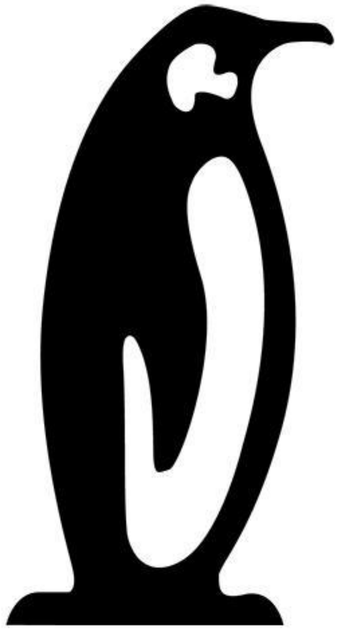


Linux Plumbers Conference

Dublin, Ireland September 12-14, 2022

Meta's CXL Journey and Learnings

Jonathan Zhang

Linux Plumbers Conference | Dublin, Ireland Sept. 12-14, 2022

Acknowledgements

Meta Reviewers

- Chris Petersen / Johannes Weiner / Anil Agrawal / Sai Dasari / Abhishek Dhanotia / Hao Wang / Paul Mckenney / Hiral Patel / Rik van Rel / Kevin Vigor / Chris Mason

Intel Reviewers

- Reddy Chagam / Dan Williams / Vishal Verma

AMD Reviewers

- Yazen Ghannam / Robert Richter

Microchip Reviewer

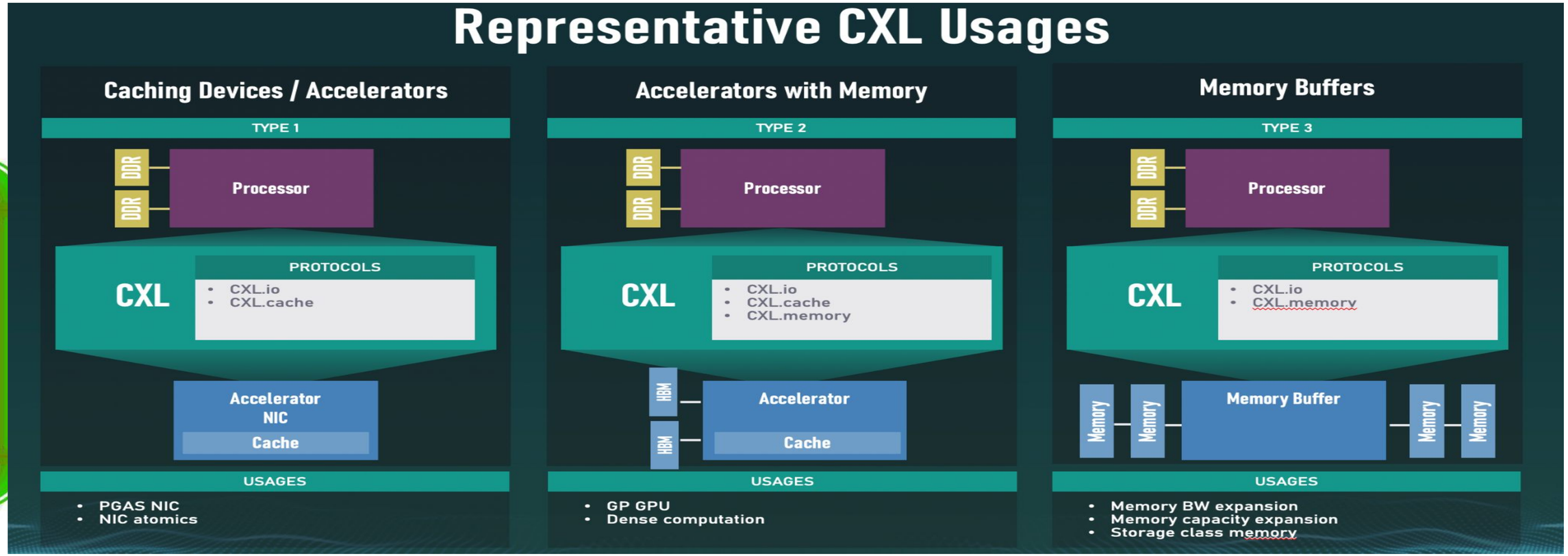
- Ariel Sibley

Agenda

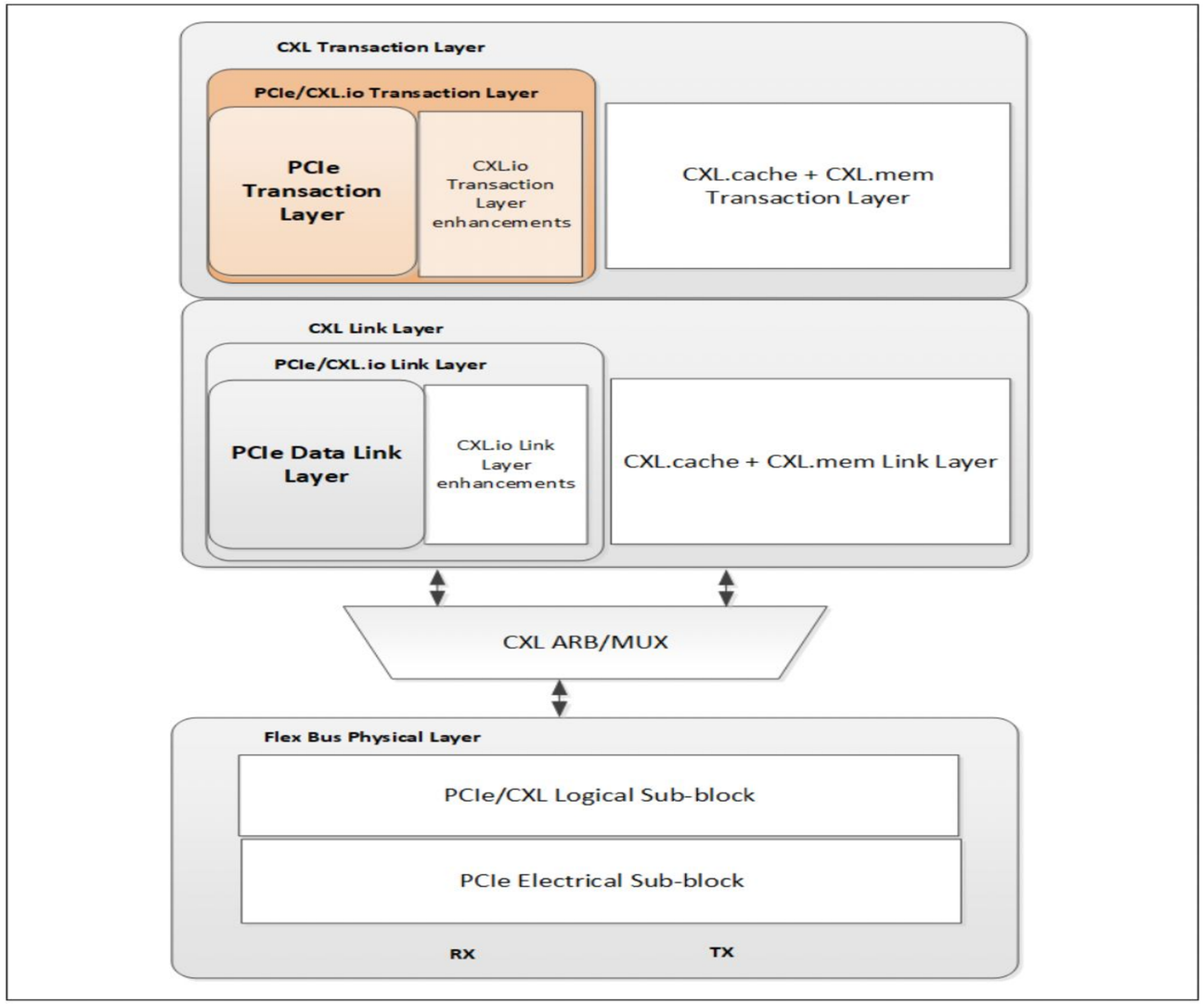
- CXL Memory Technology
- Meta Status / Plan
- Kernel Memory Management and CXL
- CXL Memory Device At-scale Management
- Call to Action / Discussion Points

