV), and Container to Container communication may take place through shared packet memory [i.4].

In a common Container Infrastructure, the node represents a physical host. The pod represents the basic unit for orchestration and management which is able to host multiple containers. For internal communication, the L2 bridge is used to connect all the pods. For external communication, the L3 router may be used to manage the traffic in and out of the node.

In figure 6.2-3 Container2Container, the containers running in the same pod may communicate with each other through shared network namespace. The pod only has one virtual Ethernet interface. When one pod is created in the node, the L2 bridge will create a virtual Ethernet interface (VethX) to communicate with the pod's virtual Ethernet interface (Veth0).

Pod2Pod in figure 6.2-3 is also designed for multiple VNFs and VNFCs scenario. Based on the security or privacy consideration, it may be not allowed to deploy multiple containers in the same pod. When each container is deployed in different pod, the communication between pods in same node is also managed by the L2 bridge.

It is important to note that Pod2Pod and Container2Container networking scenarios are a topic of active development and many other connectivity options will be possible, so the current diagrams should not be interpreted as mandating specific setups.

In Figures 6.2-1, 6.2-2, and 6.2-3, the Test Device is shown connected to only two physical ports for simplicity, and this scenario is also common in production. In reality, most NFVI deployments will involve many physical ports, with varied activation of paths between them. However, some physical ports may serve control or storage-related traffic.

The term "mesh" is used to describe the completeness of paths utilized between the ports. IETF RFC 2889 [3] requires tests of a Full mesh of traffic, which means that the each physical port under test receives frames intended for all other physical ports under test. Use of Link Aggregation and Channel Bonding will often create a full-mesh physical ports and VMs. A Partial mesh of traffic (section 3.3 of IETF RFC 2285 [2]) means that at least one port under test (a physical or logical port) receives traffic for two or more other (physical or logical) ports as shown in figure 6.2-2, and may be further described as connectivity of one-to-many ports, many-to-one port, and many-to-many ports, where input and output ports are mutually exclusive (as with Full mesh traffic). Partial meshes shall be described using the number of input ports and the number of output ports along with the connectivity between them. If different types of connectivity are used in the same test set-up, the number of ports involved with each type shall be described separately.

Figures 6.2-1 through 6.2-3 also allow the option for traffic on physical ports to be tagged with VXLAN or other tags. Many of the real world scenarios (Data Center Interconnect (DCI), or VXLAN Tunnel End Point (VTEP) on the host) would require such a performance measurement. A VXLAN VTEP may be placed in pairs of VMs or on vSwitch. In the former case it is transparent to vSwtich; the latter case is an example of a partial mesh for VXLAN.

Another form of communications path exists within the NFVI: the physical relationship among the compute, cache, memory resources and PCIe for network I/O, referred to here as Physical Compute Environment Allocation. In a simple example, figure 6.2-4 illustrates allocation of all aspects of a PVP test setup to a single NUMA node in a dual processor system.

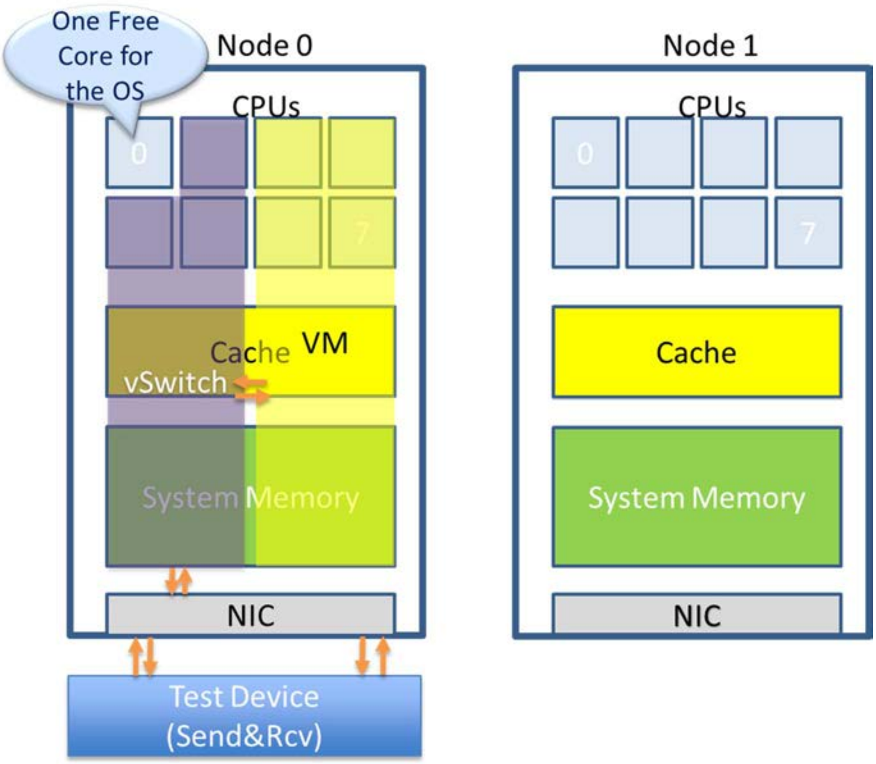

Figure 6.2-4: Example Physical Compute Environment Allocation to a Single NUMA node

Of course, allocation of either the vSwitch or VM to NUMA node 1 would change the performance measured, but cross-NUMA deployment may be required due to placement limitations, resource limitations (ports), High Availability designs, or economic limitations. It is recommended to test some scenarios where the tested resources are allotted to multiple NUMA nodes, as they can produce higher latency and lower throughput. Scenarios could involve some cross node traffic or where all traffic traverses the NUMA nodes. Examples include vSwitch data plane threads on NUMA node 0, VNF data plane threads on NUMA node 1, or VNFs and NICs spread across two nodes. The process of explicit environment allocation (or, the process of determining what resources were allocated after the NFVI completes an abstract set of requests), is more akin to configuration (see below).

6.3 Configuration

The configuration of measured NFVI and the measurement systems shall be completely specified in the dimensions of model numbers, firmware versions, software versions, and any other aspects that influence the results (such as physical location of the components within a datacentre's racks and shelves). For example, the fixed frequency of the physical CPU clock in Hz, which governs the rate that the CPU executes instructions, is one important descriptor of the NFVI. Clock Speed may depend on other CPU settings, such as energy-saving power control and clock acceleration features. For one list of NFVI platform descriptors, see clause 5.2 of ETSI GS NFV-INF 003 [i.1]. Knowledge of the configuration parameter list has been found to be a critical contributor to repeatable results.

The following points should be taken into consideration as part of Pre-test activities:

1) Management of the large number of configuration variables could be assisted by automated collection of software and hardware configuration in all meaningful dimensions, with simplified comparison between the current and previously recorded configurations to highlight any differences. This task is likely to be more practical with a simplified SUT having the minimum software packages installed, and comparatively more difficult with a full production deployment on an SUT (however, it appears that both scenarios need working solutions). It may be necessary to develop an advanced difference-seeking method, in order to make the configuration comparisons useful.

2) It is important to identify the difference between absolute configuration parameters with deterministic implementation, and abstract configuration variables where resource allocation can be arbitrary or include placement with affinity and other concepts, and prefer the use of absolute configuration whenever possible. The abstract form of configuration may be satisfied using one set of resources which are selected when the configuration/command is executed, and different resources when executed a second time, or after a re-boot, etc. The actual implementation of an abstract configuration might be discovered using the automated collection capability described above. Some ways to ensure setup consistency in benchmarking tests:

   - Use automated setup with an installer that guarantees a deterministic and identical software stack every time.
   - Read the actual (absolute) configuration and compare with blueprint or with SUT instances. There work required to automate this process, and practitioners could consider using an approach involving configuration modelling.

3) Measuring/calibrating SUT sub-system runtime properties as part of pre-test activities. This could include measuring following:

   - System level core jitter - measure actual duration of core interrupts in clock cycles, and how often the cores are interrupted by the OS, with a jitter tool, [i.5] for example.
   - Memory latency - with [i.6] or Intel's [i.7], for example.
   - Memory bandwidth - with [i.7], for example.
   - Cache latency at all levels (L1, L2 and Last Level Cache) - with [i.7], for example.

7 Test Device/Function Capabilities

7.1 Traffic Generator Requirements

The following requirements for Traffic Generator capabilities are considered necessary for the most commonly used benchmarks and metrics:

1) The Generator shall be capable of accurately generating and transmitting a range of constant frame rates, including the maximum allowable frame rate for the physical interface links.
2) The Generator shall be capable of accurately generating and transmitting a range of bursty traffic streams with configurable burst length in frames, up to the maximum link frame rate (with inter-frame gaps) for the physical interface links plus time between bursts.
3) The Generator shall be capable of generating transmission timestamps for each frame in the measured sample ("selected frames" in the benchmark definition) to support accurate latency measurements, and shall communicate each time stamp to the Receiver in a way that facilitates determining the correspondence between frames and their timestamps. Timestamps shall be applied as close to actual transmission as possible.

In addition, the present document specifies the following additional measurement-related capabilities:

1) The Traffic Generator should be capable of constructing all the streams required by the use cases.
2) The address field variation methods should be specified in the Traffic Generator configuration. The distribution of consecutive packets from a flow sent on the wire should also be specified using the inter-packet time interval (the time between the first bit of successive packets).
3) The Traffic Generator should be capable of constructing all the packet types required by the use cases.
4) The distribution or sequence of different frame sizes sent on the wire should also be specified in the Traffic Generator configuration, as per IETF RFC 6985 [4].

Test Traffic Generators should have the following user configurable Parameters:

- **Constant Frame Rate Stream:** specified in Frames per second.

