



INTEL EPA (ENHANCED PLATFORM AWARENESS) UPDATES FOR NFV

Alex, Wooram, Kim

CCIE#9351/VCP5-DCV/RHCSA
Service Provider Solution Architect
alex.kim@intel.com

AGENDA

- INTEL EPA TECHNOLOGY UPDATE FOR NFV
 - INTRODUCTION OF NFV
 - WHY EPA FOR NFV?
 - EPA ROADMAP
- Q/A

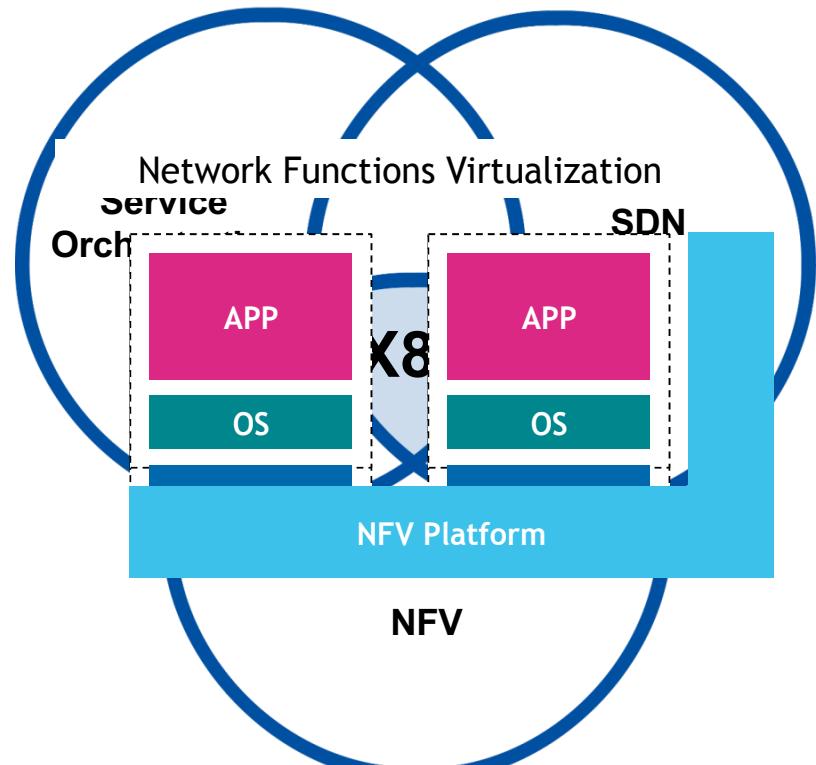


Intel EPA Technology Update



: Introduction of NFV

- Initiated December 2012 from Telco/Service Provider
- Objective
 - : Virtualization as part of operators network transformation with CAPEX/OPEX reductions
- NFV(Network Function Virtualization)
 - : decouples the network functions**
(CPE, Router, NAT, Firewall, ADC, DPI, IPS/IDS, WAAS, SBC, BGP RR...) from ASIC, NPU appliances. to
 - industry standard server hardware(X86 COTS)** as Software!

