CXL3 Fabric –
Introduction and use cases

Tony Brewer, Micron & Nathan Kalyanasundharam, AMD
• Motivation & Use Cases
• Why CXL 3.0 Fabric
• Fabric Basics and Scope
• Port Based Routing
• GFAM Device Transaction Flows
• Host Address Map
• Address Interleaving
• Device Media Partitions
• GFAM Device Media Access Protection
• Summary
• Explosive data growth
• Data stored and analyzed continuously
• Data volume and velocity demand large clusters for timely analysis
• Significant power to move and copy data
• There is a need for an efficient low latency and high bandwidth solution which minimizes data movement
CXL3 Fabric – Use Case Topology

Tightly coupled cluster

Local Scope
- Dram on the Motherboard
- <100 ns

System Scope
- Local CXL mem within the Chassis
- <200ns

Rack Scope
- Disaggregated global memory within the Rack
- <600 ns

Global Shared Memory
CXL3 Fabric – Memory Centric Computing

- HPC/AI/HPDA workloads consume significant amount of power
- Optimize compute to happen near the data
- Enable near and in-memory compute

![CXL Fabric Diagram](image-url)
Why CXL3 Fabric

- CXL 2.0 mechanisms scale to 16 hosts (MLD) per CXL memory device targeting pooling and limited sharing

- CXL 3.0 fabric addresses the large, scalable system space. 100’s of hosts, 1000’s of memory devices

- CXL 3.0 extends capabilities and introduces scalable mechanisms to support rack and pod scale systems
  - Expanded switching topologies
  - Enhanced coherency capabilities
  - Globally shared fabric attached memory
  - Peer-to-peer resource sharing

- CXL 3.0 defines the foundation for CXL fabric-based systems, future ECNs and specification releases are planned to standardize additional aspects
Port Identifiers
- Source Port (SPID)
- Destination Port (DPID)
- 4096 edge ports (12-bits)

Each edge port assigned a unique port identifier

G-FAM devices (GFD)
- Scalable memory resource
- Accessible by all host and devices in the cluster

Load/store memory semantics
CXL 3.0 augments the previously defined Hierarchy Based Routing with Port Based Routing.

Limited compatibility is defined for fabrics with a combination of Hierarchy and Port Based Routing capable CXL switches.

Acronyms
- HBR = Hierarchy Based Routing
- PBR = Port Based Routing
- SLD = Single logical device
- MLD = Multi logical device
- GFD = Global Fabric Attached Memory Device
• Previous revisions of CXL required an upstream port and a separate downstream port

• An Inter-Switch Link has been defined for CXL 3.0 allowing a single CXL switch port to act as both an upstream and downstream port

• Allows fewer ports used for inter-switch connections and better bandwidth utilization across switch ports
• Diagram shows a simple mesh topology with all CPU-to-device accesses being 1 or 2 switch hops

• Limiting the switching hops to at most two will help minimizing FAM access latency

• Eight CPUs are shown in the diagram, but with high radius CXL switches, much larger configurations are possible and still achieve at most two switch hops
• GFD is accessed using CXL.mem or CXL.io UIO

• Master to Subordinate Request
  • Switch Ingress Edge Port
    • Decodes address to identify target
    • Adds 12-bit SPID and DPID
    • Generate PBR Flit
  • Switch Egress Edge Port
    • Forward PBR Flit to the device
  • Device Ingress
    • Normalize address and access protection checks

• Subordinate to Master Response
  • Device Egress
    • Format response packet (native PBR)
    • Use request SPID as response DPID
  • Switch Egress Edge Port
    • Drop DPID and return response to the host
• CXL 2.0 supported Host / Accelerator data sharing with limited additional functionality

• CXL 3.0 Back Invalidate enables inclusive snoop filters and peer-to-peer memory requests
• Each host system has private local memory
  • Dram attached to local sockets and
  • Memory expansion over CXL
  • Memory typically owned by OS or Hypervisor running on the system

• New Fabric address space within the host physical address (HPA) space
  • Contiguous address range
  • Each host may map a subset or the entire global shared memory space
  • Address space typically owned by a central resource manager

• Fabric address space divided into ‘N’ segments
  • Host is not required to be aware of segments
- Source ID (SPID) identifies the requestor.
- Destination ID (DPID) identifies the end point of request.
- SPID and DPID added to request by switch ingress port.
- Switch uses DPID to route requests and responses to their destination.
- Fast Address Segment Table (FAST) and Interleave DPID Table (IDT) are setup by secure firmware running on host or a fabric manager.
Host may interleave on any interleave granularity

Switch edge port will decode and interleave requests to the target GFD port

Device decodes and normalizes from HPA -> DPA

Device maintains knowledge of the number of GFDs in the interleaved set
• Four independent switch planes

• Host interleaves traffic across four host links

• Each switch further interleaves across four devices

• Device maintains knowledge of interleaved set of 16

• Host interleave in this configuration must use bits within the GFD interleave granularity
• G-FAM device (GFD) may support multiple device media partitions (DMPs)
  • Fundamental attribute of DMP is the media type

• G-FAM device has up to eight decoders per host (SPID). The decoders map
  portions of HPA to device media partitions.

• CXL 3 supports unique per host physical address ranges to be mapped to each
  G-FAM device. However, the device physical address is common for all hosts.
• SPID is used to access the GFD decoder table

• Request HPA compared against all decoders associated with the requester

• DPABaseN added to the offset to derive final DPA

• Zero or multi-range hit results in access error
Requestor’s translation provides:

- **1st level of access protection and process separation**
- VA → Guest PA → Host PA

Switch request ingress port provides:

- **2nd level of access protection**
- FAST and IDT must be programmed to access the target GFD

G-FAM device provides:

- **3rd level of access protection**
- Device implements Source ID (SPID) based structures to enable access to DC Regions within device media
• GFD may have multiple DMPs with each DMP divided into fixed size blocks, and each block is assigned a memory group identifier.

• SPID access table (SAT) identifies the memory groups a SPID (host) is allowed access.

• The memory group table (MGT) identifies the memory group for each DMP block.

• MGT and SAT structures are programmed by Fabric Manager.
• Storage and compute requirements continue to grow
• CXL3.0 enables a scalable rack scale memory fabric
• Starting the journey towards realizable memory centric computing
• Large opportunities lies ahead for standards-based rack scale compute and chiplet ecosystem (UCIE)

• Get Involved!