- **Bursty Traffic Rate:** specified in terms of burst length and overall transmission rate in Frames per second.
- **Maximum Offered Load:** limit on Frame count transmitted during a Trial.
- **Constant Frame Size:** the fixed length of frames in octets (or 8-bit bytes), all headers included shall be specified.
- **Mixture of Frame Sizes:** the mixture of frame sizes, the proportion of frames of each size in each 10 second interval, and ideally the exact sequence of transmitted frame sizes, as specified in IETF RFC 6985 [4].
- **IP Address Family:** either IPv4, IPv6, or both in conditions with a mixture of frame sizes.
- **Multiple Flow Definitions:** configurable sets of n-tuples in terms of the header fields involved, and the value ranges and sequence of their changes for the designated header fields.
- **Minimum Trial Repetition Interval:** the minimum time to conduct a trial in seconds, which involves preparation of the SUT, offering load, waiting for queues to empty, and computing the measured result.
- **Trial Duration:** the time interval when the Offered Load is transmitted to the SUT, expressed in seconds.
- **Maximum Number of Trials:** the limit on trials attempted to achieve the measurement goal of the test. Reaching this limit implies an error condition or unstable measurements, and it is required to report this event.

7.2 Traffic Receiver Requirements

The following requirements for Traffic Receiver capabilities are considered necessary for the most commonly used benchmarks and metrics:

1) The Receiver shall be capable of loss-less reception of a range of constant frame rates, including the maximum allowable frame rate for the physical interface links, and bursty traffic streams.
2) The Receiver shall be capable of applying reception timestamps for each frame it receives (synchronized with the Generator's timestamp source) to support accurate latency measurements after determining the correspondence between received frames and their transmission timestamps. Timestamps shall be applied as close to actual reception as possible.

Traffic Generators and Traffic Receivers shall be capable of performing their functions simultaneously, to support both unidirectional and bi-directional testing.

Test Traffic Receivers should have the following user configurable Parameter:

- **Maximum X % Loss Ratio:** the configured allowable packet loss ratio for the Capacity with X% Loss metric. If all digits of the configured value are zero, the configuration is equal to zero loss where all packets sent shall have been received. For the Throughput Benchmark, X = 0. Lost frames shall be measured consistent with IETF RFC 2544 [1] and the Loss metric is defined in clause 11.

7.3 Test Device/Function Requirements

Test Devices/Functions claiming to measure a specific Benchmark (such as Throughput, Latency Frame Delay Variation, and Loss Ratio) and/or metric shall meet all requirements in the corresponding Benchmark clauses below, as well, especially the Method sub-clause for each Benchmark.

Test Devices/Functions performing automated execution of the Methods of Measurement shall meet all requirements in clause 12, as well.

8 Throughput

8.1 Background

Throughput is a key Benchmark for the NFVI. It is valuable to assess the maximum capacity of a SUT. This information can assist capacity planning and network engineering when conducted under circumstances relevant to the production network of interest.

8.2 Name

There is only one required Benchmark of Throughput, which requires zero-loss as a measurement goal and is consistent with IETF RFC 2544 [1]. An auxiliary metric of Capacity may be measured using a related method and procedure:

1) Throughput.

2) Capacity with X % Loss Ratio.

The X % Loss is a parameter described below.

8.3 Parameters

The following parameters shall be supported for the Benchmark and the Capacity metric variant:

- **Offered Load Frame Size:** the fixed frame size (or description of the mixture of sizes, or IMIX) transmitted by the measurement system when searching for the measurement goal.
- **Offered Load Step Size:** this parameter determines resolution or minimum variation between trials in the test. The step size is the minimum change configured in the measurement system when searching for the measurement goal. The step size forms part of the search termination criteria, by determining when the results are sufficiently close to an unknown ideal value. The step size may be specified in frames per second (an absolute number) or as a percentage of the Maximum Offered Load for the links connecting the traffic generator to the SUT, or as a percentage of load used in a previous trial.
- **Minimum Trial Repetition Interval:** the minimum time to conduct a trial in seconds, which involves preparation of the SUT, offering load, waiting for queues to empty, and computing the measured result.
- **Trial Duration:** the time interval when the Offered Load is transmitted to the SUT, expressed in seconds.
- **Maximum X % Loss Ratio:** the configured allowable packet loss ratio for the Capacity with X % Loss metric. If all digits of the configured value are zero, the configuration is the equal to zero loss where all packets sent are received. For the Throughput Benchmark, X = 0.
- **Maximum Number of Trials:** the limit on trials attempted to achieve the measurement goal of the test. Reaching this limit implies an error condition or unstable measurements, and it is required to report this event.

8.4 Scope

The list of one or more infrastructure resources which are included in the values reported, and whether the workload is physical or virtual.

8.5 Units of Measure

The Offered Load in Frames per second. The units of Packets per second would be equal to Frames per second.

NOTE: It is also acceptable to include the bits per second equivalent of the Offered Load and the percentage of the Maximum Offered Load.

8.6 Definition

The values of each metric at time T (for a given scope and measurement time) shall be assessed as the total Offered Load meeting the criteria as specified in table 8.6-1.

Table 8.6-1: Requirements for Benchmarks and metrics

Requirement number	Metric Name	Requirement
Throughput-01	Throughput	Reported values of this Benchmark shall quantify the maximum amount of Offered Load at the frame size selected for which the count of frames sent is equal to the frames received. This Benchmark is Throughput with Loss Ratio = 0 %, or zero packet loss.
Throughput-02	Capacity with X % Loss Ratio	Reported values of this metric variant shall quantify the maximum amount of Offered Load at the frame size selected for which the Loss ratio calculated as the quantity (frames sent minus frames received) over frames sent is less than or equal to X %.

8.7 Method of Measurement

This Benchmark and related metric variant employ the basic method and procedure defined in clause 12.

8.8 Sources of Error

The sources of error for this Benchmark and metric variant are listed below:

1) The Offered Load Step Size determines the resolution of the result. Larger step sizes will result in faster searches with fewer trials, but the error appears in a larger under-estimate of Throughput or Capacity.

2) Synchronization of the time-of-day clock with an external reference usually ensures sufficiently accurate timestamps, but loss of synchronization for an extended period will cause time accuracy to suffer.

8.9 Discussion

None.

8.10 Reporting Format

Ideally, the results of testing to assess Throughput or Capacity with X% Loss should be presented using a Figure with a two-dimensional graph.

The title of the graph shall include the metric name(s) and the Units of the y-axis (such as Throughput in Frames per second).

The Units of the x-axis shall be Frame Size in octets.

The Units of the y-axis shall be Frames per second, representing the Offered Load sent from the Traffic Generator to the SUT (and not the frame rate received by the Traffic Receiver). If the test traffic is bi-directional, then the sum of both traffic directions shall be plotted on the graph for each frame size tested. The y-axis may also present the corresponding bits per second scale, as an option.

For tests with variable Frame size [4], the results shall be plotted according to the average Frame size of the traffic stream (which includes the factors of Frame frequency and Frame size). If the fixed Frame size testing includes a size equal to the IMIX average size (an interesting comparison), then the results at these two sizes shall be distinguished with labelling when reported, such as 512 and IMIX512.

Along with the measured Frames per second for each Frame size, the Maximum theoretical Frames per second for all ingress interfaces used shall be plotted for each Frame size (for simple comparison purposes).

The caption of the Figure shall describe all relevant aspects of the test, in sufficient detail to inform a reader and to distinguish each Figure from similar Figures with results from different test conditions conducted as part of the same campaign (such as different test set-ups and protocols used). There shall also be a legend, identifying the benchmark and any other metric-variants reported (examples are shown in table 8.10-1).

If the graphical results are summarized as a single number, then the result of metric assessment for the smallest Frame size shall be reported, with Units of measure.

If the results are summarized in a table, then the example table 8.10-1 should be used.

Table 8.10-1: Example Throughput Tabular Reporting

Requirement number	Metric Name	Frame Size, octets			
		64	256	IMIX	1 518
Throughput-01	Throughput				
Throughput-02	Capacity with X % Loss				

The example above assumes IPv4 addresses, and IPv6 addresses would increase the frame size (64 octets becomes 84 octets, although smaller frame sizes are possible). Some forms of frame encapsulation may increase the minimum frame size, as well, such as VXLAN with its 50 octet header.

9 Latency

9.1 Background

Latency is another key Benchmark for the NFVI. It is valuable to assess the one-way frame transfer time of a SUT at various levels of offered load. This information can assist capacity planning and network engineering when conducted under circumstances relevant to the production network of interest.

9.2 Name

There are several metric variants and one required Benchmark of 99^{th} percentile of one-way Latency, which is measured at the Throughput Benchmark offered load level, and is consistent with IETF RFC 2544 [1]. The Latency Benchmark assumes that the latency will vary over the trial, and a high percentile is often more stable than the true maximum. Additional metric variants of Latency at other offered load levels may be measured using a related method and procedure.

The X % of Throughput Benchmark offered load level is a parameter described in clause 9.3.

9.3 Parameters

The following parameters shall be supported for the Latency Benchmark and other metric variants:

- **Offered Load Frame Size:** the fixed frame size (or description of the mixture of sizes, or IMIX) transmitted by the measurement system when searching for the measurement goal.
- **X % of Throughput Benchmark Offered Load:** the fraction of the offered load corresponding to the Throughput level as determined in prerequisite testing. For the Latency Benchmark, X = 100 %. When testing investigates conditions that may represent production deployment, X = 80 % or X = 50 % may be good choices for a metric variant.
- **Minimum Trial Repetition Interval:** the minimum time to conduct a trial in seconds, which involves preparation of the SUT, offering load, waiting for queues to empty, and computing the measured result.
- **Trial Duration:** the time interval when the Offered Load is transmitted to the SUT, expressed in seconds.
- **Sample Filter:** the description of the filter characteristic used to select frames from the population of all frames sent as offered load. For the Latency Benchmark, the sample filter selects all frames. If the Traffic

Generator & Receiver cannot measure latency on all frames, then the sample filter shall be configured to select the maximum number of frames and the result shall be reported as a metric variant.

- **Maximum Number of Trials:** the limit on trials attempted to achieve the measurement goal of the test. Reaching this limit implies an error condition or unstable measurements, and it is required to report this event.

9.4 Scope

The list of one or more infrastructure resources which are included in the values reported, and whether the workload is physical or virtual.

9.5 Units of Measure

The Latency or transfer time shall be expressed in microseconds.

NOTE: It is also acceptable to use metric adjectives to make the values more relevant, such as nanoseconds.

