Tiresias : Optimizing NUMA Performance with CXL Memory and Locality-Aware Process Scheduling

Wenda Tang[1,2], Tianxiang Ai[1], and Jie Wu[2]
[1]China Telecom eSurfing Cloud
[2]China Telecom Cloud Computing Research Institute

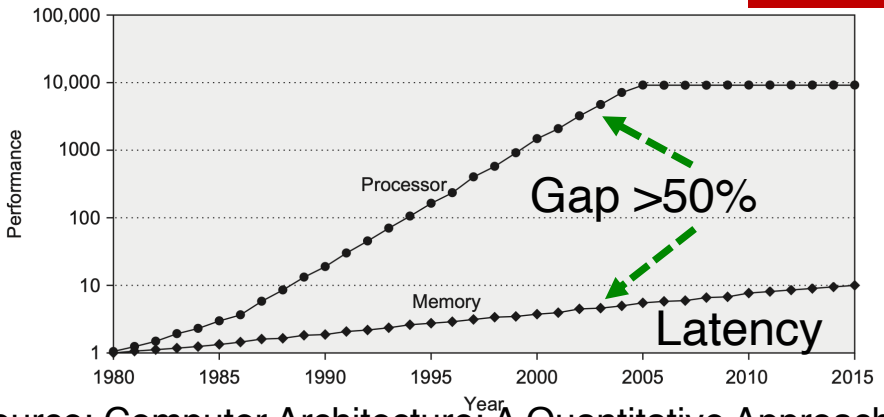
2024 年 7月

Presentation Outline

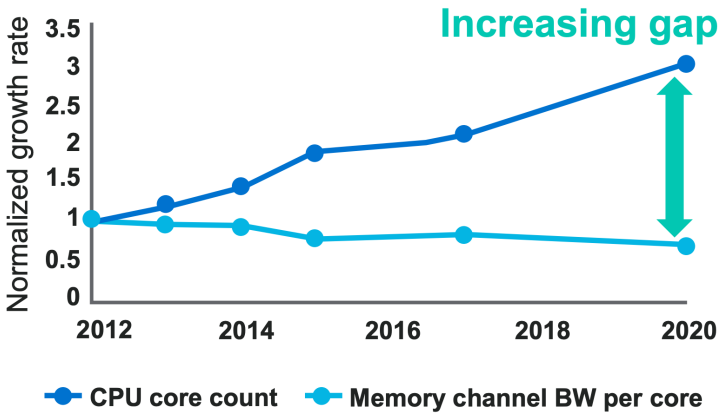
- Introduction
- Motivation & Background
- *Tiresias* Design
- Performance Analysis
- Conclusion & Future Work

