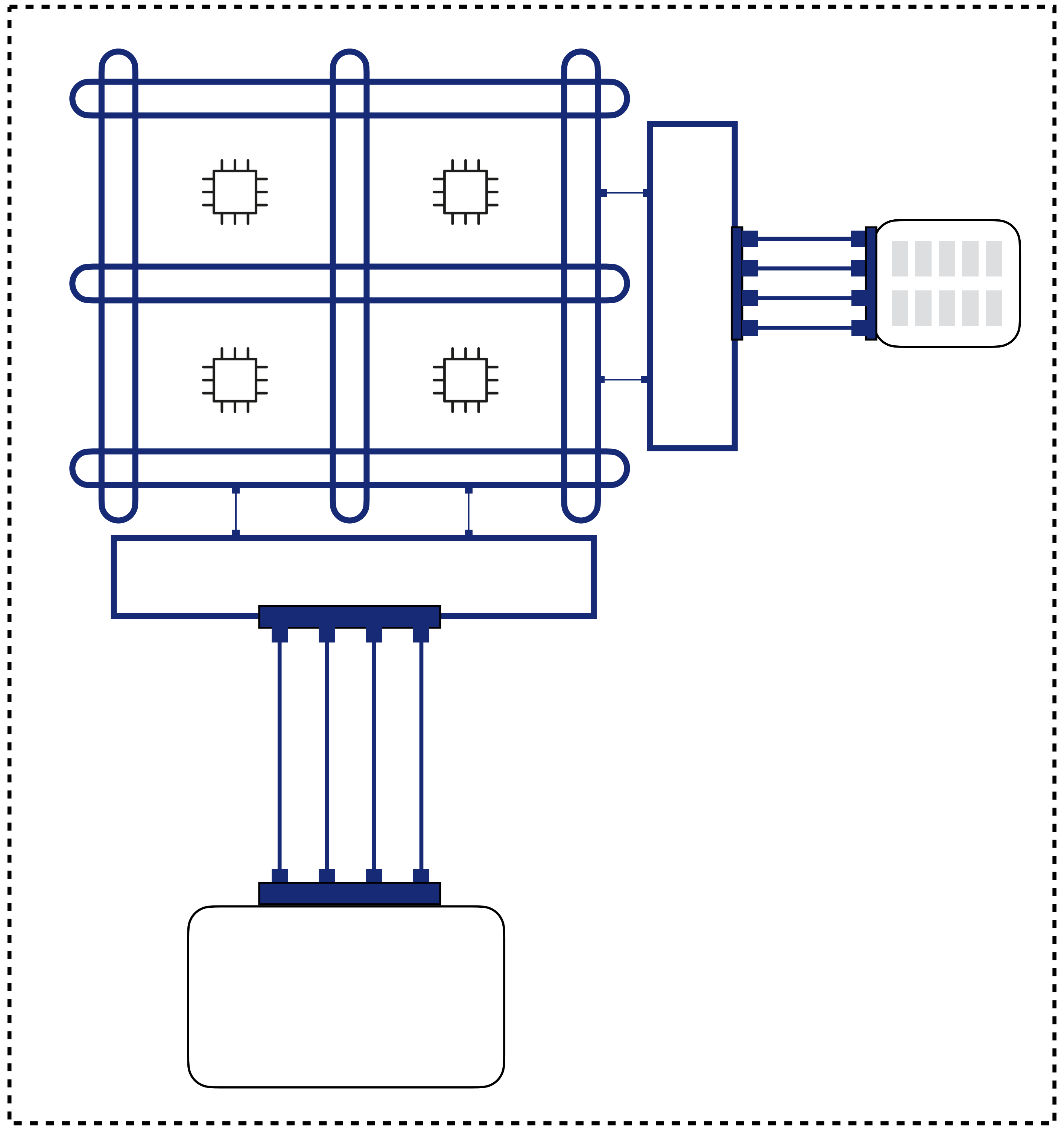
Understanding the Host Network



Midhul Vuppalapati

Saksham Agarwal

Henry Schuh

Baris Kasikci

Arvind Krishnamurthy

Rachit Agarwal

Cornell University

ILLINOIS

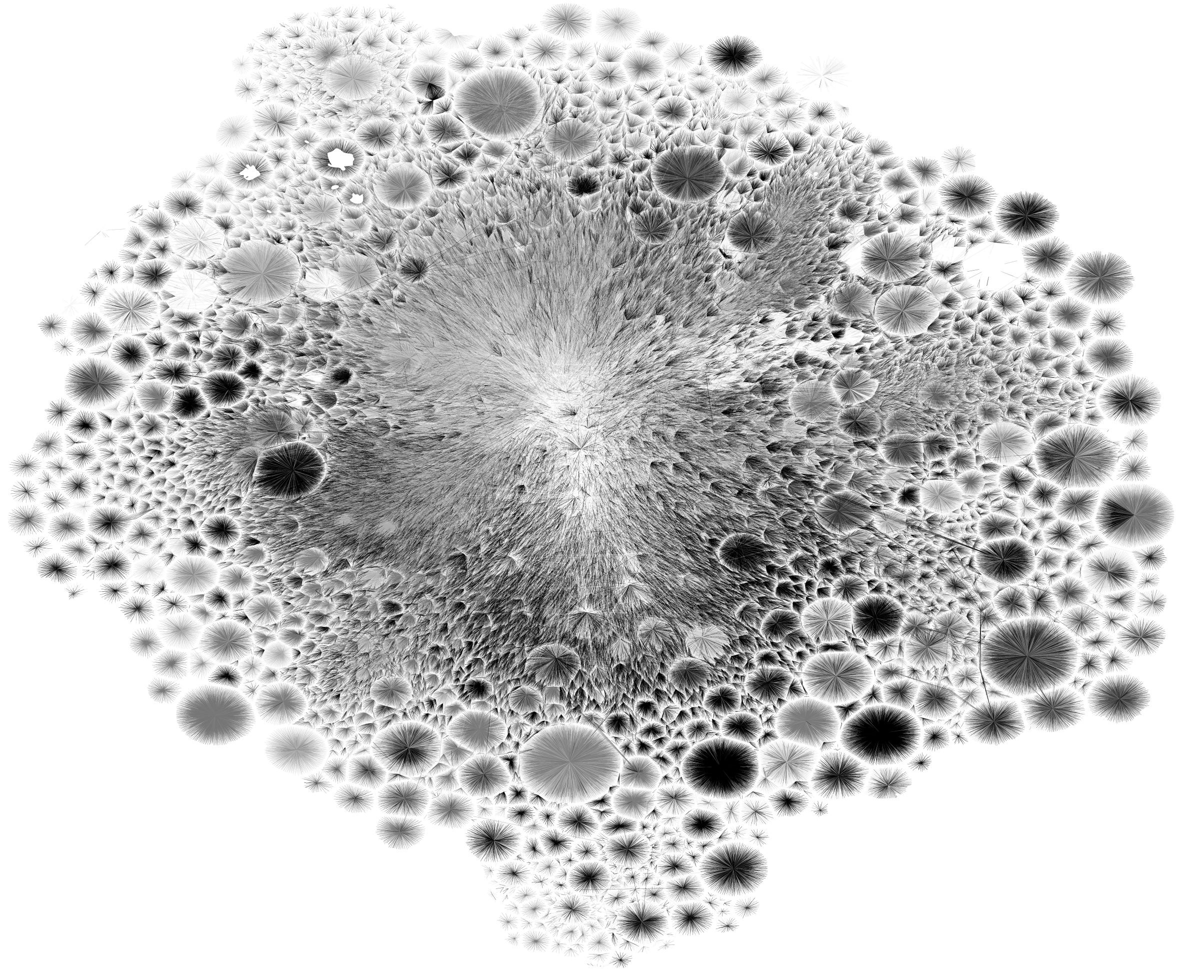
UNIVERSITY of WASHINGTON

The Host Network: Network within a single host

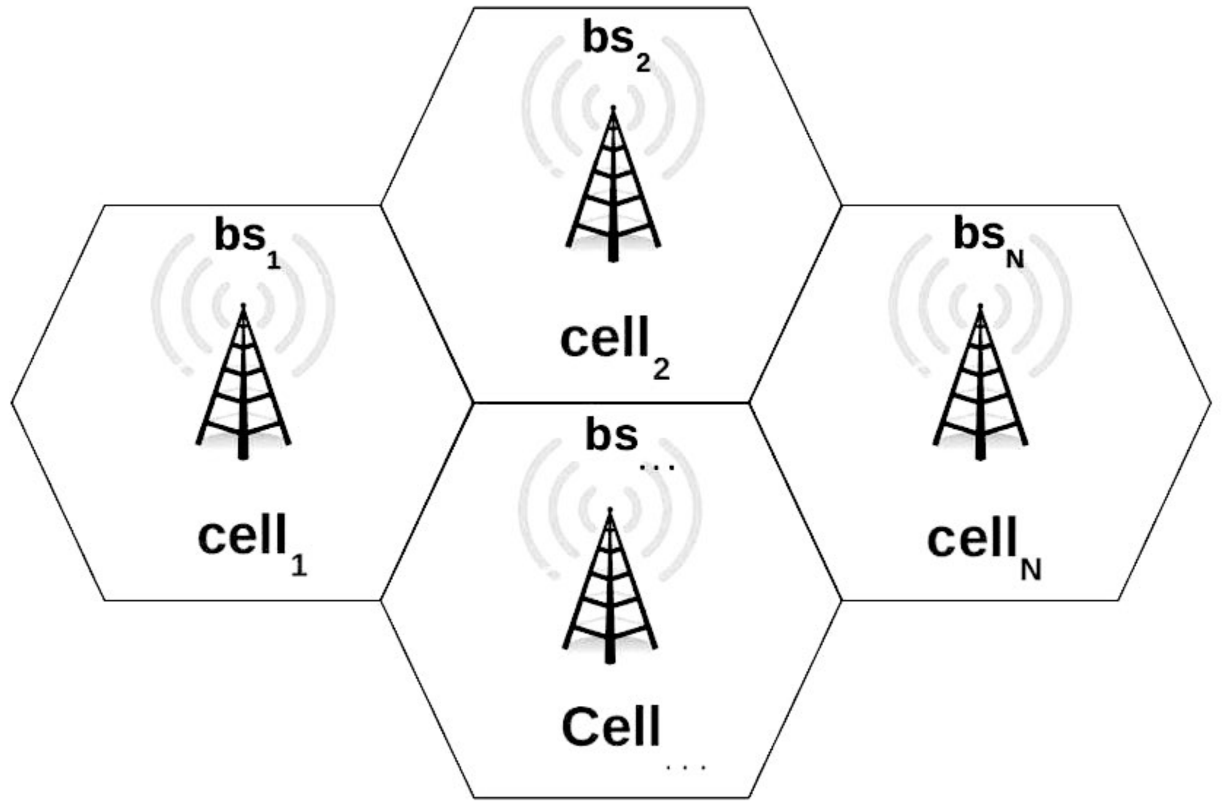
Our community has studied many different kinds of networks