9.6 Definition

The values of each metric at time T (for a given scope and measurement time) shall be assessed as the statistical summary of filtered frame Latency meeting the criteria as specified in table 9.6-1.

Table 9.6-1: Requirements for Benchmarks and metrics

Requirement number	Metric Name	Requirement
Latency-01	Latency	Reported values of this Benchmark shall quantify the 99th percentile of one-way Latency for all filtered transfer times when the Offered Load is equal to the Throughput level.
Latency-02	Transfer Time (Yth percentile, X % Thpt, Filter)	Reported values of this metric variant shall quantify the Yth percentile of the selected one-way frame transfer times across the DUT/SUT when the Offered Load is equal to the X % of the Throughput level.
Latency-03	Minimum Transfer Time (X % Thpt, Filter)	Reported values of this metric variant shall quantify the Minimum of the selected one-way frame transfer times across the DUT/SUT when the Offered Load is equal to the X % of the Throughput level.
Latency-04	Mean Transfer Time (X % Thpt, Filter)	Reported values of this metric variant shall quantify the Mean of the selected one-way frame transfer times across the DUT/SUT when the Offered Load is equal to the X % of the Throughput level.
Latency-05	Maximum Transfer Time (X % Thpt, Filter)	Reported values of this metric variant shall quantify the Maximum of the selected one-way frame transfer times across the DUT/SUT when the Offered Load is equal to the X % of the Throughput level.

9.7 Method of Measurement

This Benchmark and related metric variant employ the basic method and procedure defined in clause 12.

9.8 Sources of Error

The sources of error for this Benchmark and metric variant are listed below:

1) The Offered Load Step Size of the prerequisite test determines the resolution of the result. Larger step sizes will result in faster searches with fewer trials, but the error appears in a larger under-estimate of Throughput or Capacity.

2) Synchronization of the time-of-day clock with an external reference usually ensures sufficiently accurate timestamps, but loss of synchronization for an extended period will cause time accuracy to suffer.

3) A calibration test could measure transit latency between traffic generator ports, and report it as a constant latency measurement error. The measurement can be completed by looping traffic generator ports and measuring latency at a set offered load, e.g. 10 % of link rate. This is especially important with software generators.

9.9 Discussion

The benchmark for frame Delay Variation and related metric variants are specified in clause 10.

9.10 Reporting Format

Ideally, the results of testing to assess Latency should be presented using a Figure with a two-dimensional graph.

The title of the graph shall include the metric name(s) and the Units of the y-axis (such as one-way Latency in microseconds, although a different scientific prefix may be used, such as nanoseconds).

The Units of the x-axis shall be Frame Size in octets.

The Units of the y-axis shall be one-way Frame transfer time in seconds (although scientific prefix may be used, such as microseconds). If the test traffic is bi-directional, then results for each of the directions may be plotted separately on the same graph for each frame size tested.

For tests with variable Frame size [4], the results shall be plotted according to the average Frame size of the traffic stream (which includes the factors of Frame frequency and Frame size).

Along with the measured one-way Latency in seconds for each Frame size, the Minimum Latency should be plotted (if measured) for each Frame size.

The caption of the Figure shall describe all relevant aspects of the test, in sufficient detail to inform a reader and to distinguish each Figure from similar Figures with results from different test conditions conducted as part of the same campaign (such as different test set-ups and protocols used). There shall also be a legend, identifying the benchmark and any other metric-variants plotted.

If the results are summarized as a single number, then the result of metric assessment for the smallest Frame size shall be reported, with Units of measure.

If the results are summarized in a table, then the example table 9.10-1 should be used.

Table 9.10-1: Example Latency Tabular Reporting

Requirement number	Metric Name	Frame Size, octets			
		64	256	IMIX	1 518
Latency-01	Latency Benchmark				
Latency-02	Transfer Time (Y^{th} percentile, X % Thpt, Filter)				
Latency-03	Minimum Transfer Time (X % Thpt, Filter)				
Latency-04	Mean Transfer Time (X % Thpt, Filter)				
Latency-05	Maximum Transfer Time (X % Thpt, Filter)				

The example above assumes IPv4 addresses, and IPv6 addresses would increase the frame size (64 octets becomes 84 octets).

10 Delay Variation

10.1 Background

Delay Variation is a key Benchmark for the NFVI, closely related to Latency. It is valuable to assess the consistency of frame transfer times of a SUT at various levels of offered load. This information can assist capacity planning and network engineering when conducted under circumstances relevant to the production network of interest.

Each individual measurement of Delay Variation is derived from two measurements of one-way delay on individual frames (also called latency or transfer time). The Delay Variation Benchmark and metric variations are distinguished by the criteria for selection of the two specific one-way delay measurements.

As Delay Variation is a measure of consistency, or the spread of the numerical distribution of one-way delay, the distribution may be summarized using familiar statistics or by presenting the complete histogram for the distribution.

10.2 Name

There is only one required Benchmark of Delay Variation, the 99th percentile of Frame Delay Variation (FDV), which is measured at the Throughput Benchmark offered load level, and is an extension of benchmarks defined in IETF RFC 2544 [1]. An auxiliary metric of Inter-Frame Delay Variation (IFDV) may be measured using a related method and procedure:

1) Frame Delay Variation (see section 4.2 of IETF RFC 5481 [i.11] for the required Selection Function, packet and frame definitions have a one to one correspondence)

2) Inter-Frame Delay Variation (see section 4.1 of IETF RFC 5481 [i.11] for the required Selection Function, packet and frame definitions have a one to one correspondence) at X % of Throughput

Additional metric variants of Delay Variation at other offered load levels may be measured using a related method and procedure. The X % of Throughput Benchmark offered load level is a parameter described in clause 10.3.

10.3 Parameters

The following parameters shall be supported for the Latency Benchmark and other metric variants:

- **Selection Function:** the required pair of frames used in each individual Delay Variation measurement.
- **X % of Throughput Benchmark Offered Load:** the fraction of the offered load corresponding to the Throughput level as determined in prerequisite testing. For the Latency Benchmark, X = 100 %.
- **Offered Load Frame Size:** the fixed frame size (or description of the mixture of sizes, or IMIX) transmitted by the measurement system when searching for the measurement goal.
- **Minimum Trial Repetition Interval:** the minimum time to conduct a trial in seconds, which involves preparation of the SUT, offering load, waiting for queues to empty, and computing the measured result.
- **Trial Duration:** the time interval when the Offered Load is transmitted to the SUT, expressed in seconds.
- **Sample Filter:** the description of the filter characteristic used to select frames from the population of all frames sent as offered load. For the Latency Benchmark, the sample filter selects all frames.
- **Maximum Number of Trials:** the limit on trials attempted to achieve the measurement goal of the test. Reaching this limit implies an error condition or unstable measurements, and it is required to report this event.

10.4 Scope

The list of one or more infrastructure resources which are included in the values reported, and whether the workload is physical or virtual.

10.5 Units of Measure

The FDV and IFDV shall be expressed in microseconds.

NOTE: It is also acceptable to use metric adjectives to make the values more relevant, such as nanoseconds.

10.6 Definition

Briefly, the Selection function for FDV compares the delay of each frame to the minimum delay over all frames. The Selection Function for IFDV compares the delay of each frame to the delay of the previous frame in sending order [i.11].

The values of each metric at time T (for a given scope and measurement time) shall be assessed as the statistical summary of filtered frame Delay Variation meeting the criteria as specified in table 10.6-1.

Table 10.6-1: Requirements for Benchmarks and metrics

Requirement number	Metric Name	Requirement
DV-01	Frame Delay Variation	Reported values of this Benchmark shall quantify the 99th percentile of FDV for all filtered transfer times when the Offered Load is equal to the Throughput level.
DV-02	FDV (Yth percentile, X % Thpt, Filter)	Reported values of this metric variant shall quantify the Yth percentile of the frame delay variation (of filtered frames) across the DUT/SUT when the Offered Load is equal to the X % of the Throughput level.
DV-03	Inter-Frame Delay Variation (X % Thpt, Filter)	Reported values of this metric variant shall quantify the Range (Maximum-Minimum) of the inter-frame delay variation (of filtered frames) across the DUT/SUT when the Offered Load is equal to the X % of the Throughput level.
DV-04	Mean FDV (X % Thpt, Filter)	Reported values of this metric variant shall quantify the Mean of the frame delay variation (of filtered frames) across the DUT/SUT when the Offered Load is equal to the X % of the Throughput level.
DV-05	Mean IFDV (X % Thpt, Filter)	Reported values of this metric variant shall quantify the Mean of the absolute value of the inter-frame delay variation (of filtered frames) across the DUT/SUT when the Offered Load is equal to the X % of the Throughput level.

10.7 Method of Measurement

This Benchmark and related metric variant employ the basic method and procedure defined in clause 12.

IETF RFC 5481 [i.11] describes many measurement considerations and procedures that sound methods should include.

10.8 Sources of Error

The sources of error for this Benchmark and metric variant are listed below:

1) The Offered Load Step Size of the prerequisite test determines the resolution of the result. Larger step sizes will result in faster searches with fewer trials, but the error appears in a larger under-estimate of Throughput or Capacity.

2) Synchronization of the time-of-day clock with an external reference usually ensures sufficiently accurate timestamps, but loss of synchronization for an extended period will cause time accuracy to suffer. However, as FDV and IFDV are differential time measurements, any fixed time offset between traffic generator and receiver will cancel-out (a completely different situation from Latency).

10.9 Discussion

There is no possible measurement of IFDV for the first frame transmitted, since two frames are required. However, the first frame produces a measurement of FDV, since the frame with minimum delay value may be determined after all frames in the population are received.

The results of individual IFDV measurements take both positive and negative values. Usually the bound on negative values is determined by the inter-frame spacing at the traffic generator. However, if the sending order is changed by the DUT/SUT, then there is no effective negative bound on IFDV values.

10.10 Reporting Format

Ideally, the results of testing to assess Frame Delay Variation should be presented using a Figure with a two-dimensional graph.

The title of the graph shall include the metric name(s) and the Units of the y-axis (such as one-way Frame Delay Variation in microseconds, although a different scientific prefix may be used, such as nanoseconds).

The Units of the x-axis shall be Frame Size in octets.