What is CXL

- CXL specs: https://www.computeexpresslink.org/spec-landing



CXL Protocol

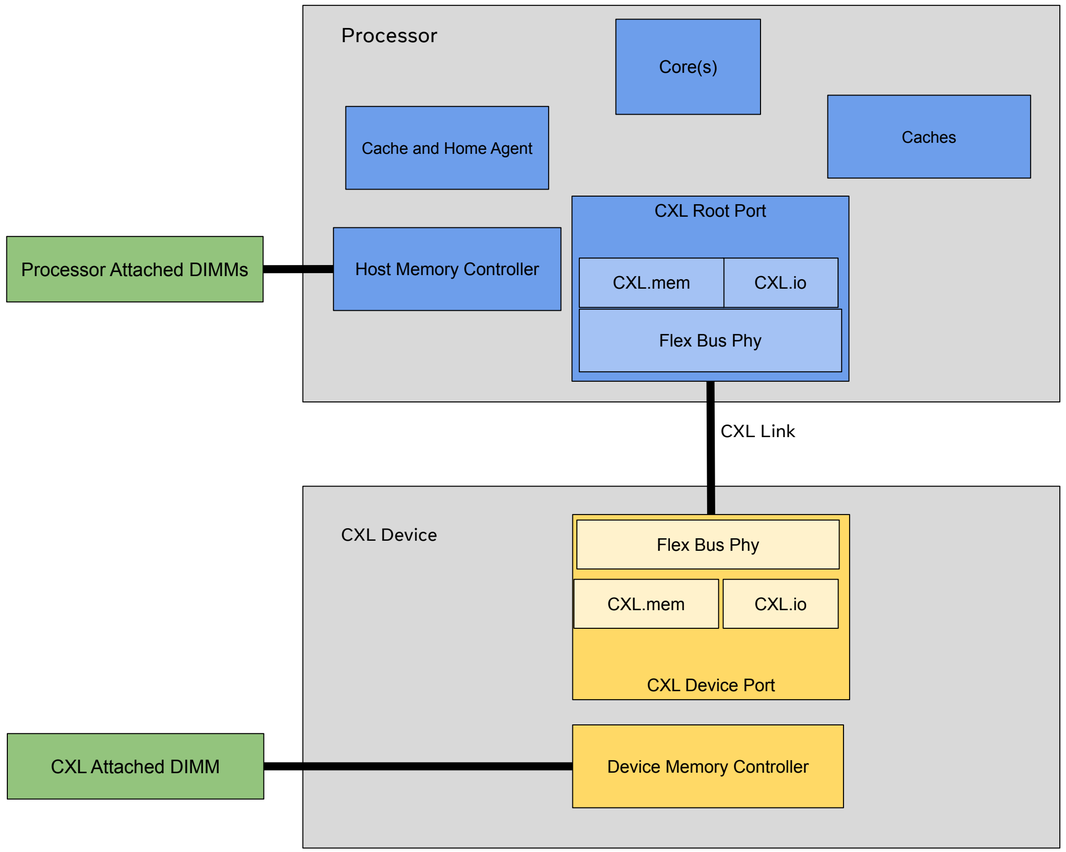


Industry Status

- Spec
  - CXL 2.0: released in Dec. 2020
  - CXL 3.0: released in Aug. 2022
    https://www.computeexpresslink.org/_files/ugd/0c1418_a8713008916044ae9604405d10a7773b.pdf
- Processors (Root Port)
  - CXL 1.1 support will be product launched soon
  - CXL 2.0 support being worked out
- CXL Memory Devices
  - CXL 2.0 support will be product launched soon

Features	CXL 1.0 / 1.1	CXL 2.0	CXL 3.0
Release date	2019	2020	1H 2022
Max link rate	32GTs	32GTs	64GTs
Flit 68 byte (up to 32 GTs)	✓	✓	✓
Flit 256 byte (up to 64 GTs)			✓
Type 1, Type 2 and Type 3 Devices	✓	✓	✓
Memory Pooling w/ MLDs		✓	✓
Global Persistent Flush		✓	✓
CXL IDE		✓	✓
Switching (Single-level)		✓	✓
Switching (Multi-level)			✓
Direct memory access for peer-to-peer			✓
Enhanced coherency (256 byte flit)			✓
Memory sharing (256 byte flit)			✓
Multiple Type 1/Type 2 devices per root port			✓
Fabric capabilities (256 byte flit)			✓