This talk

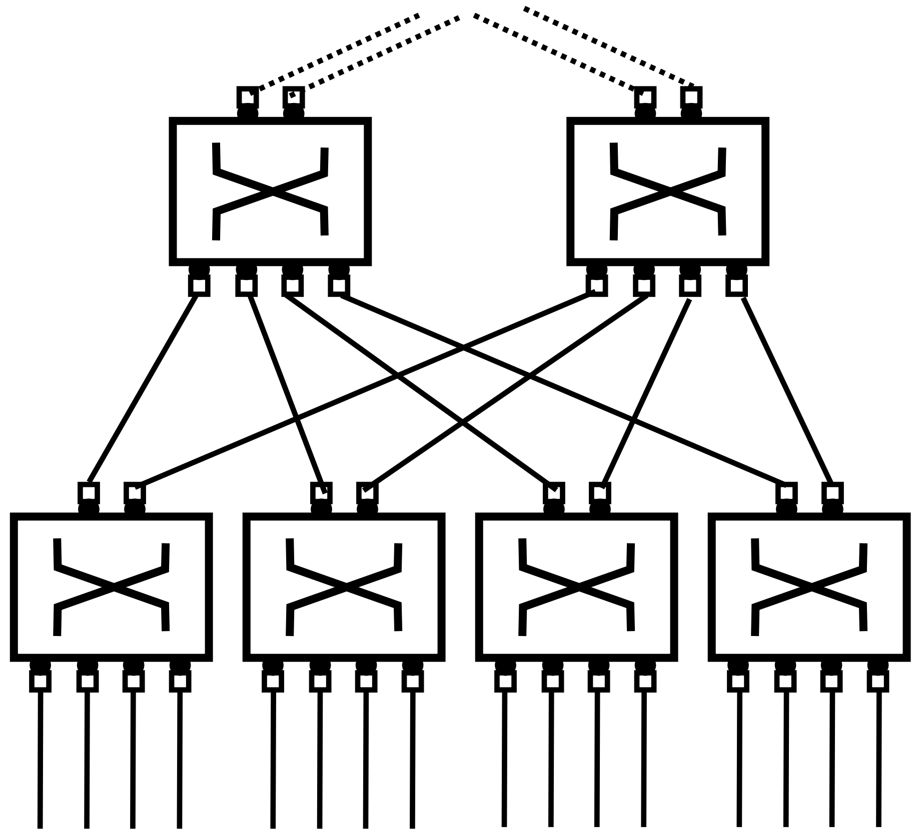
Internet



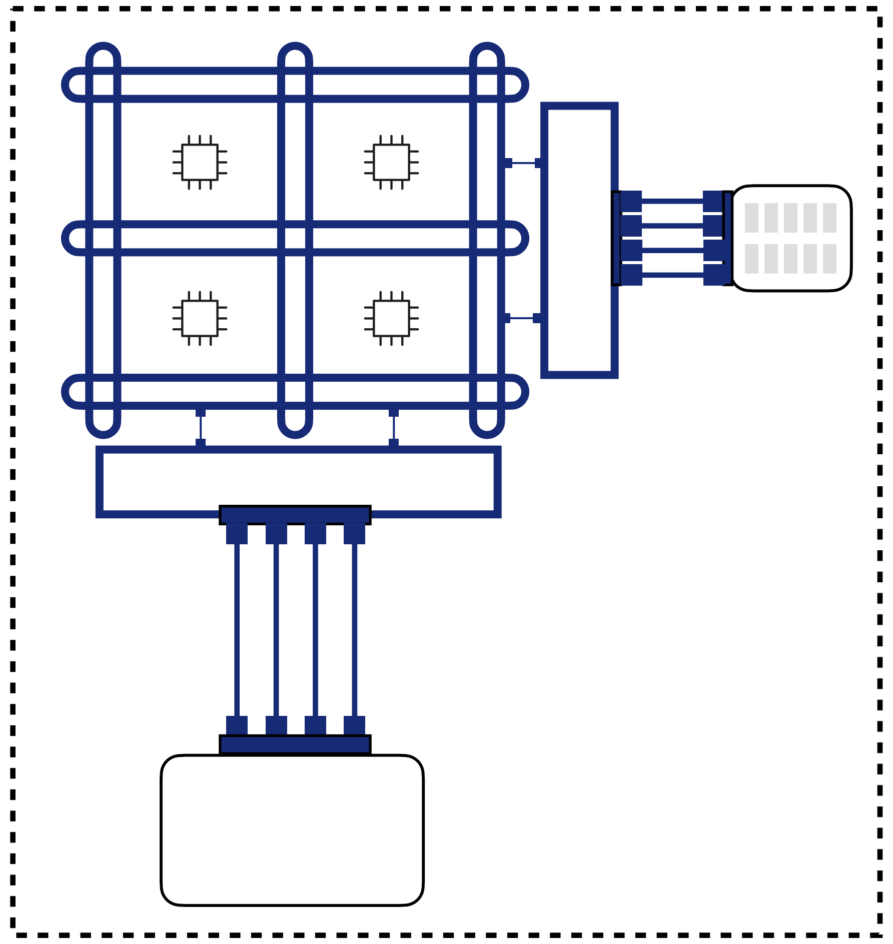
Mobile Network



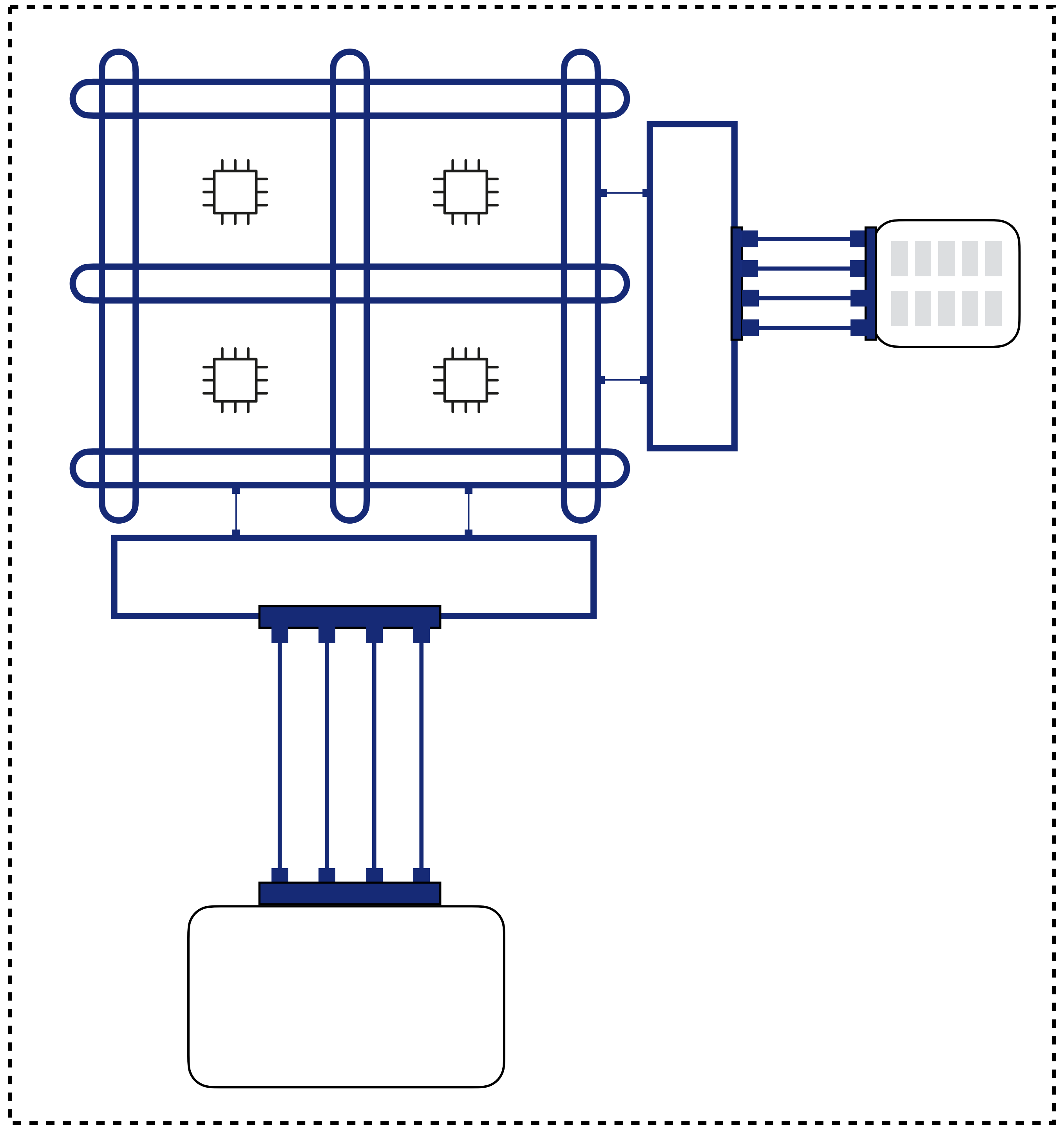
Datacenter Network



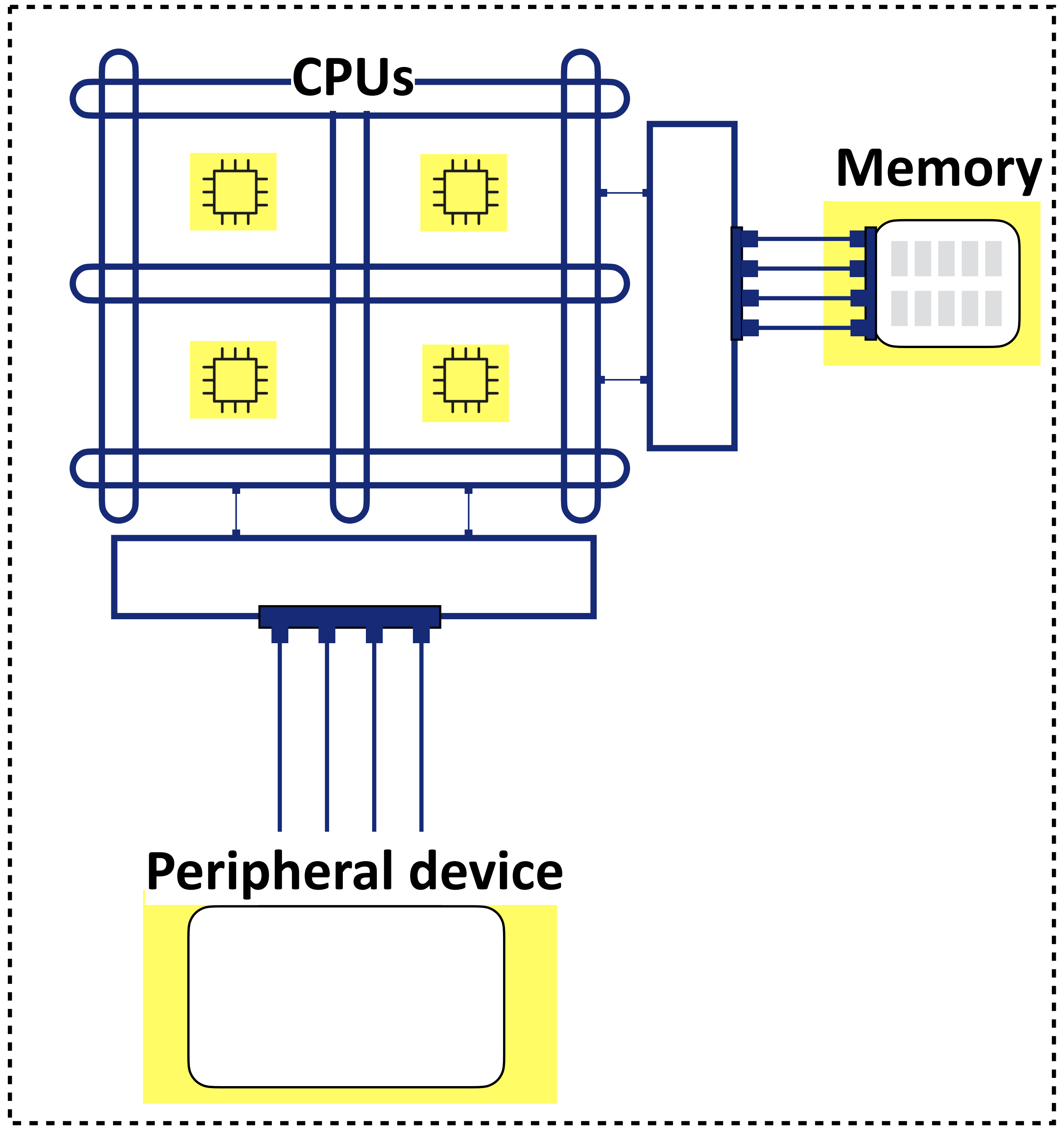
Host Network



The Host Network: An inter-network within a host



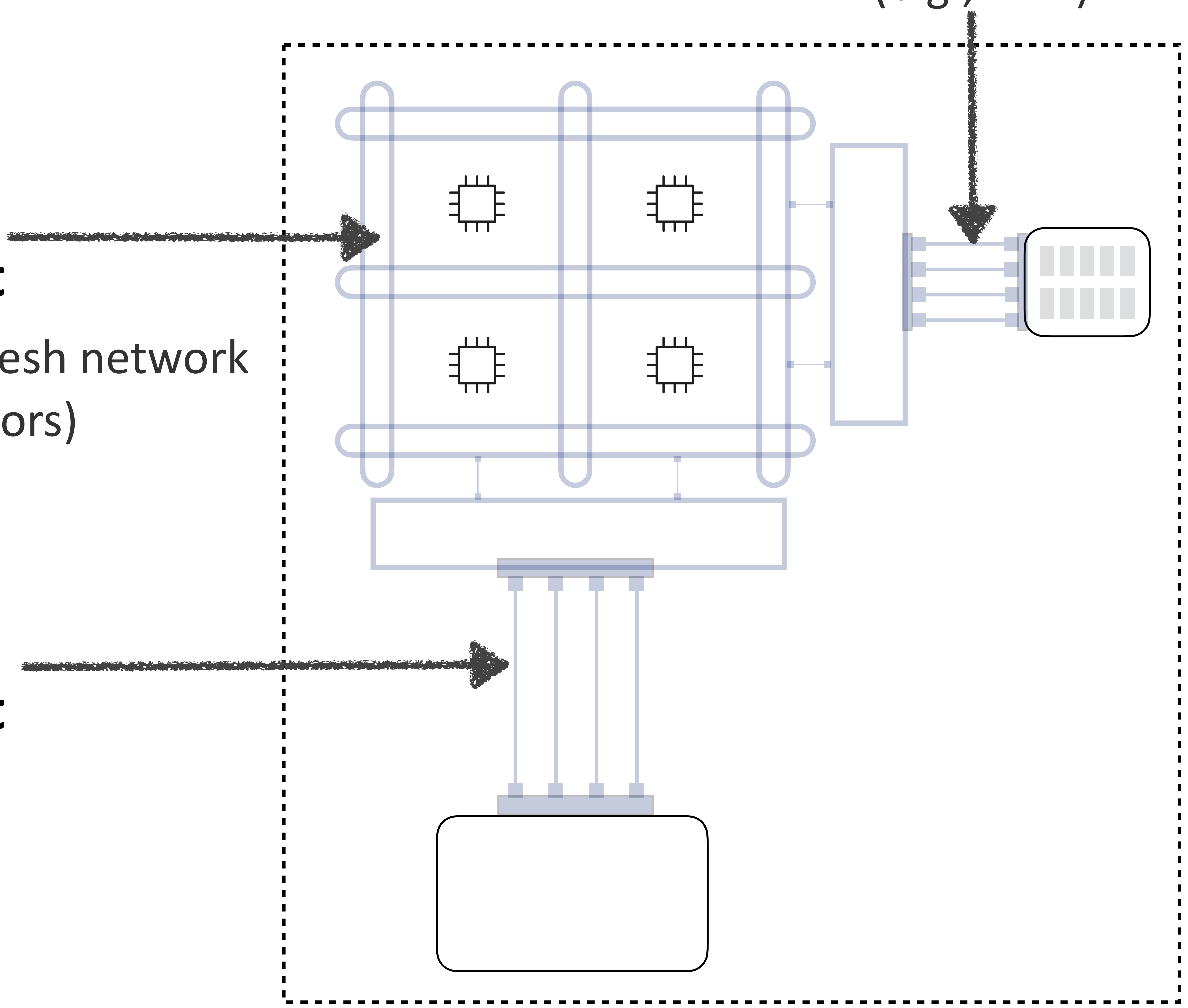
The Host Network: An inter-network within a host



Different devices

CPUs, Peripherals (e.g. NICs, SSDs), Memory

The Host Network: An inter-network within a host



Memory Interconnect

(e.g., DDR)

Processor Interconnect

(e.g., on-chip mesh network on Intel processors)

Peripheral Interconnect

(e.g., PCIe)

Different devices

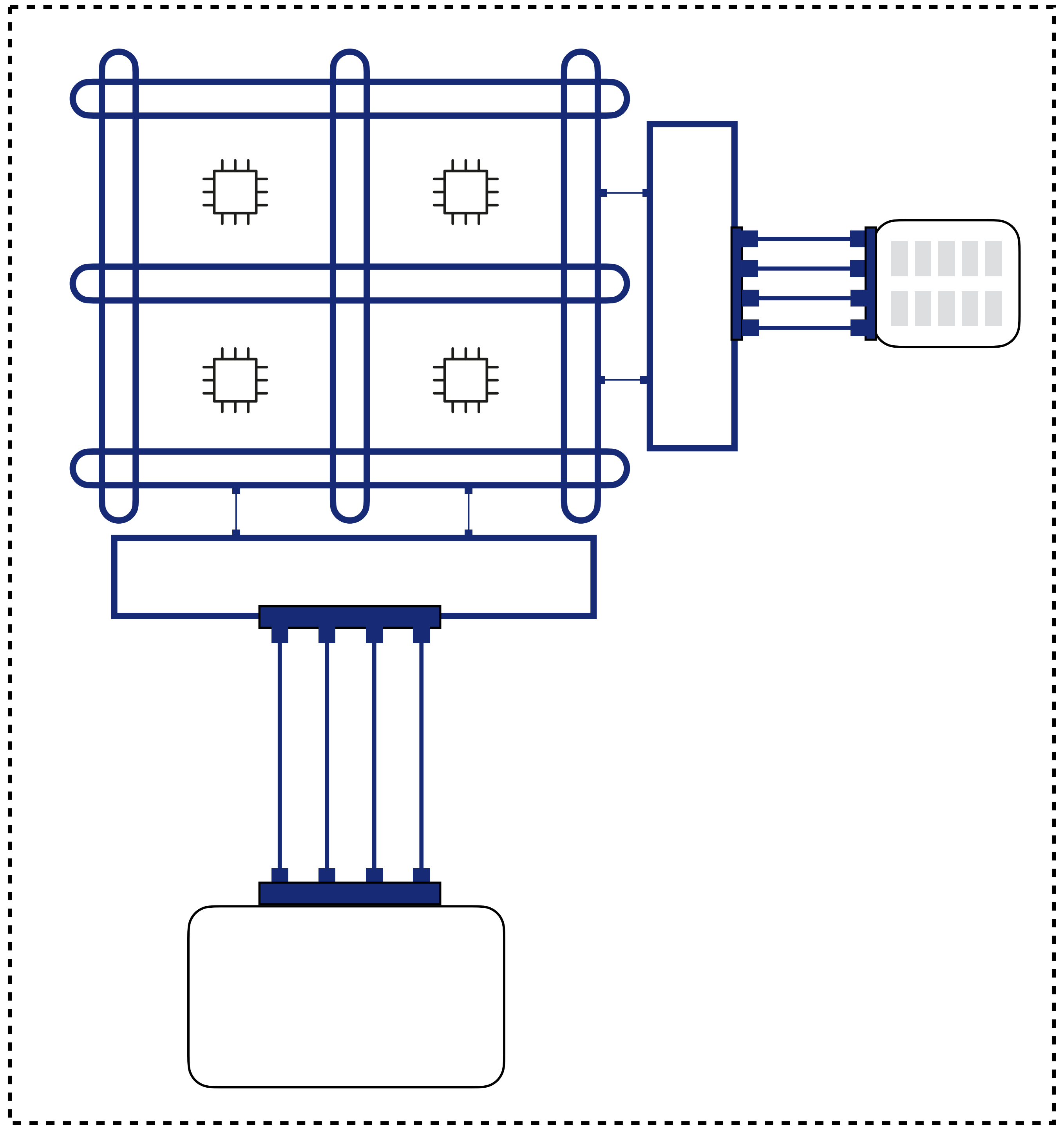
CPUs, Peripherals (e.g. NICs, SSDs), Memory

Different interconnects

Different latency and bandwidth characteristics

Different protocols

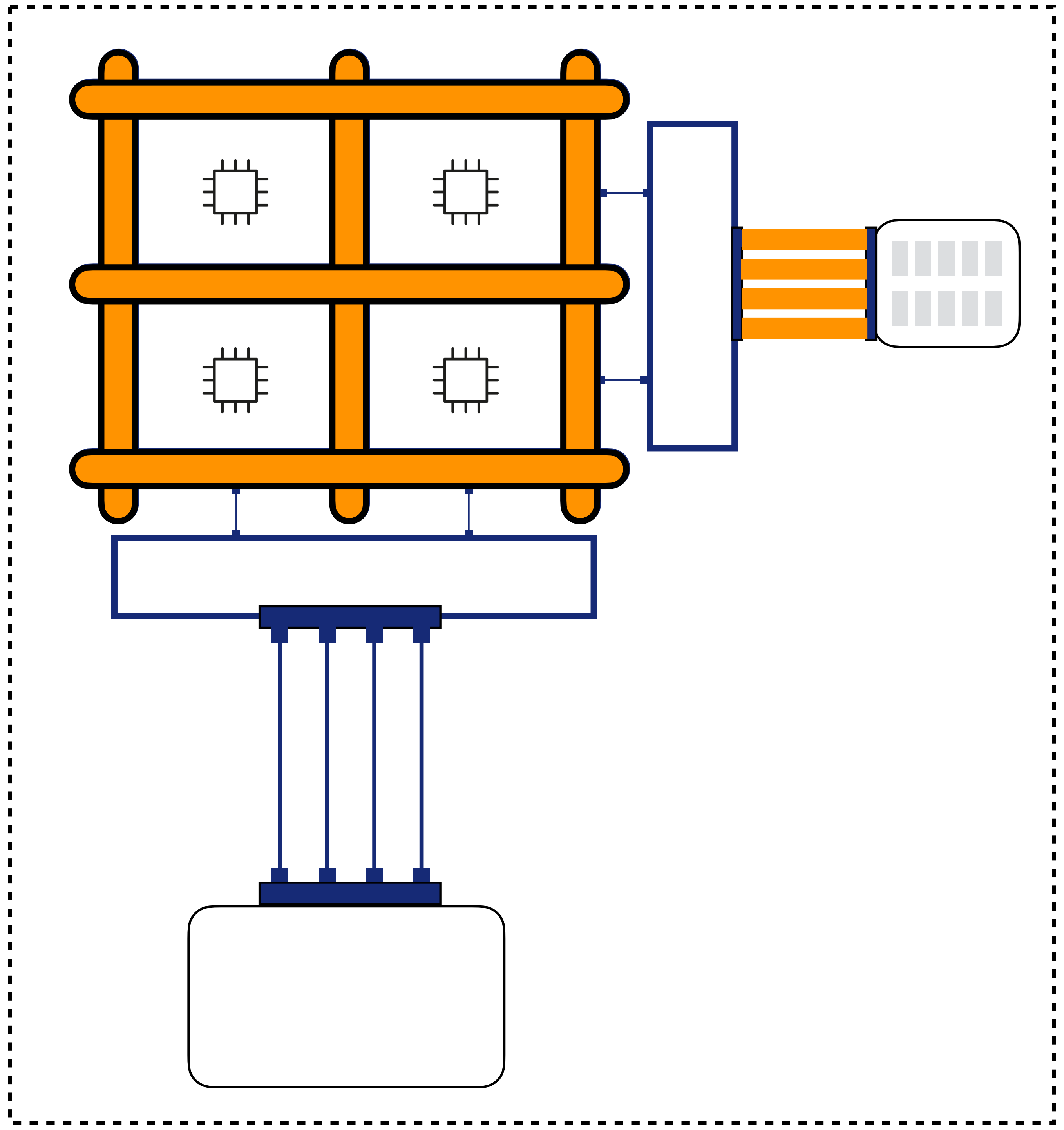
The Host Network: Example data transfers