Introduction

Memory Wall

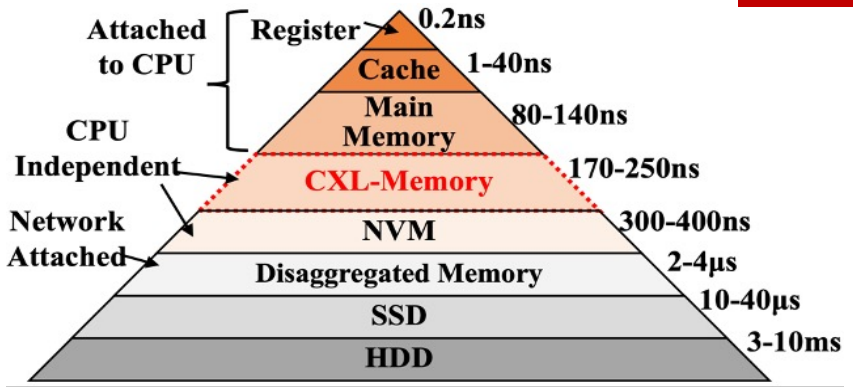


Source: Computer Architecture: A Quantitative Approach

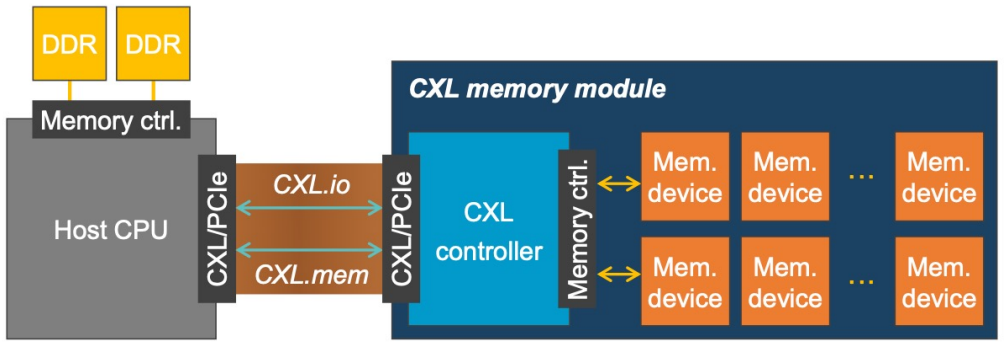


Source: Meta, OCP Summit 2021 (https://youtu.be/SIumdt2vLyo)

CXL Solution

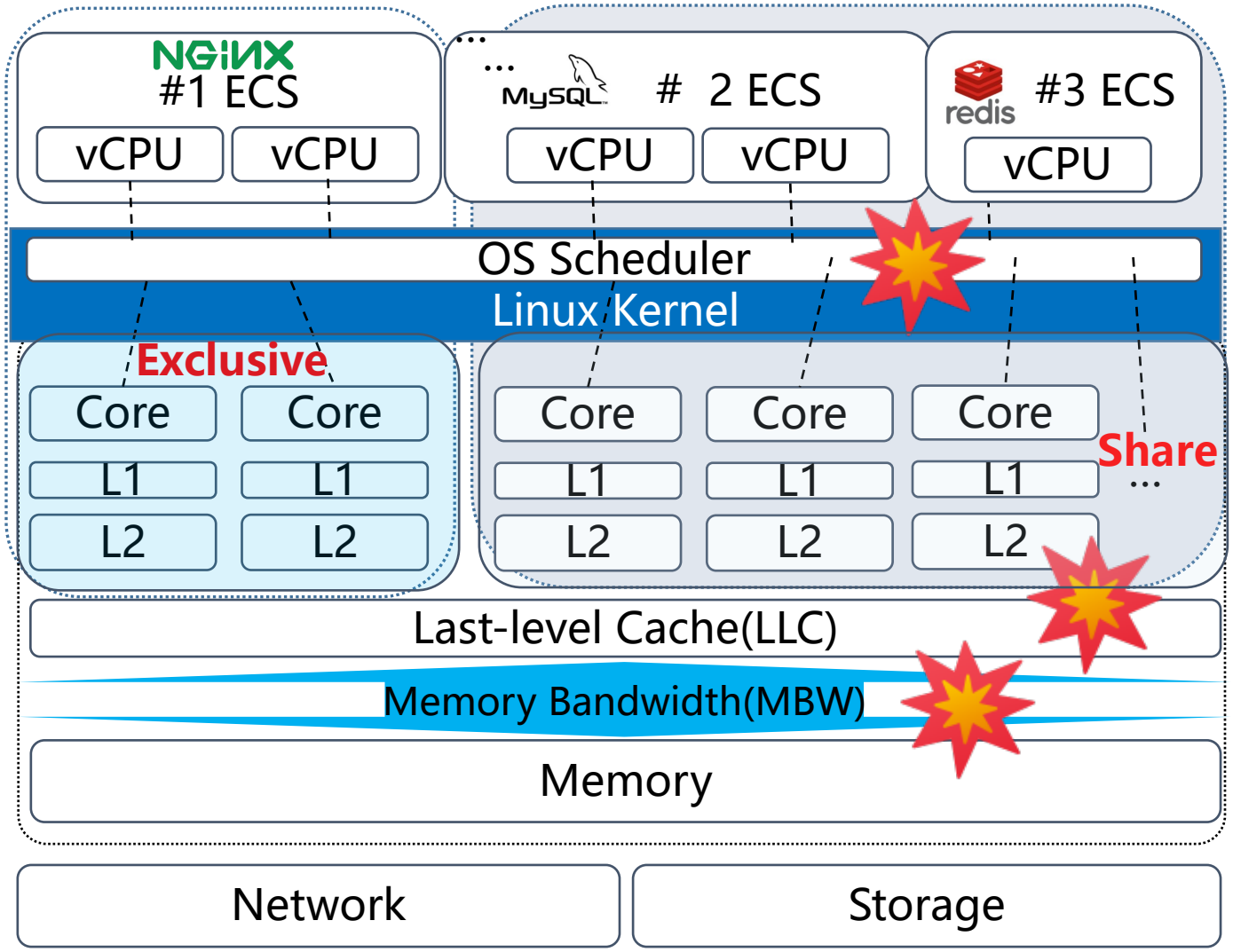


Latency characteristics of memory technologies.
—— TPP，Meta [ASPLOS'23]