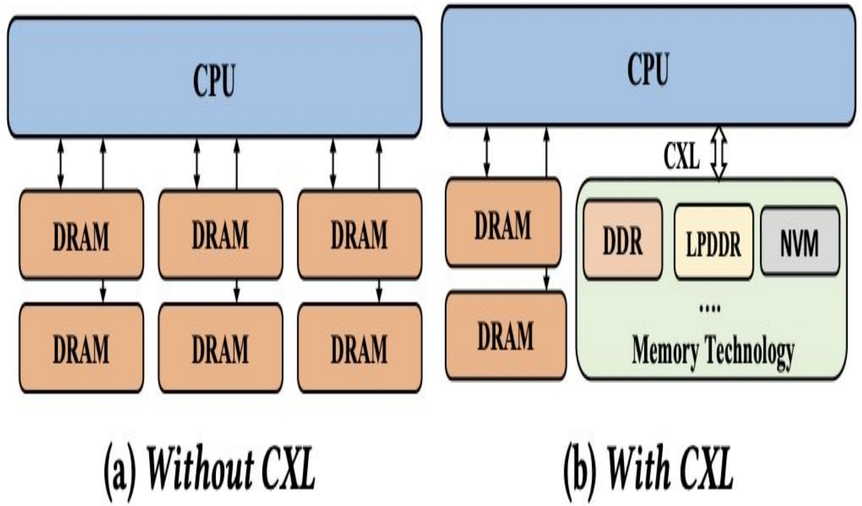
A Basic Configuration

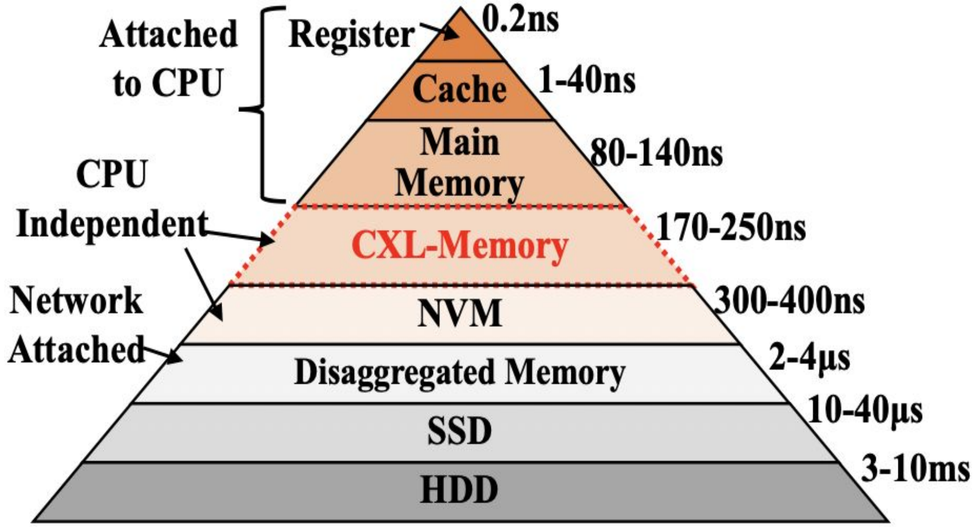



```
[root@AC-FB-03-00 ~]# numactl -H
available: 3 nodes (0-2)
node 0 cpus: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28
A
A
a

node 0 size: 15584 MB
node 0 free: 11219 MB
node 1 cpus:
Aa
A
A
A
A
A
Aa
A
A
A
A
A

node 1 free: 15092 MB
node 2 cpus:
node 2 size: 64468 MB
node 2 free: 64413 MB
node distances:
node   0   1   2
  0:  10  14  14
  1:  14  10  14
  2:  14  14  10
```

Decoupling Compute and Memory

- Bandwidth and Capacity
- Tiered Memory hierarchy
- Media diversity
- Flexible and fungible memory (Software Defined: hot plug, pooling, sharing)





Use Case Studies

- Meta:
  - General workload: https://arxiv.org/pdf/2206.02878.pdf
  - AI memory: https://www.snia.org/educational-library/ai-memory-meta-challenges-and-potential-solutions-2022
- MSFT:
  - Stranded Memory (multi-tenant use case), https://arxiv.org/abs/2203.00241
  - Multi-Tier Memory in Windows and Azure, https://youtu.be/58t_c39bMo4
- Kaist:
  - Memory disaggregation, https://www.nextplatform.com/2022/07/18/kaist-shows-off-directcxl-disaggregated-memory-prototype/

Agenda

- CXL Memory Technology
- Meta Status / Plan
- Kernel Memory Management and CXL
- CXL Memory Device At-scale Management
- Call to Action / Discussion Points

High Level Status / Plan

- System (Hardware and Software) prototype done
  - Latency/Power target met
  - Latency Impact studied, kernel patches posted upstream
- Multiple generations of system configuration being worked through
  - A couple generations / configurations of CXL devices (1.1, 2.0)
  - A couple generations / steppings of processors (1.1, 2.0)
- Memory management – kernel patches posted upstream
- At-Scale management – work in progress

Initial Use Case

- System configuration
  - CXL Memory as System Memory
  - Static configuration
- Kernel / OS (Kudos to CXL kernel/OS community)
  - Memory management
    - Meta CXL paper: https://arxiv.org/pdf/2206.02878.pdf
    - Kernel patches for CXL memory capacity expansion - Transparent Page Placement for Tiered-Memory https://lore.kernel.org/all/cover.1637778851.git.hasanalmaruf@fb.com/T/
    - Kernel patches for CXL memory bandwidth: N:M interleave policy for tiered memory nodes https://lore.kernel.org/linux-mm/20220607171949.85796-1-hannes@cmpxchg.org/
  - Device management
    - CXL driver
    - Other kernel changes
    - User space

Agenda

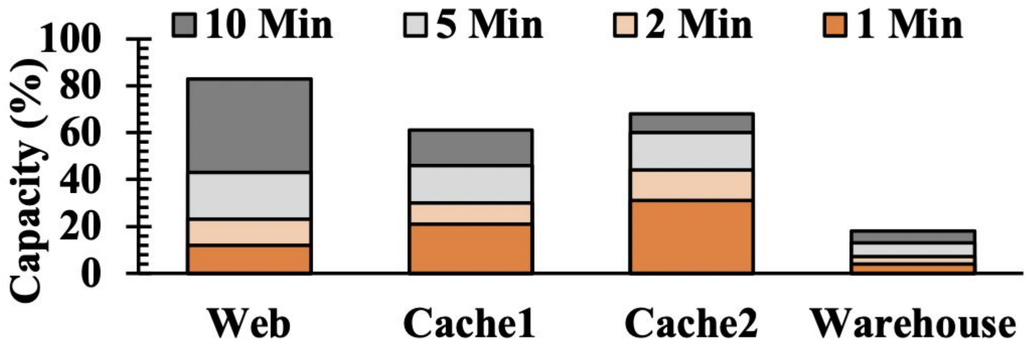
- CXL Memory Technology
- Meta Status / Plan
- Kernel Memory Management and CXL
- CXL Memory Device At-scale Management
- Call to Action / Discussion Points