The Host Network: Example Data Transfers

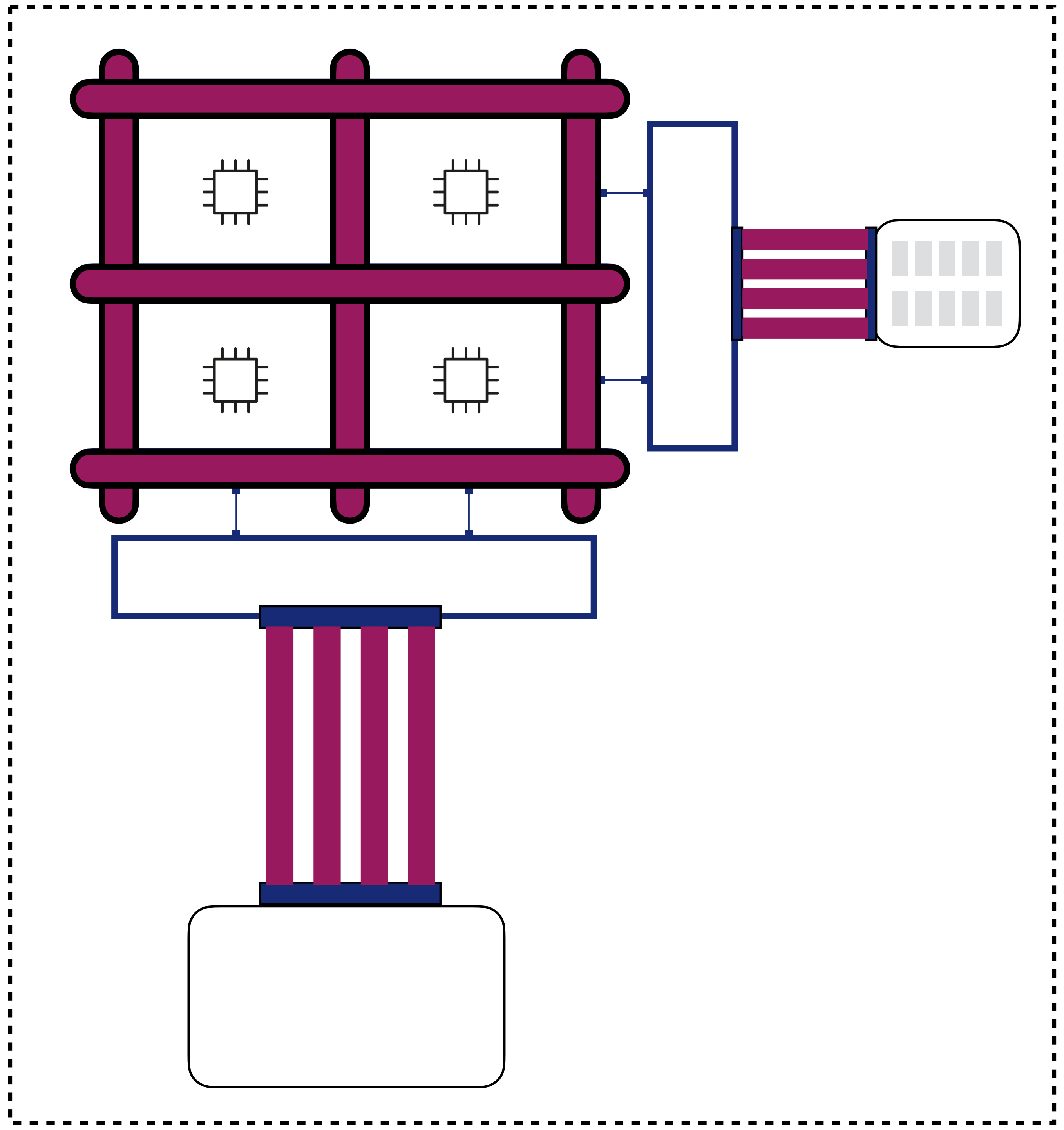
CPU <-> Memory Traffic (C2M)

Traverses processor, memory interconnects

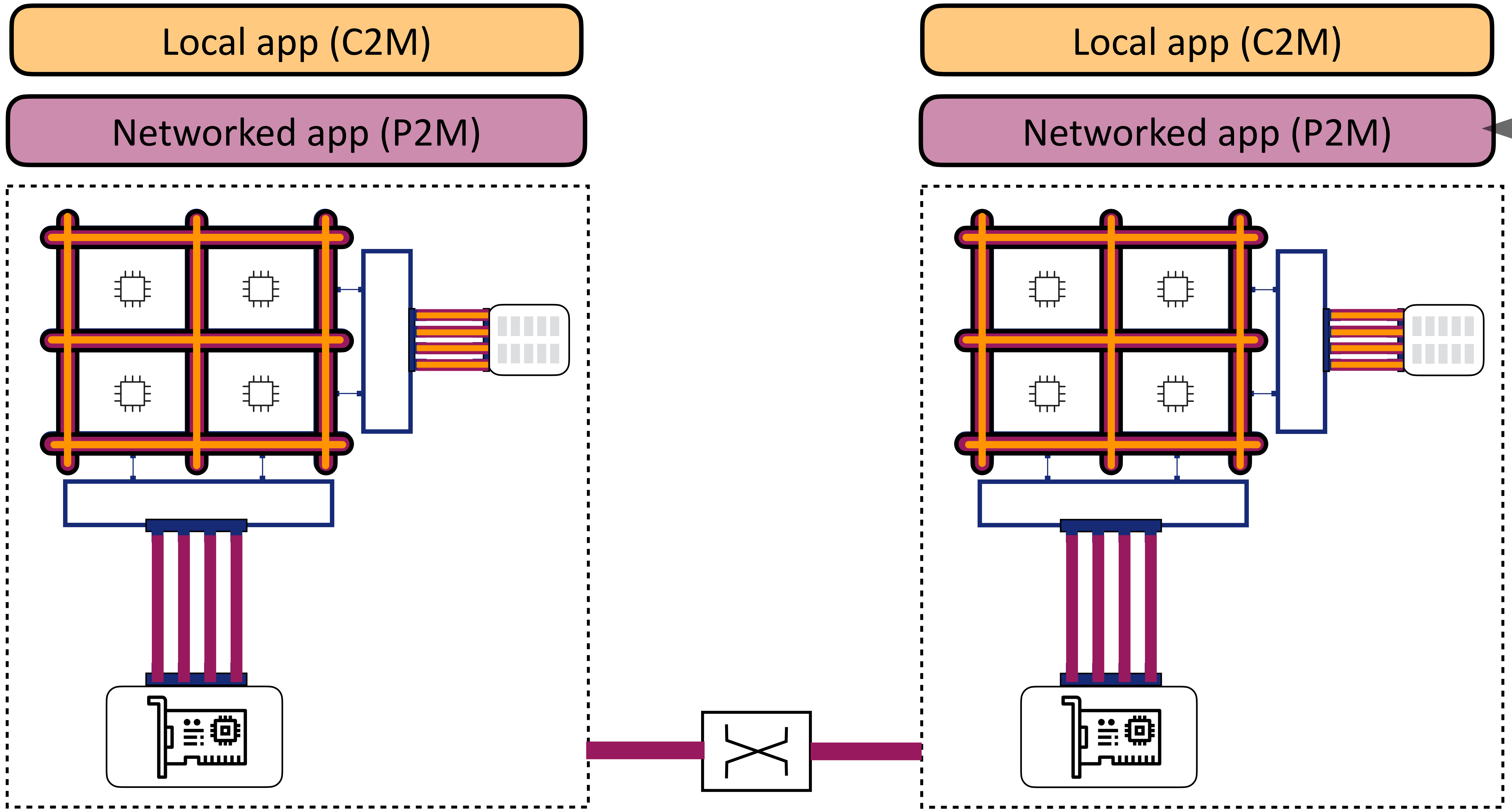


Peripheral <-> Memory Traffic (P2M)

Traverses peripheral, processor, memory interconnects



Host network contention: Impact on networked applications



Local app (C2M)

Networked app (P2M)

Local app (C2M)

Networked app (P2M)

Networked app (P2M) performance suffers
when colocated with Local (C2M) apps

G	Alibaba	ByteDance	hostCC
[HotNets'22]	[FAST'23]	[NSDI'23]	[SIGCOMM'23]

Throughput degradation

Upto 2.2x degradation

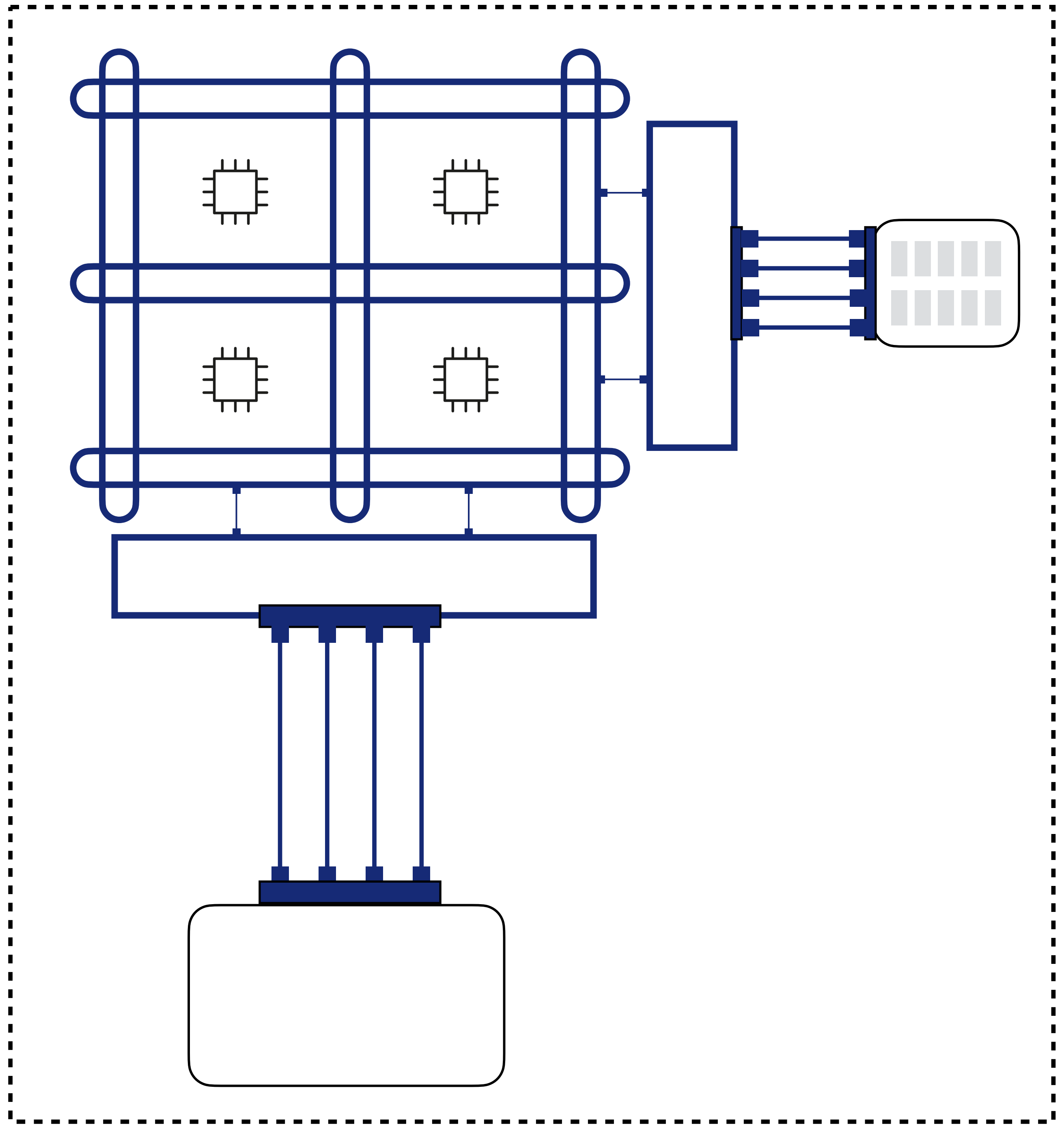
Tail latency inflation

Orders-of-magnitude inflation

Isolation violation

A large fraction of packets dropped at host

Our study: Understanding the Host Network



Building an understanding of the host network contention and its root causes

New, previously unreported, host network contention regimes

Poor interplay between processor, memory and peripheral interconnects

New lens: Conceptual abstraction to study the host network

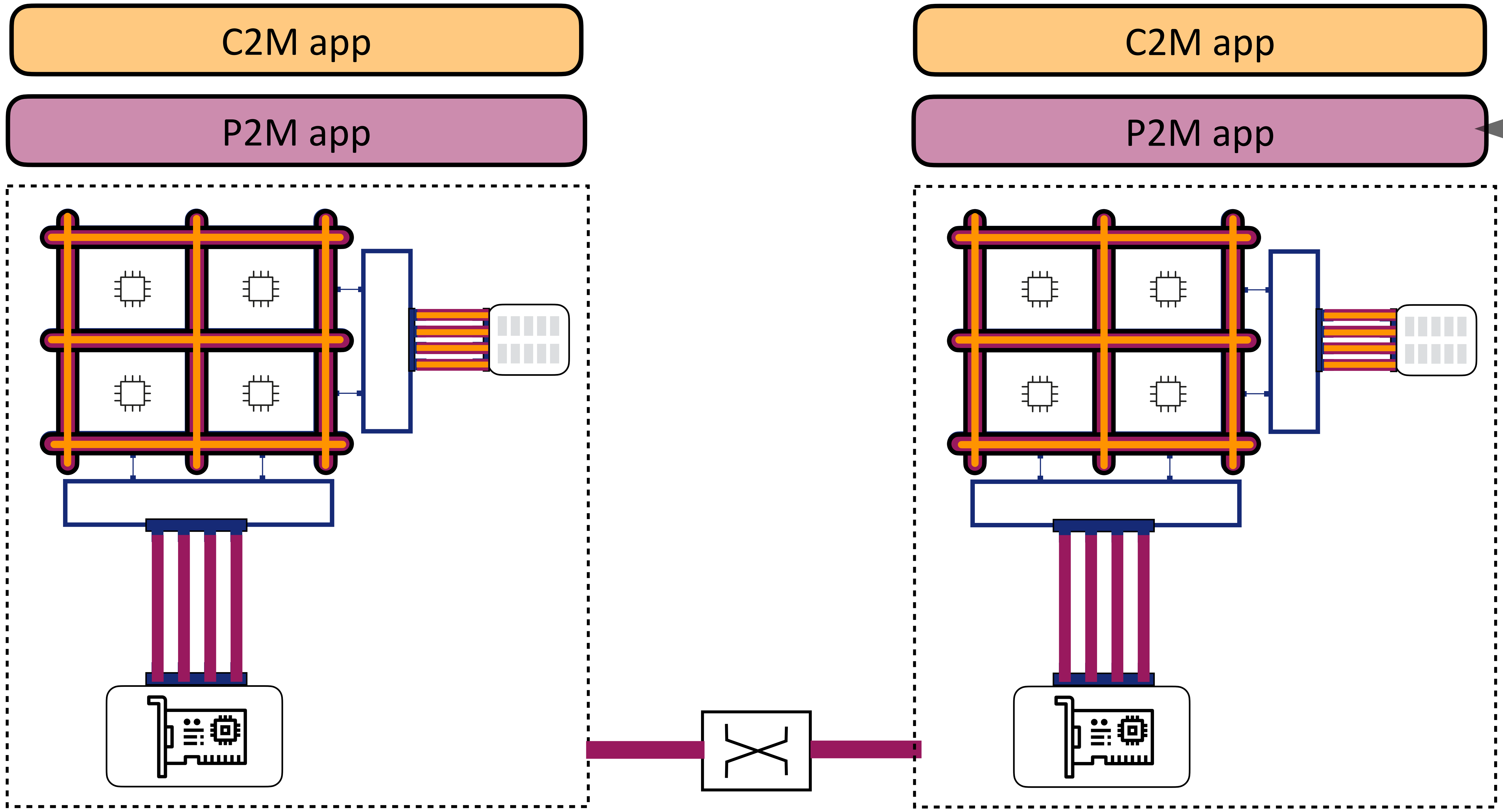
Domain-by-domain credit-based flow control

Captures the subtle interplay between different interconnects

Host network as a standalone network

All our results and observations apply even when all traffic is contained within a single host

Host network contention regimes



C2M app

P2M app

C2M app

P2M app

Prior work: P2M app performance suffers when colocated with C2M app

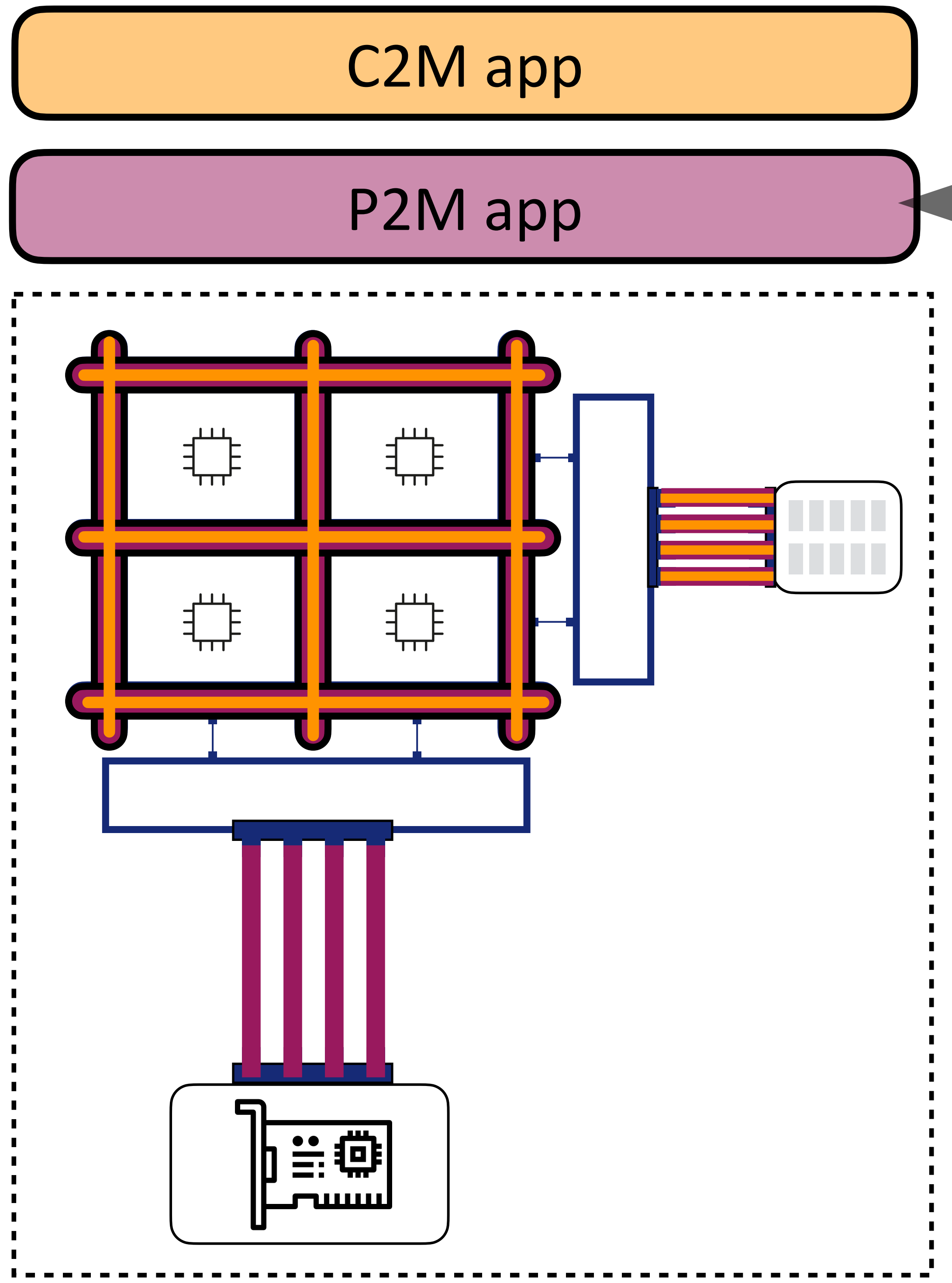
G	Alibaba	ByteDance	hostCC
[HotNets’22]	[FAST’23]	[NSDI’23]	[SIGCOMM’23]

