Ethernet Messaging Overview

Introduction

Gen-Z is a new data access technology that not only enhances memory and data storage solutions, but also provides a framework for both optimized and traditional messaging solutions. This document builds upon the features and capabilities described in the *Gen-Z Messaging Overview* document. Messaging via the transport of Ethernet packets over Gen-Z fabrics enables rapid adoption of Gen-Z technology with no changes to existing operating systems or protocol stacks, yet enables Gen-Z enabled compute components to reap many of the benefits offered by this technology when used for messaging.

Why Ethernet over Gen-Z?

The *Gen-Z DRAM and Storage-Class (SCM) Memory Overview* and *Gen-Z Messaging Overview* documents describe the enhanced features and capabilities of Gen-Z as a data access and a general purpose messaging technology, respectively. Figure 1 illustrates a simple infrastructure example that includes both a Gen-Z fabric for data access to fabric-attached memory and an existing network for messaging between servers. The memory module and the CPUs have integrated Gen-Z logic and interfaces for low-latency, high-bandwidth, resilient, and scalable byte-addressable data access to volatile or storage class (persistent) memory. But using a separate network for messaging has performance and cost challenges. First, there are two independent fabrics that each include switches, cabling, and management software. Networking often requires one or more NICs per server, and the mechanical packaging of the fabric components can be quite different. In addition, network packet latency between CPUs will be higher than the data access latencies because of the additional protocol translations at the PCIe® and network interfaces. Furthermore, the NICs are burdened with both heavy east-west as well as the more moderate north-south messaging traffic flows, requiring higher bandwidth per port. Finally, this configuration incurs higher cost, power consumption, and operational management overhead.

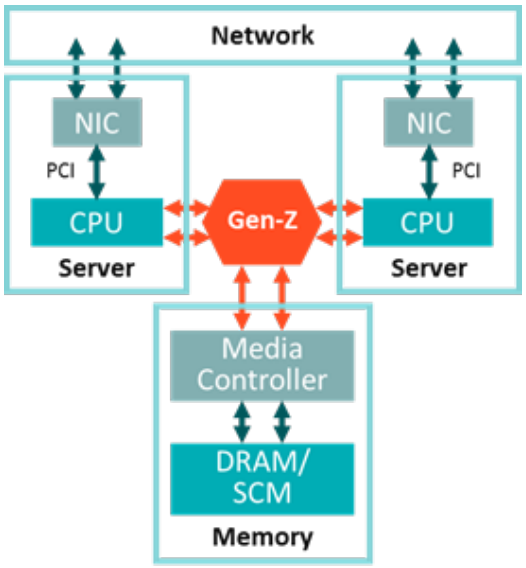

Figure 1: Gen-Z with Existing Networking

By leveraging Gen-Z for messaging as well as data access to memory/storage, platforms can be built to efficiently support legacy communication software stacks while providing numerous benefits over traditional networking approaches. There are two primary use cases for utilizing Ethernet over Gen-Z for messaging. Firstly, deployments that use existing software APIs and protocol stacks (e.g., Sockets API, TCP/IP, hypervisor-based virtual switches, etc.) will use Ethernet over Gen-Z to enable

unmodified operating systems and applications to transparently operate across Gen-Z. Secondly, most deployments will need to connect to Ethernet networks to be transparently integrated into the existing data center infrastructure.

Figure 2 demonstrates a solution that enables Ethernet packets to be transported over Gen-Z between CPUs and one or more Network Gateways. The east-west traffic flows between CPUs benefit from the high-bandwidth, low-latency, low-power Gen-Z protocol and do not burden external networks, reducing the bandwidth of the Ethernet connections to these external networks. Coupled with the fact that north-south traffic bandwidth between CPUs and the Network Gateways is moderate for many workloads, this architecture yields a reduction in the number of Network Gateway uplink ports and the associated CAPEX and OPEX costs. This architecture also provides the benefit of flexible network connectivity options when using Gen-Z. The amount of north-south traffic flows required by targeted workloads can be dynamically managed by adding and removing Network Gateway components of various sizes and bandwidth capacity. Just like fabric-attached memory resources, network connectivity is no longer dependent on the number and types of CPUs deployed in the fabric.

