Corigine

WHITE PAPER

Accelerating Contrail vRouter

WHEN DEPLOYED WITH THE JUNIPER NETWORKS CONTRAIL CLOUD NETWORKING PLATFORM, THE CORIGINE AGILIO VROUTER SOLUTION DELIVERS ACCELERATED PERFORMANCE THAT ENABLES INCREASED BUSINESS AGILITY AND REVENUE GROWTH.

CONTENTS

EXECUTIVE SUMMARY

Juniper Networks Contrail Cloud is an open, standards-based software solution that delivers network virtualization and service automation for federated cloud networks. It provides self-service provisioning, improves network troubleshooting and diagnostics, and enables service chaining for dynamic application environments across enterprise virtual private cloud (VPC), managed infrastructure as a service (IaaS) and NFV use cases. Contrail vRouter is the software entity that forwards network packets between physical and virtual interfaces on a specific host. While other software entities, such as Open vSwitch (OVS), forwards packets on Layer 2/switching (L2), vRouter operates on Layer 3/routing (L3).

When deployed with the Juniper Networks Contrail Cloud networking platform, the Corigine Agilio vRouter solution delivers accelerated performance that enables increased business agility and revenue growth. This document will illustrate the advantages of a scalable, standards based, hardware-accelerated Contrail vRouter solution and present performance comparisons between the accelerated Linux kernel implementation and the user space Intel Data Plane Development Kit (DPDK) implementation.

1. ABOUT CONTRAIL 4.1

Contrail simplifies the creation and management of virtual networks to enable policy-based automation, greatly reducing the need for physical and operational infrastructure typically required to support network management. In addition, it uses mature technologies to address key challenges of large-scale managed environments, including multitenancy, network segmentation, network access control and IP service enablement. These challenges are particularly difficult in evolving dynamic application environments such as the web, gaming, big data, cloud and the like.

Contrail allows a tenant or a cloud service provider to abstract virtual networks at a higher layer to eliminate device-level configuration and easily control and manage policies for tenant virtual networks. A browser-based user interface enables users to define virtual network and network service policies, then configure and interconnect networks simply by attaching policies. Contrail also extends native IP capabilities to the hosts (compute nodes) in the data center to address the scale, resiliency and service enablement challenges of traditional orchestration platforms.

Contrail can be used with open cloud orchestration systems such as OpenStack. It can also interact with other systems and applications based on operations support system (OSS) and business support systems (BSS), using northbound APIs. This allows customers to build elastic architectures that leverage the benefits of cloud computing — agility, self-service, efficiency and flexibility — while providing an interoperable, scale-out control plane for network services within and across network domains.

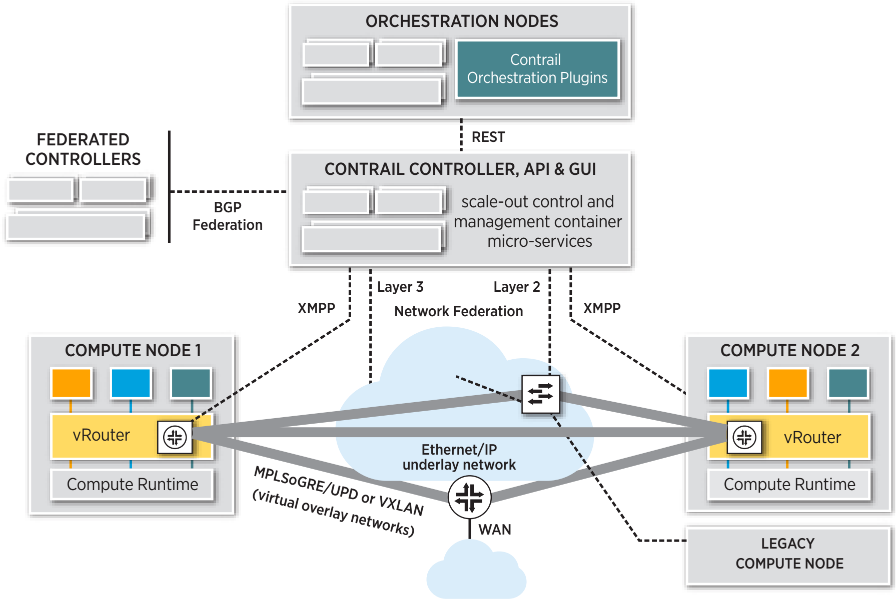

Figure 1: Contrail 4.1 system overview

2. CONTRAIL VROUTER

Contrail vRouter is the software entity that forwards network packets between physical and virtual interfaces on a specific host. While other software entities, such as Open vSwitch (OVS), forwards packets on Layer 2/switching (L2), vRouter operates on Layer 3/routing (L3).

Contrail vRouter is comprised of the forwarding plane (data plane) and the agent which is the interface to the control plane and can be summarized with the diagram below.

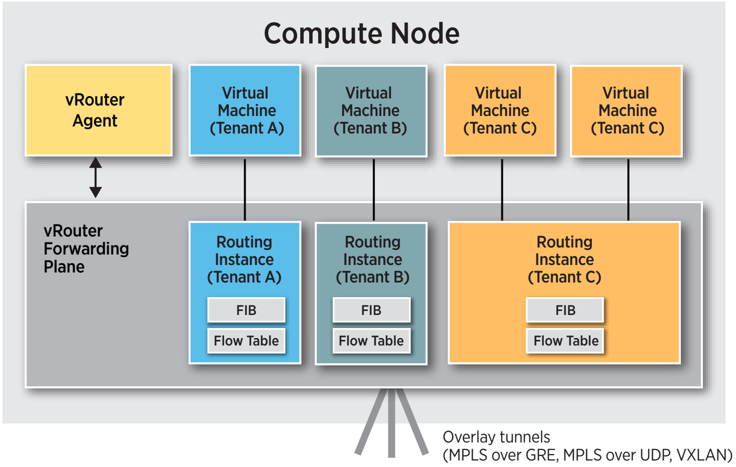

Figure 2: Contrail vRouter system overview

Contrail vRouter runs on the compute nodes of the cloud or NFV infrastructure. It gets network tenancy, VPN and reachability information from the control function nodes and ensures native L2 or L3 services for the Linux host on which it runs or for the containers or virtual machines (VMs) of that host. Each vRouter is connected to at least two control nodes to optimize system resiliency. The vRouters run in one of a few high-performance implementations: as a Linux kernel module, as a DPDK-based process, or embedded into a Smart network interface card (NIC).