Throughput degradation

Tail latency inflation

Isolation violation

Host network contention regimes



Prior work: P2M app performance suffers when colocated with C2M app

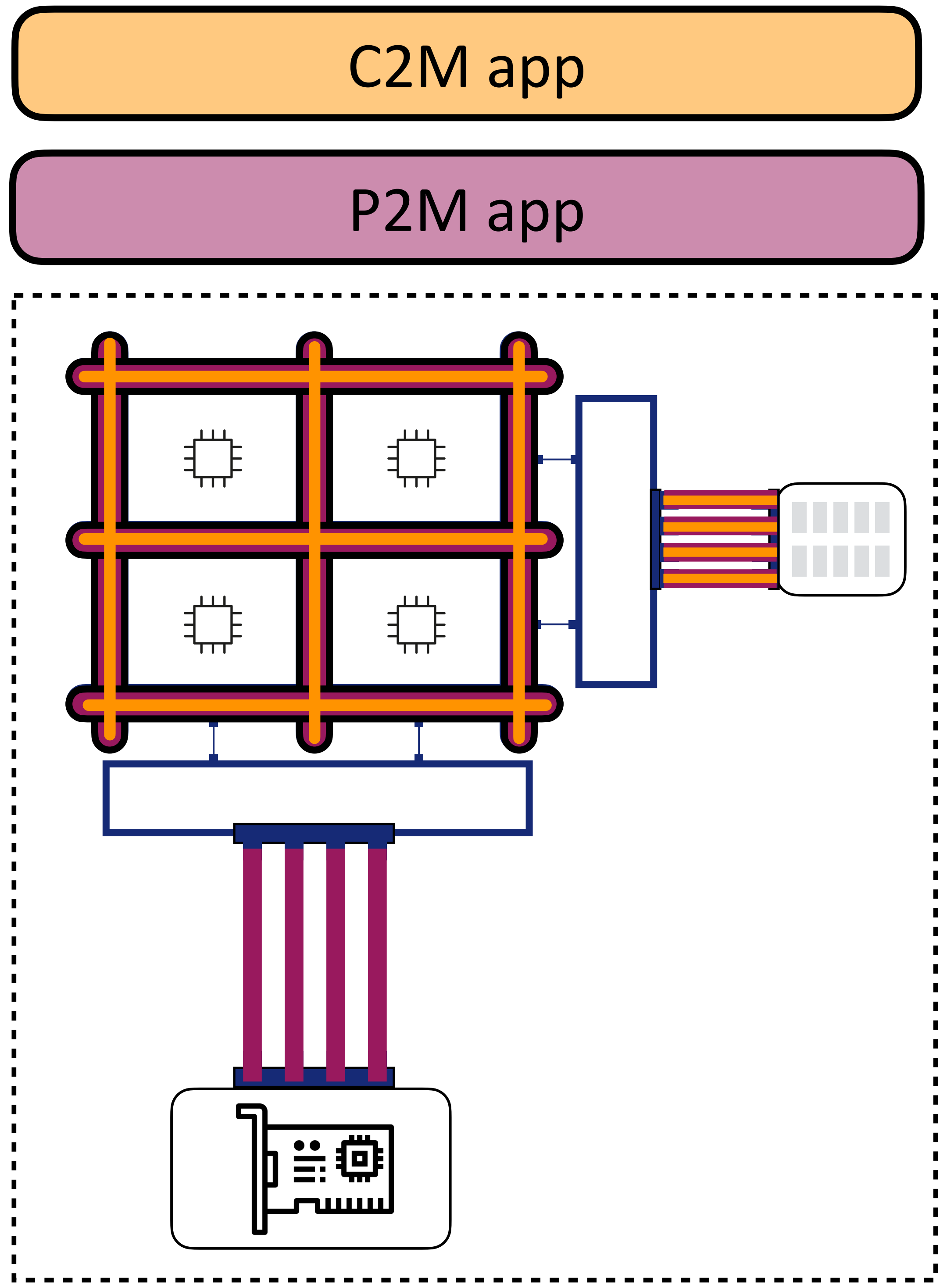
Google [HotNets'22]   Alibaba [FAST'23]   ByteDance [NSDI'23]   hostCC [SIGCOMM'23]

Throughput degradation

Tail latency inflation

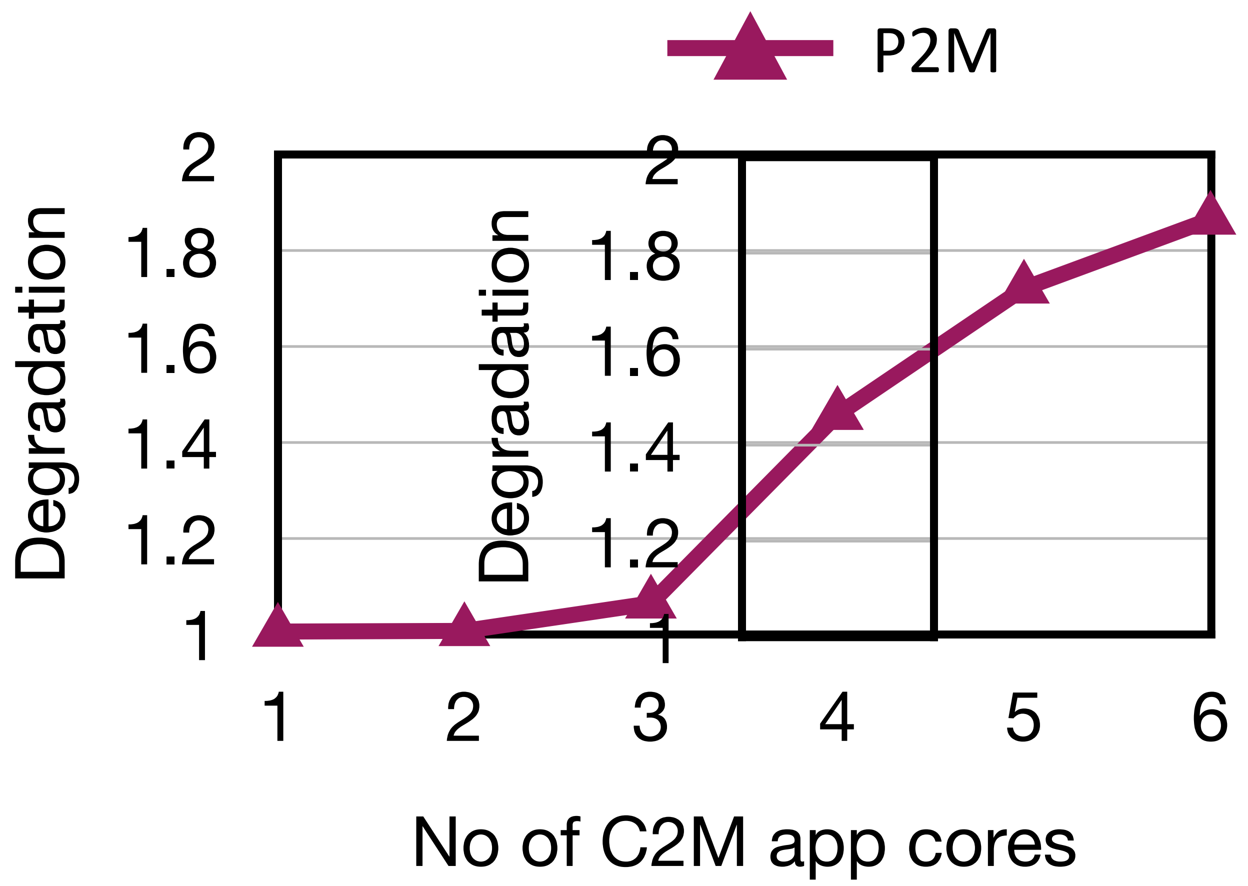
Isolation violation

Host network contention regimes

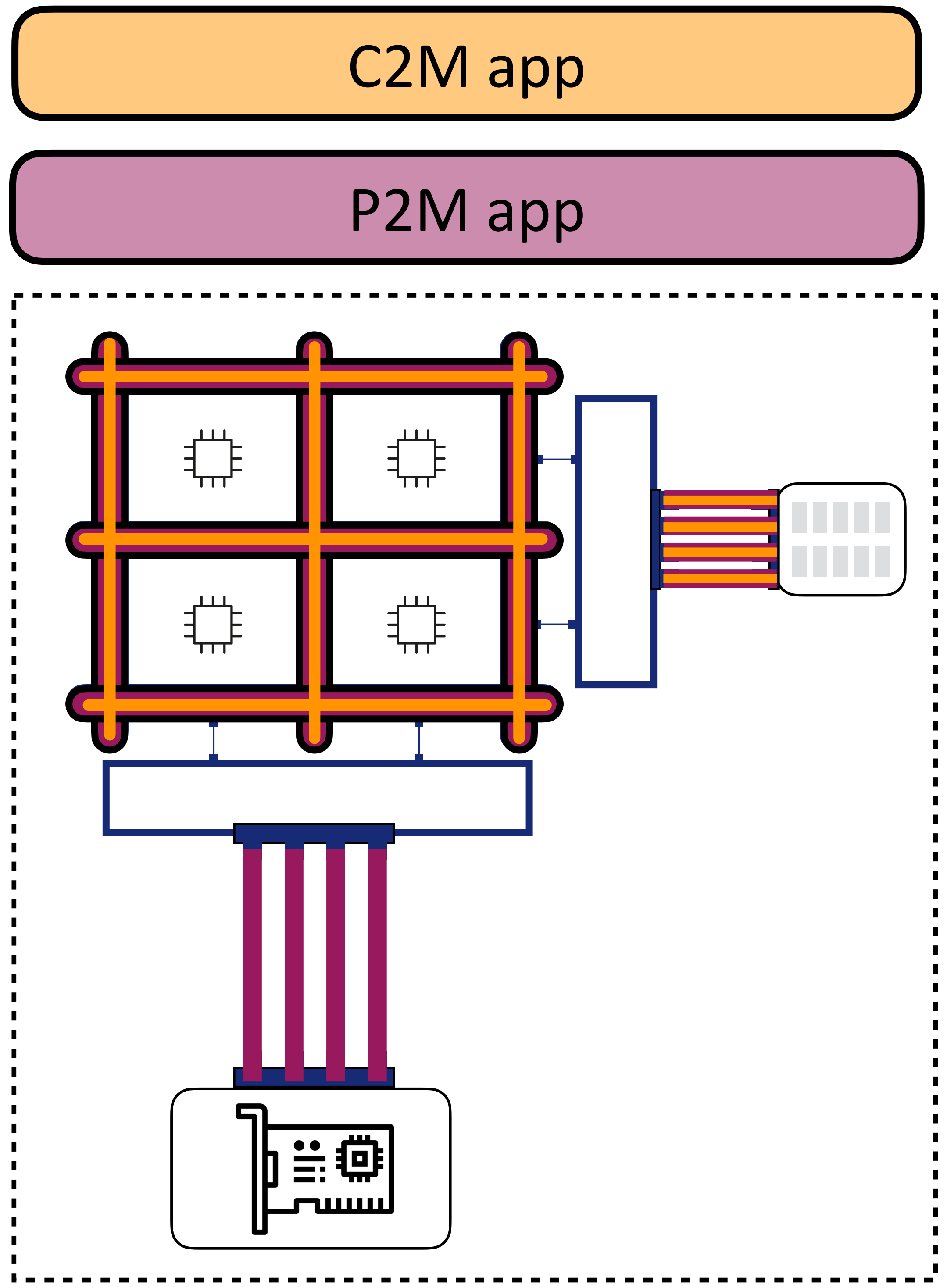


Prior work: P2M app performance suffers when colocated with C2M app

$$\text{Degradation} = \frac{\text{Throughput (isolated)}}{\text{Throughput (colocated)}}$$