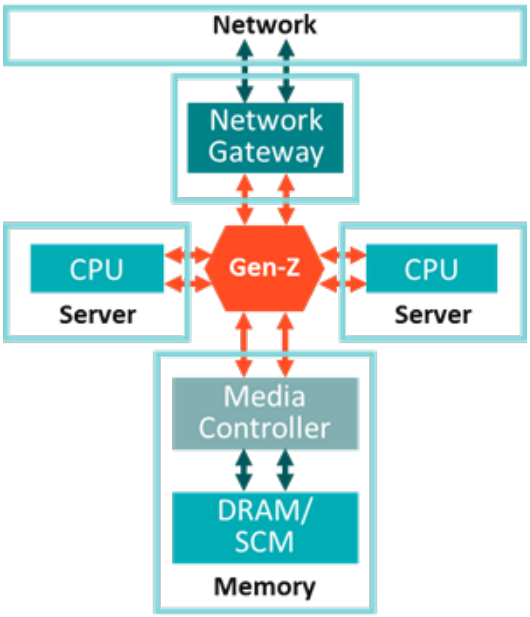

Figure 2: Gen-Z for Messaging & Data Access

To summarize, utilizing Ethernet over Gen-Z messaging offers the following benefits over alternative approaches:

- Supports unmodified operating systems and protocol stacks (e.g., TCP/IP)
- Lower Latency
- Scalable bandwidth
- Reduced power consumption
- Integrated management framework
- Simple, optimized, low-cost system designs
- Reduced capital and operating costs

Ethernet over Gen-Z Implementation

Enabling Ethernet over Gen-Z entails creating an emulated Ethernet NIC (eNIC) driver to support existing network stacks, and defining the packet protocol for transporting (a.k.a. tunneling) Ethernet packets using the Gen-Z protocol. An eNIC driver is placed in the system software exactly as a traditional Ethernet driver. The eNIC driver implements the Ethernet over Gen-Z tunneling protocol such that the packet format presented to upper layer protocols looks exactly like Ethernet with no protocol

conversions necessary, maintaining absolute compatibility. The eNIC driver takes full advantage of Gen-Z hardware mechanisms to facilitate efficient, resilient, high performance communication between Gen-Z components. By using an Ethernet over Gen-Z tunneling protocol, Network Gateways can be implemented in a lightweight manner with minimal protocol conversions or translations between the physical Ethernet uplink ports and the eNICs operating in Gen-Z components. This yields low latency, high performance communication between Gen-Z components and native Ethernet infrastructure external to the Gen-Z fabric.

Conceptually, Ethernet over Gen-Z operates in a similar manner to a hypervisor's software emulated vNIC and vSwitch model. Figure 3 shows a typical virtualized server environment that includes virtual machines with virtual NICs (vNICs) and a virtual Switch sharing a pair of physical NICs. Packet movement between vNICs is merely a series of memory-semantic data copies between the various VM socket buffers and the internal vSwitch buffers. The software emulated nature of the vNIC allows it to adapt quickly to new IP and Ethernet protocols as they emerge.