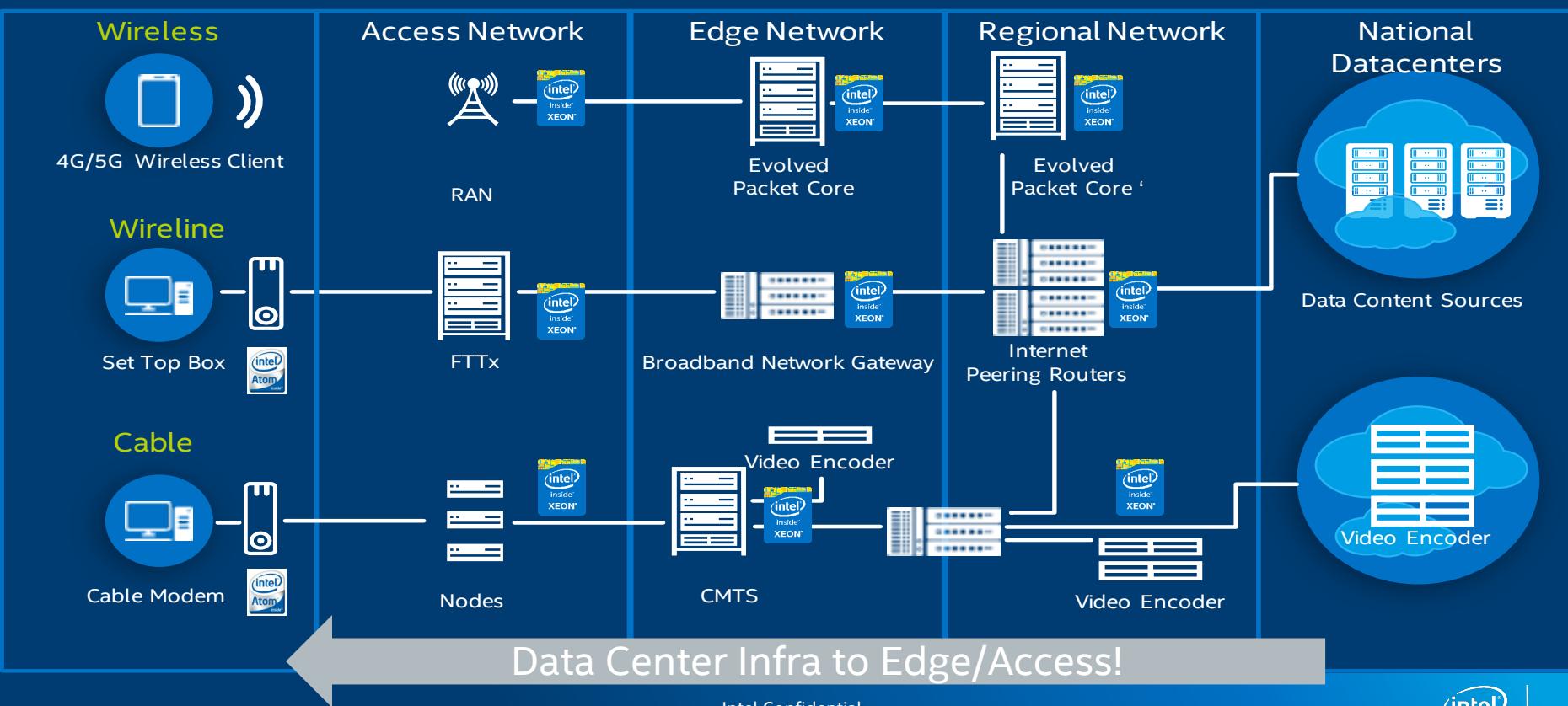


Sources:<https://www.sdncentral.com/which-is-better-sdn-or-nfv/> http://portal.etsi.org/nfv/nfv_white_paper.pdf

Intel EPA Technology Update

: Introduction of NFV

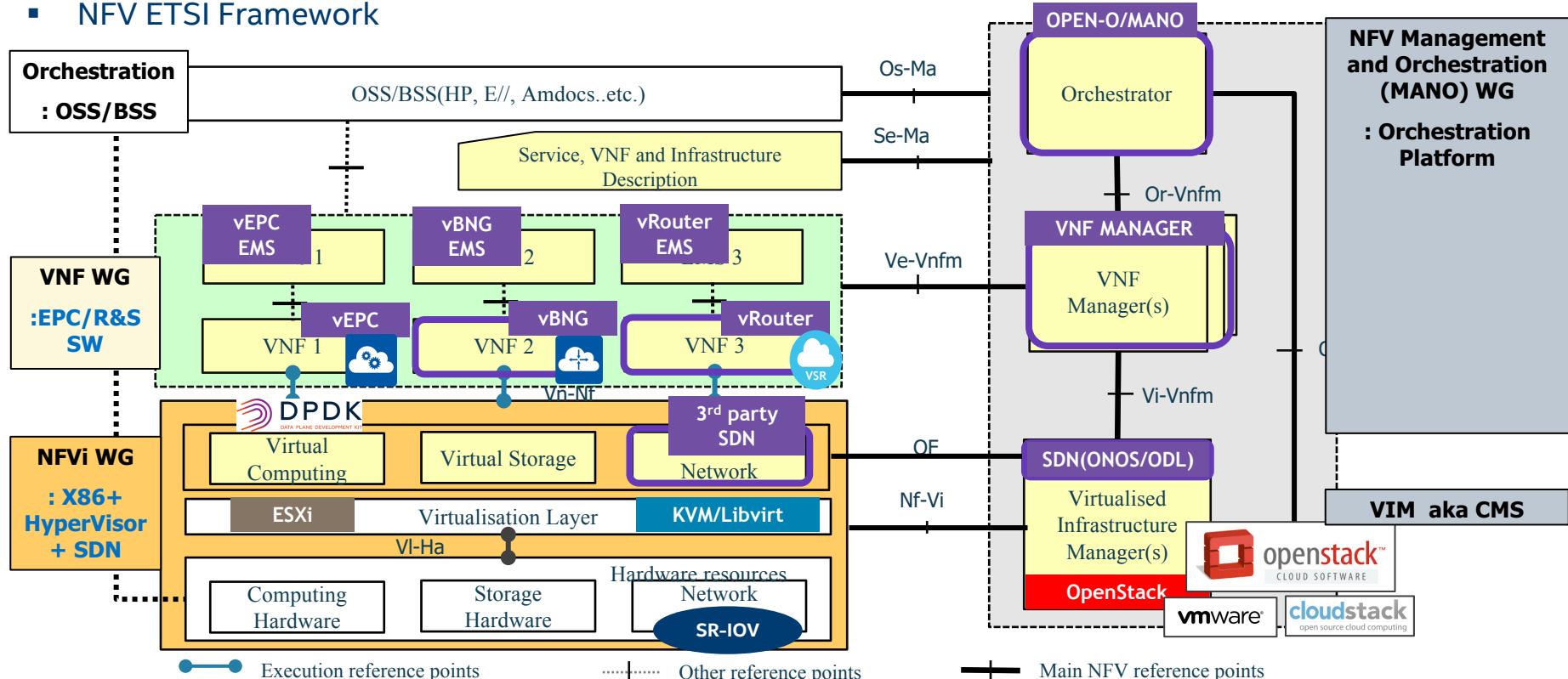
- Comparison of Wireless, Telco Broadband and MSO Network Architecture