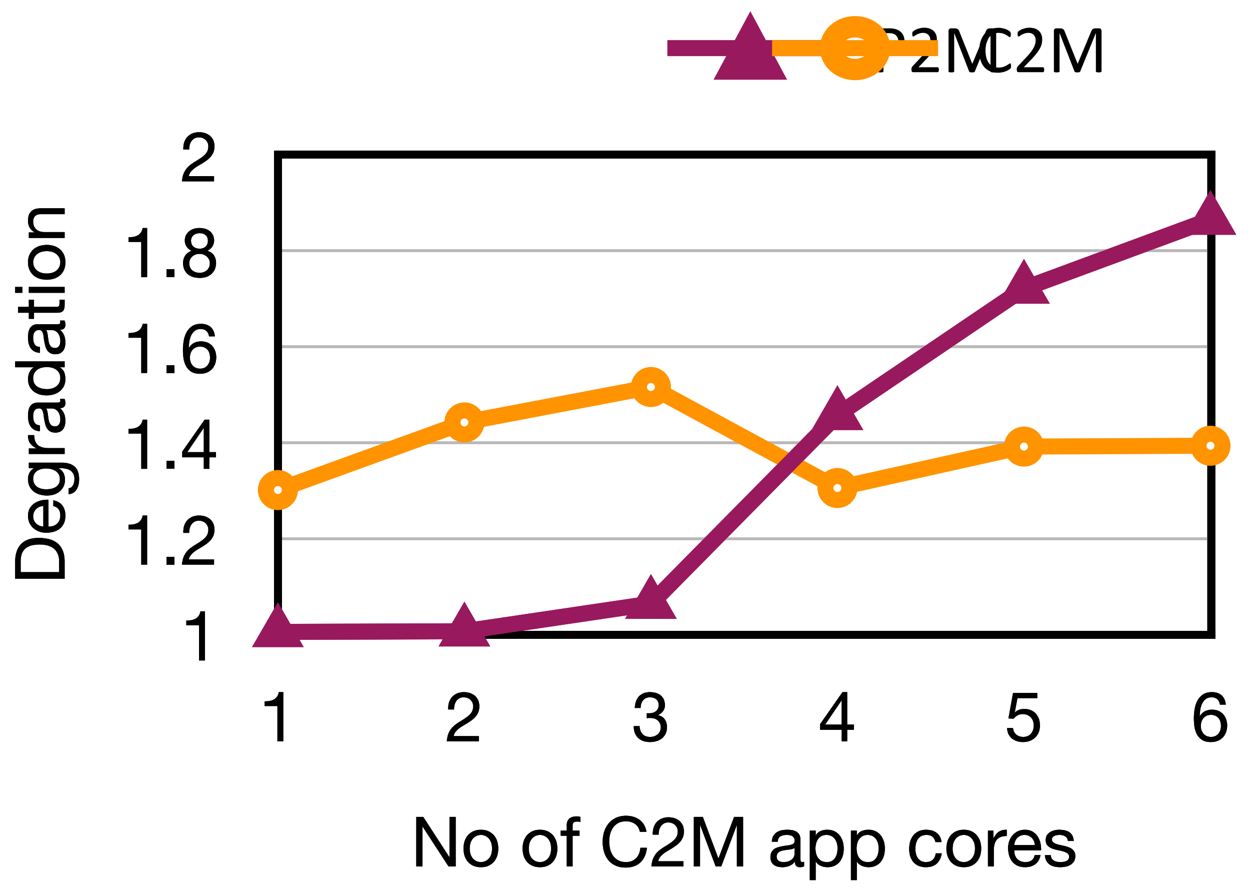


Host network contention impacts both C2M and P2M apps

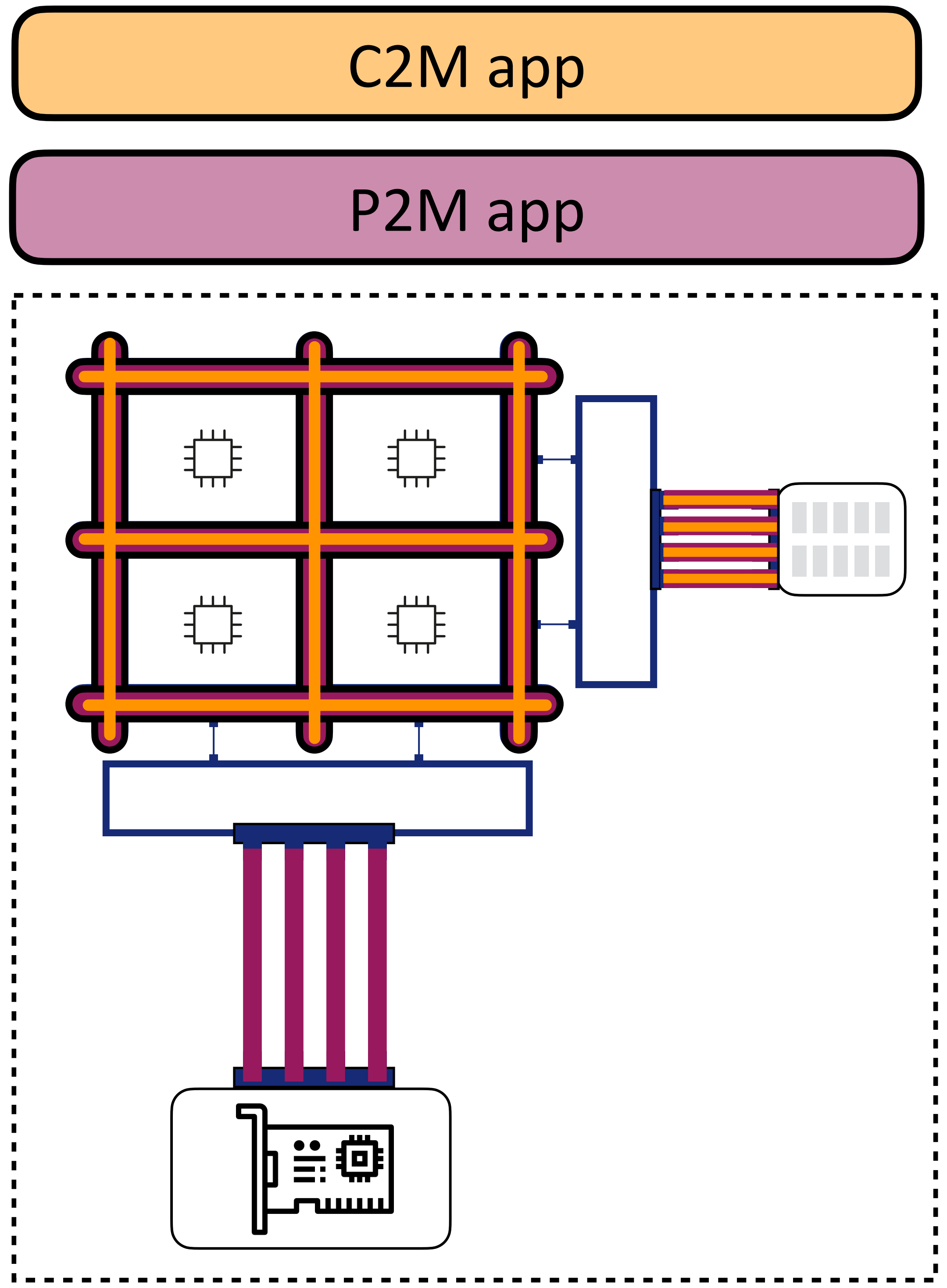


Prior work: P2M app performance suffers when colocated with C2M app

Observation #1: C2M app performance also suffers

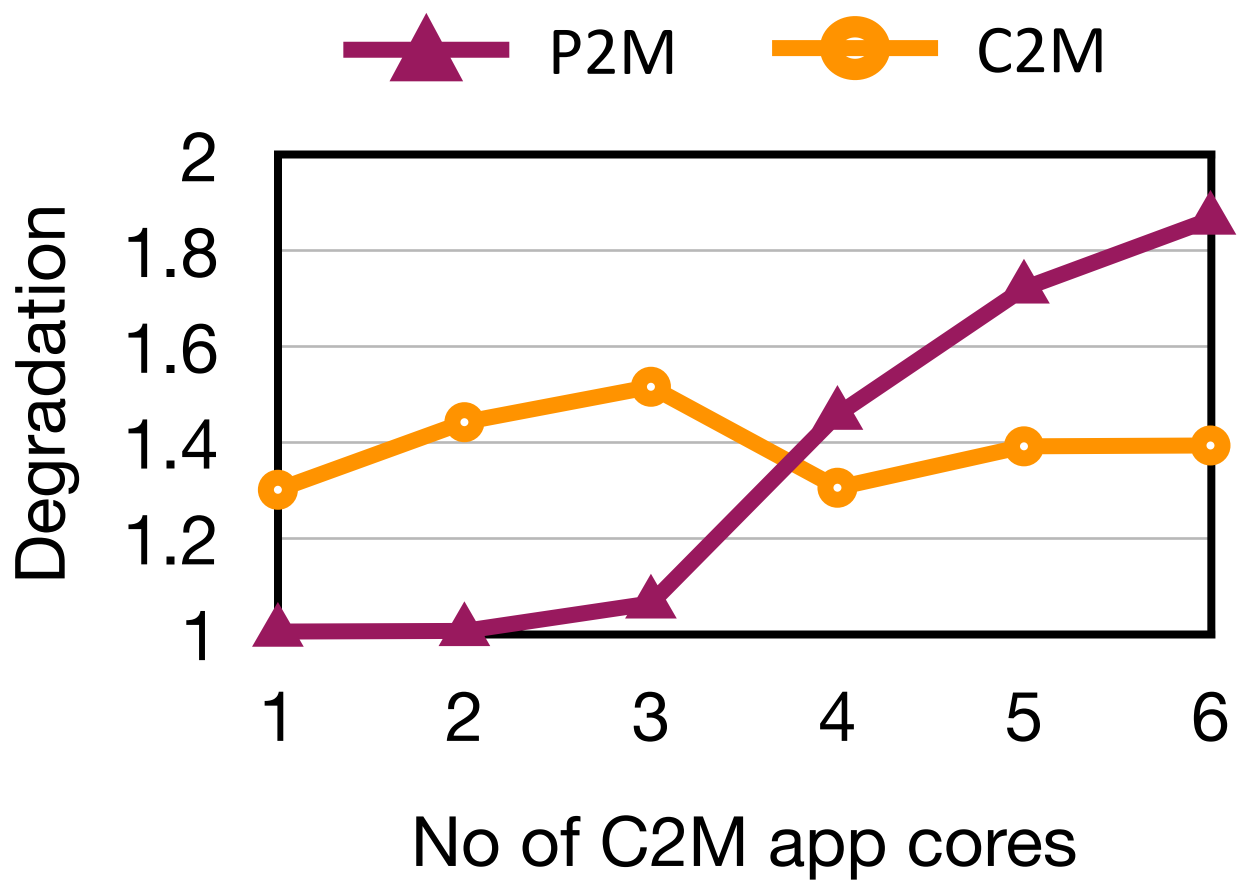


Host network contention: The full picture

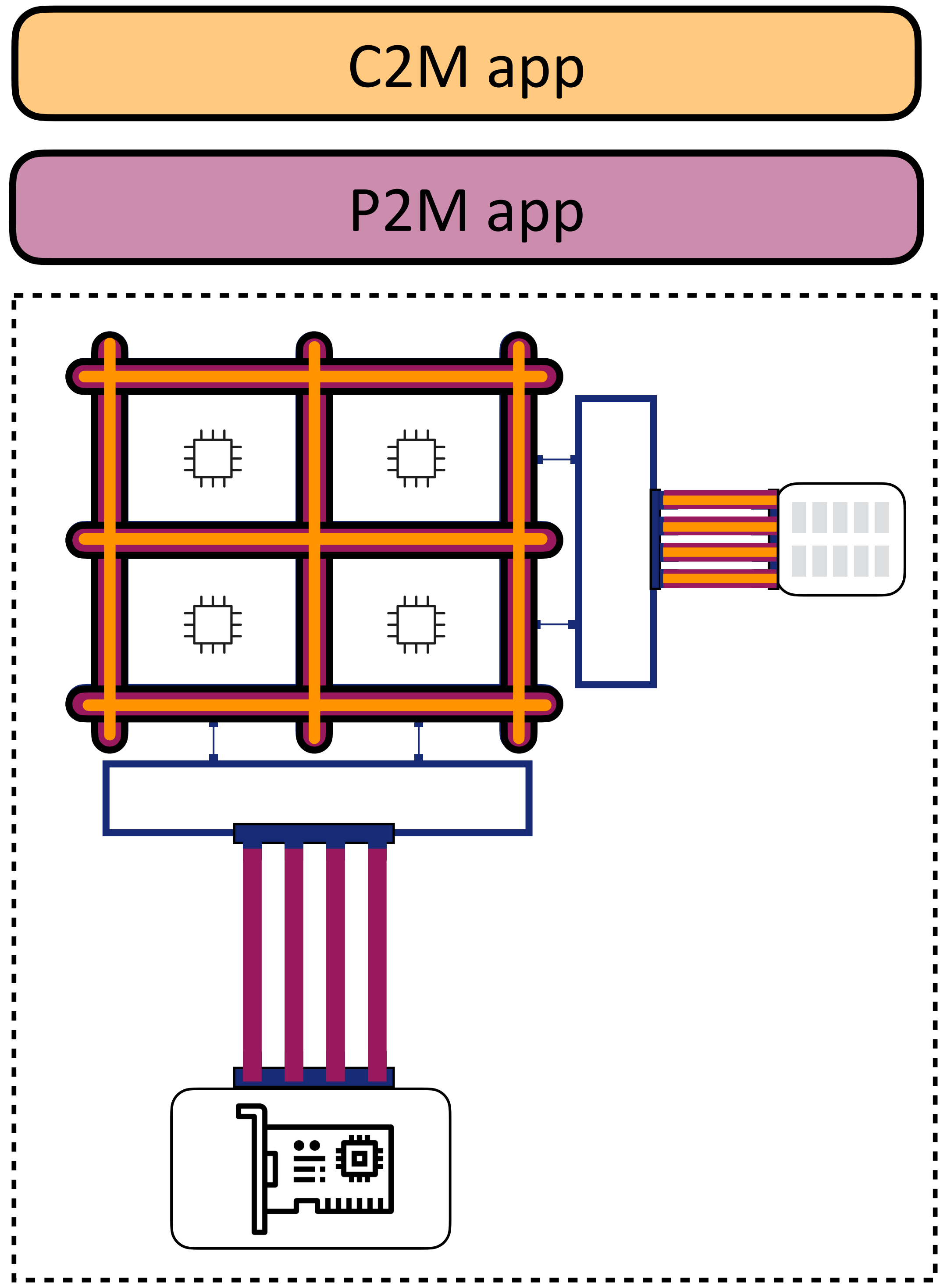


Observation #1: C2M app performance also suffers

Observation #2: (in most cases) P2M app causes severe degradation for C2M app



Host network contention: The full picture

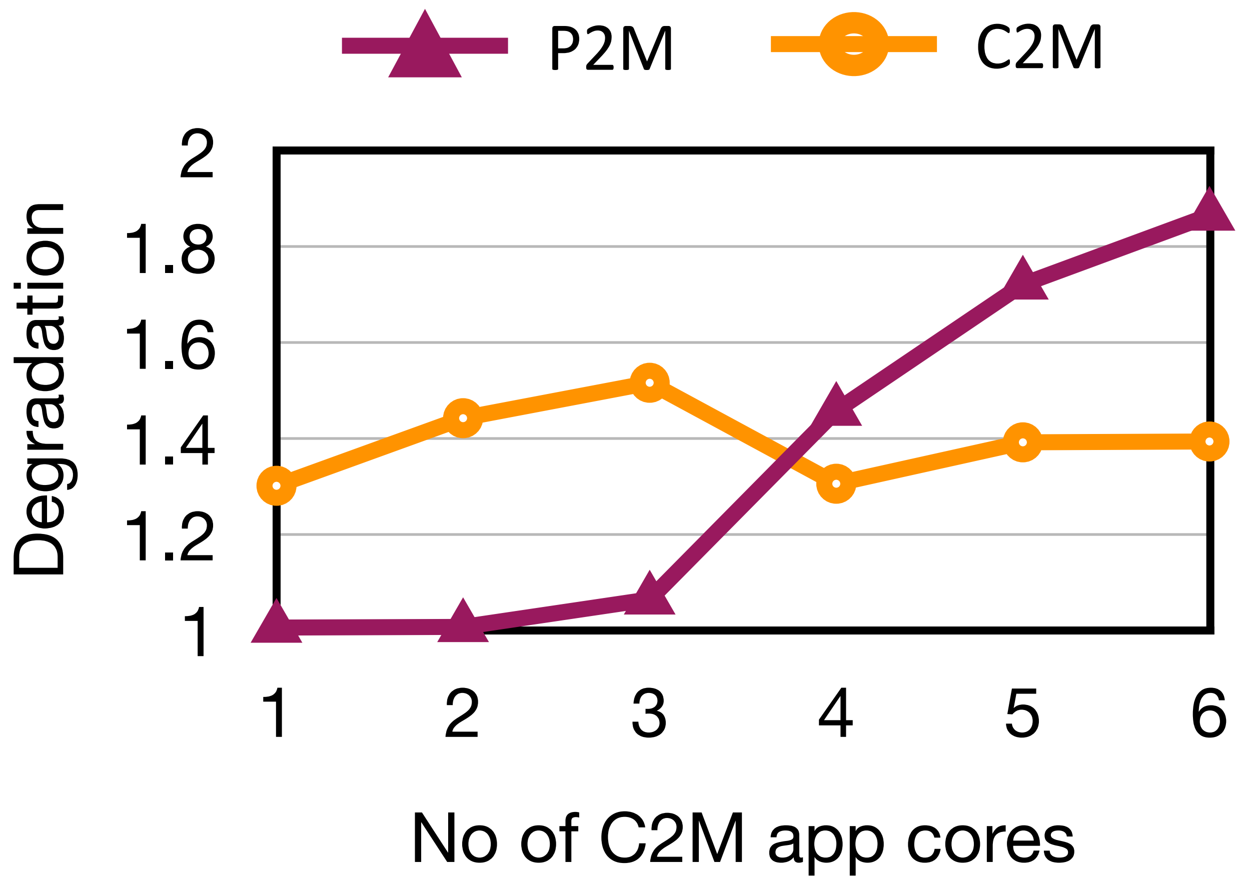


Observation #1: C2M app performance also suffers

Observation #2: (in most cases) P2M app causes severe degradation for C2M app

P2M Write

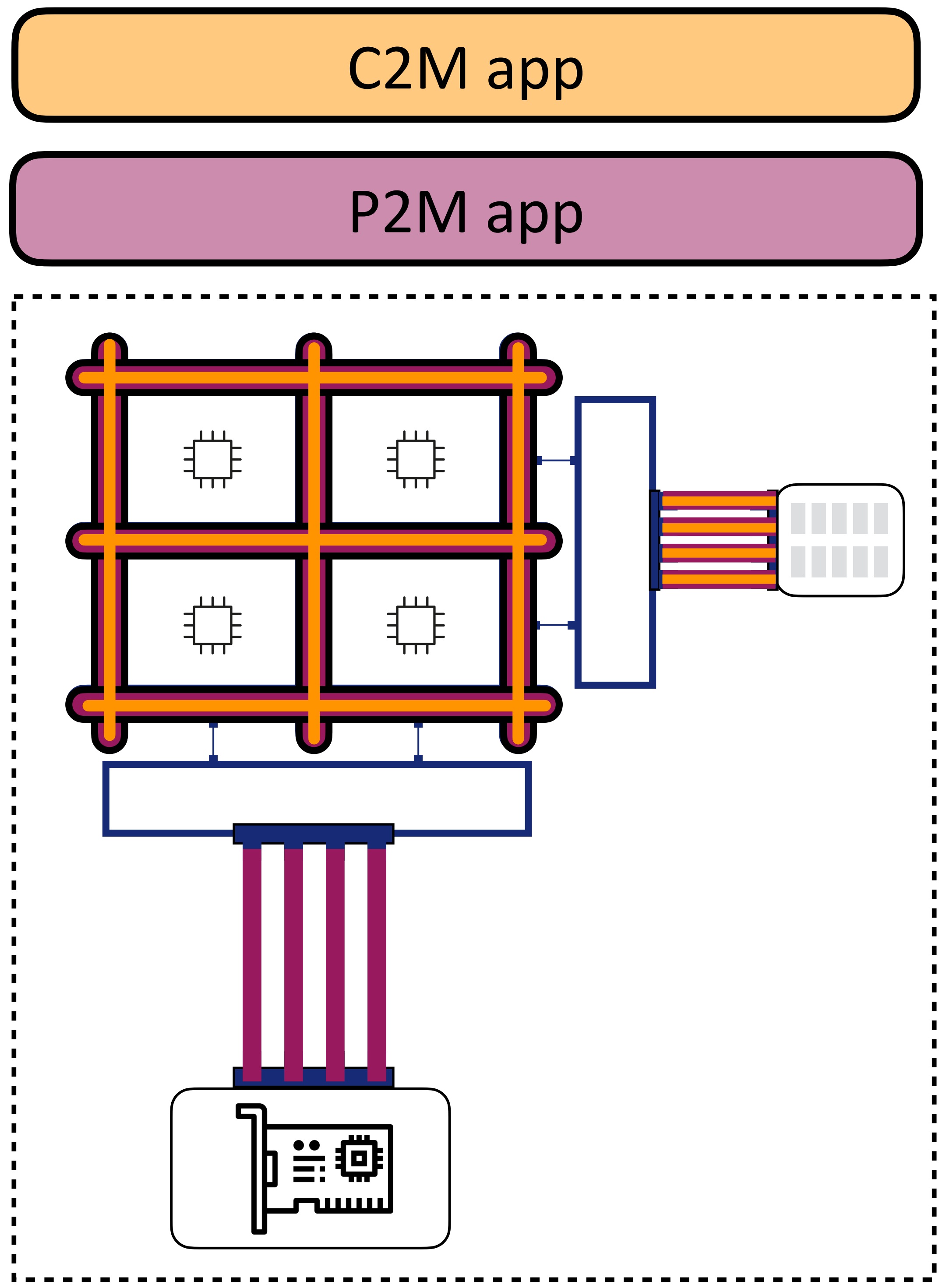
C2M Write

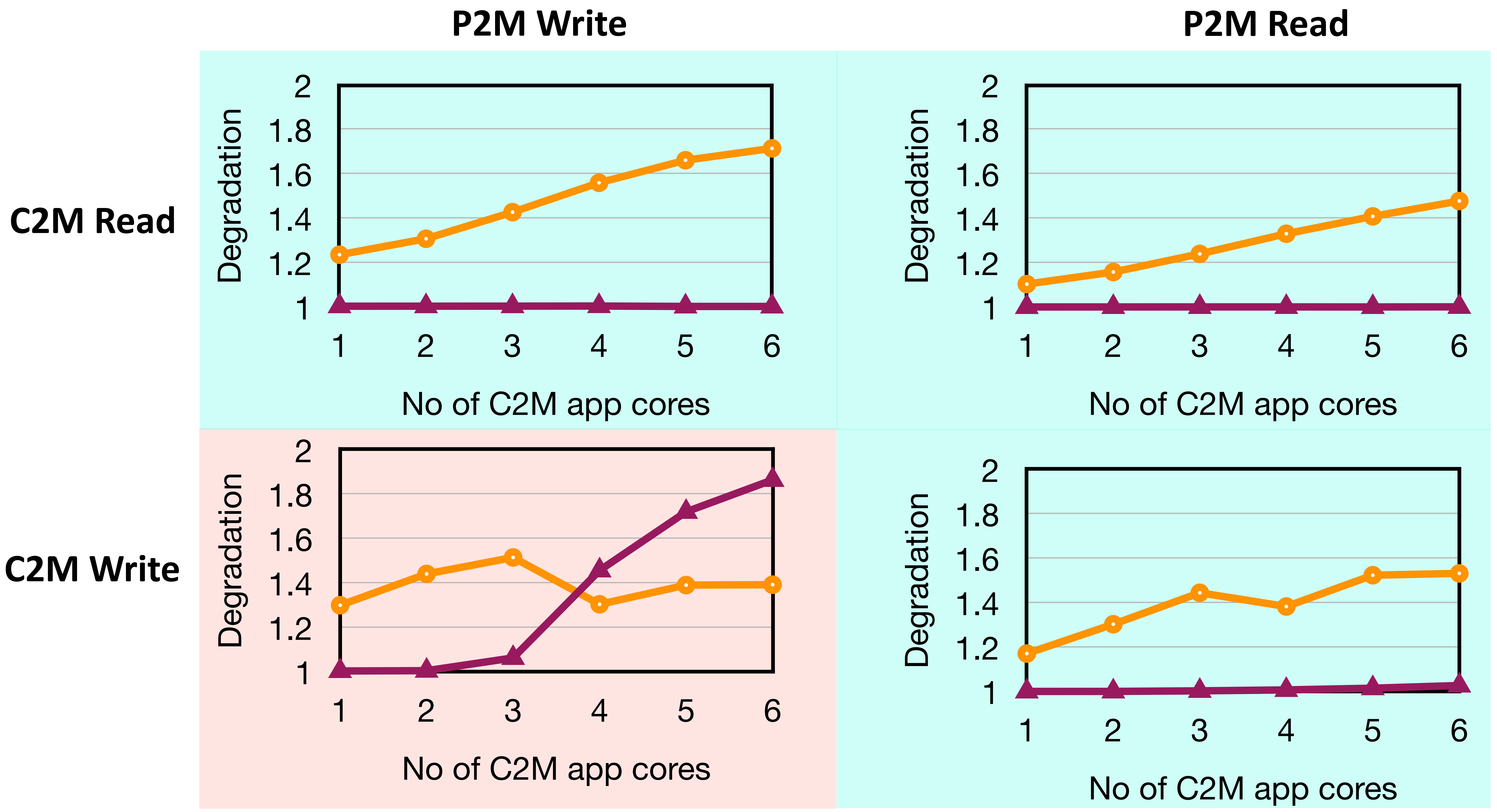


Host network contention: The full picture

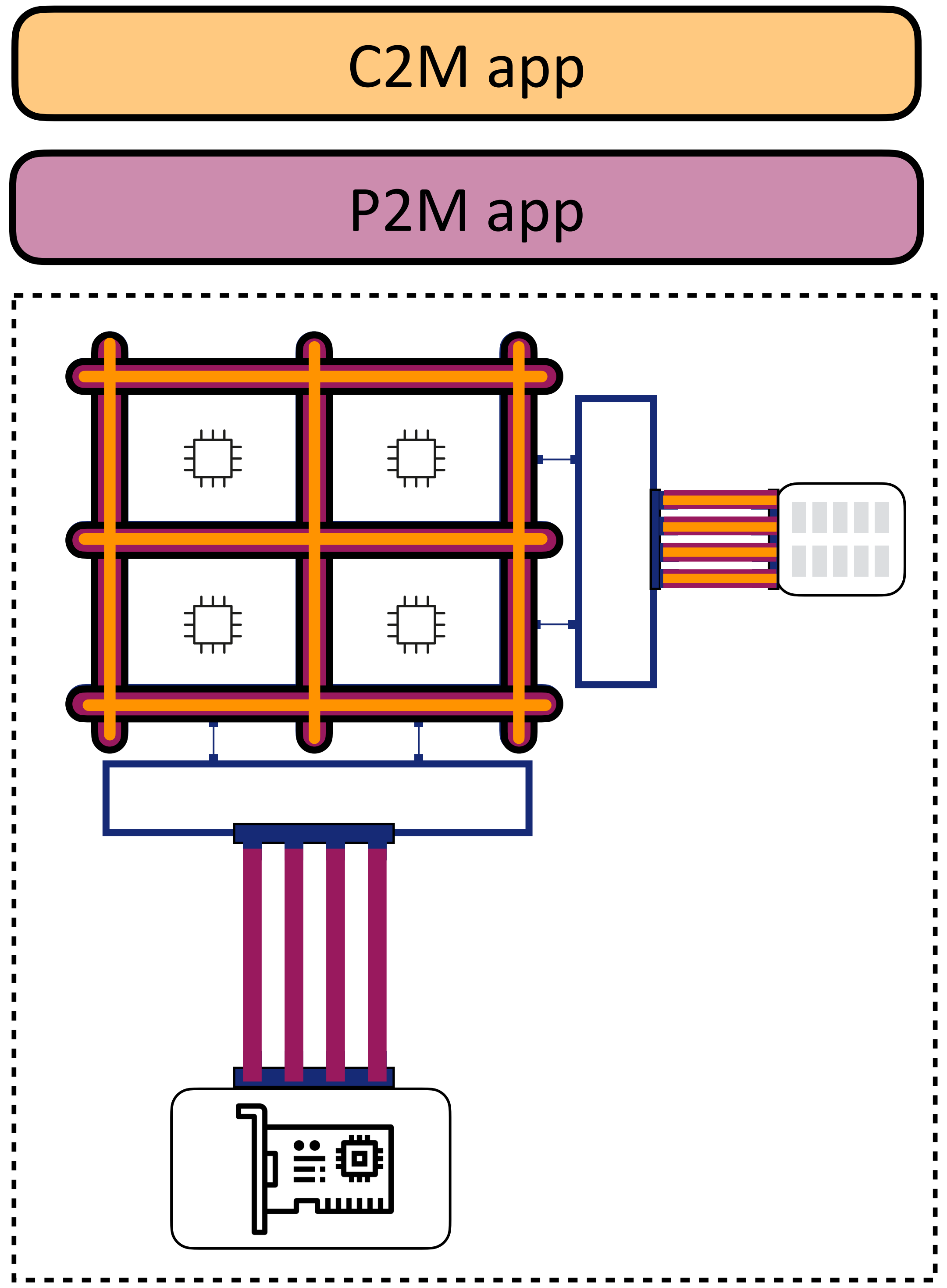
Observation #1: C2M app performance also suffers

Observation #2: (in most cases) P2M app causes severe degradation for C2M app





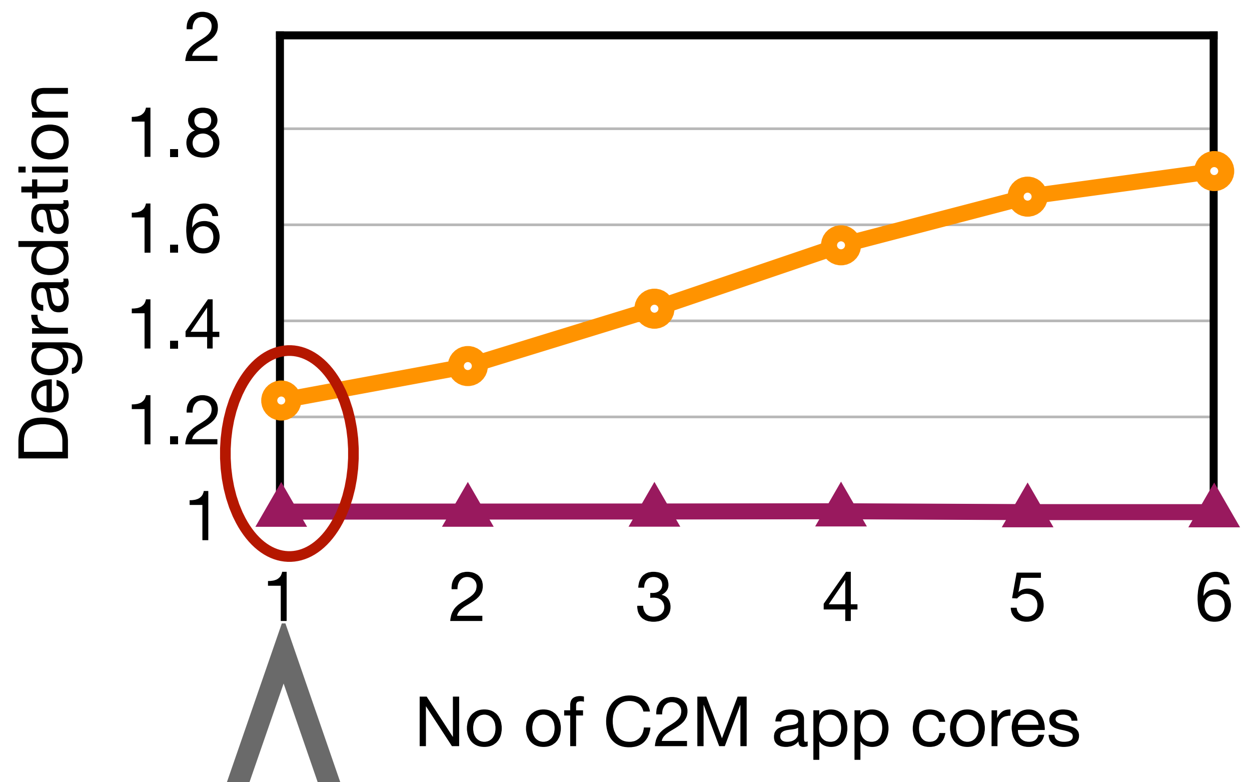
Host network contention: Not merely due to limited resources



