Linux Plumbers Conference | Richmond, VA | Nov. 13-15, 2023

CXL Emulation in QEMU: Progress, Status and most importantly… What's next?

Jonathan Cameron - Huawei, Fan Ni - Samsung

Agenda

What landed since LPC 2022

Major topics in flight.

Discussion: What next?

Jump in at any time with questions!

Usual warning - we will only discuss published specifications.

Status

Pre LPC 2022

Basic enablement!

- CFMWS, Root Bridge, Root Port, Switch USP/DSP, Type 3

Today (so landed upstream in last year)

- Volatile and mixed Type 3 devices
- Multiple HDM decoders everywhere (more complex setups)
- CDAT (plus PCI DOE) - discoverable performance characteristics
- RAS error injection (event records)
- Poison injection

Under review / RFCs posted

- Dynamic Capacity Devices*
- CCI rework / Fabric Management Features*
  - Switch CCI
  - FM-API over MCTP over I2C
- Scan media

In staging tree, but not actively developed

- ARM support
- Performance Monitors

Posted but no plan to upstream (yet)

- Niagara MHD support
- Type 2 device support.

* More details follow!

Dynamic Capacity Devices

Prior to the introduction of DCD, adding or releasing memory capacity is very disruptive

- The host needs to reprogramming the HDM decoders
- Outstanding traffic must be quiesced
- System reset is needed

DCD is a memory device implementing dynamic capacity allowing memory capacity changes dynamically without reprogramming the HDM decoders

- Presenting its maximum capacity to each host
- HDM decoders are programmed for the entire DPA range
- DCD command set is implemented to control actual memory allocation/deallocation
  - Through DC extents

How we emulate DCD in Qemu

- Augmenting type3 memory device with Dynamic Capacity
  - 1-8 DC Regions
  - Extent list representing extents accepted by the host
  - Read/Write to the the DC Region
- Mailbox command support
  - Get Dynamic Capacity Configuration (4800h)
  - Get Dynamic Capacity Extent List (4801h)
  - Add Dynamic Capacity Response (4802h)
  - Release Dynamic Capacity (4803h)
- Using QMP interface to initiate DC extent add/release request.
  - FM is not implemented yet in Qemu*

What we miss now for DCD related?

- Only add dynamic capacity capability to type 3 device
  - No Multiple headed device for DCD
  - No LD-FAM, GFD DCD
- DC region is set to be non-volatile only
- No shared extents
  - A device is only used by a single host
  - Tag is not used
- Generation number is not really used
- Add/release capacity is prescriptive
  - Extent list based
- DCD Management Command Set not implemented

Issue of CXL Spec r3.0 for DCD

Issue 1:

- FM can initiate to add **multiple** extents in one request (5604h)
  - Table 7-62: Initiate Dynamic Capacity Add Request Payload
  - *"The processing of the actions initiated in response to this command **may or may not result in a new entry** in the Dynamic Capacity Event Log."*
- However, each Dynamic Capacity Event Record can hold **only one extent** (8.2.9.2.1.5 Dynamic Capacity Event Record)

Issue 2:

- The host responses a DC add event with exact one **Add Dynamic Capacity Response (4802h)**
  - The response holds an extent list
- Extents accepted by the host can be a **subset** of what the device offers for a DC Extent Add Request

Fabric Management

What is Fabric Management?

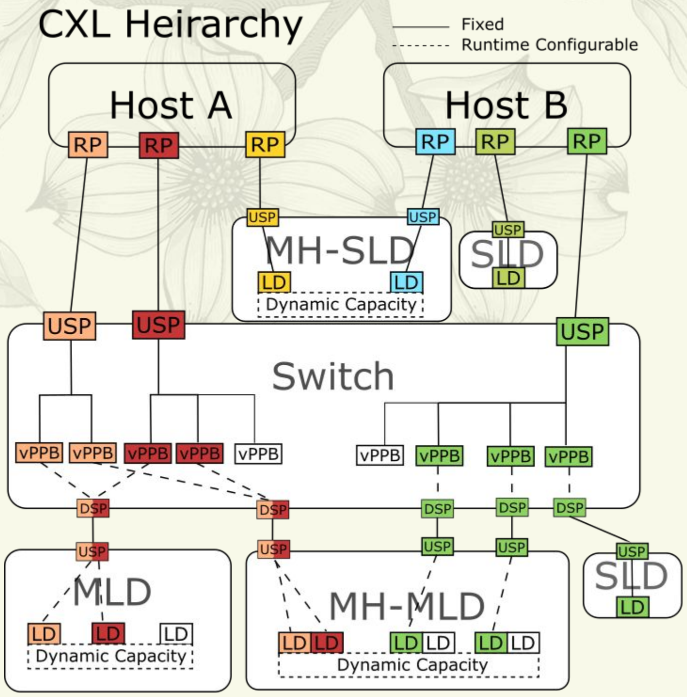
CXL Virtual Hierarchy fabrics (PCIe like ones) enable dynamic reconfiguration.