Intel EPA Technology Update

: Introduction of NFV

- NFV ETSI Framework



Intel EPA Technology Update

: Why EPA for NFV?

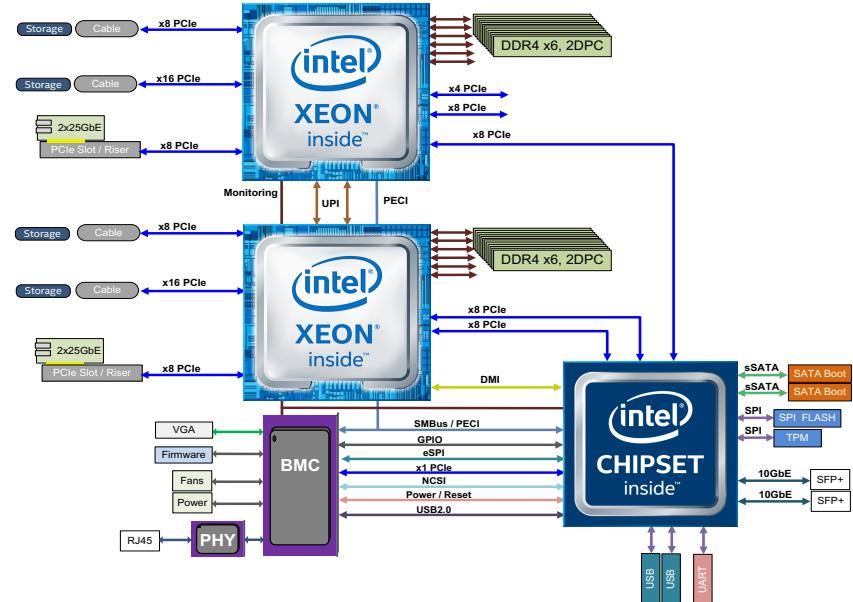
▪ Requirements on NFV infrastructure, not IT

- : deterministic behavior with High performance
- : small packet processing
- : low latency (<10µs for CPE and access functions), jitter
- : availability (detect failed VMs in <1s and autorestart, recover host failures)
- : advanced management (OSS/BSS)

▪ Major Bottleneck in NFV

- : CPU Interrupt,
- : Context switching overhead!
- : Memory copies !

▪ Resolution for NFV



But, How to Discover, Provision and Assign right workload placement in Telco Cloud Environment?

Intel EPA Technology Update

: Why EPA for NFV?

- **Openstack EPA**

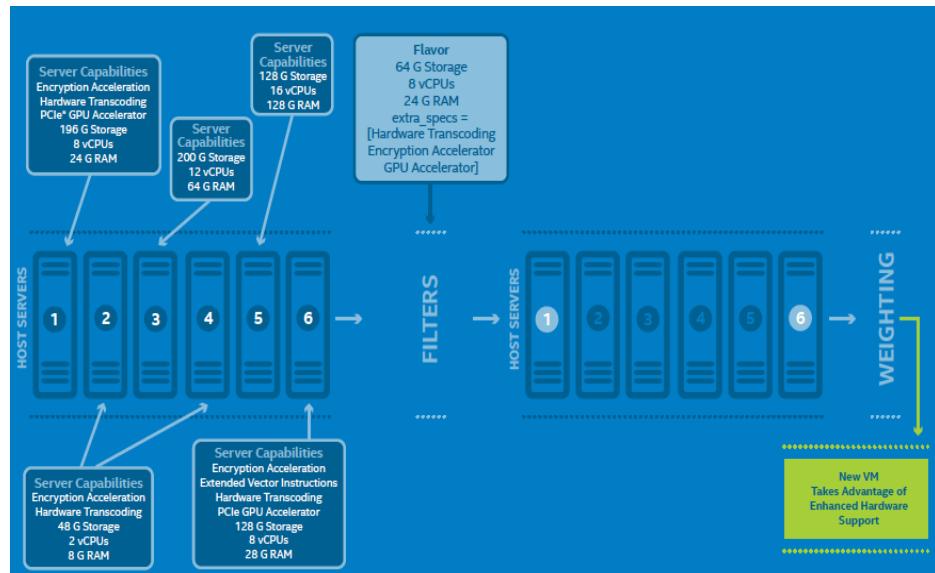
- Umbrella term for Intel's contributions to OpenStack

- Exposes platform capabilities to OpenStack
 - : better view of underlying platform capabilities

- OpenStack intelligence to filter platforms with specific capabilities
 - : match VM workload with platform capabilities
 - : finer grained view of the server platform
 - assign the cloud application workload to the best virtual resource

- **EPA focus areas:**

- : Detecting NFVi platform(COTS) capabilities
 - : Configuring NFVi platform(COTS) capabilities



EPA provide design advantages in Discovery, Provisioning for High Performance, Management!