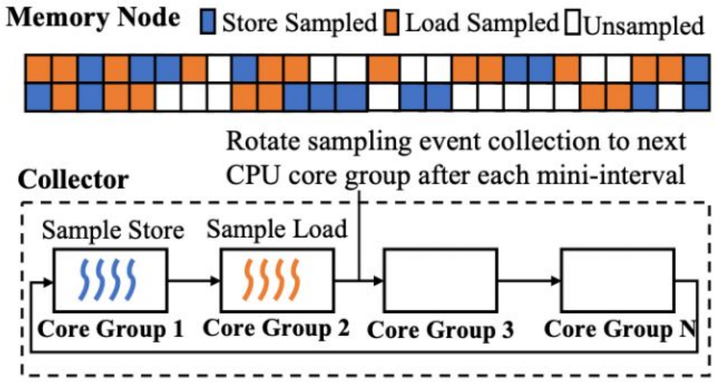
Problem Statements

- Goals
  - Utilize increased memory bandwidth.
  - Minimize impact to workload performance despite increased memory latency:
    - Without prior knowledge of application behavior
    - Without in-depth application tuning

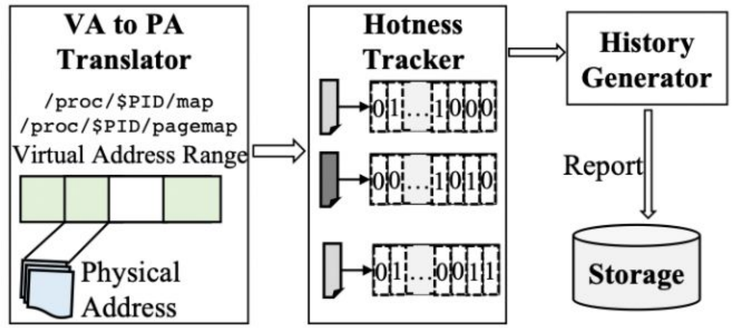
- Impact of current kernel
  - Paging mechanism being latency intensive.
  - NUMA balancing not able to move pages to CPU-less memory node.
  - Failing to maintain a head room of free pages under memory pressure.

User Space Profiling Tools

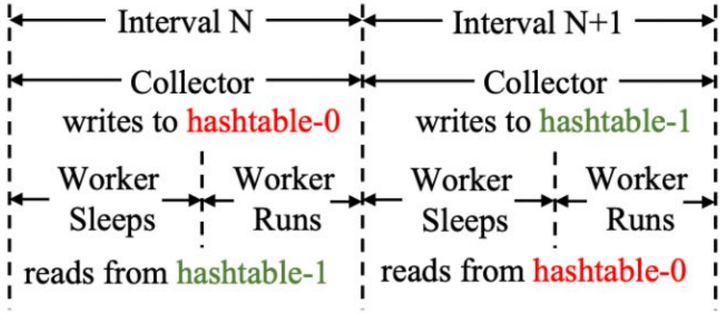




(a) *Collector*



(b) *Worker*



(c) *Workflow within a Cycle*

TPP (Transparent Page Placement)

- Design
  - Migration for lightweight reclamation
    - demotion of cold pages
    - promotions of hot pages
  - Decoupling allocation and reclamation
  - Observability
- Effectiveness (Comparing to baseline – all capacities provided by local memory)
  - Baseline: 96GB processor attached DRAM
  - CXL config: 64 GB processor attached DRAM + 32GB CXL DRAM

WorkLoad	Current kernel	TPP
Web	-17%	-0.5%
Caching 1	-10%	-0.4%

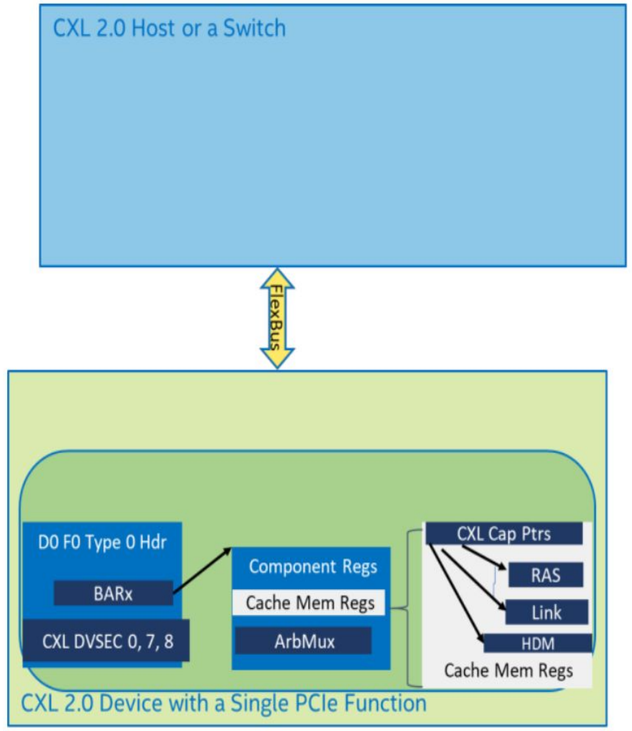
Interleave Improvement

- https://lore.kernel.org/linux-mm/20220607171949.85796-1-hannes@cmpxchg.org/
- Suitable for bandwidth-intensive workloads
- Only applies to processes and vmas with an "interleave" Policy
- Effectiveness (Comparing to baseline – all capacities provided by local memory)
  - Default 1: 1 interleaving, 40% drop
  - 5:1 interleaving, 8% increase
- The optimal ratio (N: 1) mostly depends on
  - Hardware (The shape of latency vs. bandwidth curves)
  - But not workload