Observation #1: C2M app performance also suffers

Observation #2: (in most cases) P2M app causes severe degradation for C2M app

Observation #3: Performance degrades even when resources are not bottlenecked



CPU resources are isolated

Shared interconnects:

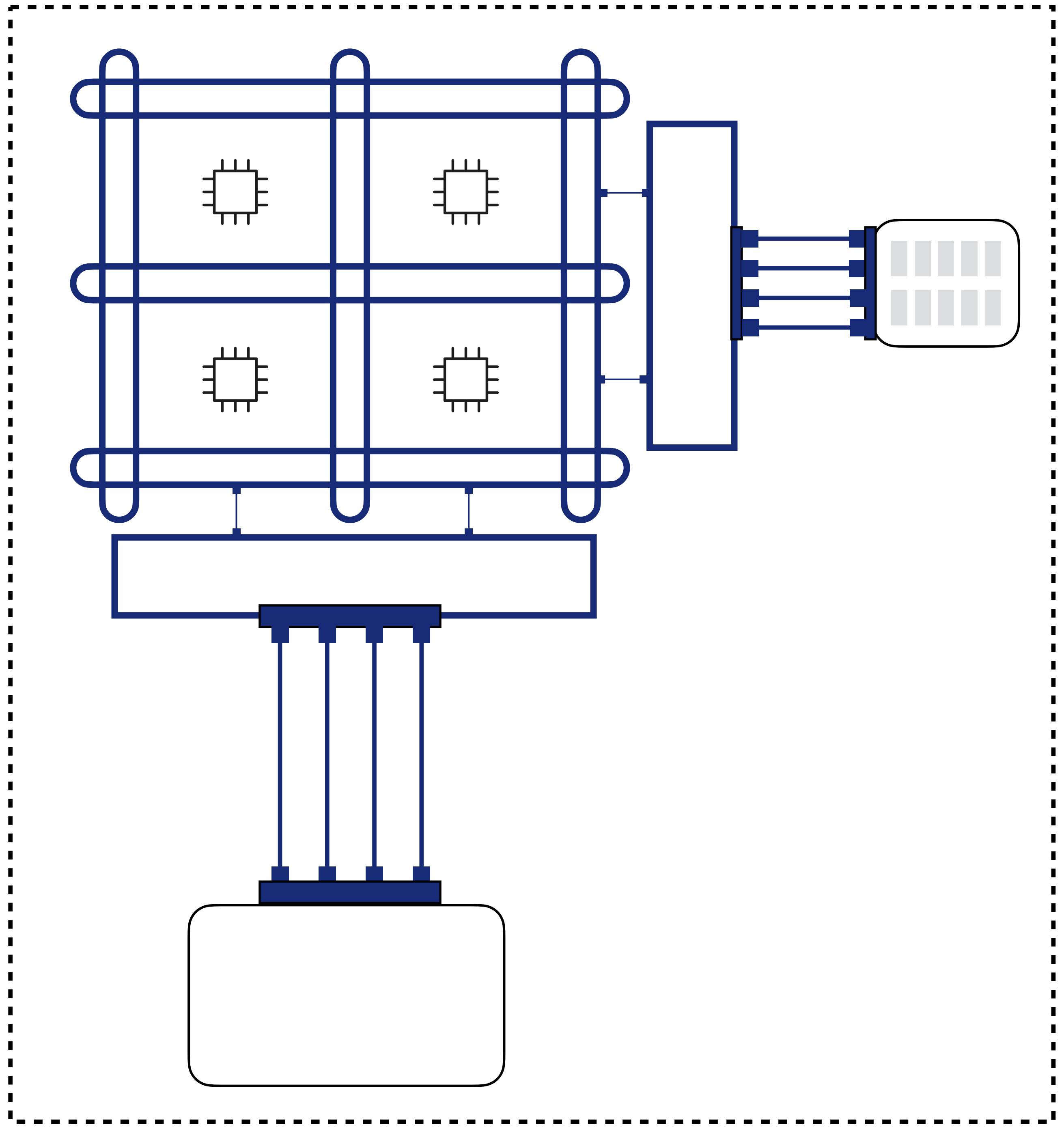
Processor interconnect < 10% bandwidth utilization

Memory interconnect 42% bandwidth utilization

Yet performance degrades!

Host network contention is rooted in interplay between processor, memory, peripheral interconnects

Our study: Understanding the Host Network



Building an understanding of the host network contention and its root causes

New, previously unreported, host network contention regimes

Poor interplay between processor, memory and peripheral interconnects

New lens: Conceptual abstraction to study the host network

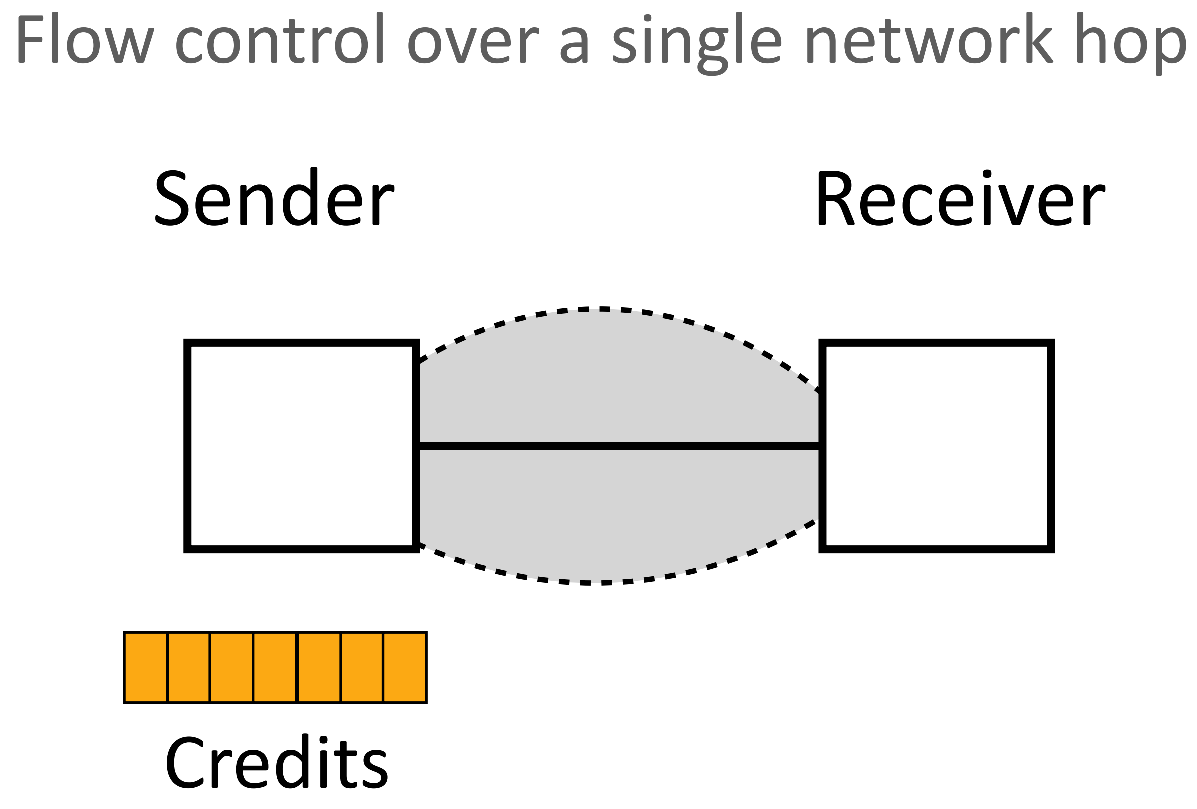
Domain-by-domain credit-based flow control

Captures the subtle interplay between different interconnects

Host network as a standalone network

All our results and observations apply even when all traffic is contained within a single host