The Units of the y-axis shall be one-way Frame Delay Variation in microseconds (although scientific prefix may be used, such as nanoseconds). If the test traffic is bi-directional, then results for each of the directions may be plotted separately on the same graph for each frame size tested.

For tests with variable Frame size [4], the results shall be plotted according to the average Frame size of the traffic stream (which includes the factors of Frame frequency and Frame size).

Along with the measured one-way Frame Delay Variation in seconds for each Frame size, the Inter-Frame Delay Variation should be plotted (if measured) for each Frame size.

The caption of the Figure shall describe all relevant aspects of the test, in sufficient detail to inform a reader and to distinguish each Figure from similar Figures with results from different test conditions conducted as part of the same campaign (such as different test set-ups and protocols used). For example, the actual X % of Throughput achieved should be reported. There shall also be a legend, identifying the benchmark and any other metric-variants plotted.

If the results are summarized as a single number, then the result of metric assessment for the smallest Frame size shall be reported, with Units of measure.

If the results are summarized in a table, then the example table 10.10-1 should be used.

Table 10.10-1: Example FDV Tabular Reporting

Requirement number	Metric Name	Frame Size, octets			
		64	256	IMIX	1 518
DV-01	Frame Delay Variation				
DV-02	FDV (Y^{th} percentile, X % Thpt, Filter)				
DV-03	Inter-Frame Delay Variation (X % Thpt, Filter)				
DV-04	Mean FDV (X % Thpt, Filter)				
DV-05	Mean IFDV (X % Thpt, Filter)				

The example above assumes IPv4 addresses, and IPv6 addresses would increase the frame size (64 octets becomes 84 octets).

11 Loss

11.1 Background

Frame or packet loss is a new Benchmark for the NFVI, closely related to Throughput. It is valuable to assess the loss ratio of a SUT at various levels of offered load (above the Throughput level, which requires zero loss). This information can assist capacity planning and network engineering when conducted under circumstances relevant to the production network of interest.

The Loss benchmark and its metric variations are primarily distinguished by the specificity used to describe the characteristics of frame or packet losses and their correlation (in time). In the following clauses, definitions and descriptions using the term "frame" should be considered equally applicable to "packet" when measurements are conducted at the packet layer. The metric-variants should be useful for summarizing Long Duration testing, where the focus is on characterizing loss bursts and loss-free periods and reporting their temporal distribution in order to identify any regular interference from the system e.g. Linux kernel.

11.2 Name

There is only one required Benchmark of Loss Ratio, the ratio of lost frames to total frames transmitted (during the trial), and is an extension of benchmarks defined in IETF RFC 2544 [1].

A set of metric-variants of Loss may be measured using a related method and procedure:

1) Loss Ratio at 110 % of Throughput Benchmark offered load level (Loss Benchmark).

2) Loss Ratio at X % of Throughput Benchmark offered load level.

3) Maximum consecutive loss count at X % of Throughput Benchmark offered load level.

4) Loss-free seconds count at X % of Throughput Benchmark offered load level.

The X % of Throughput Benchmark offered load level is a parameter described below.

11.3 Parameters

The following parameters shall be supported for the Loss Benchmark and other metric variants:

- **X % of Throughput Benchmark Offered Load:** the fraction of the offered load corresponding to the Throughput level as determined in prerequisite testing. For the Loss Benchmark, X = 110 %. When testing investigates conditions that may represent production deployment, X = 80 % or X = 50 % may be good choices for a metric variant.

- **Offered Load Frame Size:** the fixed frame size (or description of the mixture of sizes, or IMIX) transmitted by the measurement system when searching for the measurement goal.

- **Minimum Trial Repetition Interval:** the minimum time to conduct a trial, in seconds, which involves preparation of the SUT, offering load, waiting for queues to empty, and computing the measured result.

- **Trial Duration:** the time interval when the Offered Load is transmitted to the SUT, expressed in seconds.

- **Sample Filter:** the description of the filter characteristic used to select frames from the population of all frames sent as offered load. For the Loss Benchmark, the sample filter selects all frames.

- **Maximum Number of Trials:** the limit on trials attempted to achieve the measurement goal of the test. Reaching this limit implies an error condition or unstable measurements, and it is required to report this event.

11.4 Scope

The list of one or more infrastructure resources which are included in the values reported, and whether the workload is physical or virtual.

11.5 Units of Measure

The Loss Ratio shall be expressed as a percentage. Counts of consecutive losses shall be expressed in frames. Counts of Loss-free seconds shall be expressed in seconds.

11.6 Definition

The values of each metric at time T (for a given scope and measurement time) shall be assessed as the statistical summary of filtered frame Loss meeting the criteria as specified in table 11.6-1.

Table 11.6-1: Requirements for Benchmarks and metrics

Requirement number	Metric Name	Requirement
Loss-01	Loss Ratio	Reported values of this Benchmark shall quantify the ratio of lost frames to total frames transmitted during the trial when the Offered Load is equal to 110 % of the Throughput level (no filter on transmitted frames).
Loss-02	Loss Ratio (X % Thpt, Filter)	Reported values of this metric variant shall quantify the ratio of lost frames to total frames transmitted (all frames meeting the filter criteria) during the trial when the Offered Load is equal to X % of the Throughput level.
Loss-03	Max Loss Count (X % Thpt, Filter)	Reported values of this metric variant shall quantify the Maximum count of consecutive frame losses (all frames meeting the filter criteria) during the trial when the Offered Load is equal to X % of the Throughput level.
Loss-04	Loss-Free Seconds (X % Thpt, Filter)	Reported values of this metric variant shall quantify a count of each contiguous second constituting the trial, during which the traffic receiver observed zero frame losses (with all frames meeting the filter criteria) when the Offered Load is equal to X % of the Throughput level.

11.7 Method of Measurement

This Benchmark and related metric variants employ the basic method and procedure defined in clause 12.

For the Loss-free seconds metric variant, the trial duration shall include at least 60 seconds of offered load (excludes idle and preparation times). The time scale for one second boundaries shall be established at the traffic receiver beginning with reception of the first frame of the offered load during the trial, and a partial second of traffic at the end of a trial shall be discarded for counting purposes.

11.8 Sources of Error

The sources of error for this Benchmark and metric variant are listed below:

1) The Offered Load Step Size of the prerequisite test determines the resolution of the result. Larger step sizes will result in faster searches with fewer trials, but the error appears in a larger under-estimate of Throughput or Capacity.

2) Synchronization of the time-of-day clock with an external reference usually ensures sufficiently accurate timestamps, but loss of synchronization for an extended period will cause time accuracy to suffer.

11.9 Discussion

The intent of the Loss-count and Loss-free seconds count is to provide additional insights on the nature of frame losses when they occur. For example, a trial with all losses occurring in a single consecutive loss event may indicate an interruption of forwarding in the data plane, or a single buffer overflow event during the trial.

11.10 Reporting Format

Ideally, the results of testing to assess Loss Ratio should be presented using a Figure with a two-dimensional graph.

The title of the graph shall include the metric name(s) and the Units of the y-axis (such as one-way Loss Ratio in percent).

The Units of the x-axis shall be Frame Size in octets.

The Units of the y-axis shall be Loss Ratio in percent. If the test traffic is bi-directional, then results for each of the directions may be plotted separately on the same graph for each frame size tested.

For tests with variable Frame size [4], the results shall be plotted according to the average Frame size of the traffic stream (which includes the factors of Frame frequency and Frame size).

Alternatively, a two dimensional graph may be prepared for each Frame size (or IMIX), where the Units of the x-axis should be the Offered Load in Frames per second, or in X % of Throughput level, or both.

Along with the measured benchmark one-way Loss Ratio for each Frame size, the Loss Ratio at X % Throughput may also be plotted (if measured) for each Frame size.

The caption of the Figure shall describe all relevant aspects of the test, in sufficient detail to inform a reader and to distinguish each Figure from similar Figures with results from different test conditions conducted as part of the same campaign (such as different test set-ups and protocols used). For example, the actual X % of Throughput achieved should be reported. There shall also be a legend, identifying the benchmark and any other metric-variants plotted.

If the results are summarized as a single number, then the result of metric assessment for the smallest Frame size shall be reported, with Units of measure.

If the results are summarized in a table, then the example table 11.10-1 should be used.

Table 11.10-1: Example Loss Ratio Tabular Reporting

Requirement number	Metric Name	Frame Size, octets			
		64	256	IMIX	1 518
Loss-01	Loss Ratio				
Loss-02	Loss Ratio (X % Thpt, Filter)				
Loss-03	Max Loss Count (X % Thpt, Filter)				
Loss-04	Loss-Free Seconds (X % Thpt, Filter)				

The example above assumes IPv4 addresses, and IPv6 addresses would increase the frame size (64 octets becomes 84 octets).

12 Methods of Measurement

12.1 Pre-Test and Measurement Procedure

This clause provides a set of commonly-used methods to conduct measurements that determine the values of benchmarks and metrics.

Table 12.1-1: Pre-Test and Measurement Steps

Requirement number	Name	Requirement
Pre-Test-01	Select Use Cases and Benchmarks	The Use Cases for the NFVI shall be determined and listed. The relevant Benchmarks shall be selected and listed.
Pre-Test-02	Select and confirm/calibrate Test Device/Function	The Test Traffic Generator and Receiver selected shall meet all requirements of clause 4, clause 6 regarding connectivity to the test setup, clause 7, and all clauses describing the selected Benchmarks and metrics.
Pre-Test-03	Implement Test Setup(s)	The Test Setups relevant to the selected Use Cases shall be selected and implemented using the NFVI according to the requirements of clause 6.
Pre-Test-04	Implement and Record Test configuration	The test configuration shall be applied to the Test Setup and the Test Traffic Generator and Receiver, and recorded as required in clause 6. The Traffic Generator configuration will include a range of values for some parameters such as frame size, and each are used in separate tests and trials.
Pre-Test-05	Determine Parameters for Benchmarks and Metrics	Select and implement Parameter values for the selected Benchmarks and Metrics, as specified in clause 8 for the Throughput Benchmark (for example).
Pre-Test-06	Configure address learning time-outs	Configure address time-outs such that they shall not expire during the Minimum Trial Repetition Interval.