- Configurable Switches / Multi Logical Devices
- Dynamic Capacity (MLD, MH-SLD, MH-MLD)

Why emulate it?

- Test bench for Fabric Managers (?)
- CXL standards prove out.
- Standard interfaces to drive host tests (CI!)
- Some interfaces may be exposed to hosts

Note we aren't talking about large scale CXL fabrics (r3.0+)



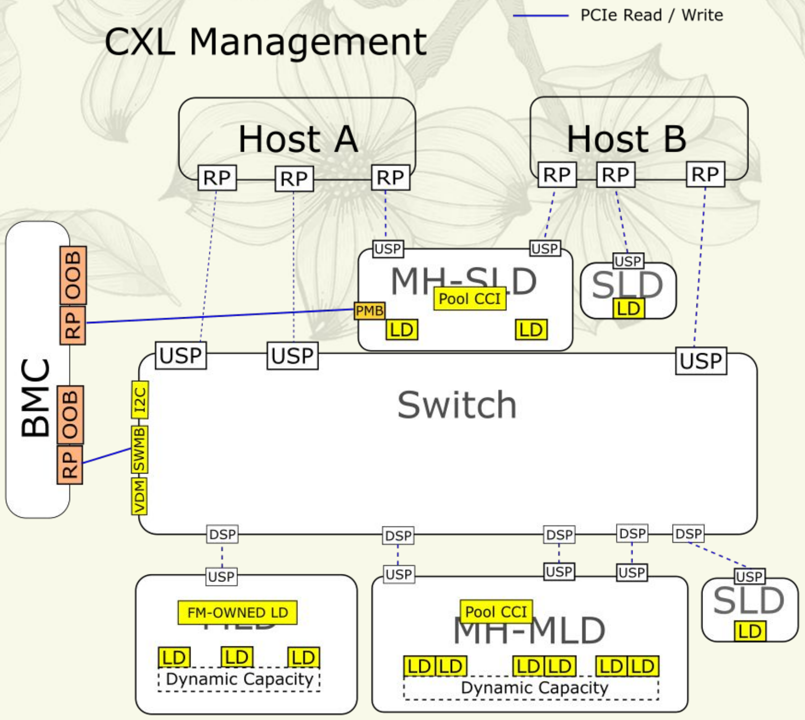
Dashed Lines == Configurable

Fabric Management

What are control paths? In band PCIe

- (Primary Mailbox)

- Switch CCI
  - Configure switch



Leverage existing in band mailbox in new ways

Fabric Management

What are control paths? + OoB MCTP (e.g. I2C)

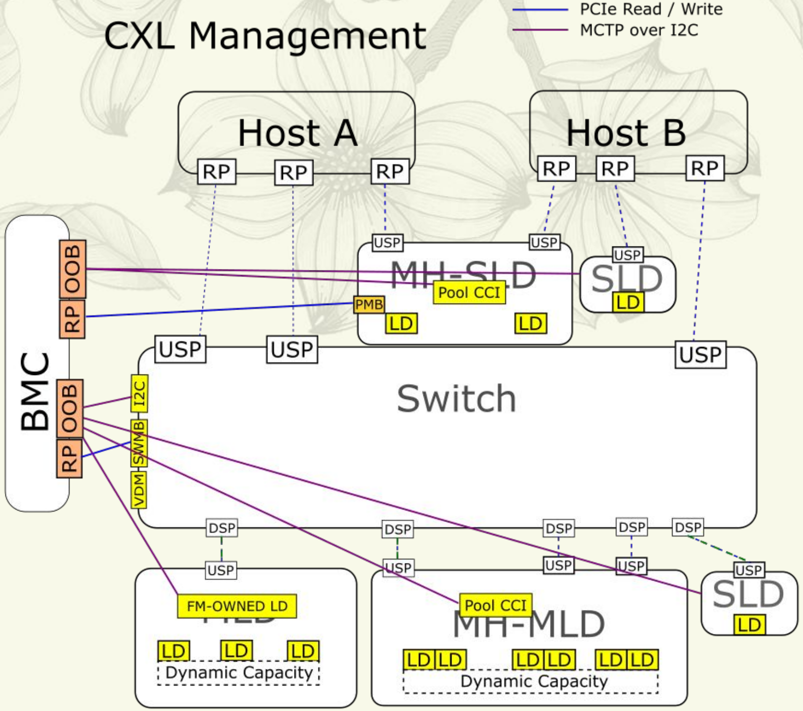
- (Primary Mailbox)