2.1 KEY FEATURES

- Routing and Bridging: The vRouter forwarding plane provides overlay L3 routing and L2 bridging in multi-tenant environments. Overlay orchestration is agnostic to the underlay network. When Contrail is using Ethernet VPN (EVPN) to orchestrate an underlay physical switching fabric it can work with any switches supporting the EVPN open standard and VXLAN data plane. This has been qualified on Juniper Networks switches.
- Load Balancing: Equal-cost multipath (ECMP) load balancing with session affinity is built into the vRouter's forwarding plane. It distributes traffic across endpoints like VNF network services.
- Security and Multi-tenancy: The use of tenant domains and L3 VPNs to create virtual networks inherently provides a secure, segregated environment. Virtual networks can only talk to each other when policies provide for it. vRouter has built-in distributed L3 and L4 firewall capabilities to build and manage security policies.
- Gateway Services: Contrail works to gateway the overlays on most physical or VM-based routing and switching equipment that supports MP-BGP control plane protocols and a VXLAN, MPLSoUDP or MPLSoGRE data plane. It can support multiple, active gateways.
- High Availability: Active/active redundancy is built in to all control and management components.
- API Services: REST APIs for configuration, operation and analytics provide seamless integration

with popular or customized orchestration systems.

- Plugins for Cluster Orchestrators: Set up networks manually or with APIs. Plugins for third-party cluster orchestrators use the API to automate network and security in conjunction with workload orchestration for Kubernetes, OpenStack, OpenShift and Mesos.

KEY BENEFITS:

OFFLOAD AND ACCELERATION OF CONTRAIL VROUTER HIGH-PERFORMANCE VROUTER DATAPATH SUPPORTING 20-30MPPS

USES ONLY ONE CORE FOR CONTRAIL VROUTER DATAPATH OFFLOADING X86 CPU CORES

RECLAIMED CPU CORES FROM CONTRAIL VROUTER OFFLOAD ALLOW MORE VMs AND VNFs TO BE DEPLOYED PER SERVER

NATIVE NETWORKING I/O PERFORMANCE TO VMs AND VNFs THROUGH THE USE OF SR-IOV

HARDWARE INDEPENDENCE AND VM MOBILITY VIA XVIO

3. AGILIO VROUTER SOFTWARE

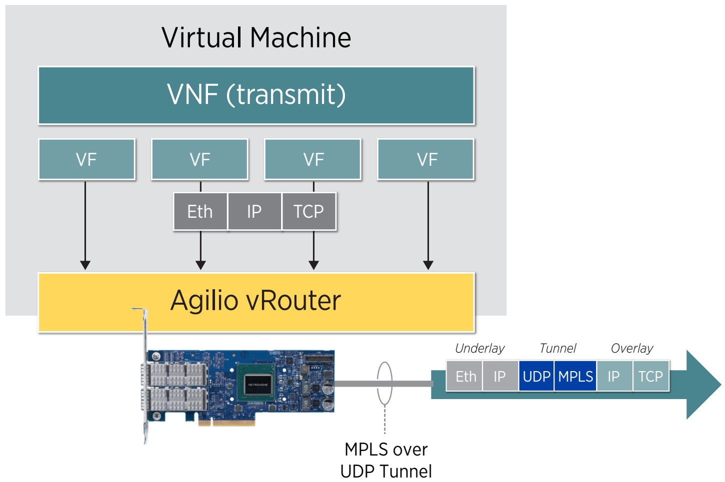

Figure 3: Agilio vRouter encapsulation for overlay networking

Corigine enables acceleration of Contrail vRouter to preserve valuable host CPU resources by offloading the Contrail vRouter datapath to Corigine's Agilio CX SmartNICs. The vRout-er lookup tables are mirrored on the SmartNIC to facilitate fast lookup and routing decisions which improves the packet delivery performance to VNFs and VMs from an average of 3Mpps to 20-23Mpps when comparing the non-accelerated versus the accelerated kernel vRouter implementations. The user space DPDK vRouter implementation remains less than 6Mpps and uses valuable host CPU cores. By increasing network performance and efficiency, dramatic improvements to cloud workload capacity are realized. Corigine's Agilio vRouter software provides the ability to decrease the number of servers required to support the same level of service, thereby decreasing data center TCO.