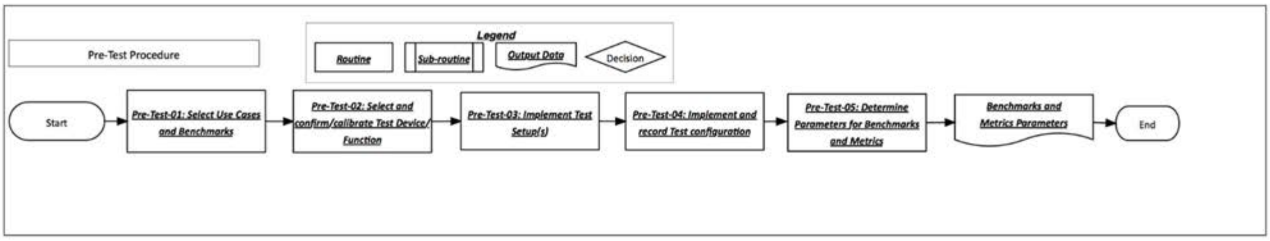

Figure 12.1-1: Pre-Test Procedures

12.2 Core Procedures

This clause provides a hierarchy of procedures to assess the values for Benchmarks and Metrics.

A trial is one measurement of frame transmission and reception using a single Offered Load (Traffic Generator Configuration) and single configuration of the NFVI Test Setup.

A test consists of one or more trials, typically uses the results of a previous trial to determine the Traffic Generator Configuration of Offered Load for the next trial, and follows a Search Strategy to achieve the Measurement Goal required by the Benchmark or Metric.

A Set consists of two or more repeated tests, and summarizes the overall results to determine the values for Benchmarks and Metrics.

A Method consists of multiple Sets of tests, where a relevant aspect of the test setup configuration is changed between Sets, or a significant aspect of the Traffic Generator Configuration was changed in a way that cannot be directly combined with another Set of tests (such as increasing the number Layer 3 flows in the Offered Load, or employing a mixture of frame sizes in the same trial).

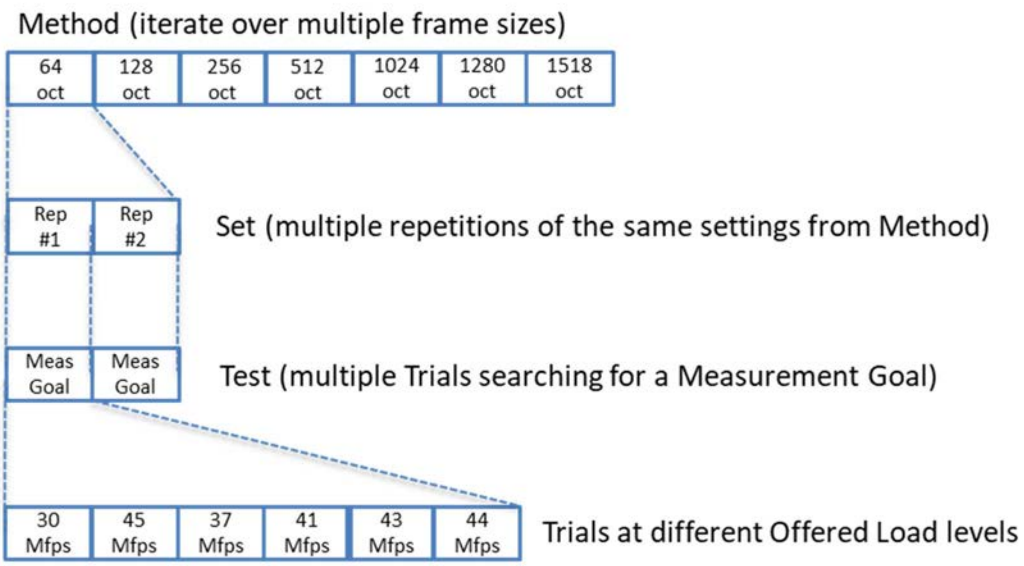

Figure 12.2-1: Hierarchy of Core Procedures

Table 12.2-1 describes the procedural requirements for a Trial, based on section 23 of IETF RFC 2544 [1].

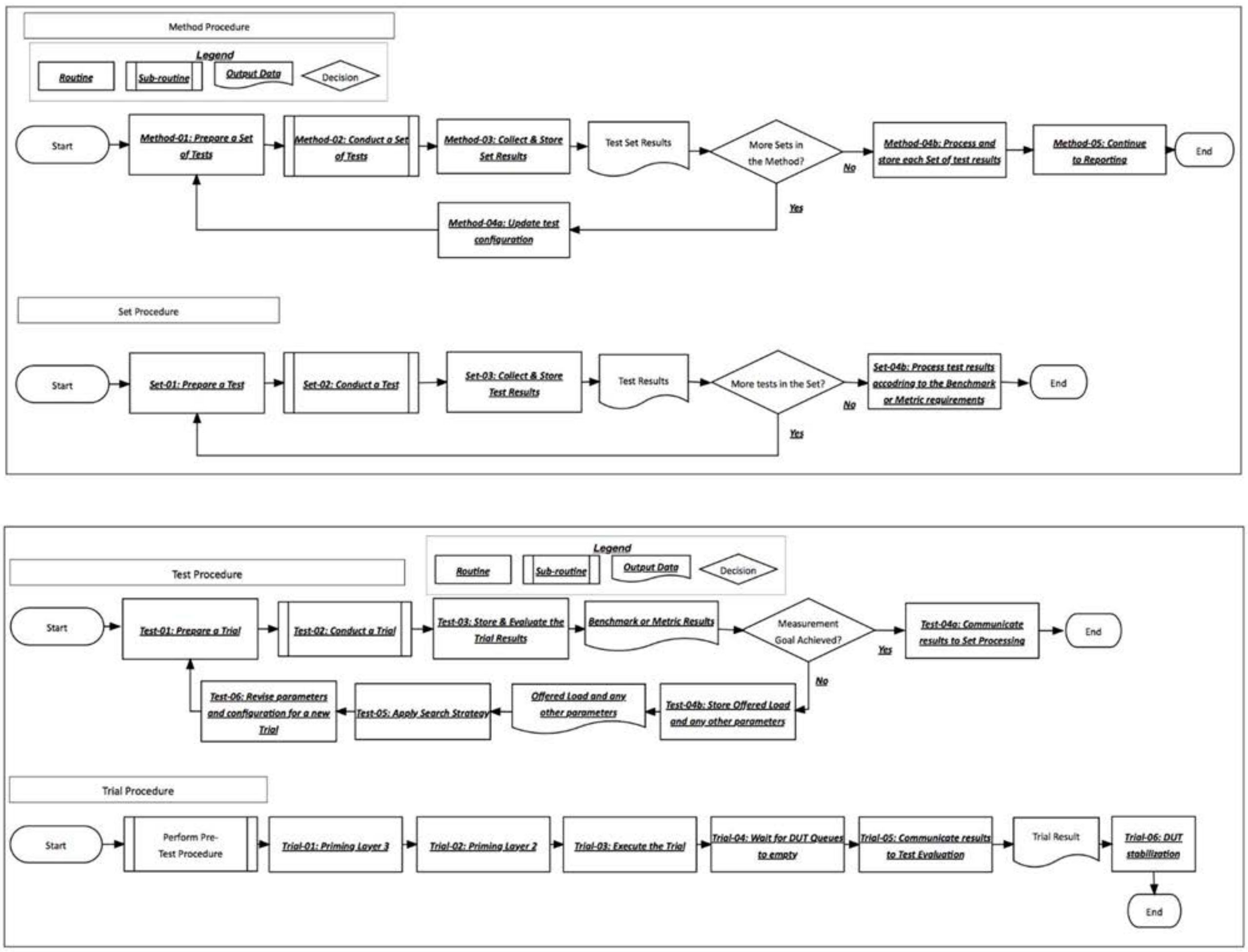

Figure 12.2-2: Overview of Procedures

Table 12.2-1: Steps for Each Trial

Step number	Name	Requirement
Trial-01	Priming Layer 3	The Test Traffic Generator shall send routing updates as necessary to ensure data plane readiness for packet transfer.
Trial-02	Priming Layer 2	The Test Traffic Generator shall send Layer 2 Learning Frames as necessary to ensure data plane readiness for frame transfer.
Trial-03	Execute the Trial	The Test Traffic Generator shall send Frames in accordance with the Offered Load parameters.
Trial-04	Wait for DUT Queues to empty	The Test Traffic Receiver shall process all incoming frames (which may include counting, timestamping, latency calculations, or other processing as dictated by the Benchmark or metric), for 2 seconds after the Test Traffic Generator ceases transmission for the Trial. (see section 23 of IETF RFC 2544 [1]).
Trial-05	Communicate results to Test Evaluation	Evaluate the results in Step Test-03 of the Test procedure.
Trial-06	DUT stabilization	The DUT is left idle and allowed to stabilize for at least 5 seconds before another trial begins. (see section 23 of IETF RFC 2544 [1]).

Table 12.2-2 describes the procedural requirements for a Test.

Table 12.2-2: Steps for Each Test

Step number	Name	Requirement
Test-01	Preparing a Trial	The Test Device/Function shall control the Traffic Generator parameters and configuration for the impending trial, typically the level of Offered Load.
Test-02	Conduct a Trial	The requirements for Trial Steps 01 through 06 apply.
Test-03	Evaluate and Store the Trial Results	The Test Device/Function shall evaluate the results and determine if the Measurement Goal has been attained, or if more Trials are necessary.
Test-04a	IF: Measurement Goal Achieved	The Test Device/Function shall communicate the results for the Benchmark or Metric to the Set Processing procedure (step Set-03), and terminate the Test.
Test-04b	IF: Measurement Goal not yet Achieved	The Test Device/Function shall store the Offered Load and any other parameters. Proceed to Step Test-05.
Test-05	Apply Search Strategy	Determine the Traffic Generator parameters and configuration for the next trial, according to the Search Strategy employed.
Test-06		Return to Test-01 with revised parameters and configuration for a new Trial.