- Switch CCI
  - Configure switch

- **MCTP to FM owned CCI in MLD**
  - **Configure LD allocations**
  - **Configure DCD**

- **MHD Pool CCI**

- **Out of Band MCTP to pretty much anywhere!**

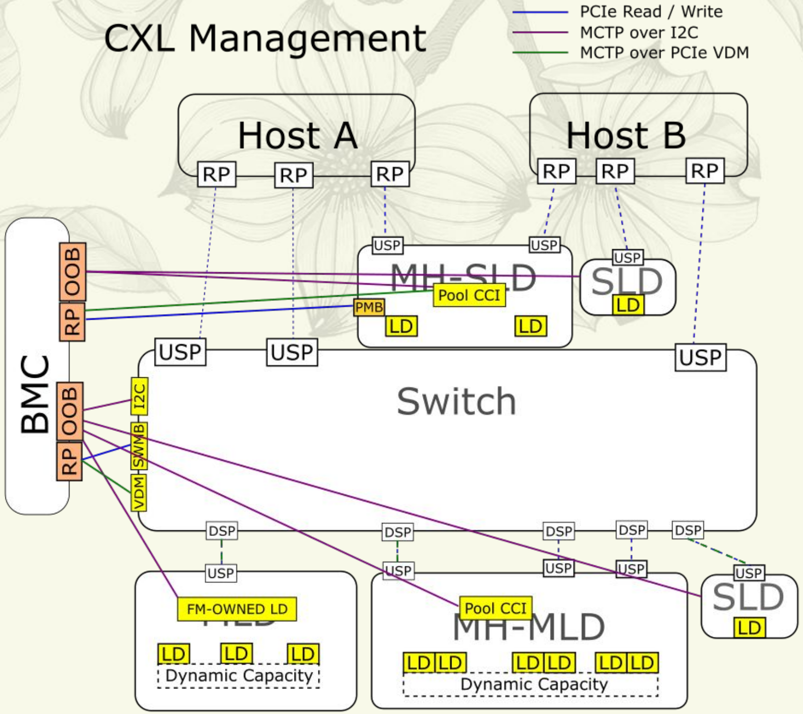


Direct control channels to devices

Fabric Management

What are control paths? MCTP over PCIe VDM

- (Primary Mailbox)
- Switch CCI
  - Configure switch
- MCTP to FM owned CCI in MLD
  - Configure LD allocations
  - Configure DCD
- MHD Pool CCI
- Out of Band MCTP to pretty much anywhere!

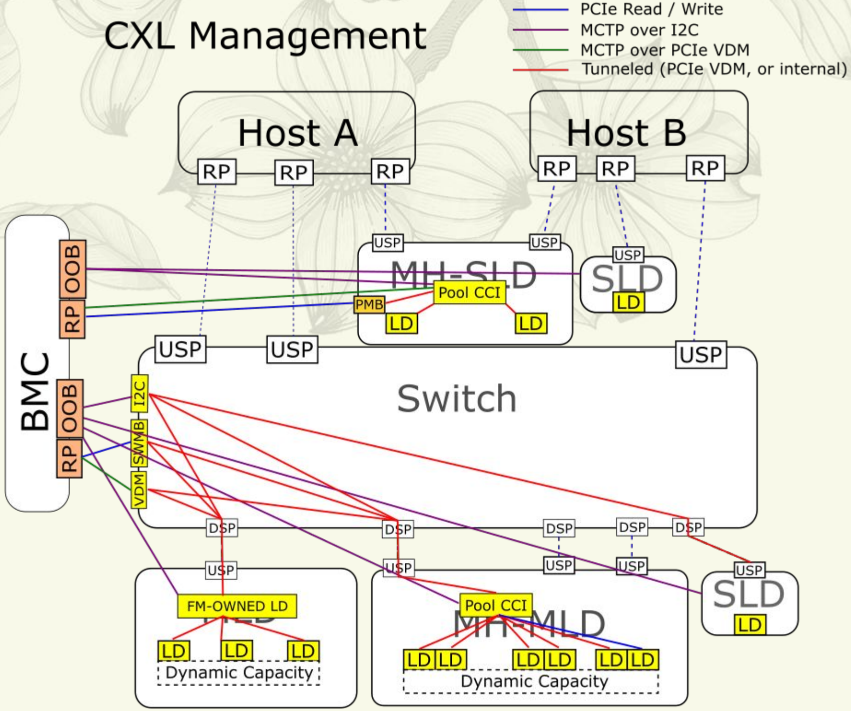


Nothing new (yet!)