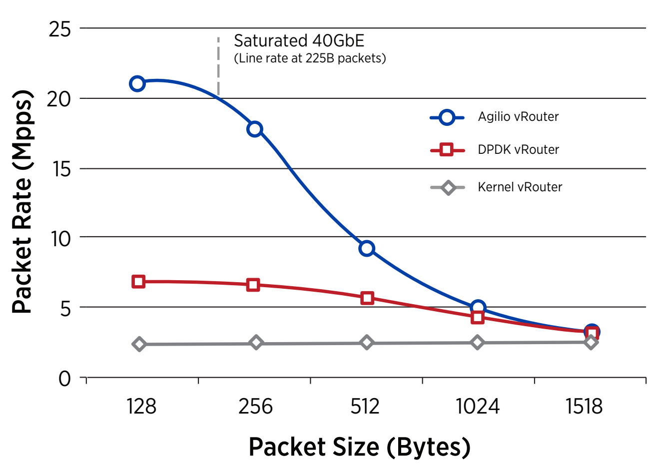

Figure 4: Agilio vRouter performance versus DPDK vRouter and kernel vRouter

Corigine's Agilio vRouter software significantly increases network performance while reducing resource usage by offloading and accelerating the data plane and making use of SR-IOV and Corigine Express Virtio (XVIO) interfaces to effectively and efficiently handle network traffic. New connections are initially processed by the Contrail vRouter agent before being offloaded to the Agilio vRouter forwarding plane. This means that the setup rate performance with the Agilio hardware offload is practically identical to the setup rate of the software-based Contrail vRouter.

3.1 AGILIO VROUTER ARCHITECTURE

The Agilio SmartNICs and software will transparently receive configuration settings directly from the vRouter datapath hosted in the kernel. As a result, the vRouter functionality, configuration, and control channels are completely preserved. This transparent offload allows the Agilio solution to integrate with the Contrail controller and vRouter subsystems transparently.

Figure 5 depicts the vRouter data plane being offloaded to the Corigine Agilio CX Smart-NIC. The figure also shows SR-IOV and XVIO interfaces to the VMs. More detail regarding the architecture of these interfaces will be discussed in the next section.

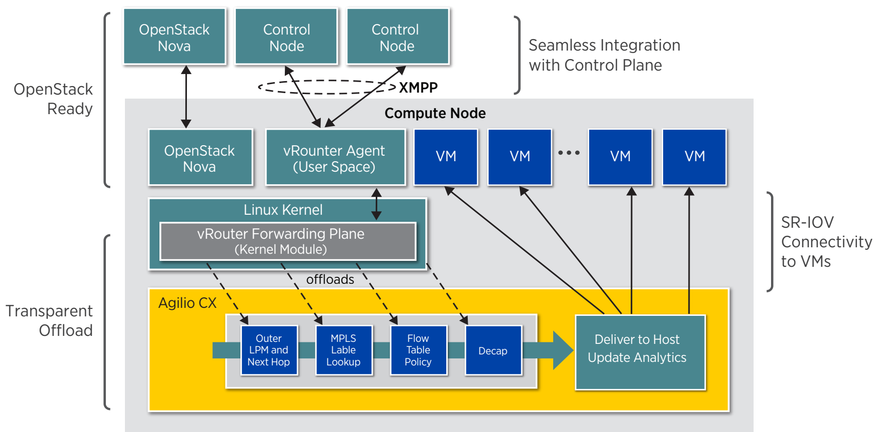

Figure 5: Agilio vRouter architectural overview

The common tunneling configurations that Agilio vRouter supports for L2 and L3 overlay networking include MPLS over GRE, MPLS over UDP and VXLAN. This is accomplished by offloading and accelerating lookups and actions for several Contrail vRouter tables including:

- Interface Tables
- Next Hop Tables
- Ingress Label Manager (ILM) Tables
- IPv4 FIB
- IPv6 FIB
- L2 Forwarding Tables
- Flow Tables

In the case of encapsulation, a VM or VNF would transmit a packet on a VF that is connected to the Agilio CX SmartNIC. The Agilio vRouter datapath receives the packet and executes the required lookups. In the case of an L3 overlay and network tunneling with MPLS over UDP (see figure 3), the inner Ethernet header is stripped and the MPLS over UDP headers are added for the tunnel. Finally, the underlay network headers (Ethernet and IP) are added based on MPLS label lookups and the final frame is transmitted on the network.

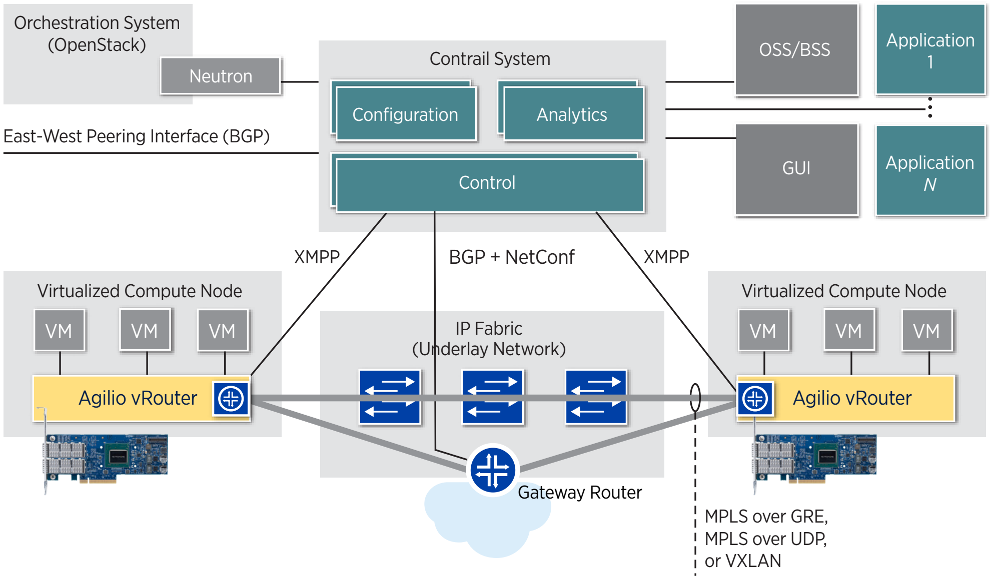

Figure 6: Contrail architecture with Agilio vRouter solution

3.2 AGILIO VROUTER SR-IOV AND XVIO INTERFACES

Agilio vRouter allows the VMs to connect to the offloaded data plane via either SR-IOV or Corigine's XVIO interface, which presents a standard virtio over vhost-user interface to the VM. The I/O bandwidth delivered through the datapath to VMs is affected by the datapath decisions made on the host or hypervisor and the I/O interface between the VM and the host or hypervisor (virtio, DPDK, SR-IOV).