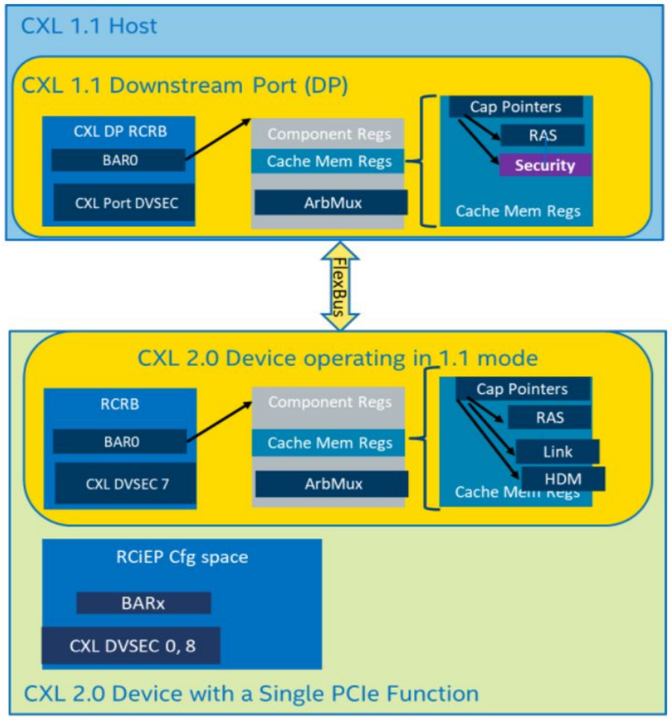
Agenda

- CXL Memory Technology
- Meta Status / Plan
- Kernel Memory Management and CXL
- CXL Memory Device At-scale Management
- Call to Action / Discussion Points