Intel EPA Technology Update

: Why EPA for NFV?

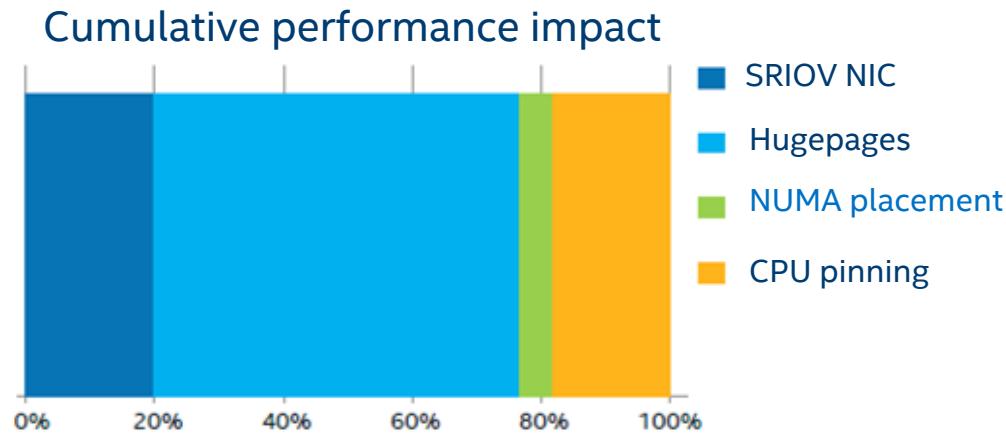
- EPA Characterizations

- L3 forwarding Benchmark on Each Feature

- SR-IOV NIC
 - Huge Pages
 - NUMA placement
 - CPU pinning

- Test Condition

- DPDK based virtual BNG(BRAS) with OpenStack
 - 2 * Intel® Xeon® processor E5-2690 v2 @ 3.0 GHz
 - L3 forwarding, 64B
 - Max throughput with no packet loss
(Non-Drop Rate)



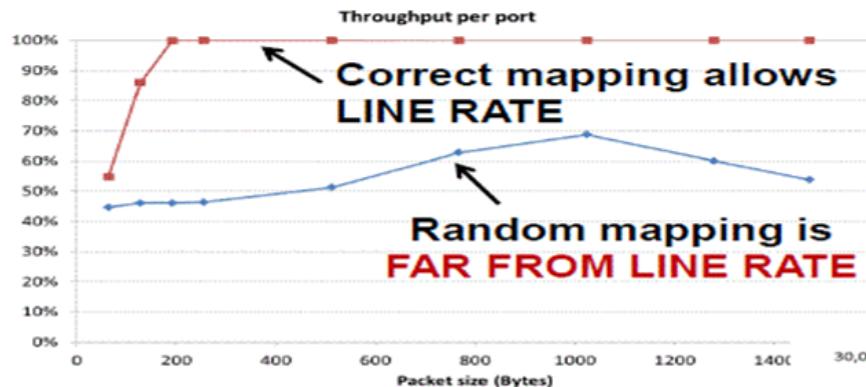
Source: Intel white paper QoS in BRAS with Linux* and IA, August 2014

Mandatory Key Features is Hugepages, NUMA, CPU Pinning for packet processing!

Intel EPA Technology Update

: Why EPA for NFV?

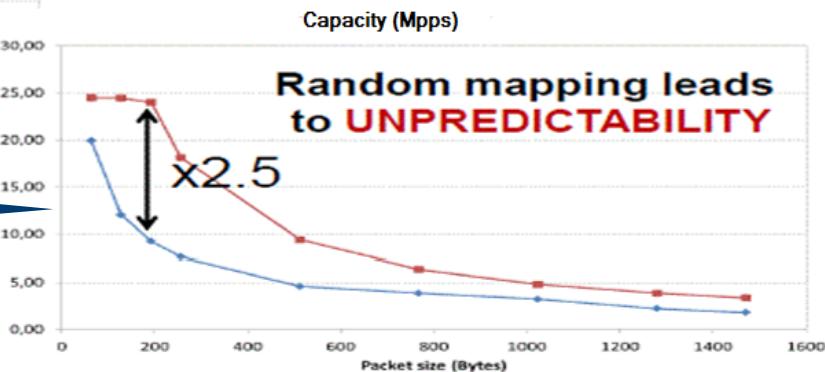
- Brocade 5600 vRouter EPA Characterizations – Throughput/port & System Capacity (Packet per Sec.)