Using CXL memory as a bandwidth expander.
—— Caption，Intel [MICRO'23]

Motivation & Background



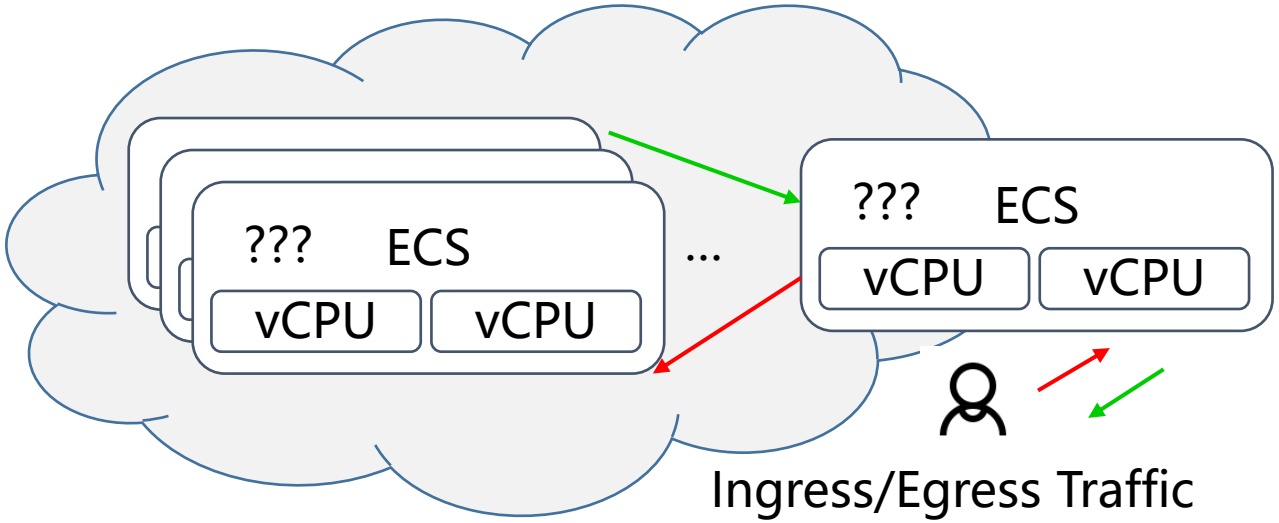
Interference is Everywhere in Public Clouds

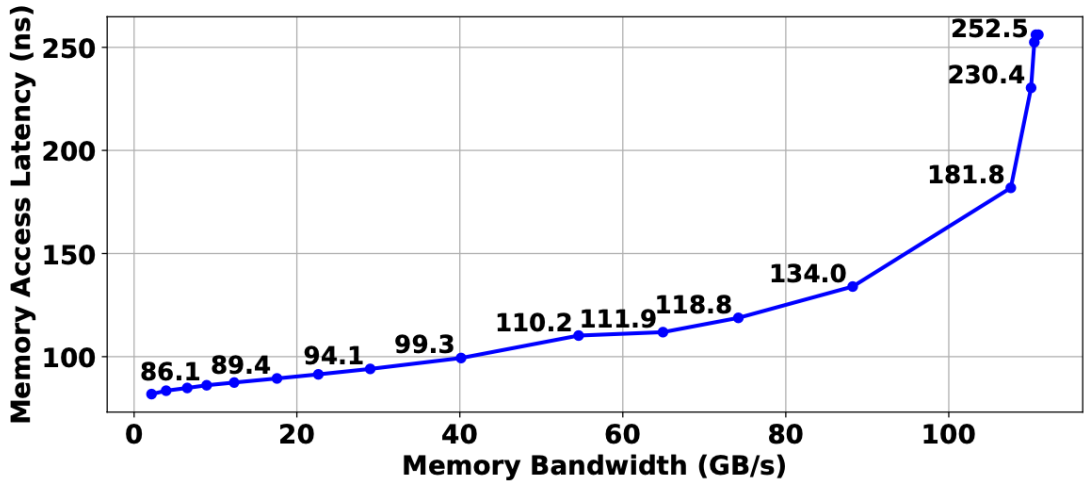
Interference in each layer

1. CPU Contention->Scheduling Latency
2. L1/L2 Cache Interference
   run-queue-rebalance warm-up
3. LLC Cache Interference
4. Memory Bandwidth Contention
5. Network, Storage I/O Contention...