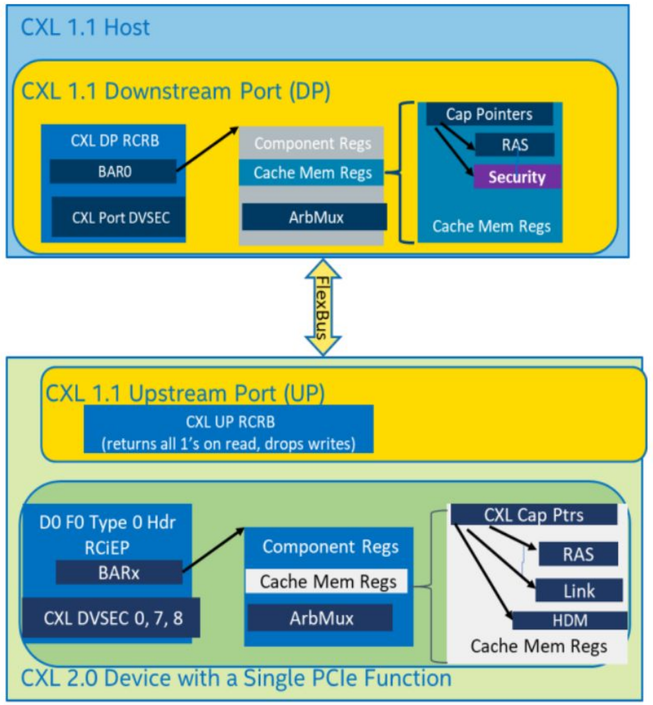
RCRB Mode



2.0 Host + 2.0 Device



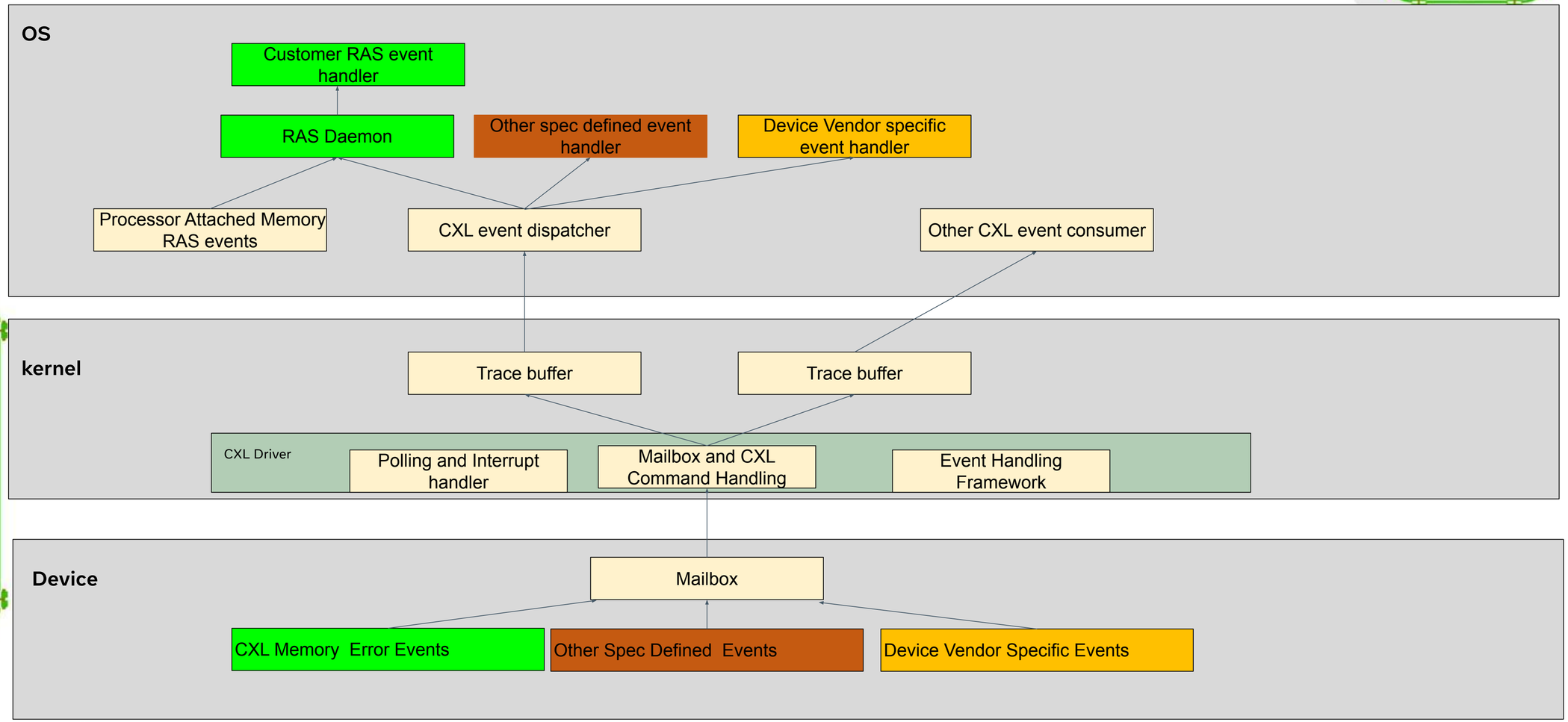
1.1 Host + 2.0 Device (RCRB mode)



1.1 Host + 2.0 Device (non-RCRB mode)

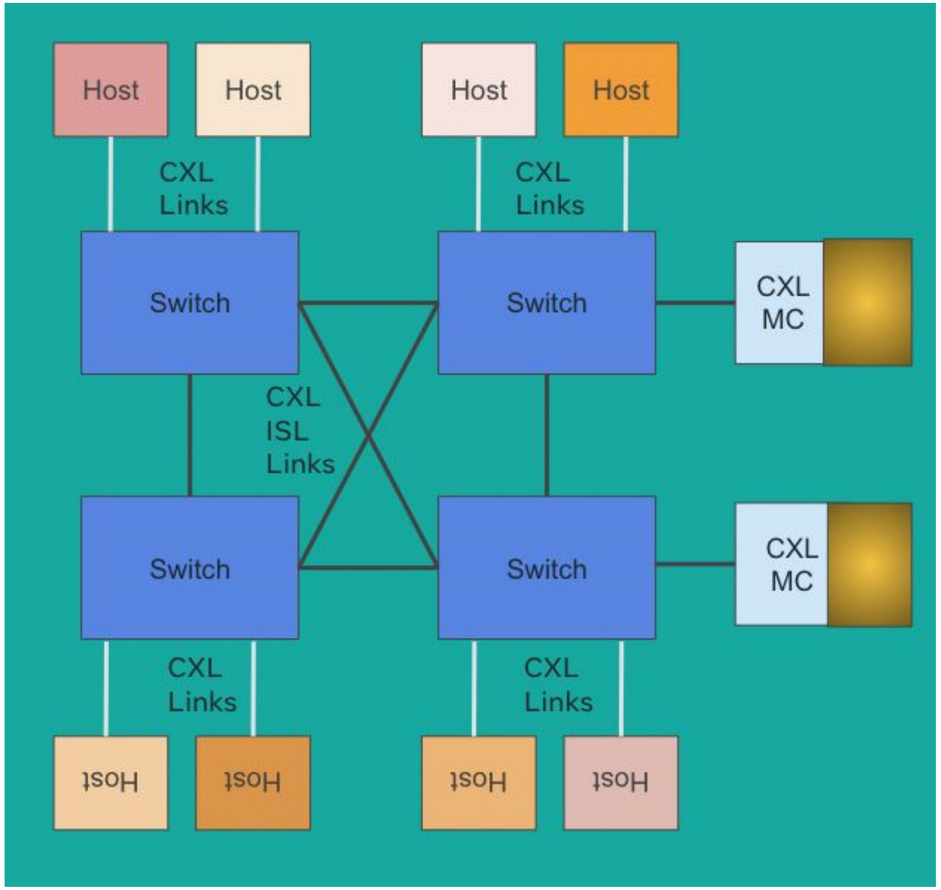
A number of production-planned CXL 2.0 devices only support RCRB mode!