Correct mapping allows LINE RATE

Random mapping is FAR FROM LINE RATE

UP to 50% TP degradation!



250% System capacity degradation!

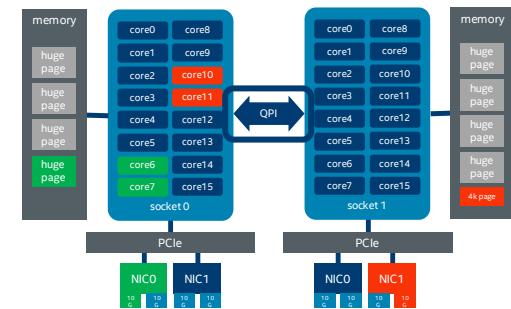
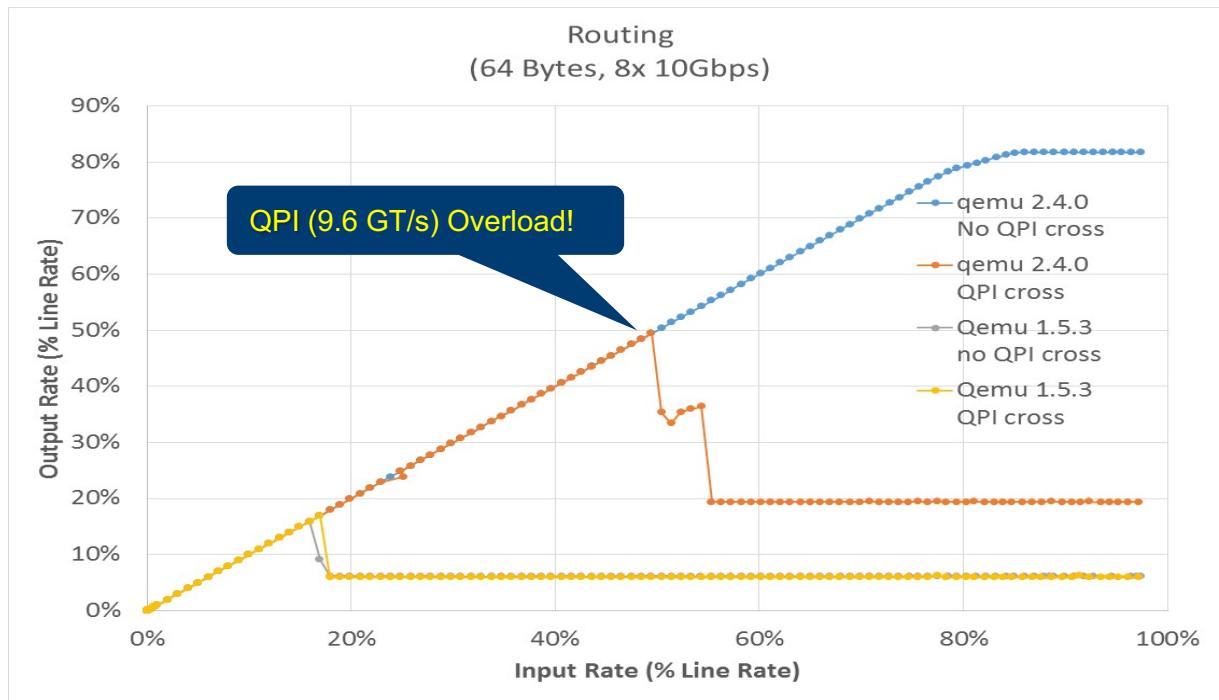
Random mapping leads to UNPREDICTABILITY

x2.5

Intel EPA Technology Update

: Why EPA for NFV?

- Brocade 5600 vRouter EPA Characterizations - Quick Path Interconnect (QPI), CPU to CPU connection