Motivation & Background

Differentiated Memory QoS Requirements

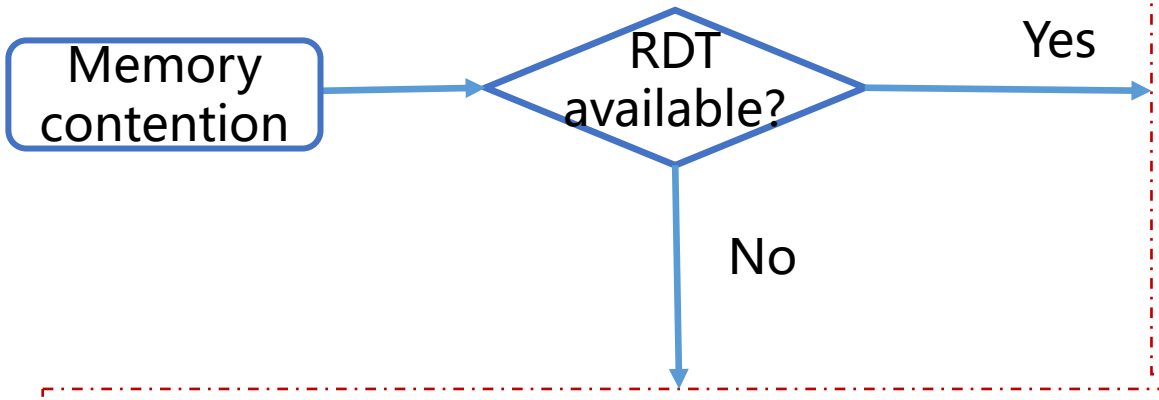




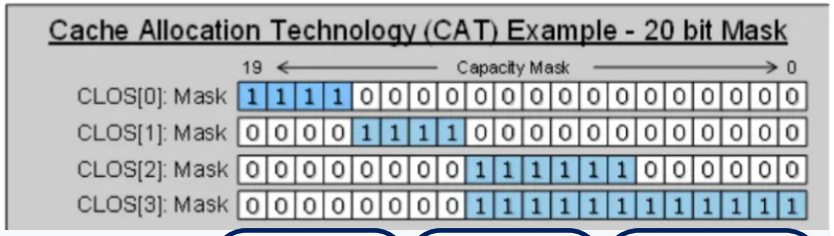
- Latency-Critical (LC) Performance Sensitive, Metric: P99 Latency…
  e.g., Web Search, Social Media
- Best-Effort (BE) Performance Insensitive, Metric: Job Finish Time…
  e.g., Offline Analysis

Memory access latency increases monotonically as the memory bandwidth pressure increases.