(End-to-end) Credit-based Flow control: A brief primer



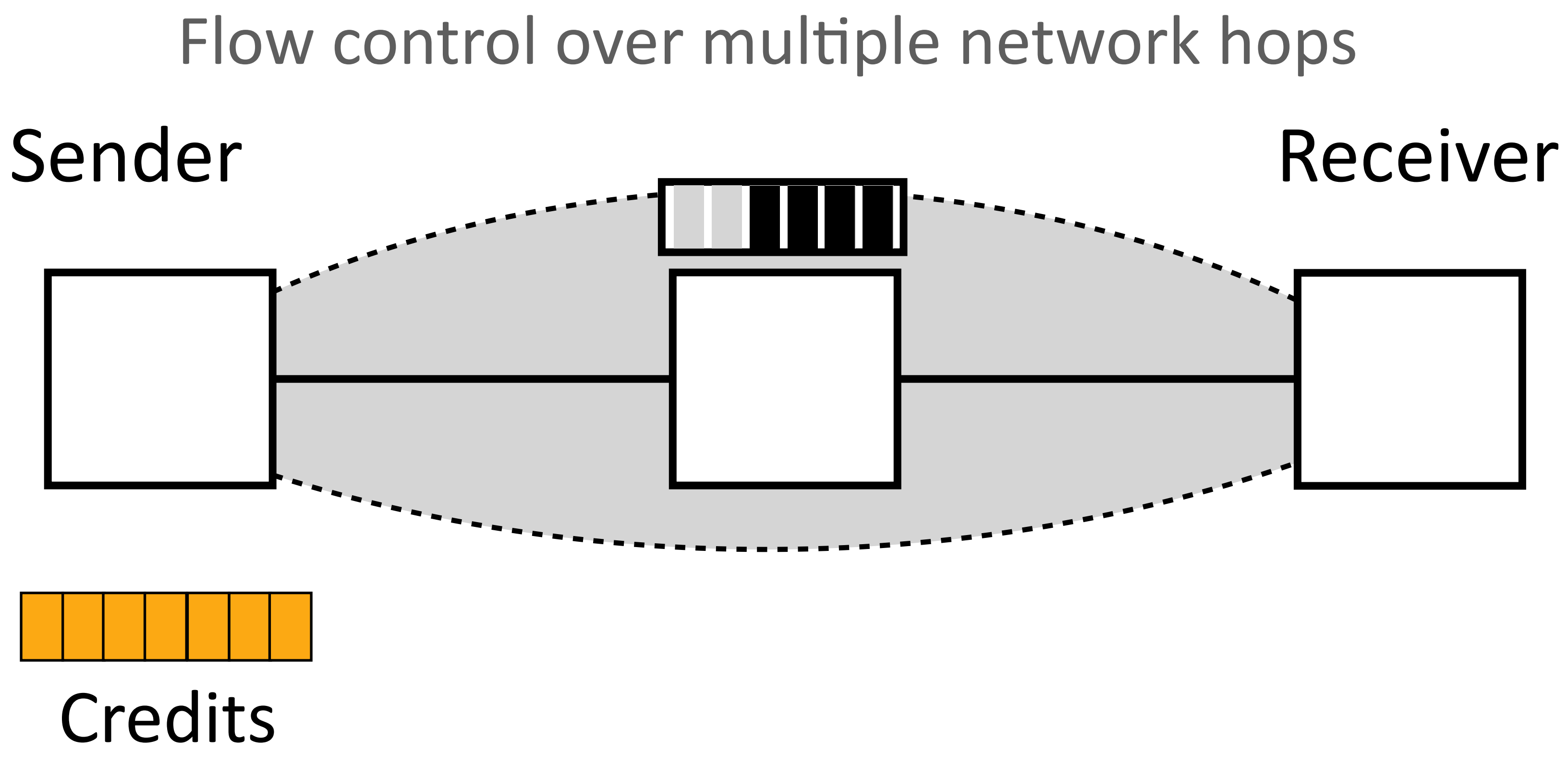
Sender is assigned credits (limits # in-flight requests)

Sender consumes a credit to send a message

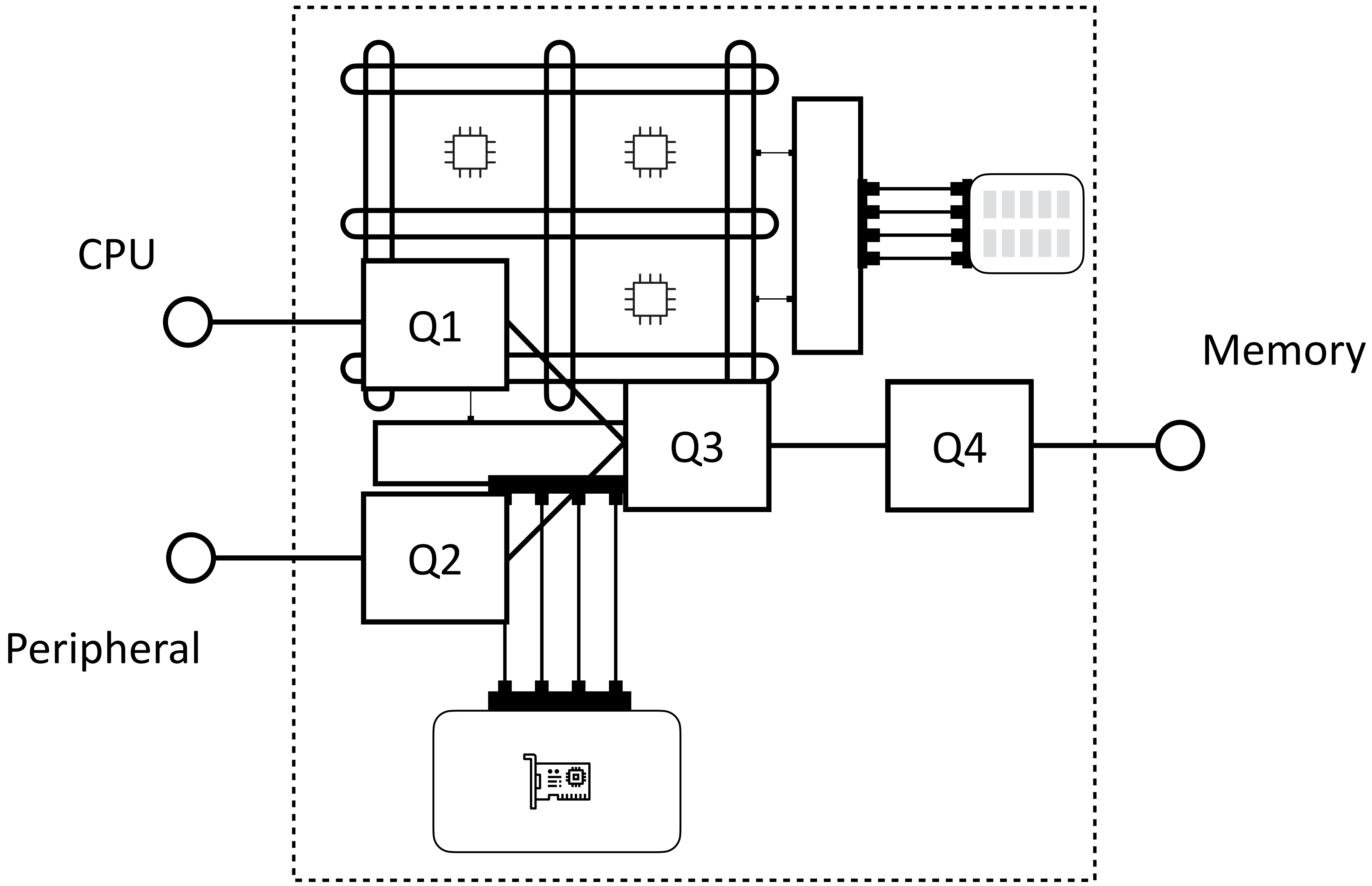
Credit replenished when message receipt is acknowledged by receiver

$$\textbf{Throughput} \leq \frac{\textbf{\# Credits}}{\textbf{Latency}}$$

(Latency: Time between credit allocation and replenishment)



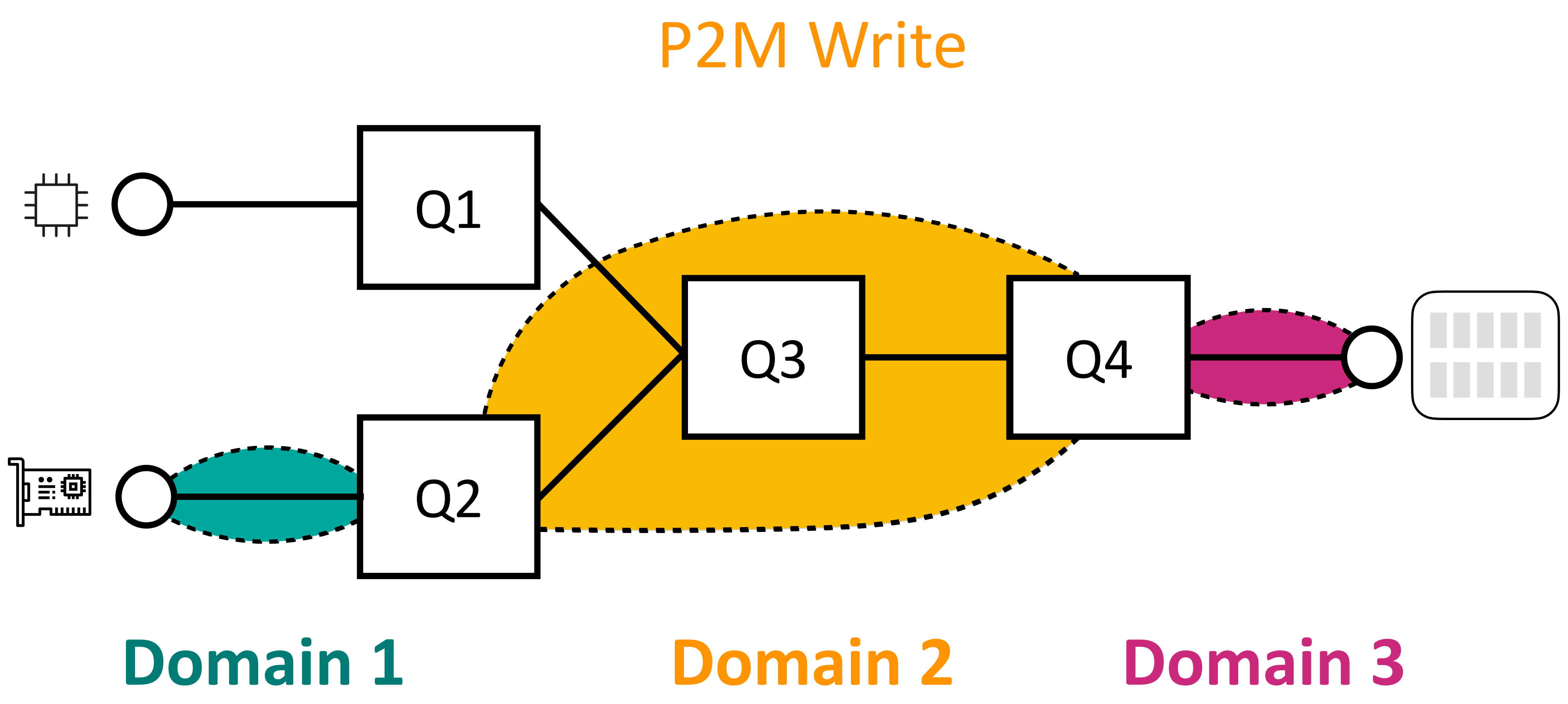
An abstract representation of the host network



Host network nodes: potential queueing points

Domain-by-domain credit-based flow control

Domains: Sub-networks of host network



Different domains: Different credits and different latency

Credits: Peripheral Interconnect (PCIe) credits

Latency: Peripheral <-> Q2

Credits: Q2 buffer size

Latency: Q2 <-> Q4

Credits: Q4 buffer size

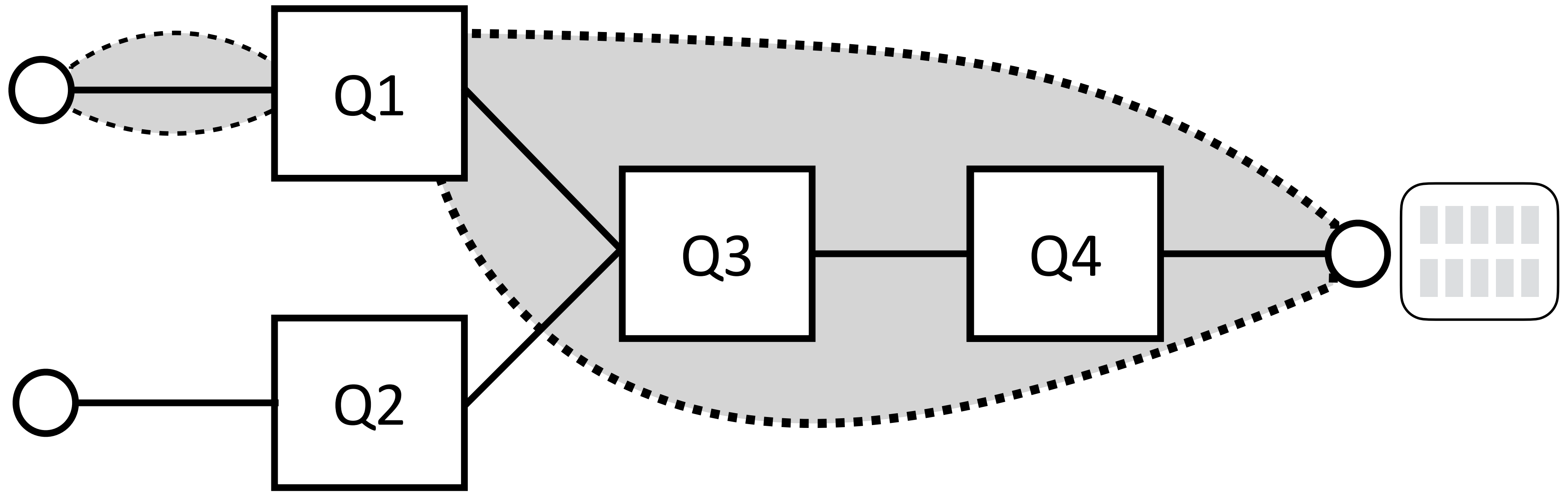
Latency: Q4 <-> Memory

End-to-end throughput = min (Domain 1 throughput, Domain 2 throughput, Domain 3 throughput)

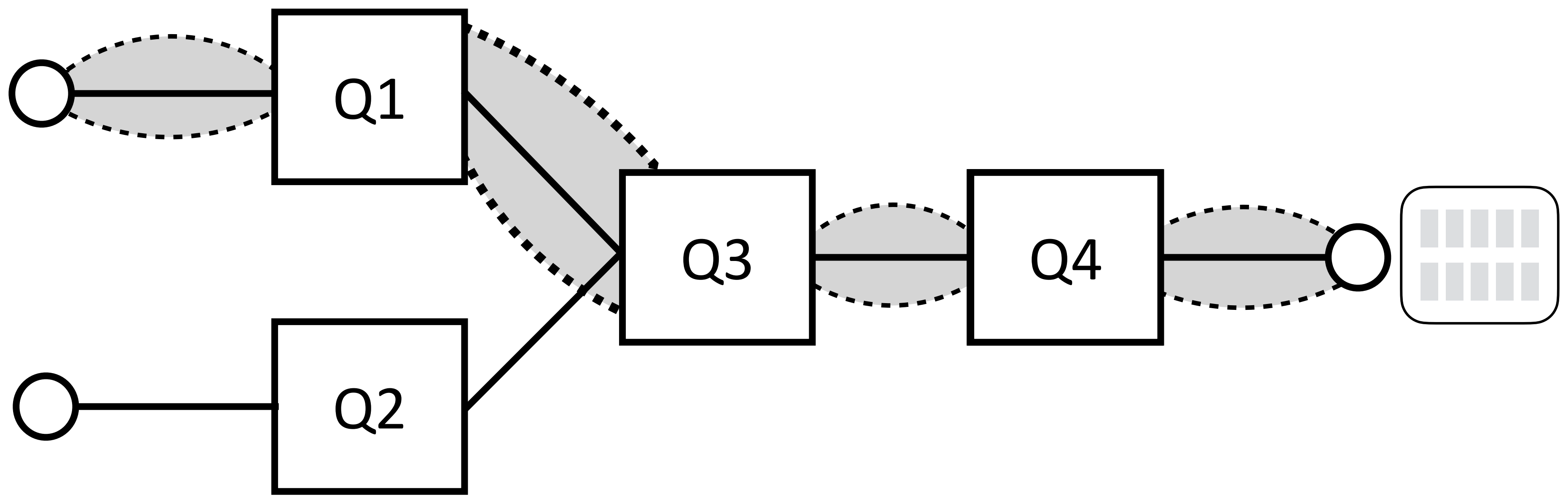
Domains in the Host Network

Depending on source/type, different requests traverse different domains in the host network