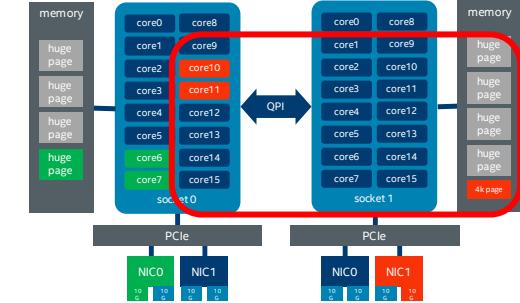
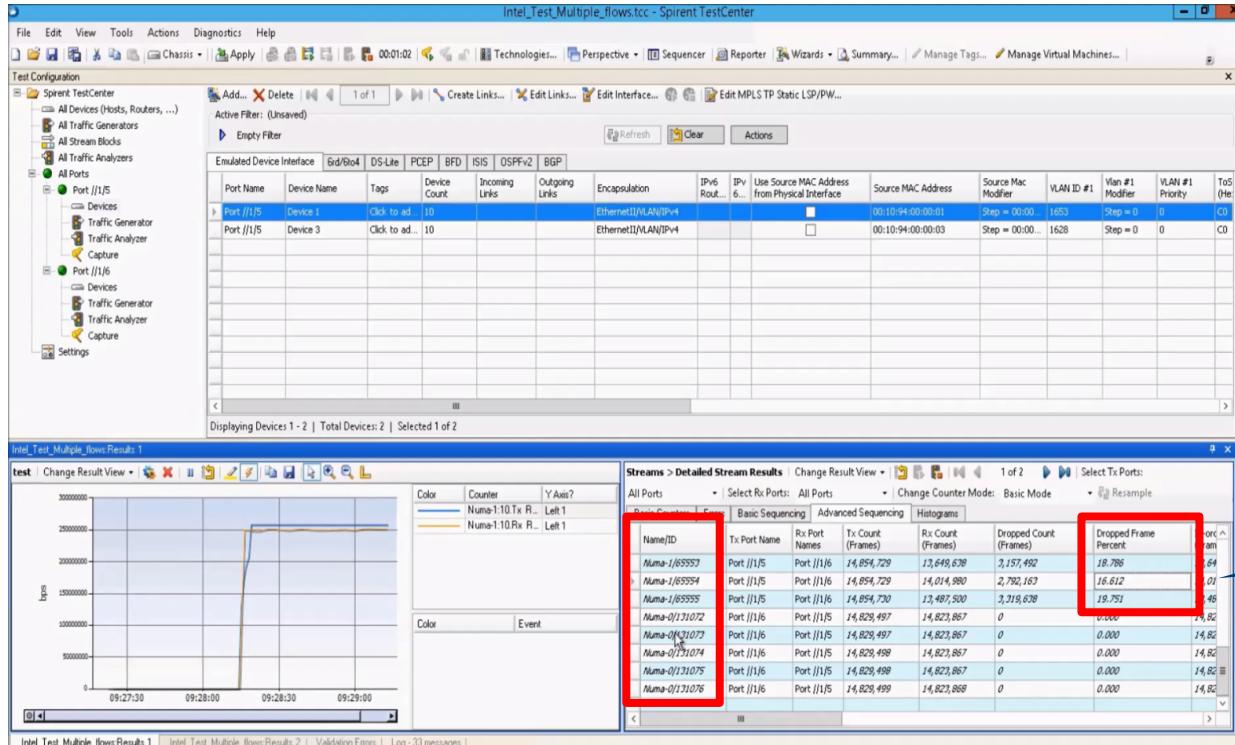


Crossing QPI causes problematic overload behavior!

Intel EPA Technology Update

: Why EPA for NFV?

- Brocade 5600 vRouter EPA Characterizations - NUMA(Non Uniform Memory Access)

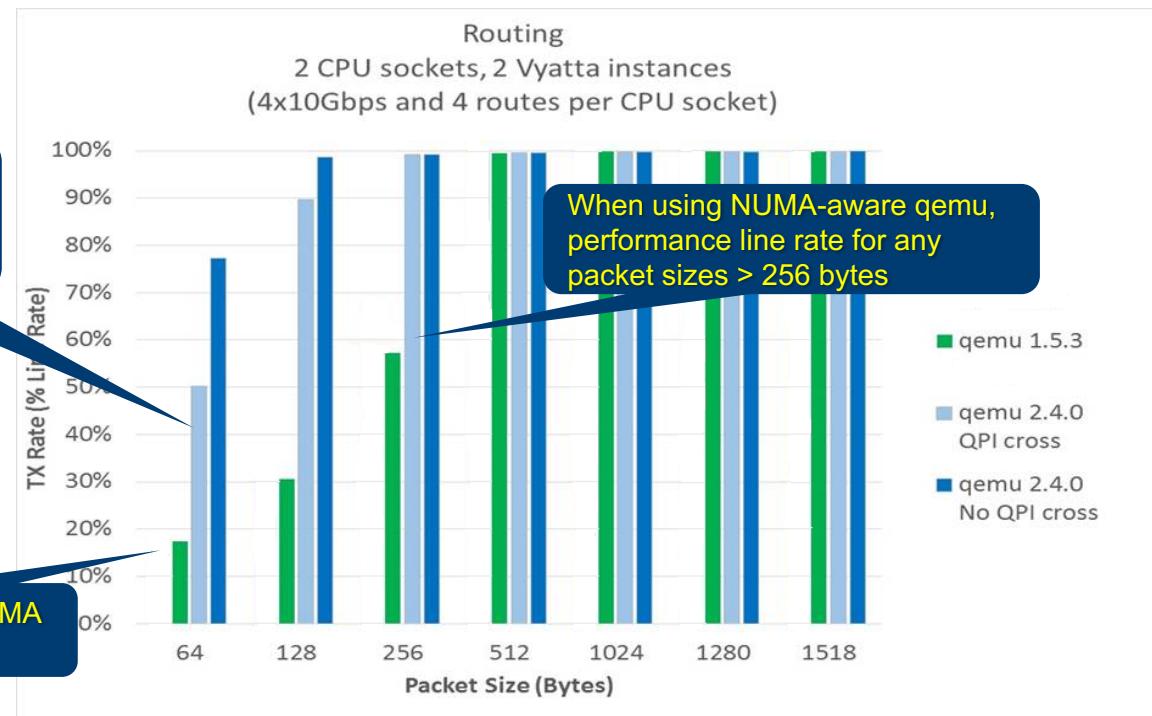


NUMA unaware VNF causes problematic packet drop behavior!