Table 12.2-3 describes the procedural requirements for a Set of Tests. When test repetition is required, the Set Procedure governs the number of repetitions.

Table 12.2-3: Steps for Each Set

Step number	Name	Requirement
Set-01	Prepare a Test	Configure the Traffic Generator Fixed Parameters for the impending Test, and enable control to the Test and Trial procedures.
Set-02	Conduct a Test	The requirements for Test Steps 01 through 06 apply.
Set-03	Collect & Store Test Results	For each repeated Test in the Set of Tests, collect and store the results for the Benchmark and/or Metric.
Set-04a	IF: More Tests in the Set?	Proceed to Step Set-01.
Set-04b	IF: All Test Repetitions Complete	Process the set of test results according to the Benchmark or Metric requirements. For example, the statistical summary of the Set of Test Results may include the Minimum, Maximum, Average, and Standard Deviation of the Benchmark results. Return to Method Procedure.

Table 12.2-4 describes the procedural requirements for each Method. The Method Procedure dictates the relevant aspects of the Test Procedure to change, including Test Device/Function parameters for the Traffic Generator or configuration of components in the Test Setup which affect frame processing and therefore, data plane performance.

Table 12.2-4: Steps for Each Method

Step number	Name	Requirement
Method-01	Prepare a Set of Tests	Configure the Traffic Generator Fixed Parameters and the configuration of components in the Test Setup which affect frame processing for the impending Test, and enable control to the Set, Test and Trial procedures.
Method-02	Conduct a Set of tests	The requirements for Set Steps 01 through 06 apply.
Method-03	Collect and Store Set Results	For each Completed Set of Tests, collect and store the results for the Benchmark and/or Metric associated with the Traffic Generator Fixed Parameters and the configuration of components in the Test Setup.
Method-04a	IF: More Sets in the Method?	Proceed to Step Method-01, with new Traffic Generator Fixed Parameters and/or the configuration of components in the Test Setup.
Method-04b	IF: All Sets of Tests are Complete	Process and store each Set of test results, arranging for simple comparison across iterations of Traffic Generator Fixed Parameters and Setup Configurations.
Method-05	Continue to Reporting	Method Procedure complete.

Further, the Method Procedure:

1) Should permit collection of measurements of system resource utilization alongside the benchmark measurements. Examples of systems resources are processor utilization and memory usage, see clauses 6 and 8 of ETSI GS NFV-TST 008 [5], respectively.

2) Should encourage test repetition with and without system resource measurements, to assess any effect on performance and quantify the extent where possible.

12.3 Search Algorithms

12.3.1 Introduction

The assessment of most networking benchmarks requires multiple measurements conducted over a range of stimuli to achieve a goal. The most common case requires searching over a range of offered load levels to find the highest offered load level that meets the parameters of the benchmark (or Measurement Goal). The most common search algorithms are the Binary and Linear algorithms. Once a trial has been completed, the test procedure evaluates the result alternatives (frame loss observed, or all frames received successfully) and determines the parameter settings for the next trial (step Test-03).

In earlier specifications [1], laboratories provided an isolated test environment where the sole source of frame loss was considered to be the resource limitations of the DUT. For example, routing updates and address learning frames are sent in advance of the measured streams in the trial (section 23 of IETF RFC 2544 [1]) to ensure stable measurement conditions, and the frame-forwarding processes are not interrupted. Frame loss from lab facilities or cabling could not be tolerated. As a result, the outcome of every trial was evaluated with full confidence, and search decisions were conducted on the basis of measurement truth.

In the NFVI, there are many processes running besides those that accomplish frame forwarding. Frame loss can be caused by infrequent background processes running (unavoidably) in the DUT, in addition to DUT resource limitations. These two categories of demand on DUT resources during benchmarking were first identified in [i.8] as Transient (the infrequent background processes) and Steady-state. The loss caused by background processes when they consume resources necessary for frame forwarding may not be distinguishable from loss experienced when offered load exhausts the frame forwarding resource. Therefore, all trial outcomes observing loss are suspect, since loss heralds either the background process events and/or resource exhaust (see clause A.1.2 for additional discussion). However, trial outcomes with no loss may be accepted as truth without verification, because this is a scenario where the background process produces half-lies [i.14].

With the new search algorithms specified in the present document, the infrequent nature of these background processes is exploited to achieve a more accurate characterization of the resource limitations, without appreciably increasing the number of trials required to achieve a measurement goal. Of course, the background process frequency and loss characteristics can be assessed with long duration testing.

12.3.2 Binary search

The Binary Search [i.12] and [i.13] is well-described in the literature, but there are a few areas which benefit from explicit specification, beginning with the search space. The Offered Load is the most common test stimuli and it is the basis for the requirements below, but other stimuli should be treated in an equivalent way. The Binary Search for benchmarking seeks to identify a sequential pair of array elements, one with measurement outcome "Loss = 0" and the other with "Loss > 0". The search for this pair is the main difference from the classic Binary search for an array element that matches a target value. Another difference is that classic search algorithms emphasize speed and reduce the number of conditional evaluations for efficiency, but the time required to conduct a test Trial is the dominant factor in each iteration of a benchmarking search.

The Offered Load Step Size Parameter divides the search space into a sorted array of Offered Load levels in Frames per second, where the Maximum Frames per second summed for all ingress interfaces used on the SUT shall be divided by the step size in Frames per second to produce the number of elements in total search space array (rounded up to the nearest integer). If ingress interfaces use different rates, then the total traffic shall be distributed to each interface proportionally to their rate. Each array element shall be referenced by its array index in the procedures that follow.

The steps in the optional Binary search are specified below, where:

- S is defined as a sorted array of test stimuli, for example Offered Load levels.
- n is defined as a calculated variable giving the number of elements in the search array, S.
- m is defined as the integer index of array elements, [0, 1,…, n-1].
- L is defined as the lower index of the current search window, initially set to 0.
- R is defined as the higher index of the current search window, initially set to m = n-1.
- Calculations using L and R will be converted to integers by taking the integer part of the resulting value, int( ).

Table 12.3.2-1: Steps to create the Search Array, S

Step number	Name	Requirement
Array-01	Find the number of elements in the search space, n.	The Maximum Frames per second summed for all ingress interfaces shall be divided by the Offered Load Step Size Parameter in Frames per second to produce the number of elements in the complete search space (rounded up to the nearest integer), n.
Array-02	Create an Array of size n	The Array elements shall be identified by an index, m [0…n-1]. The Array shall have n element-pairs, (Offered Load, Loss).
Array-03	Assign stimuli values and initial Loss value	Each array element shall contain an Offered Load value in Frames per second, calculated as (m + 1) × Step Size in Frames per second. This is a sorted array of (Offered Load, Loss) where Loss shall be set to NA for all elements at the start.

Table 12.3.2-2: Steps in the Binary Search

Step number	Name	Requirement
B-01	Select the stimulus (Offered Load Level)	Calculate m = int((L+R)/2)
B-02	Check the stimulus (Offered Load Level)	IF: Loss == NA THEN: Communicate the Offered load Level of array element m to the Test Procedure (Test-01) ELSE: continue
B-03	Evaluate trial result	Loss outcome shall be communicated from the Test Procedure (Step Test-03) See note
B-04	IF: L == R	IF: L == R THEN: The Measurement Goal is found at m or m - 1 Select the higher of the two elements with Loss == 0 and return that Offered Load to the Test Procedure. ELSE: continue
B-05	IF: Loss == 0	IF: Loss == 0 THEN: The Measurement Goal lies in the upper half of the current search array. Set L = m + 1 Return to step B-01 ELSE: continue
B-06	IF: Loss > 0	IF: Loss > 0 THEN: The Measurement Goal lies in the lower half of the current search array. Set R = m - 1 Return to step B-01
NOTE: The outcome for m may already be stored.		

Under the circumstances of highly repetitive testing where the duration of a complete Method is a critical aspect, it may be possible to optimize the search range by conducting several trials at different Loads, and setting L and R parameters to search over a smaller range of expected outcomes.

Of course, there is the possibility to apply the Binary search strategy where the Loss target is X % and greater than 0, by adding additional conditional evaluations in B-04, B-05, and B-06, especially for the case where the Measurement goal is achieved (Loss == X %). Also, see clause 12.4 for an alternate evaluation of loss.

12.3.3 Binary search with loss verification

This clause defines a new search algorithm, binary search with loss verification, specifically designed to address the issues encountered for NFVI benchmarking. Procedures for physical device benchmarking ensured that transient processes (such as address learning and routing updates) were conducted before conducting trials to discover resource limits. However, a general purpose computing platform has many forms of transient processes that run in the background, and they cannot be disabled completely or system health may suffer. Therefore, benchmarking procedures that recognize and mitigate frame loss caused by background processes will be needed to assess the resource limits, and additional tests should characterize the effects and frequency of background processes. Splitting the DUT/SUT characterization in two categories matches the optimization activities that often accompany benchmarking: identifying bottlenecks that limit data plane capacity is different from identifying the source of transients and seeking to reduce their frequency. Testing to evaluate the new search algorithm has been conducted [i.15] and the benefit to test repeatability was found to be substantial.

The possibility for errors to appear in search results is described in the literature [i.14]. The new procedure specified here benefits from recognizing that frame loss resulting from the background processes represents a form of communications error. Communications error correction involves adding redundant information to increase robustness to error. In the case of NFVI benchmarking, the simplest form of redundancy is applied, recognizing that only the trials with Loss > 0 may be in error (these are the half-lies referred to above), and the strategy repeats those trials using the same stimulus. Message repetition is the simplest way to add error robustness to a communication system, and it is particularly effective in systems with binary messages where only one message type is error-prone. Annex B describes this analysis in more detail.

The Offered Load is the most common test stimuli and it is the basis for the requirements below, but other stimuli should be treated in an equivalent way.