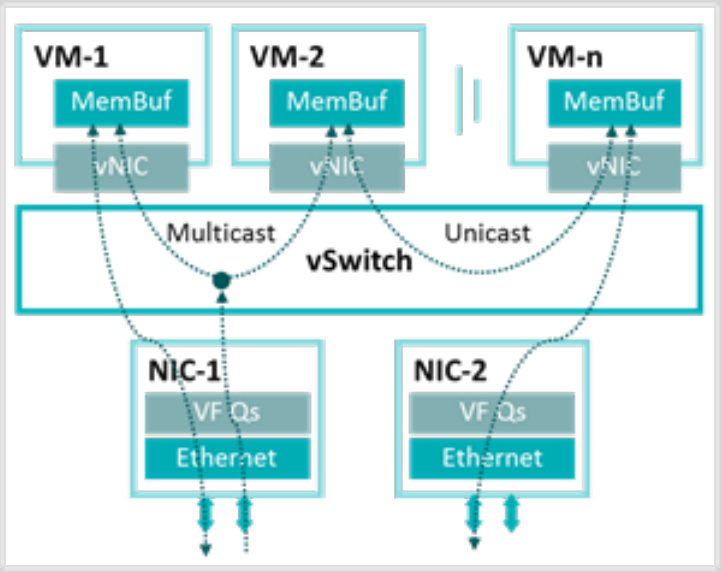

Figure 3: Existing vNIC & vSwitch Model

Ethernet over Gen-Z expands on these emulated virtual NIC/switch concepts such that Gen-Z interconnect enables memory semantic data copies across a wide range of computing systems spanning an enclosure, a rack, and beyond. In addition. Gen-Z facilitates both CPU load/store as well as hardware-assisted message data movers to provide low-latency small messaging or high-bandwidth messaging, respectively. Thus, Gen-Z enables software-defined networking concepts to extend beyond virtualized servers and be applied to physical infrastructure. Figure 4 depicts a Gen-Z environment that includes several Gen-Z enabled SoCs connected to each other and a pair of Network Gateway devices over an arbitrary Gen-Z fabric. Within each SoC, one or more eNIC drivers transmits and receives Ethernet packets using hardware-assisted messaging. To the operating system (OS) network stack, the eNIC driver behaves like a high performance Ethernet Adapter and requires no modification to the OS. The eNIC drivers and Fabric Management Services support all modern Internet and Transport Layer protocols needed by OS network stacks. Some eNIC drivers may also support advanced services like the Data Plane Development Kit (DPDK) for Network Function Virtualization (NFV) workloads that benefit from Gen-Z's low-latency, high bandwidth support.

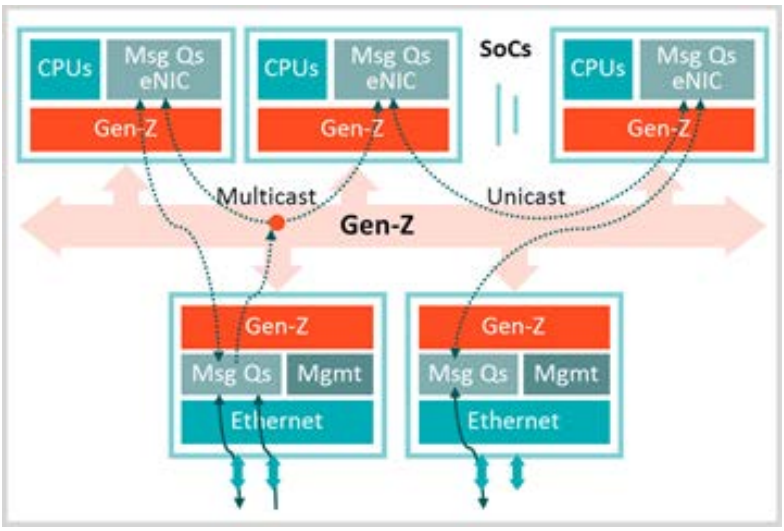

Figure 4: Gen-Z eNICs expand on software-based vNIC & vSwitch Concepts

Network Gateways

Specialized components called "Network Gateways" provide connectivity to external networks. These Network Gateways may be software entities on Gen-Z enabled SoCs that implement Gen-Z over Ethernet messaging using internal or external NICs for Ethernet uplink connectivity. An alternative solution is to implement Network Gateways using hardware based devices (e.g. ASICs or FPGAs) that are designed for specific gateway and/or functionality (e.g. protocol acceleration, etc.). A Network Gateway acts as either a simple layer 2+ bridge or a layer 3+ router between Ethernet and Gen-Z to provide SoCs with connectivity to external Ethernet networks. For small configurations, a simple L2+ Network Gateway would act like a shared NIC device with 2-4 external Ethernet ports. The Gen-Z side would leverage the eNIC messaging queues and data movers and support any number of Gen-Z processor components depending on the bandwidth needs and gateway component design. For configurations with relatively large numbers of SoCs, the Network Gateway could be built as a port expander box or option card for an existing network Switch/Router, allowing the customers to scale Gateway bandwidth as needed.

Network Virtualization

The eNIC drivers and Network Gateways also support virtualized environments and overlay networks protocols like VXLAN and GENEVE. Figure 5 demonstrates that eNIC drivers can be instantiated either in the kernel or as a vNIC on a VM. Figure 5-A depicts a standard bare metal server with one or more kernel level eNIC drivers supporting multiple applications. Figure 5-B depicts a virtualized server where a kernel level eNIC driver operates in the hypervisor and generic virtual NICs in the VMs are supported through a vSwitch. Figure 5-C depicts a virtualized server which has an eNIC for every VM, allowing Ethernet traffic to bypass the hypervisor and transfer more directly to/from the Gen-Z fabric. Because eNICs are software-defined with direct access to Gen-Z hardware, no kernel level software components are necessary for high performance data path communications.