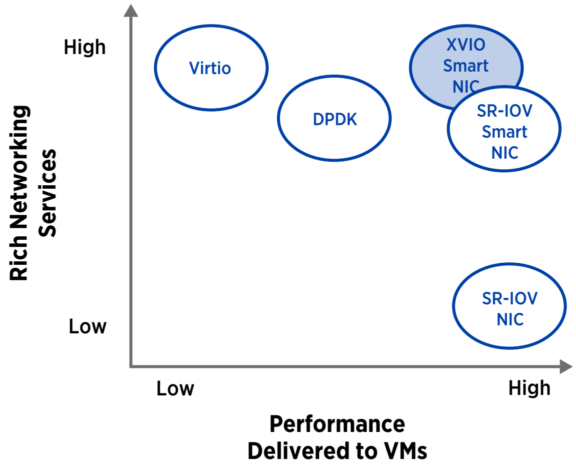

Figure 7: Services versus performance graph

The tests will provide evidence that a SmartNIC can maintain software defined flexibility while delivering performance over both virtio or SR-IOV. This is something that a basic NIC with SR-IOV cannot achieve.

In the figure below the SR-IOV passthrough scenario is depicted where the SmartNIC VF is made available to the VM as a hostdev.

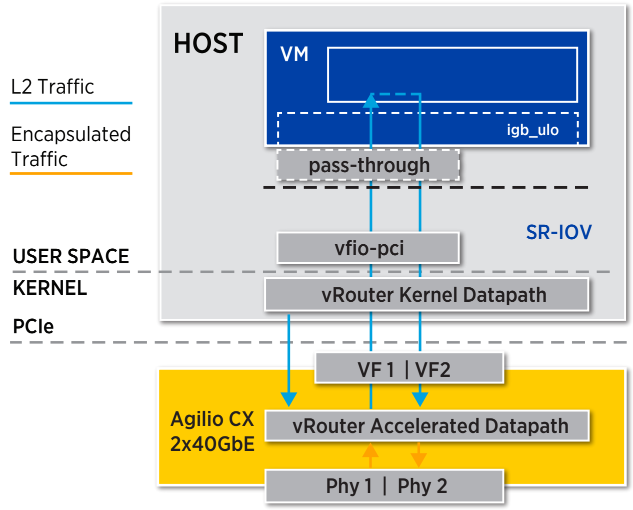

Figure 8: Agilio vRouter architectural overview of the SR-IOV passthrough interface

Figure 9 depicts the XVIO scenario where a DPDK virtio forwarding application using the Poll-Mode Driver (PMD) presents the VM with a generic virtio interface. Using virtio interfaces through XVIO enables live migration of guest VM instances which is something not possible when using SR-IOV alone.

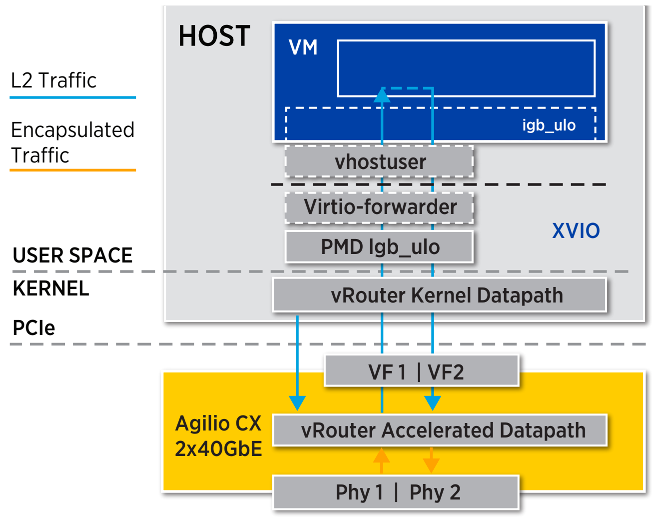

Figure 9: Agilio vRouter architectural overview of the XVIO interface

4. PERFORMANCE

4.1 TEST CASE 1: TWO-NODE VROUTER WITH IXIA CONFIGURATION

For this test case, a two-node Contrail vRouter was used with a 40GbE switch. The Ixia was then connected to the switch via a 40GbE link. Traffic was generated by the Ixia and directed to one of the compute node's VMs, where it was turned around by a DPDK application and sent back to the Ixia.

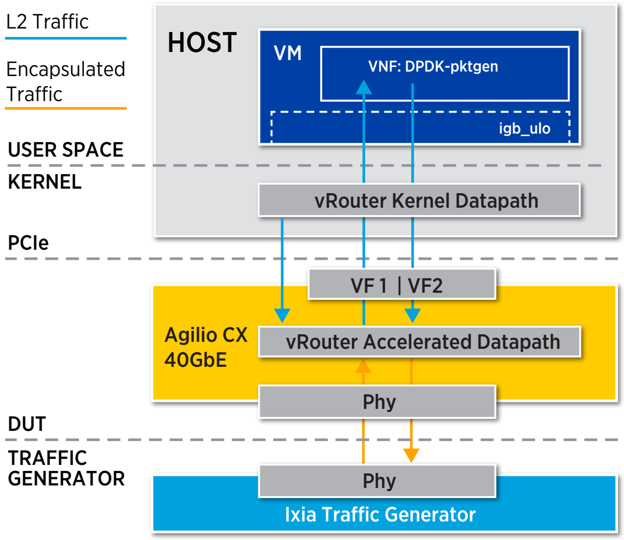

Figure 10: Single compute Ixia test-case configuration

A single PCIe SR-IOV VF was used as the VM network interface over which all traffic to the VM was directed. Table 1 contains the hardware and software specification of each compute node for the Ixia test.