The steps in the Binary search with Loss Verification shall be used as specified below, where all variables assigned in clause 12.3.2 are re-used except for the Array S. Additionally, the following variables are used:

- V is defined as a (n element) sorted array of test stimuli, for example Offered Load levels, and serves to store trial outcomes as well.
- r is a variable tracking the number of verifications of a particular Offered Load level (The recommended value of max(r) = 2). r is initially set to 1.
- z is threshold for Loss(r) to override Loss Verification when the count of lost frames is very high and avoid unnecessary verification trials. z = 10 000 frames has been used in practice. There is risk of negating the value of Loss Verification from setting z too low. If the high loss threshold is not wanted, then set z = infinity and all Loss(r) > 0 outcomes will be repeated.

Note that the search array, V, records all instances of repeated testing, and therefore allows the tester to quantify the number of trials (with Loss(r) > 0) where the eventual outcome was reversed by a single Loss(r) == 0 outcome. However, a minimal array, S, where the Loss(r) == 0 outcome overwrites previous Loss outcomes may be used as an alternative to array, V.

Table 12.3.3-1: Steps to create the Search Array, V

Step number	Name	Requirement
AV-01	Find the number of elements in the search space, n.	The Maximum Frames per second summed for all ingress interfaces shall be divided by the Offered Load Step Size Parameter in Frames per second to produce the number of elements in the complete search space (rounded up to the nearest integer), n.
AV-02	Create an Array of size n	The Array elements shall be identified by an index, m in [0…n-1]. The Array shall have n element (max(r)+1)-tuples, (Offered Load, Loss(1), …, Loss(max(r)).
AV-03	Assign stimuli values and initial Loss value	Each array element shall contain an Offered Load value in Frames per second, calculated as (m + 1) * Step Size in Frames per second. This is an array of sorted Offered Load levels and Loss(r) outcomes, where Loss(r) shall be set to NA for all elements at the start.

Table 12.3.3-2: Steps in the Binary Search with Loss Verification

Step number	Name	Requirement
BLV-01	Select the stimulus (Offered Load Level)	Calculate m = int((L+R)/2)
BLV-02	Check the stimulus (Offered Load Level)	IF: Loss(1) == NA OR r > 1 THEN: Communicate the Offered load Level of array element m to the Test Procedure (Test-01) ELSE: continue
BLV-03	Evaluate trial result	Loss outcome shall be communicated from the Test Procedure (Step Test-03) See note
BLV-04	IF: L == R	IF: L == R THEN: The Measurement Goal is found at m or m-1 Select the higher of the two elements with Loss(r) == 0 and return that Offered Load to the Test Procedure. ELSE: continue
BLV-05	IF: Loss(r) == 0	IF: Loss(r) == 0 THEN: The Measurement Goal lies in the upper half of the current search array. Set L = m + 1 Set r = 1 Return to step BLV-01 ELSE: continue
BLV-06	IF: z > Loss(r) > 0 AND r<max(r)	IF: z > Loss(r) > 0 AND r < max(r) THEN: The Trial is repeated with the same stimulus (same m) Set r = r+1 Return to step BLV-02 ELSE: continue
BLV-07	IF: Loss(r) > 0 AND r == max(r) OR Loss(r) > z	IF: Loss(r) > 0 AND r == max(r) OR Loss(r) > z THEN: The Measurement Goal lies in the lower half of the current search array. Set R = m - 1 Set r = 1 Return to step BLV-01
NOTE: The Loss outcome(s) for this m may already be stored.		

12.3.4 Binary search with NDR and PDR Loss Thresholds

This form of search has been implemented and tested in [i.17], [i.18] or [i.16]. The procedure requires two Loss thresholds, a very low loss ratio (or true zero loss ratio in some usages) referred to as the No-Drop-Rate (NDR) and a higher threshold referred to as the Partial Drop Rate (PDR).

If NDR > 0, then this search can only be used with the Capacity at X % Loss metric-variant, where X equals NDR expressed as a percentage.

PDR throughput (Loss Ratio > 0) measurements allow developers, testers and users to compare with the Throughput benchmark and in cases where the PDR - Throughput load difference is large, seek an explanation based on SUT setup or NFVI application code and/or the configuration tested.

Measuring Capacity at a set of Loss Ratios is important to many Internet transport and applications including telecom services (see Recommendation ITU-T Y.1541 [i.19] for a long-standing specification of Network performance objectives for IP-based services, and TCP/IP Capacity [i.20] to name a few).

It may be useful to propose a set of suggested Loss Ratios aligned with above specifications like those in Recommendation ITU-T Y.1541 [i.19].

12.4 Long Duration Testing

This clause describes Methods where assessments are conducted using very long overall durations, but intermediate results are collected at much shorter intervals. The short-term evaluations (Trials) allow for characterization of individual events, such as those originating from infrequent background processes, and allow the method to be terminated early if the current results indicate that significant instability has already been encountered. IETF RFC 8204 [i.9] describes this form of assessment as Soak tests, and provides useful background. As described in annex B, Long duration testing can be used to discover the frequency of transient events observed at Offered Load Levels corresponding to the Throughput Offered Load level, and lower levels of Offered Load.

Background processes may not necessarily cause Loss, but affect Delay Variation instead. Long Duration testing can be useful to assess the stability of Latency, Delay Variation, and their related metric-variants.

13 Follow-on Activities

This clause describes areas for follow-on development of the present document in future releases.

The container-related setups in figure 6.2-3 should be reviewed after further study, as this is an active development area.

New material for annex B, summarizing testing on Binary Search with Loss Verification.

Add NDR/PDR and MLResearch search algorithm descriptions.

New methods for Long Duration Testing, taking advantage of the Loss Benchmark and its metric-variants.

A new metric-variant for Loss, where the timestamps of loss events would be collected for correlation analysis with system information (logs of process events).

Annex A (informative): Survey of Current Benchmarking Campaigns

A.1 Overall Summary of Key Issues and Points of Learning

A.1.1 Introduction

This clause summarizes the relevant testing studies and campaigns conducted on NFVI networking technologies and capabilities, especially the lessons learned from these studies and key issues identified. Each sub-clause represents an opportunity to improve existing methods and procedures, and suggests potential mitigations for the issues.

A.1.2 Test Conditions with Non-zero Packet Loss

Existing procedures assume that loss-free transmission is achievable in the isolated laboratory test environment. However, some tested configurations may exhibit a measurable level of packet loss during the chosen trial duration, even when operating under conditions which are otherwise steady and reliable [i.16]. This is a potential issue for measuring loss-free Throughput and other benchmarks requiring zero-loss.

Potential Mitigations:

Pre-requisite tests: Seek conditions/configurations which do not exhibit steady packet loss, and restrict testing to only those conditions which meet the pre-requisite.

Procedures and Reporting:

1) Possibly allow non-zero packet loss as an acceptance threshold, but only after considering the effect of this loss in the context of the use case(s) addressed, ensuring the loss level used is sufficiently low to minimize effects, and clearly reporting the loss threshold level used with all results.

2) Since the measurable loss may cause significant variation in the result from searching algorithms, consider repeating the tests many times and reporting the range of results over many repetitions. Here, the term "many" is taken to mean the sufficient number of tests to establish the full range of variation. The number of repetitions needed is analysed further in annex B.

A.1.3 Repeatable Results Depend on many Config. Variables

Existing procedures include requirements on DUT configuration. For example, see section 7 of IETF RFC 2544 [1]. When the physical network function DUT is replaced with a general purpose computing platform and virtual networking capabilities as part of the NFVI, the number of configuration variables increases substantially (possibly by a factor between 10 and 100 more variables). Further, the instructions for the SUT or DUT may not recommend values for all the configuration variables that are present. Lastly, configuration of abstract variables may not result in repeatable configuration operations. Abstract configuration variables are non-deterministic in their eventual instantiation, while absolute configuration parameters are fully deterministic in their instantiation.

Potential Mitigations:

Benchmark Definition: Seek definitions which normalize the results over the resources tested.

Configuration: For a given benchmark, certain absolute configuration variables will be critical, and others will not change often (e.g. BIOS version).

Test Set-up:

1) Management of the large number of configuration variables could be assisted by automated collection of software and hardware configuration in all meaningful dimensions, with simplified comparison between the current and previously recorded configurations to highlight any differences. It may be necessary to develop an advanced difference-seeking method, in order to make the configuration comparisons useful.

2) It is important to identify the difference between absolute configuration parameters and abstract configuration variables, and prefer the use of absolute configuration whenever possible. The abstract form of configuration may be satisfied using one set of resources which are selected when the configuration/command is executed, and different resources when executed a second time, or after a re-boot, etc. The actual implementation of an abstract configuration might be discovered using the automated collection capability described above.

Reporting: The complete configuration should be reported, and this further encourages automated execution and collection. The category of each configured item should be identified as absolute configuration parameters or abstract configuration variables.

A.1.4 Generation of Multiple Streams

Existing methods, such as section 12 of IETF RFC 2544 [1], specify the method(s) used to generate multiple streams in tests that follow the single stream test configuration. IETF RFC 2544 [1] specifies that Destination addresses should be varied, but there are use cases for varying Source address (or both Source and Destination). Also, the results of the VSPERF testing [i.10] indicate throughput sensitivity to the fields that vary. For example, a web hosting use case would involve many Source addresses and one Destination address. Another use case could be a core network function where streams from many Source and many Destination addresses are encountered.

Bidirectional traffic should possess complimentary stream characteristics. For example, specifying many-to-one streams in one direction should be complimented by one-to-many streams in the other direction.

When considering the range of addresses for field variation, there are several methods available to choose each individual address in the stream of packets. For example, an address could be randomly selected from the allowed range (without replacement, until the entire range is exhausted), or an addresses could be incremented/decremented within the address range. Also, addresses and other fields could be selected from a stored list created prior to testing (e.g. a packet capture file).

Potential Mitigations:

Test Set-up:

1) The use case(s) addressed by the goals of the test will help to determine whether tests with multiple streams are necessary, and how the streams should be constructed in terms of the address fields that vary.

2) Bidirectional traffic should possess complimentary stream characteristics.

Test Device/Function Capabilities:

1) The Traffic Generator should be capable of constructing all the streams required by the use cases.

2) The address field variation methods should be specified in the Traffic Generator configuration. The distribution of consecutive packets from a flow sent on the wire should also be specified.