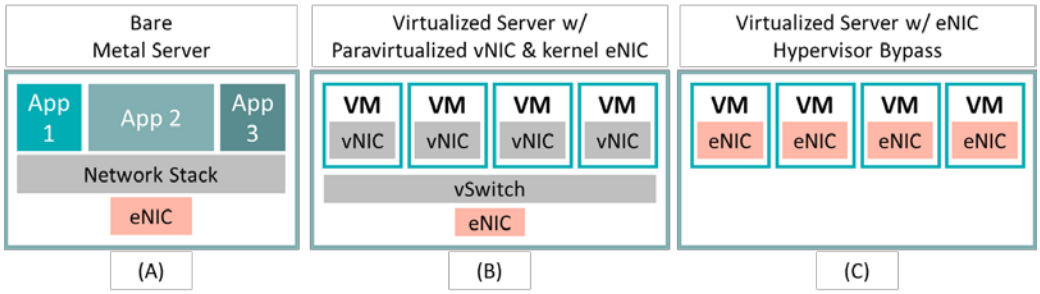

Figure 5: eNICs in Virtual and non-Virtualized Environments

Enhanced Multipath for Messaging

All Gen-Z components can support multiple link interfaces which increase aggregate bandwidth, solution resiliency, and topology flexibility. This enables Gen-Z solutions to deliver optimal performance and availability without sacrificing performance. The Gen-Z logic in each component automatically isolates and load-balances traffic to meet application performance needs. In the event of switch or link failures, Gen-Z logic transparently rebalances traffic to the remaining links and switches to minimize application performance impacts. Finally, both virtual or kernel level eNIC drivers, via data movers used for Gen-Z hardware-assisted messaging, utilize multiple Gen-Z interfaces and paths through the Gen-Z fabric, thus taking advantage of Gen-Z's automated resiliency and load balancing such that teaming and bonding drivers are unnecessary in Gen-Z deployments.

Message Models

Gen-Z supports both software and hardware-assisted messaging models as described in the *Gen-Z Messaging Overview* document. Ethernet over Gen-Z utilizes the hardware-assisted model to provide traditional networking to user and kernel applications and services. Hardware-assisted messaging uses lightweight, component-integrated data movers to move data with minimum software involvement. Data movers can use Gen-Z write message operations to move up to 64 KB of data that targets destination-managed memory. In addition to these Gen-Z operations, data movers can support queue-based programming models. This model of messaging supports all of the flexible topologies and advanced switching capabilities identified in the *Gen-Z Messaging Overview* document. By harnessing these capabilities and operations in the eNIC drivers, all existing network stacks can be supported.

Security

Gen-Z offers a combination of hardware-enforced isolation techniques and full packet authentication to prevent errant or malicious component software or hardware from communicating with unauthorized components or accessing unauthorized resources, including Gen-Z message queue memory. Gen-Z Fabric Management configures communications between SoCs and Network Gateways on both a component and eNIC basis to enforce network isolation and protection. This allows each eNIC to establish a separate memory queue pair with a Gateway, in effect a private tunnel over Gen-Z, because the queue pair's Gen-Z addresses are not in the mapping tables for any other Gen-Z device or software endpoint. Gen-Z multicast mechanisms similarly restricts which SoC or Gateway components belong to various multicast groups.

Summary

Ethernet over Gen-Z messaging is designed to facilitate existing network protocols and applications while enabling capabilities that will be difficult or impossible to achieve with existing network technology. These capabilities include:

- Higher performance with lower capital and operation costs
  - Lower latencies, scalable bandwidth, lower power, simpler management
- Software-based Emulated Ethernet architecture supports unmodified operating systems and protocol stacks
  - Supports traditional kernel eNIC drivers as well a virtual eNIC driver that support hypervisor bypass
- Network Gateways that can scale in size and performance to match required north-south traffic flows
  - Decouples Ethernet network connectivity from the number and size of compute component deployed
- Multipath, load balancing, and resiliency features that are native in Gen-Z hardware
  - Supports a variety of fabric topologies for optimizing to specific workloads
  - Multi-pathing software or redundant drivers are not required
- Leverages Gen-Z's flexible topology support and advanced switching technology
  - Reduces latency jitter and enables congestion avoidance
- Gen-Z offers security features that protect and isolate an infrastructure's messaging resources

DISCLAIMER

This document is provided 'as is' with no warranties whatsoever, including any warranty of merchantability, noninfringement, fitness for any particular purpose, or any warranty otherwise arising out of any proposal, specification, or sample. Gen-Z Consortium disclaims all liability for infringement of proprietary rights, relating to use of information in this document. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted herein.

Gen-Z is a trademark or registered trademark of the Gen-Z Consortium.

All other product names are trademarks, registered trademarks, or servicemarks of their respective owners.