Host	Network Devices	Hardware	Software
Controller and 2 Computes	**Agilio CX** 40GbE and **Intel XL710** 2x40GbE	**Motherboard** Supermicro X10DRL-LN4 **CPU** Intel(R) Xeon(R) CPU E5-2630 v4 20 CPUs @ 2.20GHz 2 NUMA nodes **Memory** 4x 16GB DIMM DDR4 @ 2400 MHz	**vRouter** Contrail 4.1.1 **OpenStack** Ocata **Linux** Ubuntu 16.04.2 (Kernel 4.4.0-62)

Table 1: Server specifications for the Ixia test-case

The performance figures in this section were captured by the Ixia device. These graphs document the performance increase achieved with Corigine's Agilio offload versus the unaccelerated kernel vRouter implementation and the DPDK vRouter implementation. The performance numbers shown are applicable to MPLSoGRE and VXLAN scenarios. Throughput measurements include the encapsulation in the bit-rate values.

4.2 TEST CASE 1: PERFORMANCE RESULTS

Figure 11 shows the frame rate performance of the Agilio CX SmartNIC with vRouter offload in comparison to the Intel XL710. MPLSoGRE encapsulation was used as the overlay encapsulation protocol. The rates include both the PHY-to-VM and the VM-to-PHY traffic and were measured with Ixia. At 92B packet size, Agilio CX delivers 21.38Mpps while the DPDK vRouter delivers 6Mpps using 4 CPU cores. This is a 3.5X performance delta.

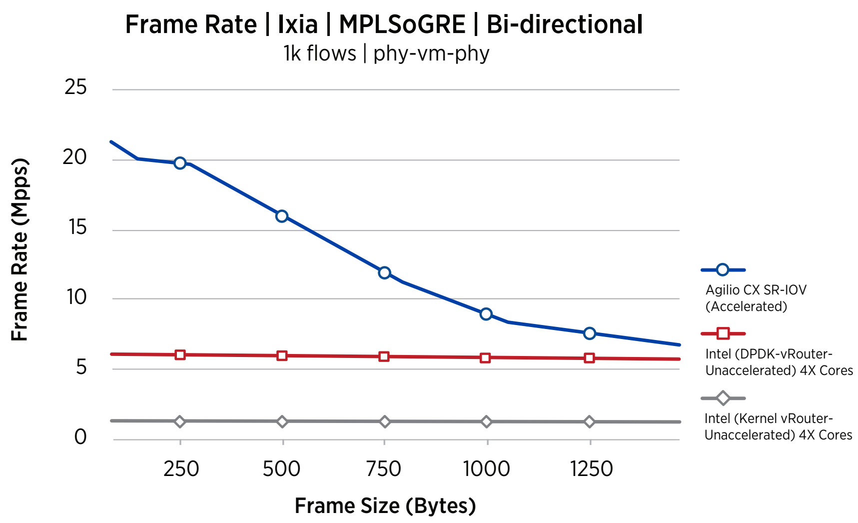

Figure 11: Accelerated vs. non-accelerated vRouter frame rate performance | SR-IOV | MPLSoGRE

Figure 12 shows the throughput (bit rate) performance of the Agilio CX SmartNIC with vRouter offload in comparison to the Intel XL710 when MPLSoGRE encapsulation is used as the overlay. The rates include both the PHY-to-VM and the VM-to-PHY traffic and were measured with Ixia. At 540B packet size, Agilio CX delivers 66.8Gb/s while Intel XL710 delivers 22.24Gb/s using 4 CPU cores.

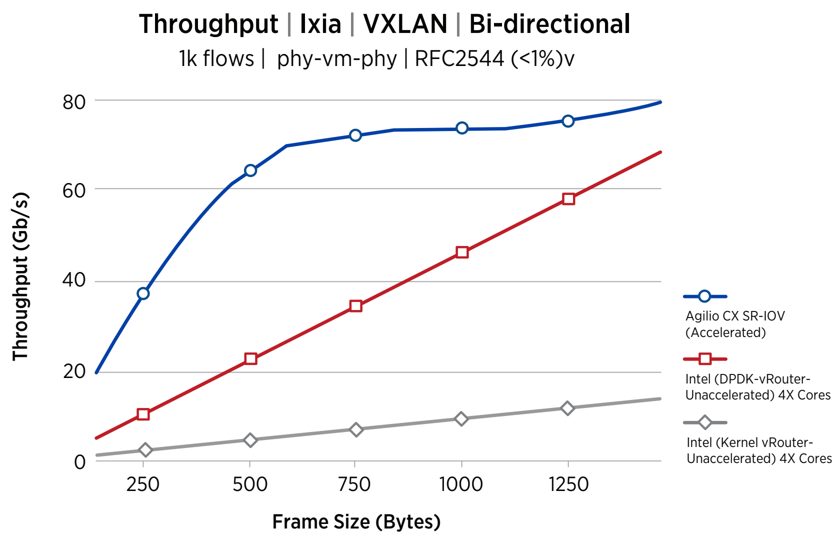

Figure 12: Accelerated vs. non-accelerated vRouter throughput performance | SR-IOV | MPLSoGRE

Figure 13 shows the frame rate performance of the Agilio SmartNIC with vRouter offload in comparison with Intel XL710. VXLAN encapsulation was used as the overlay encapsulation protocol. The rates include both the PHY-to-VM and the VM-to-PHY traffic and were measured with Ixia IxNetwork in RFC 2544 test mode ('<1%' loss). At 146B packet size, Agilio CX delivers 21.21Mpps while Intel XL710 delivers 6Mpps using 4 CPU cores.