1. Differentiated Memory QoS Guarantee



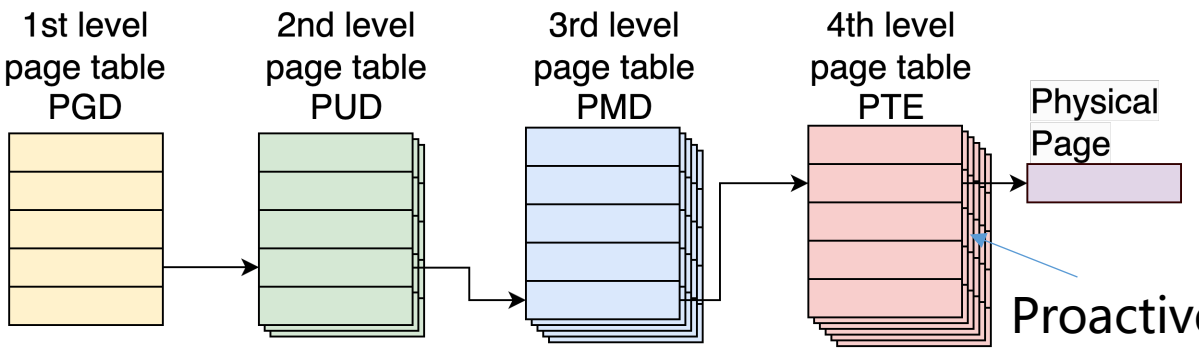
Hardware-based memory bandwidth control

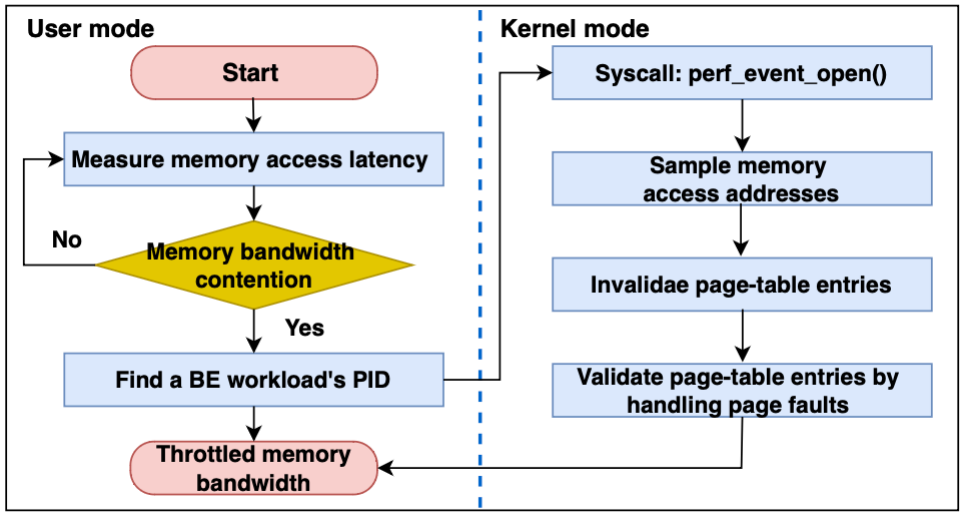


CORE	IPC	MISSES	LLC[KB]	MBL[MB/s]	MBR[MB/s]
0-2	0.28	7893k	383.2	901.2	430.8
3-5	0.28	45k	25.3	361282.6	22.4
6-8	0.26	89468k	6778.8	43904.3	4.3