Event Management



Reset Management

- Conventional reset
  - Hot Reset – Triggered via link (via LTSSM or link down)
  - Warm Reset – Triggered via external signal, PERST# (or equivalent, form factor specific mechanism)
  - Cold Reset – Involves main Power removal and PERST# (or equivalent, form factor specific mechanism)
- Functional level reset
  - No effect on the CXL.mem protocol
- CXL reset
  - Multi hosts sharing a device, while CXL reset affects all HDMs and the whole device.
  - HDM region may migrate from one host to another host
  - Mapping of a host memory region may switch from one CXL device to another CXL device.



Reset Management

- CXL driver flow to support reset
  - Prior to issuing reset
    - Offline HDM ranges, eg. quiesce and stop CXL.mem traffic
    - Make sure the host stop initiating any new CXL.io requests, including power management
    - Clear/randomize content if the device does not do so automatically
  - Issue CXL reset
    - Set the bit to clear HDM range
  - Following CXL reset
    - Re-initialize all CXL functions
    - Re-initialize HDM range if device did not do so

CXL Memory Information

- Problems with dmidecode
  - Processor attached memory data only. ← Could include CXL memory IF host firmware supports so.
  - Static info from boot time.
  - System memory only, not showing memories accessible through devices.
- How about something like lsmem, similar to lspci?
  - Current data. Updated following hot-plug, hot-remove.
  - Memory media Info: both processor attached memory and CXL memory.
  - Interleave info.
  - Memory segments mapping info
    - System memory
    - Kernel memory devices (DAX device, DirectCXL device, etc.), used by which process.
  - CXL device info: /sys/bus/cxl/devices/{regions | mems | ports | dports | decoders}
    - ndctl cxl command displays such info

Agenda

- CXL Memory Technology
- Meta Status / Plan
- Kernel Memory Management and CXL
- CXL Memory Device At-scale Management
- Call to Action / Discussion Points

Call to Action

- Contribute to kernel memory management improvement for CXL memory
  - Code Development, Review, Design discussion, use case study, benchmarking
- Contribute to kernel/OS work needed for CXL device at-scale management
  - Code Development, Review, Design discussion, use case study, benchmarking
- Design software solution to unleash the potentials brought forth by CXL technology
- Join CXL forum SSWG and MSWG

CXL Mirco Conference

- Time: Wednesday (09/14) 3pm to 6:30pm
- Room: Herbert
- Leads: Adam Manzanares (Samsung), Ben Widawsky (Intel), Dan Williams (Intel)

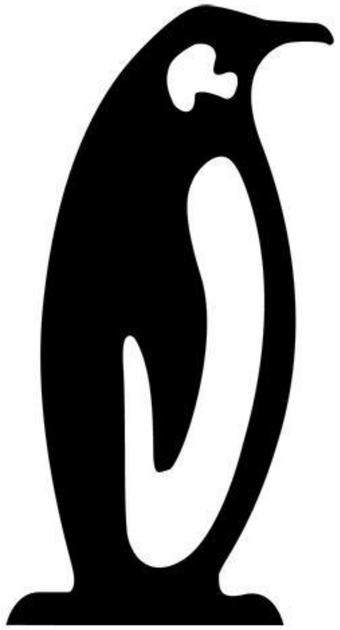
Linux Plumbers Conference | Dublin, Ireland Sept. 12-14, 2022

Discussion – Kernel Memory

- For interleaving, how to tune the interleave ratio with less tuning effort? Right now, some manual work is needed.
- TPP and interleave are exclusive technologies, some workload works better with TPP (latency sensitive), some workload works better with interleave (bandwidth sensitive). How to get kernel to train itself to decide best approach for the current workload? Could BPF be a fallback route?
- Tiered memory hierarchy
  - Actual latency/Bandwidth values not considered.
  - How to utilize the increased bandwidth to its full potential?
- Hot plug enablement
  - Traffic management (need to quiesce all traffic)
  - Device management such as CXL reset, device enumeration
  - Memory management (active, potential regions)
  - Security (reinitialize/randomize memory content)
- Memory sharing, how to utilize it

Discussion – Performance Tools

- Could some of the memory performance tuning be done in user space?
- How to change the latency/bandwidth profiles of CXL memory regions, to simulate memory hierarchy.
- How to change the behavior of benchmarking apps, such as page type ratio, memory access pattern, etc, to simulate a variety of workloads memory demand behavior.



Linux Plumbers Conference

Dublin, Ireland September 12-14, 2022