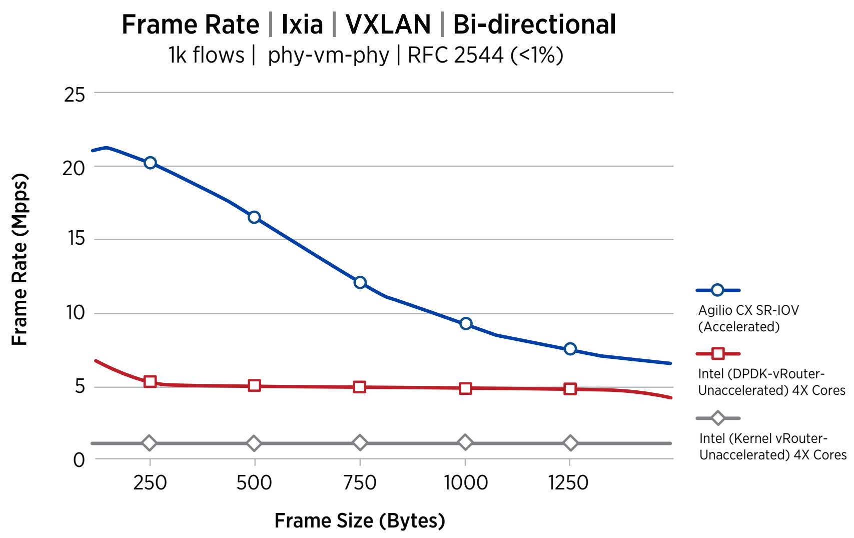

Figure 13: Accelerated vs. non-accelerated vRouter frame rate performance | SR-IOV | VxLAN

Figure 14 shows the throughput (bit rate) performance of the Agilio CX SmartNIC with vRouter offload in comparison to Intel XL710. VXLAN encapsulation was used as the overlay encapsulation protocol. The rates include both the PHY-to-VM and the VM-to-PHY traffic and were measured with Ixia IxNetwork in RFC 2544 test mode ('<1%' loss). At 562B packet size, Agilio CX delivers 69.27Gb/s while Intel XL710 delivers 22.68Gb/s using 4 CPU cores.

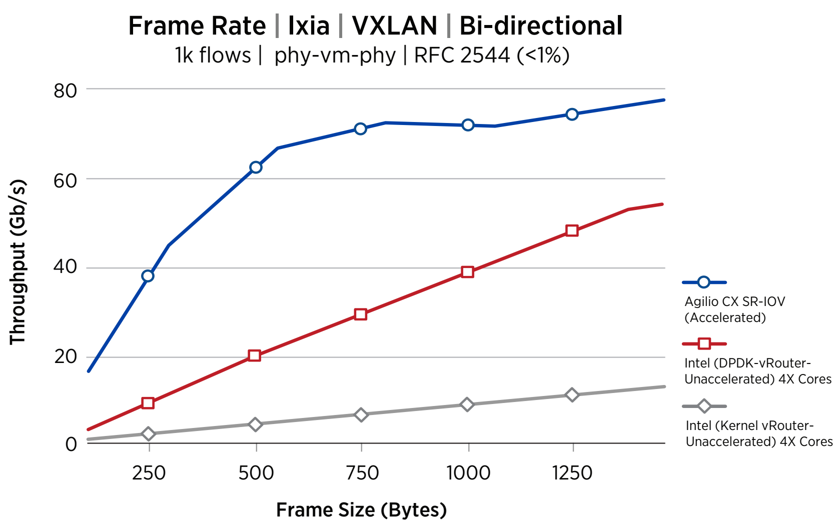

Figure 14: Accelerated vs. non-accelerated vRouter throughput performance | SR-IOV | VxLAN

Figure 15 shows the Round-Trip Latency between Ixia and a guest VM on the compute node. The full path of the traffic includes Ixia, 40GbE switch, Agilio CX SmartNIC (incl. VXLAN decapsulation), PCIe/SR-IOV to VM, DPDK application within VM rewriting IP and MAC fields, PCIe/SR-IOV from VM, Agilio CX SmartNIC (including VXLAN encapsulation), 40GbE switch, Ixia.

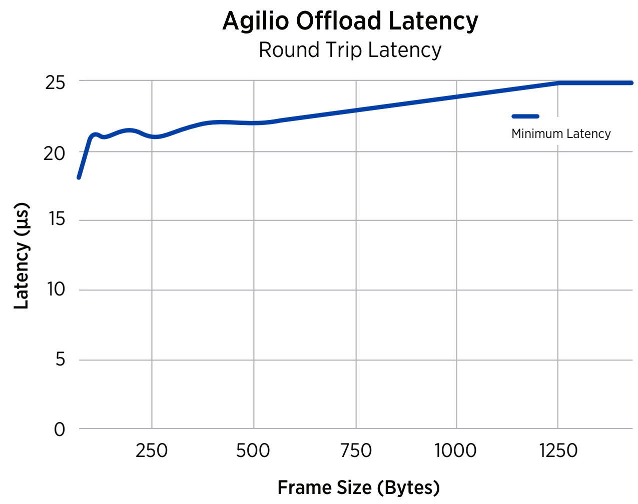

Figure 15: Accelerated vRouter latency for the Ixia test-case

4.3 TEST CASE 2: THREE-NODE BACK-TO-BACK CONFIGURATION

Three high-performance x86 servers were connected via a Cisco Nexus 3000 series switch. One server served as the controller while the remaining two were used as compute nodes. Multiple flows were created by varying L4 addresses and DPDK-pktgen was used to generate source and sync traffic between the two computes:

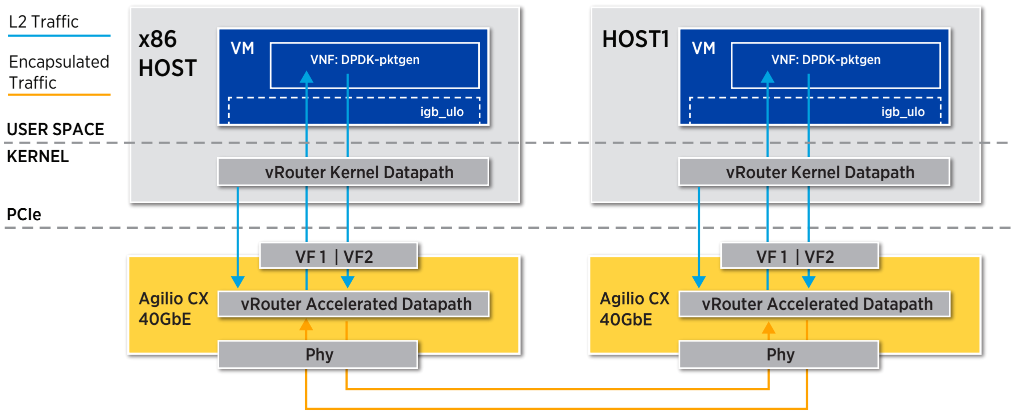

Figure 16: Back-to-back DPDK-pktgen test-case configuration

Guest machines were created on the respective compute nodes and connected to the vRouter data plane via SR-IOV and XVIO VF (Agilio vRouter) and vhostuser (kernel/dpdk-vRouter) interfaces as described in the previous section.

Table 2 contains the hardware and software specification of each compute node for the DPDK-pktgen test.

Host(s)	Network Devices	Hardware	Software
Controller and 2 Computes	**Agilio CX** 2x40GbE and **Intel XL710** 2x40GbE	**Motherboard** Supermicro X10DRi **CPU** Intel(R) Xeon(R) CPU E5-2630 v4 20 CPUs @ 2.20GHz 2 NUMA nodes **Memory** 4x 16GB DIMM DDR4 @ 2400 MHz	**vRouter** Contrail 4.1.1 **OpenStack** Ocata **Linux** Ubuntu 16.04.2 (Kernel 4.4.0-62)

Table 2 Server specifications for the back-to-back test-case

4.4 TEST CASE 2: PERFORMANCE RESULTS

For Figure 17 the traffic was exchanged between two VMs on opposite compute nodes. The rates include both PHY-to-VM and VM-to-PHY traffic, and were measured with dpdk-pktgen. Agilio XVIO and SR-IOV have similar performance and both deliver approximately 22Mpps at 64B packet size. Intel DPDK unaccelerated solution delivers 6Mpps at 64B packet size.

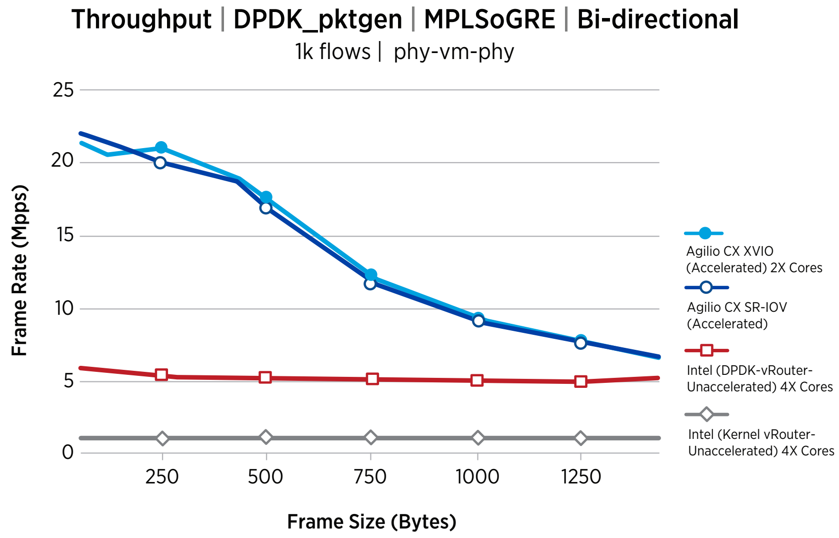

Figure 17: Accelerated vs. non-accelerated vRouter frame rate performance for the dpdk-pktgen test-case | MPLSoGRE

Figure 18 shows the throughput performance of the Agilio CX SmartNIC with vRouter offload in comparison to Intel XL710. The rates include both the PHY-to-VM and the VM-to-PHY traffic, and were measured with dpdk-pktgen. At 440B packet size, Agilio CX XVIO and SR-IOV deliver 71Gb/s while Intel XL710 DPDK unaccelerated vRouter delivers 19.73Gb/s and kernel unaccelerated vRouter delivers 4.31Gb/s.