CMT:Cache Occupancy | **MBM:Local Bandwidth** | **MBM:Remote Bandwidth**

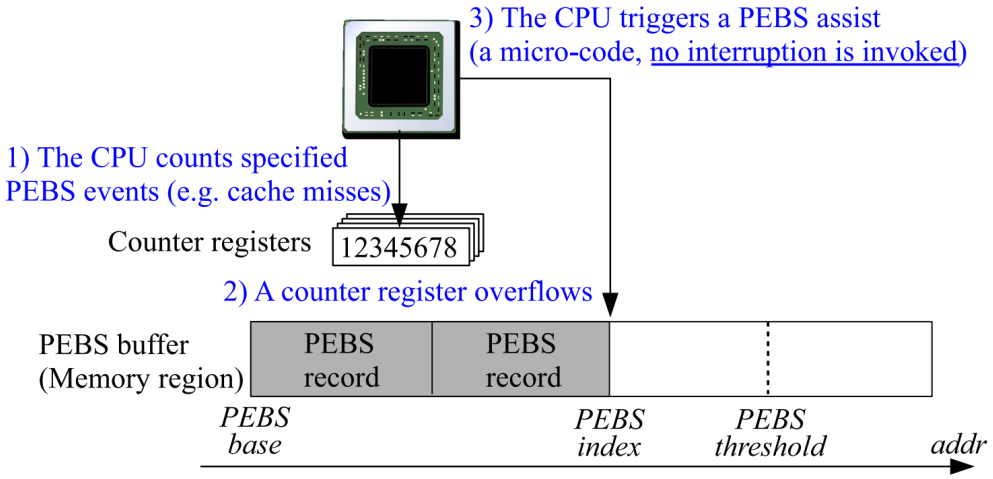
Software-based memory bandwidth control





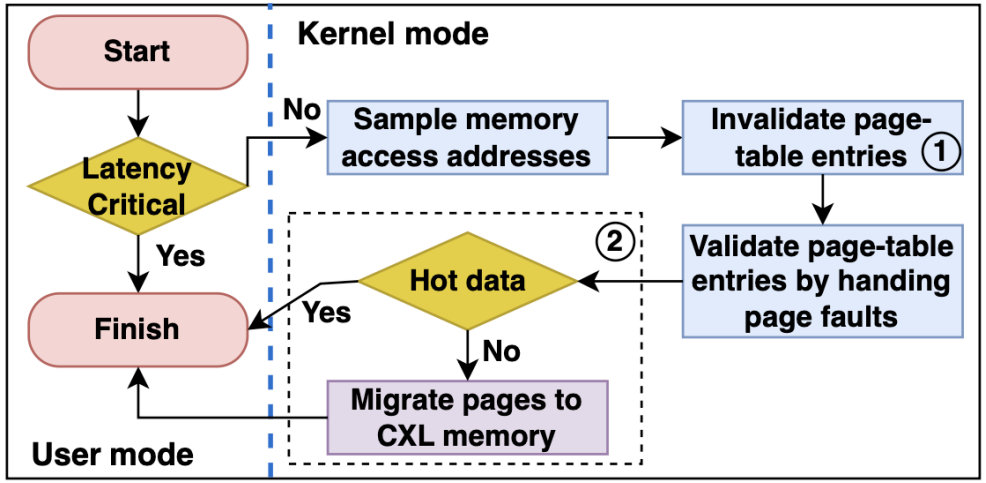
2. Bandwidth Expansion via CXL Memory

Memory Profiling



PEBS （Processor Event-Based Sampling)-based hot data identification
Sample technique used in MEMTIS[SOSP23], HeMem[SOSP21],TMTS[ASPLOS23] ...

Migrate Hotspots



Unthrottling memory bandwidth of BE workloads via CXL memory.

3. Locality-Aware Process Scheduling

PTSR

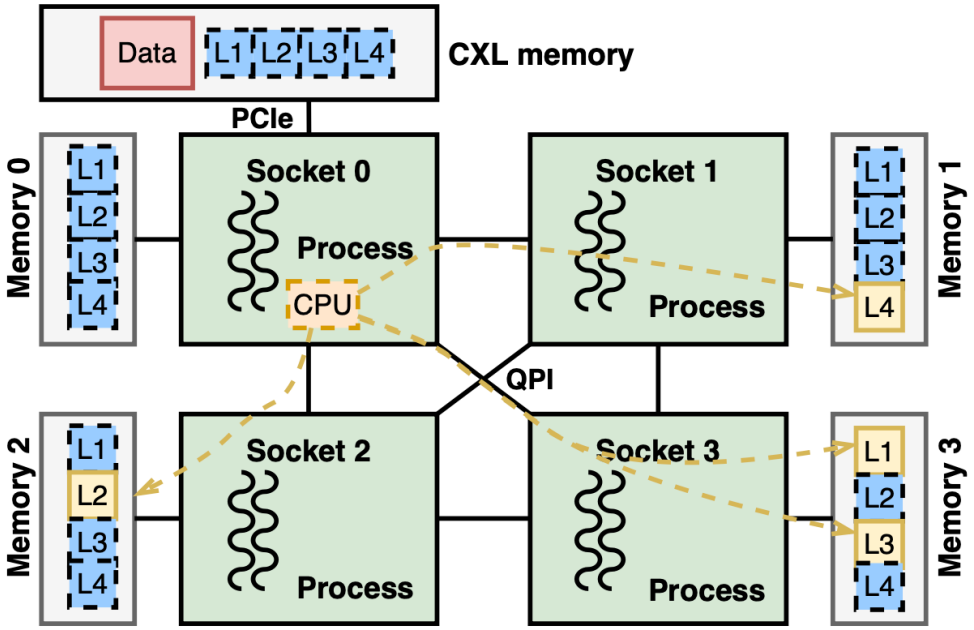
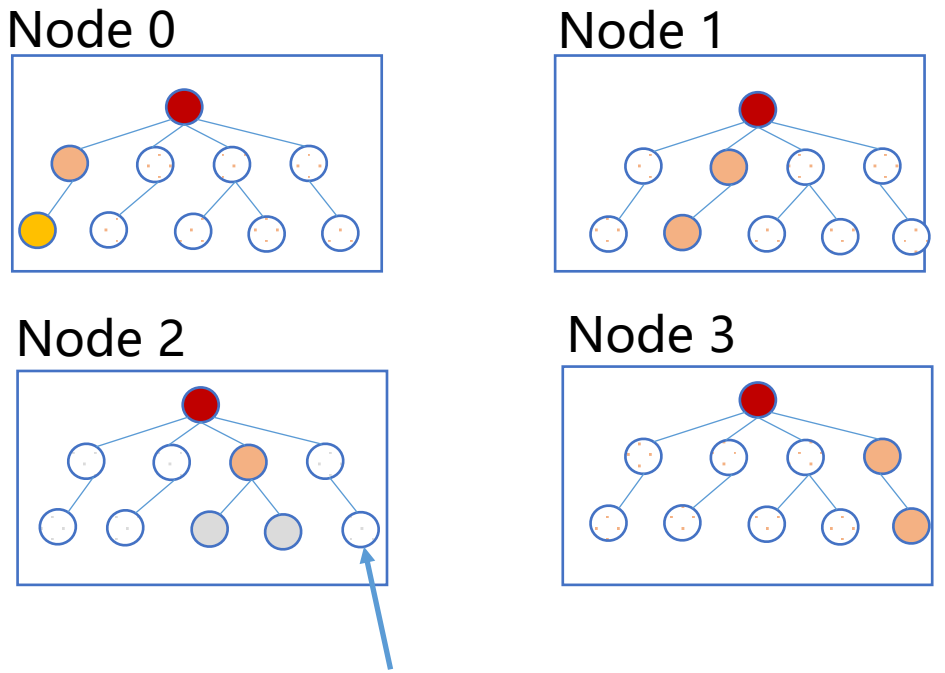