Intel EPA Technology Update

: Why EPA for NFV?

- Brocade 5600 vRouter EPA Characterizations - NUMA(Non Uniform Memory Access)

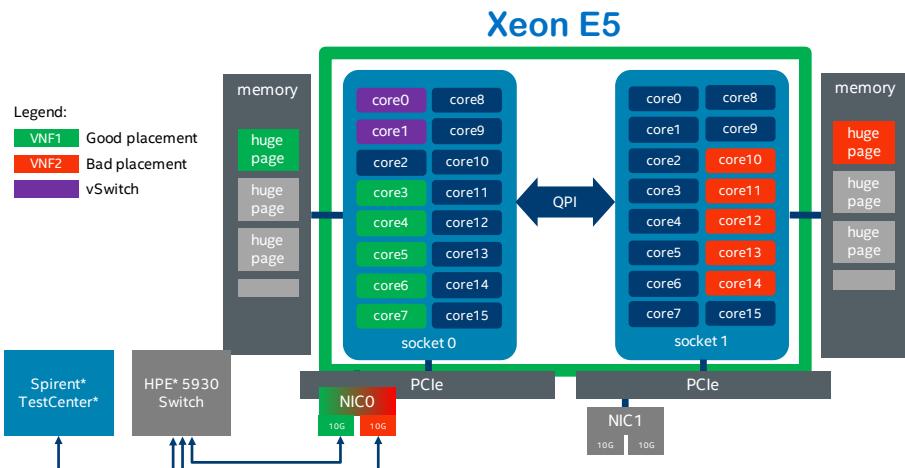


Best Known Configuration/Reference architecture is Essential!

Intel EPA Technology Update

: Why EPA for NFV?

- Best Known EPA Configuration



CONFIGURATION GUIDE



Enabling Enhanced Platform Awareness for Superior Packet Processing in OpenStack*

Intel Corporation
Datacenter Network
Solutions Group

Authors

Shivapriya Hiremath
Solution Software Engineer,
Intel Corporation

Vaidyanathan Krishnamoorthy
Solution Software Engineer,
Intel Corporation

Andrew Duignan
Solution Software Engineer,
Intel Corporation

James Chapman
Solution Software Engineer,
Intel Corporation

1.0 Introduction

Network functions virtualization (NFV) emerged in response to the problems that network operators face when developing and maintaining their infrastructures built with traditional hardware appliances. The growing demands for new services, constantly increasing network traffic, and the need for rapid service delivery are just some of the challenges that industry cannot overcome with the traditional network architectures. Moreover, the technological advancements cause the hardware life cycle to shorten. This leads to long-lasting procure-design-deploy cycles that require unique human skills.

NFV is a network architecture concept designed by a group of telecommunications industry representatives of the European Telecommunications Standards Institute (ETSI). NFV primarily aims to leverage the existing technologies to virtualize entire classes of network nodes that in turn may be connected to form compound infrastructures. Designed with modularity, reliability, and scalability in mind, such infrastructures are capable of running complex services.

The architecture of NFV consists of three blocks: virtualized network functions (VNFs), NFV infrastructure (NFVI), and NFV management and orchestration (MANO). One of the MANO components of the ETSI NFV model, virtualized infrastructure manager (VIM), is responsible for managing and orchestrating the NFVI.

https://builders.intel.com/docs/networkbuilders/EPA_Enablement_Guide_V2.pdf

Intel EPA Technology Update

: EPA in MANO



■ TOSCA EPA Example - vRouter

```
interfaces:
  Standard:
    configure: scripts/vdu/data_configure.sh # included in the CSAR
  requirements:
    - dependency:
        node: data_host
        relationship:
          type: vnfsdk.VirtualHostedOn
        properties:
          performance_hints:
            cpu_affinity:
              value: 'true'
            container:
              value: 'false'
              priority: 1.0 # hints that it's absolutely required
  capabilities:
    nfv_compute:
      # This is the "standard" way to define performance
      optimization in the # NFV profile
    properties:
      mem_page_size: large
      cpu_allocation:
        core_count: 4
      numa_node_count: 2
      numa_nodes:
        primary:
          id: 0
          mem_size: 16 GB
        vcpus:
          a: 2
          b: 3
        secondary:
          id: 1
          mem_size: 2 GB
```