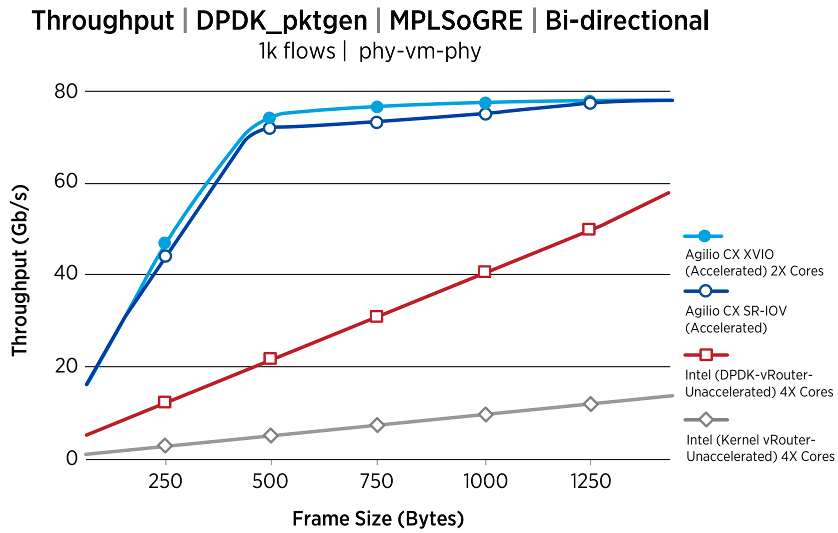

Figure 18: Accelerated vs. non-accelerated vRouter throughput performance for the dpdk-pktgen test-case | MPLSoGRE

Latency was measured between two VMs on separate compute nodes, and also between an Ixia system and a VM.

4.5 TEST CASE 3: VM-TO-VM ICMP PING LATENCY TEST

A standard ICMP Ping test was performed to determine the latency between the respective guest machines. Observe that these latencies include traversing the datapath four times. The Agilio solution delivers 6X lower latency.

Latency(us)	Minimum	Maximum	Average
Agilio CX XVIO	70	80	76
Agilio CX SR-IOV	80	110	84
Intel Kernel vRouter	450	550	487

Table 3: Solutions' latency overview

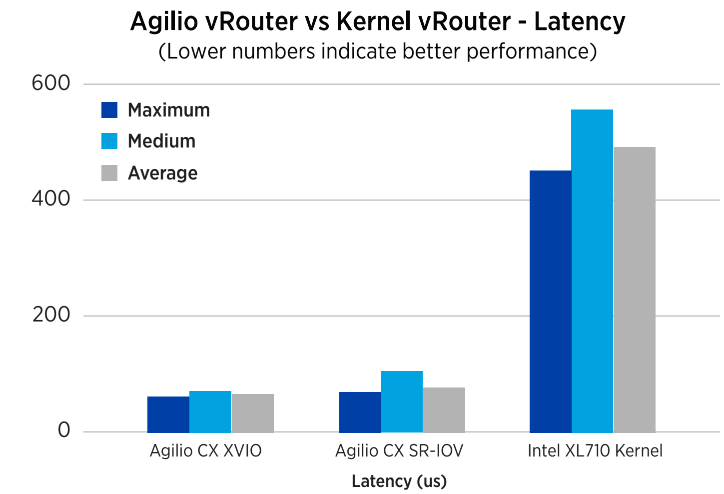

Figure 19: Accelerated vs. non-accelerated vRouter round-trip latency for ICMP pings between computes

4.6 RESOURCE USAGE

The output from the 'htop' application shows CPUs maxing out per kernel-vRouter interface actively handling traffic. Agilio removes this load from the host CPU by offloading network flow forwarding to the SmartNIC:

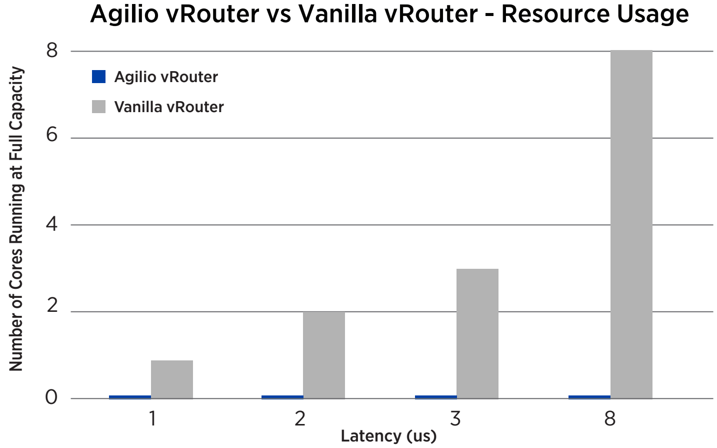

Figure 20: Accelerated vs. non-accelerated vRouter resource usage

This results in more processing power (vCPUs or cores) being available for the application running within a VM.

5. SUMMARY

The performance results presented undoubtedly show the advantage of using Agilio CX SmartNICs with Contrail vRouter offload compared to purely software-based kernel and DP-DK-based vRouter solutions. The offloading of the Contrail vRouter data plane with SR-IOV or XVIO interfacing to guest machines results in a significant performance increase while preserv-ing valuable CPU resources.

From a performance perspective, the Agilio SmartNIC solution allows operators to fully utilize compute platforms through offload due to:

1. Significantly increased PPS capacity of Contrail vRouter datapath allowing applications to reach their full processing capacity.
2. Reclaimed CPU cores from Contrail vRouter offload allowing more VMs and VNFs to be deployed per server.
3. Native networking I/O performance to VMs and VNFs through the use of SR-IOV.

Lastly, the functional benefit to Contrail users is that they can take advantage of the Agilio performance gains while retaining their pre-existing Contrail architecture. The control plane (XMPP) configuration for the Contrail vRouter data plane remains the same, and OpenStack management is unchanged. Network automation and configuration are not disrupted with Agilio, making the user experience identical to software-only deployments – the only difference is performance. Users will experience a significant gain in performance with Agilio-accelerated Contrail vRouter platforms.

Corigine

Email: sales@corigine.com

www.corigine.com.cn

WP-vRouter-Contrail-8/18