A.1.5 Test Stream Variations over Size and Protocol

In addition to tests streams with simple packet characteristics and constant packet sizes, there is sufficient evidence of NFVI processing sensitivity to variations in packet header composition and protocol type. Further, results with a mix of sizes do not always fall within the performance range established by the constant size results (over a range of packet sizes). Experience also indicates performance sensitivities and unexpected results with different mixes of packet sizes.

Potential Mitigations:

Configuration:

1) The use case(s) addressed by the goals of the test will help to determine what protocol combinations should be tested, and how the streams should be constructed in terms of the packet header composition and mix of packet sizes.

2) At least two mixes of packet size should be specified for testing. IETF RFC 6985 [4] provides several ways to encode and specify mixes of packet sizes, even repeatable packet size sequences.

Test Device/Function Capabilities:

1) The Traffic Generator should be capable of constructing all the packet types required by the use cases.

2) The distribution or sequence of packet sizes sent on the wire should also be specified in the Traffic Generator configuration.

Method:

1) Procedures should encourage test repetition over constant packet sizes, and allow for additional repetitions where the packet header composition is varied and with a mix of packet sizes.

A.1.6 Testing during dynamic flow establishment

Existing methods encourage testing under stable routing and forwarding conditions, such those defined in section 23 of IETF RFC 2544 [1]. There is anecdotal evidence that SDN controller reactive flow processing and flow table installation may cause throughput limitations (more than just the effects on new flows, in terms of loss and delay on first packet). Thus, it appears worthwhile to test while flows are being installed, as well as static case. In addition, one might test Throughput while updating the lookup table, similar to procedures for physical network devices.

Potential Mitigations:

Method:

1) Procedures should permit test repetition to assess performance during flow establishment, and define the methods to compare the dynamic and static flow cases. As an example of the dynamic flow test case, some fraction of total offered load would require "first packet" processing, while all other flows would be learned before measurements during a trial. The comparative static tests would use the same offered load level, but all flows would be learned before measurement.

A.1.7 Monitor Operational Infrastructure Metrics During Tests

Both [i.8] and [i.9] recommend measurements on infrastructure utilization (e.g. CPU and Memory utilization) during benchmarking, as auxiliary metrics. These metrics may help to explain what system resource limit accounts for the black-box results observed externally, and they may be useful in future operational monitoring and capacity planning.

Potential Mitigations:

Method:

1) Procedures should permit collection of measurements of system resource utilization alongside the benchmark measurements.

2) Procedures should encourage test repetition with and without system resource measurements, to assess any effect on performance and quantify the extent where possible.

Annex B (informative): Development of New Search Strategies

B.1 Mitigating background processes that cause infrequent loss

This clause addresses the scenario sometimes found in NFVI benchmarking, where the true zero-loss conditions are difficult to maintain, or impossible to achieve with the many essential background processes running in general purpose computing environments and operating systems.

The present benchmarking test philosophy has relied on measuring frame losses as an indicator that the resources of the DUT are exhausted, and that the Offered Load is expected to be reduced in the next step of the search. In essence, the search space is a sorted array of Offered Load levels, and each level can be paired with the binary outcome of the results of the Trial conducted at that load level: TRUE when resources have been exhausted (loss was observed), and FALSE when the resources were sufficient for the Offered Load.

Figure B.1-1: The Search Array

The number of elements in the search array is equal to Maximum fps (frames per second) divided by the traffic generator step size parameter. In the example of figure B.1-1, the step size has been configured at 1 Mfps, so the search array contains n = 12 elements at the start of the search.

In conventional physical device benchmarking, all Trial outcomes are treated as absolute truth, or if loss occurred unexpectedly the source was located and fixed. Frame loss from lab facilities or cabling could not be tolerated. One of the challenges in benchmarking the NFVI is that general purpose computing platforms may not be able to support this level of certainty. But the methods can be adapted.

This clause provides the model of the effects of background processes that demand sufficient resources to cause frame loss on an infrequent basis. Based on discussion in [i.14] the problem of searching games with errors can be modelled as a set of communication channels between the Questioner and the Responder. To make this model immediately relevant to benchmarking, the Questioner poses inquiries by setting the Offered Load level of the Traffic Generator, and responses are based on observation of the stream in the Traffic Receiver observing on a black-box basis. The Responder is the role that the DUT plays, and internal resources and processes of the DUT consider the Offered Load inquiry and formulate signals which are ultimately returned to the Questioner/Traffic Receiver.

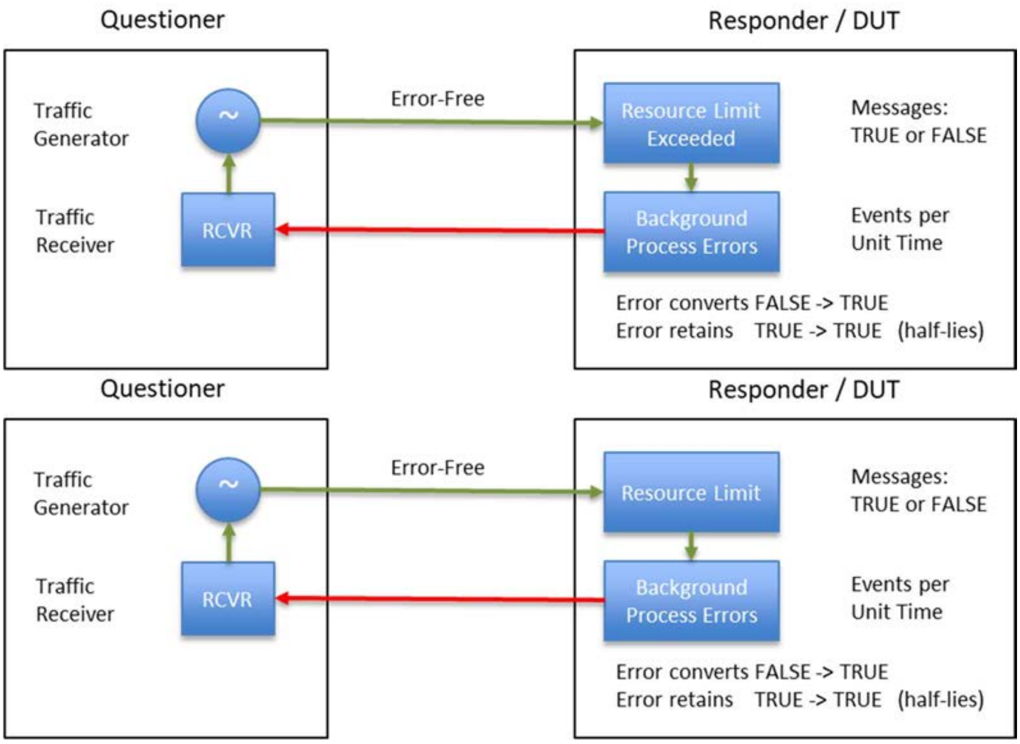

Figure B.1-2: The Search Process as a Communication System

Figure B.1-2 illustrates the model of communication between a Questioner and Responder adapted to the entities involved in benchmarking. The Responder or DUT receives the inquiry of Offered Load over an Error-free channel, and determines if a Resource Limit has been exceeded (True) or that the Limit has not been exceeded (False) and all offered frames have been accommodated. However, the DUT can corrupt a False Resource Limit message if a background process causes Frame Loss, and convert that message to True when an infrequent "error" event occurs.

Alternatively, when the Responder or DUT receives the inquiry of Offered Load over the Error-free channel, and determines that a Resource Limit has been exceeded (True), it communicates this outcome by discarding frames in practice. And, if some infrequent background process causes additional Frame Loss, the message describing the Resource Limit remains True. This is the condition of half-lies described in [i.14], and it is an important feature of this benchmarking model and the new search method defined in the present document. Recognition that only the False message can be corrupted means that:

- Only the True/Loss outcome requires verification, the False/Loss-free outcome of any Trial can be trusted. Once losses occur, they cannot be reversed by the communication channel to become Loss-free.
- Repetition of a Trial yielding the True/Loss outcome should be sufficient to obtain a more confident result, because the background processes causing error are designed to be infrequent, having a long inter-event interval, T, with respect to the time required to conduct, evaluate, and repeat automated trials. Further analysis of this point follows below.

The Trial duration is taken as t seconds.

The frequency of all background process-derived errors/losses is F events per second, and F = 1/T. The target of Long Duration testing is to characterize the frequency of the background process-derived errors/losses, and also the variation of this frequency.

The relationship $T > x \times t$ illustrates that the Trial duration is much smaller than the inter-event interval of errors, where the tester chooses $x >> 1$ so that x trials can be conducted between events.

The frequency interpretation of probabilities yields $p = t/T$, where p is the probability that background process-derived errors/losses will affect any Trial.

The Trial repetitions for verification have the effect of increasing the number of steps needed to search the array successfully. The Binary search requires $O(\log_2 (n))$ steps, and the Binary search with Loss Verification and the maximum repetitions r = 2 requires $1,5 \times O(\log_2 (n))$ steps, assuming that half the outcomes yield True or Loss, and require a repeated Trial. This is deemed to be a reasonable increase when p is non-negligible, because a more confident result is better than repeating $O(\log_2 (n))$ steps additional times when Sets of tests result in wide variation.

Annex C (informative): Authors & contributors

The following people have contributed to the present document:

Rapporteur:
Mr. Al Morton, AT&T Labs

Other contributors:
Ms. Maryam Tahhan, Intel Corporation

Mr. Pierre Lynch, Ixia Technologies

Mr. Gergely Csatari, Nokia

Mr. Bruno Chatras, Orange

Mr. Joerg Aelken, Ericsson Telefonaktiebolaget LM

Mr. Akram Al Sawaf, EANTC

Mr. Carsten Rossenhoevel, EANTC

Dr. Sridhar Rao, Spirent Communications

Mr. Trevor Cooper, Intel Corporation

Mr. Eddie Arrage, FutureWei Technologies

Mr. Alec Hothan, Cisco Systems

Mr. Maciek Konstantynowicz, Cisco Systems

Mr. Cheng Huang, Huawei Technologies Co. Ltd.

Mr. Xianming Li, Huawei Technologies Co. Ltd.

History

Document history		
V3.1.1	October 2018	Publication