• YANG EPA Example - Open Source MANO VNFD

```
732   leaf mempage-size {
733     description
734       "Memory page allocation size. If a VM requires
735       hugepages, it should choose LARGE or SIZE_2MB
736       or SIZE_1GB. If the VM prefers hugepages it
737       should choose PREFER_LARGE.
738       LARGE      : Require hugepages (either 2MB or 1GB)
739       SMALL     : Doesn't require hugepages
740       SIZE_2MB  : Requires 2MB hugepages
741       SIZE_1GB  : Requires 1GB hugepages
742       PREFER_LARGE : Application prefers hugepages",
743     type enumeration {
744       enum LARGE;
745       enum SMALL;
746       enum SIZE_2MB;
747       enum SIZE_1GB;
748       enum PREFER_LARGE;
749     }
750   }
751
752   leaf cpupinning-policy {
753     description
754       "CPU pinning policy describes association
755       between virtual CPUs in guest and the
756       physical CPUs in the host.
757       DEDICATED : Virtual CPUs are pinned to
758           physical CPUs
759       SHARED    : Multiple VMs may share the
760           same physical CPUs.
761       ANY       : Any policy is acceptable for the VM";
762     type enumeration {
763       enum DEDICATED;
764       enum SHARED;
765       enum ANY;
766     }
767     default "ANY";
768   }
```

Intel EPA Technology Update

: EPA in MANO

- EPA at NFV-Orchestrator

- EPA enabled open source NFV-O:

- Rift.io* Rift.ware*
- GigaSpaces* Cloudify*
- <http://getcloudify.org/>

The screenshot shows the Rift.ware interface for deploying an Enhanced Network Service. The process is divided into three steps: 1. SELECT NETWORK SERVICE (IP Traffic), 2. SELECT POOL (OpenStack 1), and 3. SLA PARAMETERS.

The SLA PARAMETERS section is highlighted with a red border. It contains a table of EPA parameters with On/Off switches:

EPA PARAMETERS	On	Off
TXT	On	Off
NUMA NODES	On	Off
DPDK ACCELERATED	On	Off
NIC	RNIC Rev A0	Nantic
MEMORY PAGE SIZE	On	Off
OVS ACCELERATION	On	Off
DDIO	On	Off
CAT	On	Off
QAT	On	Off
PCI-PASSTHRU	On	Off
CPU PINNING POLICY	On	Off

Below the table, there is an ADVANCED PARAMETERS section. At the bottom of the interface are buttons for SAVE FLEET and LAUNCH FLEET.

A code snippet from the deployment blueprint is visible at the bottom left:

```
yaml
xwl_eps_deployment:
    type: cloudify.nodes.BlueprintDeployment
    properties:
        blueprint_id: { get_input: rv_blueprint_id }
    interfaces:
        cloudify.interfaces.lifecycle:
            create:
                inputs:
                    deployment_inputs:
                        # cloudify plugin checks whether Network or Port are associated with an SR-IOV / PassThrough
                        network: { get_property: [ 'network', 'resource_id' ] }
                        max_bandwidth: { get_property: [ 'internal_network', 'resource_id' ] }
                        anti-affinity: { get_property: [ 'vnfs-anti-affinity', 'resource_id' ] }
                relationships:
                    - target: front-end network
                      type: cloudify.relationships.connected_to
                    - target: internal network
                      type: cloudify.relationships.connected_to
                    - target: load-echo vnf
                      type: cloudify.relationships.depends_on
                    - type: cloudify.relationships.depends_on
                      target: vnfs anti-affinity

```

Intel EPA Technology Update

: Why EPA for NFV?

- EPA supported Openstack and NFV Orchestrator



Intel EPA Technology Update

: EPA Roadmap

- EPA Feature

Feature	Software Dependencies
Host CPU feature request	Havana
PCI passthrough & SR-IOV Support	Havana
NUMA awareness and placement	Juno
Hugepage support	Kilo
CPU pinning	Kilo
Security Groups for OVS+DPDK (Stateless)	Mitaka
Telemetry Capture (via collectd)	Mitaka
OVS+DPDK (Merged with OVS agent)	Mitaka
OVS+DPDK controlled by ODL	Mitaka
OVF Meta-Data Import	Mitaka
Power management	Newton
Resource Director Technology	Newton

Intel EPA Technology Update

: Additional Resources

<https://builders.intel.com/university/networkbuilders/coursescategory/openstack-1#openstack-enhanced-platform-awareness-101>

<https://builders.intel.com/university/networkbuilders/coursescategory/openstack-1#openstack-enhanced-platform-awareness-102>

Please visit:

OSM: <https://osm.etsi.org>

OSM YouTube Channel:

<https://www.youtube.com/channel/UC5ylvBvh7Eg8deWTPzKOjg>

ONAP: <https://www.onap.org/>