illustration of PTSR
(page-table self-replication)

Page fault based process scheduling



Performance Analysis

Expected Latency

M local NUMA access
N remote NUMA access

$p_1 = \binom{M+N-2}{N} / \binom{M+N}{N}$

$p_2 = \binom{M+N-2}{M} / \binom{M+N}{M}$

$p_3 = \binom{M+N-2}{N-1} / \binom{M+N}{N}$

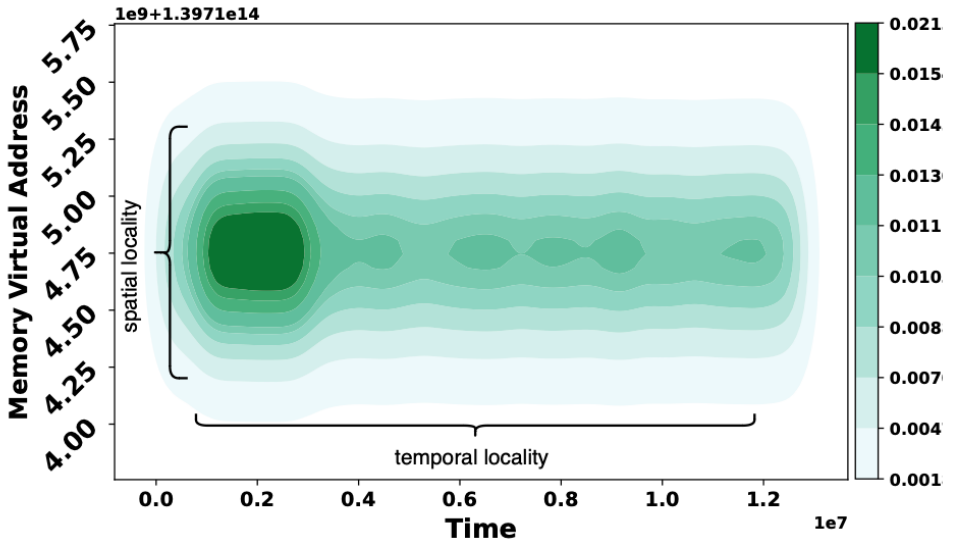
l_{access}	Local access
s_{access}	Schedule+ local access
r_{access}	Remote access

$$E = \frac{l_{access} + \sum_{i=2}^{M+N}((p_1 + p_2)l_{access} + 2p_3 s_{access})}{M+N}$$

$$E = \left(\frac{M^2+N^2}{M+N} l_{access} + \frac{2MN}{M+N} s_{access}\right)/(M+N)$$

...

Data Locality



Kernel density estimation plot of memory address accesses over time in a Memcached process.

$$E \ll r_{\text{access}}$$

Conclusion & Future Work

Conclusion

- Black-box workloads in public clouds call for new techniques for allocating memory subsystem resources （including CXL memory）.
- *Tiresias* exploits three optimization techniques: (1) workload-aware and software-based memory bandwidth management, (2) a memory page migration strategy to alleviate memory bandwidth contention by leveraging CXL memory, and (3) PTSR based locality-aware process scheduling.

Future Work

- Experiments results on real CXL hardware.
- System implementation includes CPU scheduler and page-table management in Linux Kernel.
- QoS monitoring and performance-aware strategies.

Thanks