Fabric Management

What are control paths? Tunneling!

- (Primary Mailbox)
  - **MH Pool CCI accessed via Tunnel**
- Switch CCI
  - Configure switch
  - **Tunnel via PCIe VDM to downstream devices**
- MCTP to FM owned CCI in MLD
  - Configure LD allocations
  - Configure DCD
  - **Tunnel to each LD within MLD.**
- MHD Pool CCI
  - **Tunnel to each LD within MHD.**
- Out of Band MCTP to pretty much anywhere!
  - Do everything!

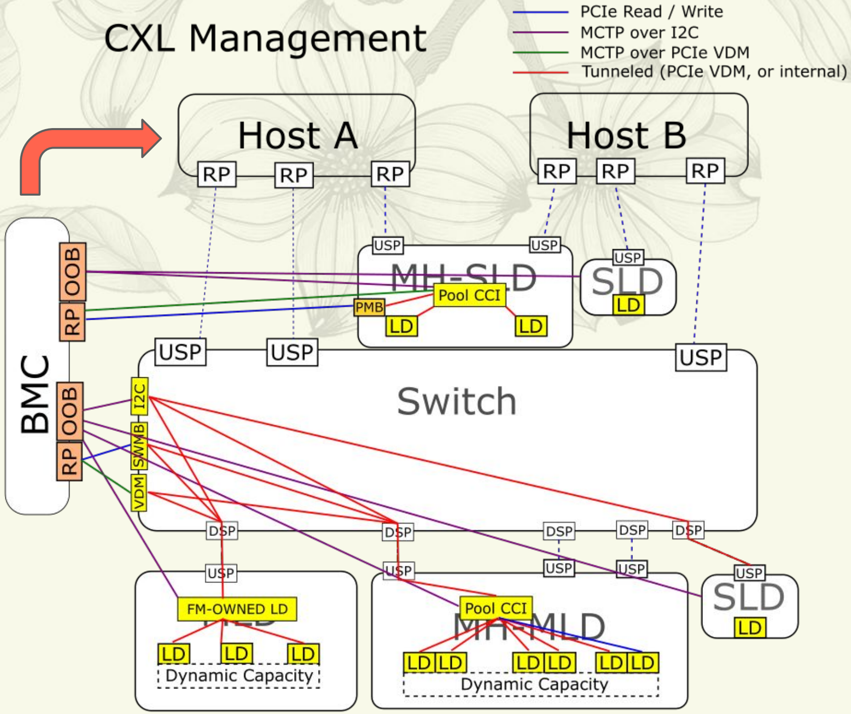


Tunneling between devices is over MCTP over PCIe VDM

Fabric Management

What are control paths? Tunneling!

- (Primary Mailbox)
  - MH Pool CCI accessed via Tunnel
- Switch CCI
  - Configure switch
  - Tunnel via PCIe VDM to downstream devices
- MCTP to FM owned CCI in MLD
  - Configure LD allocations
  - Configure DCD
  - Tunnel to each LD within MLD.
- MHD Pool CCI
  - Tunnel to each LD within MHD.
- Out of Band MCTP to pretty much anywhere!
  - Do everything!



For QEMU, HOST A == HOST B == BMC!

What's next?

What do people care about? (Set priorities)

- Dynamic Capacity Devices
  - Shared DCD regions?
  - Multi Host DCD (multiple instances of QEMU?)
- Multi Head Devices?
- Fabric Management
  - LD assignment, vPPB assignment
  - OoB interfaces (emulated MCTP host interfaces?)
  - DCD
  - Filling in all the details (there are a lot of commands!)
- Type 2 Devices?
- ARM support (could do with some help!)

What have we forgotten? (longer term!)

- Large Scale Fabrics?
  - How much should we do in QEMU?
- Performance optimization?
- IDE / TDISP etc?

What are people sitting on out of tree, that they might want to upstream?

Linux Plumbers Conference

Richmond, Virginia | November 13-15, 2023