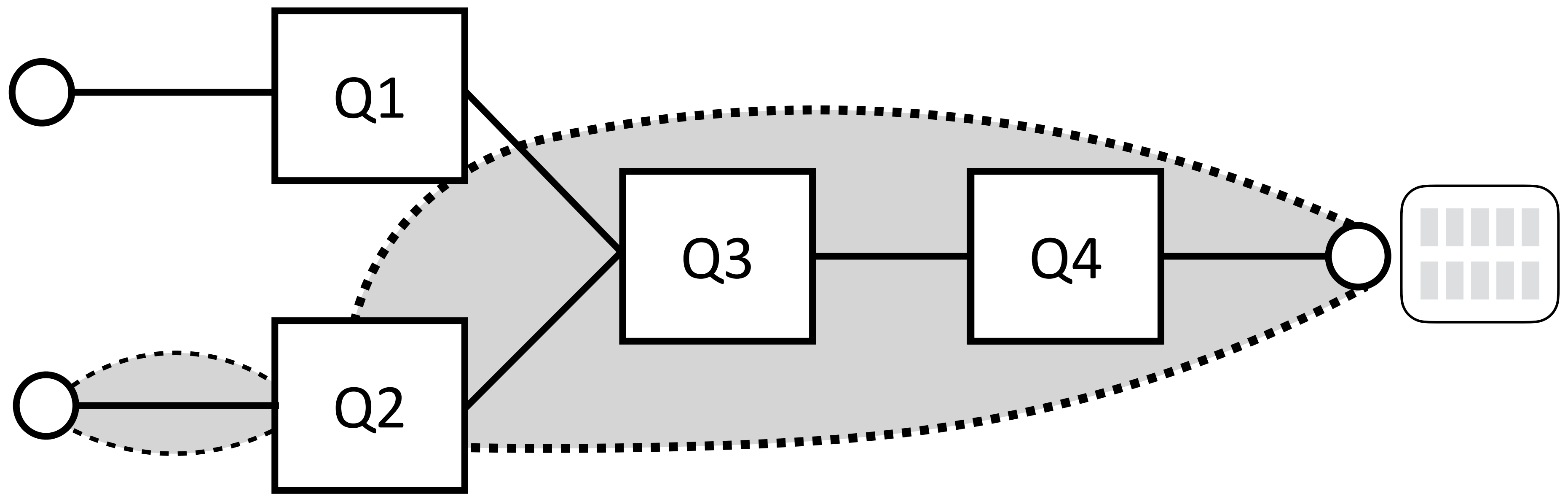
C2M Read



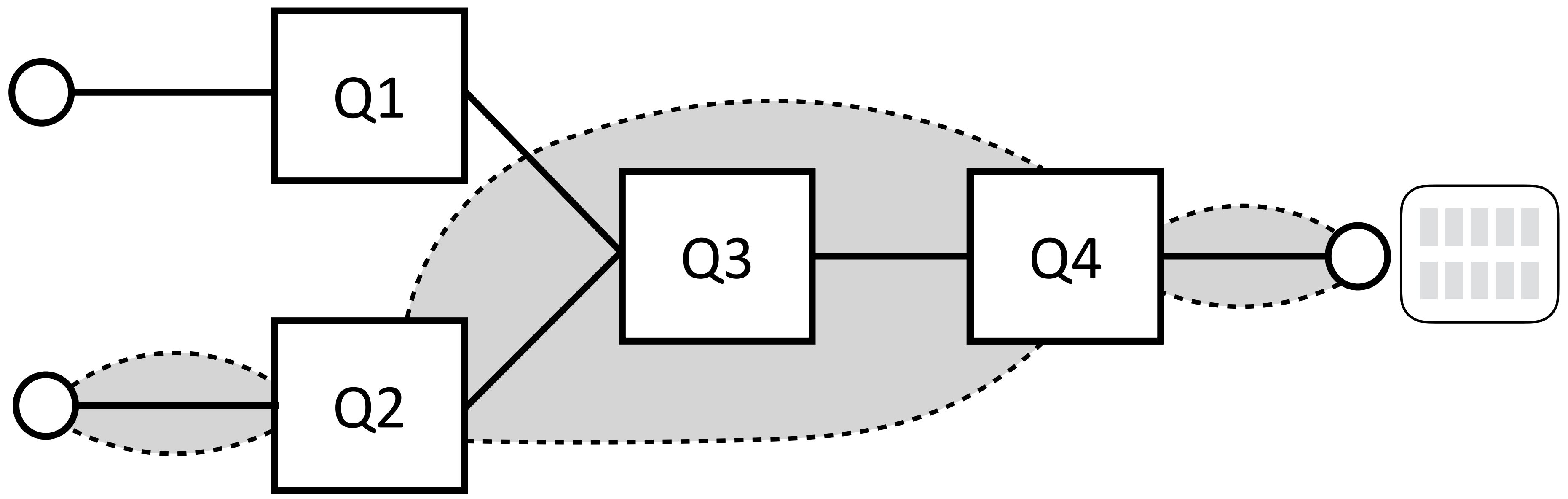
C2M Write



P2M Read



P2M Write



Reverse engineered domains and their characteristics on Intel architecture (see paper for details)

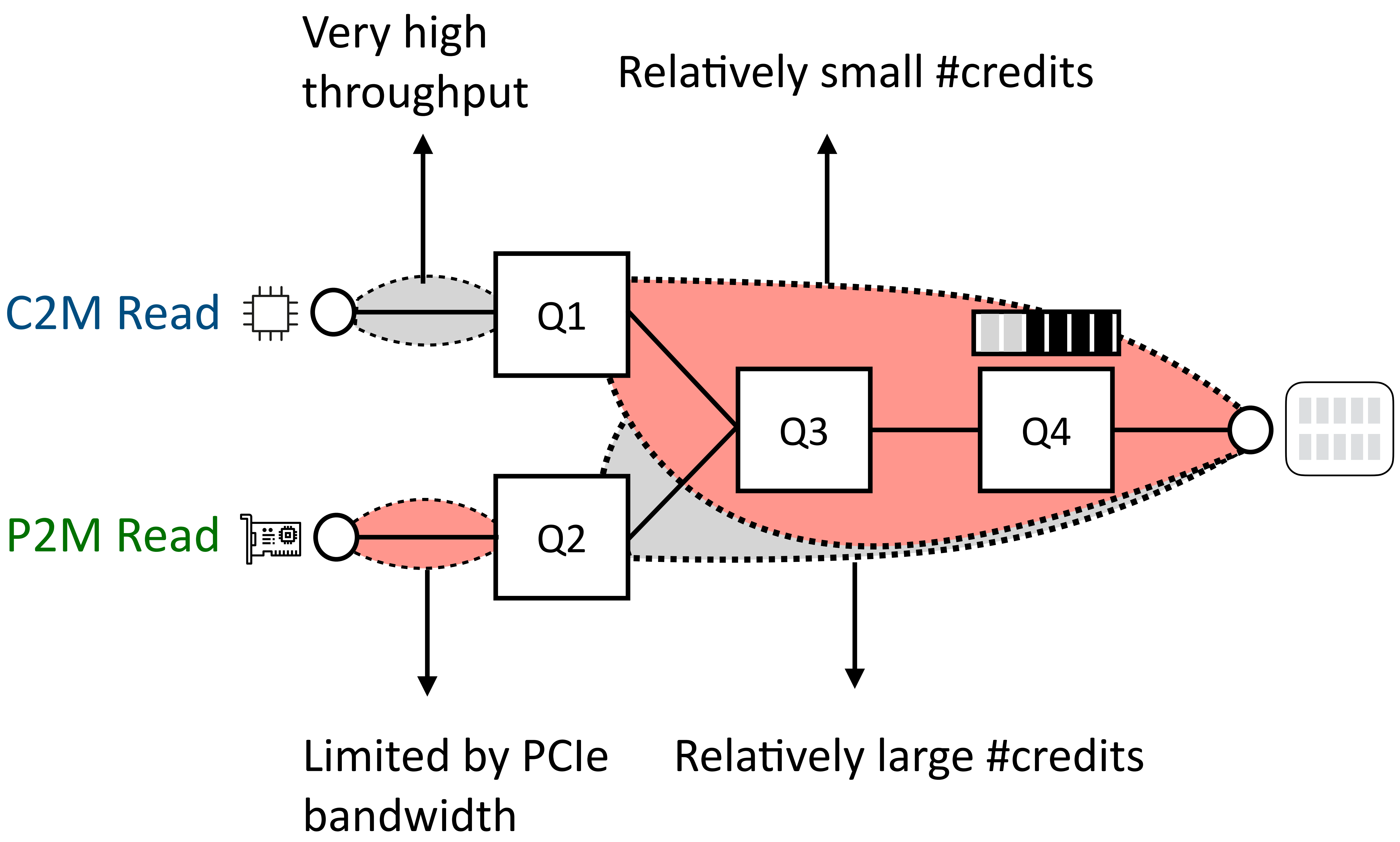
Understanding Regimes

	Core reason
Blue regime: C2M degrades but P2M does not	**Asymmetry in credits of domains**
Red regime: Both C2M and P2M degrade	**Asymmetry in latencies of domains**

Understanding the blue regime

Colocation: Latency inflation due to queueing in host network

Asymmetry in credits: P2M can better tolerate latency inflation compared to C2M


Very high throughput
Relatively small #credits
C2M Read
Q1
Q3
Q4
Q2
P2M Read
Limited by PCIe bandwidth
Relatively large #credits

C2M read: Latency inflation => throughput degradation

[Q1 <-> Memory] domain is the bottleneck due to small credits

P2M read: Latency inflation ⇏ throughput degradation

[Q2 <-> Memory] domain is not the bottleneck due to large credits

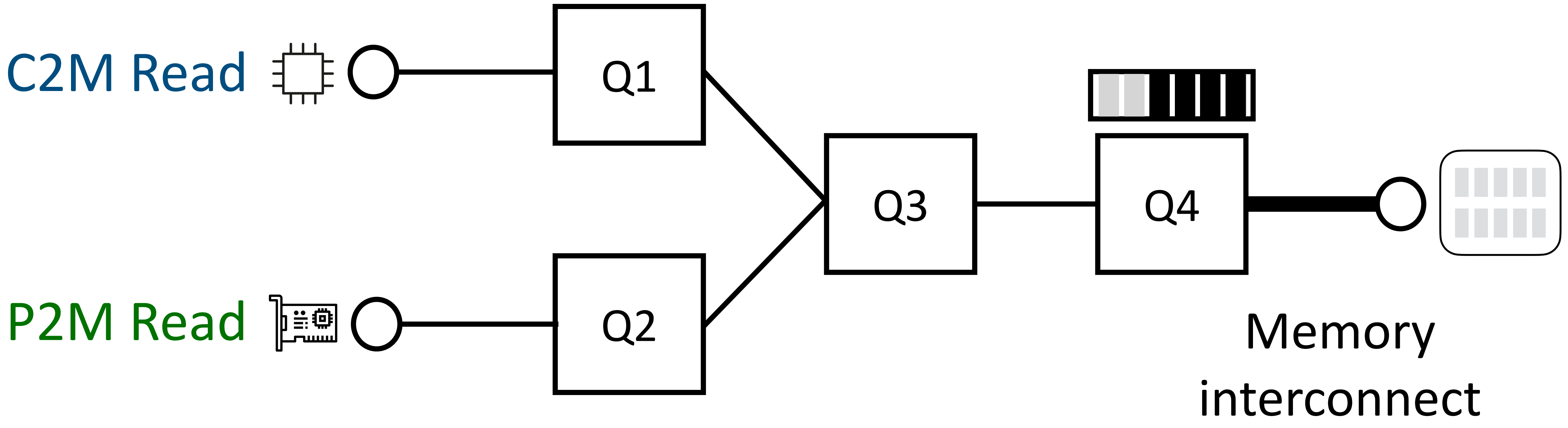
Understanding the blue regime

Colocation: Latency inflation due to queueing in host network

Asymmetry in credits: P2M can better tolerate latency inflation compared to C2M

C2M read: Latency inflation => throughput degradation

[Q1 <-> Memory] domain is the bottleneck due to small credits



P2M read: Latency inflation ≠> throughput degradation

[Q2 <-> Memory] domain is not the bottleneck due to large credits

Causes of queueing

Contention at memory interconnect

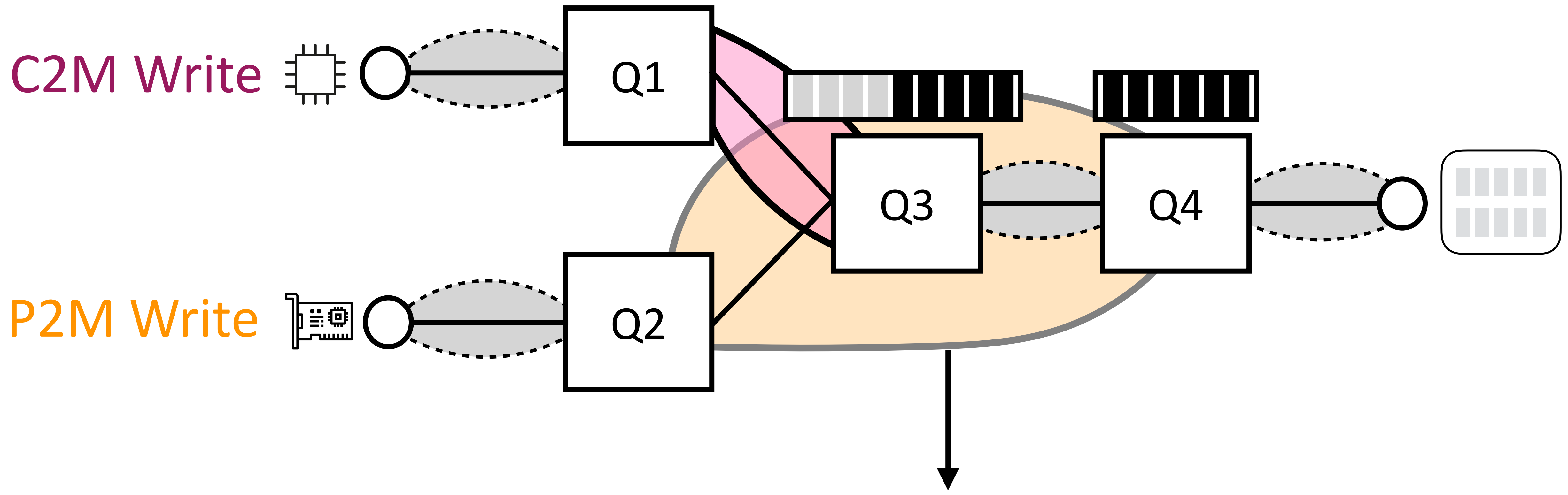
Contention within the memory modules

(even when memory interconnect is not saturated)

e.g., load imbalance across banks

Understanding red regime

Asymmetry in latency inflation: P2M write throughput degrades despite having large credits



Relatively large #credits

Large latency inflation

Poor interplay between P2M and C2M write domains

Backpressure from Q4 impacts P2M writes, but not C2M writes

Large latency inflation for P2M due to “unfair” backpressure

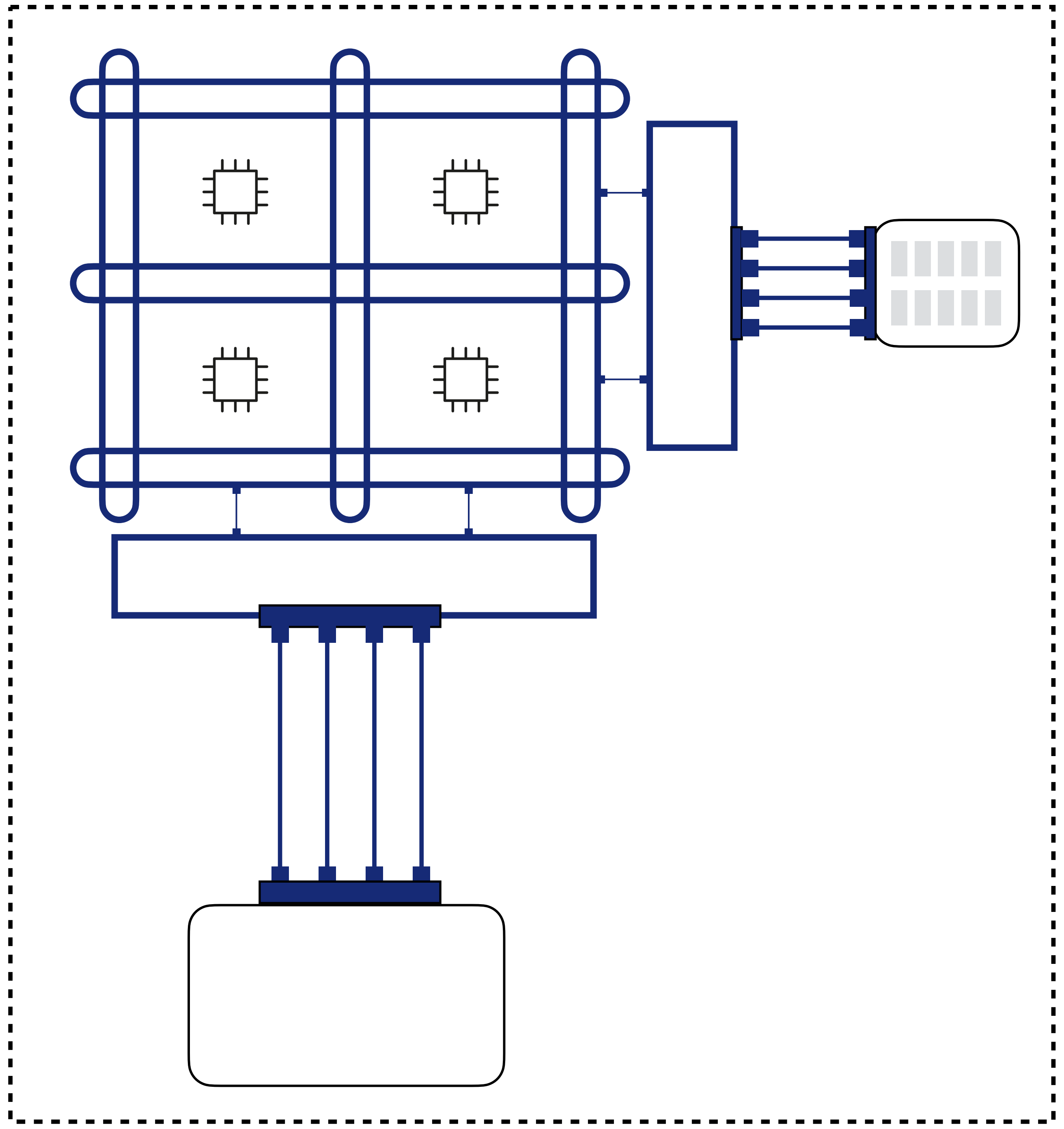
Understanding Regimes

	Core reason
Blue regime: C2M degrades but P2M does not	**Asymmetry in credits of domains**
Red regime: Both C2M and P2M degrade	**Asymmetry in latencies of domains**

Domain-by-domain credit-based flow control enables explaining different host network contention regimes

(Please see paper for precise explanations and quantitative validation)

Our study: Understanding the Host Network



Building an understanding of the host network contention and its root causes

New, previously unreported, host network contention regimes

Poor interplay between processor, memory and peripheral interconnects

New lens: Conceptual abstraction to study the host network

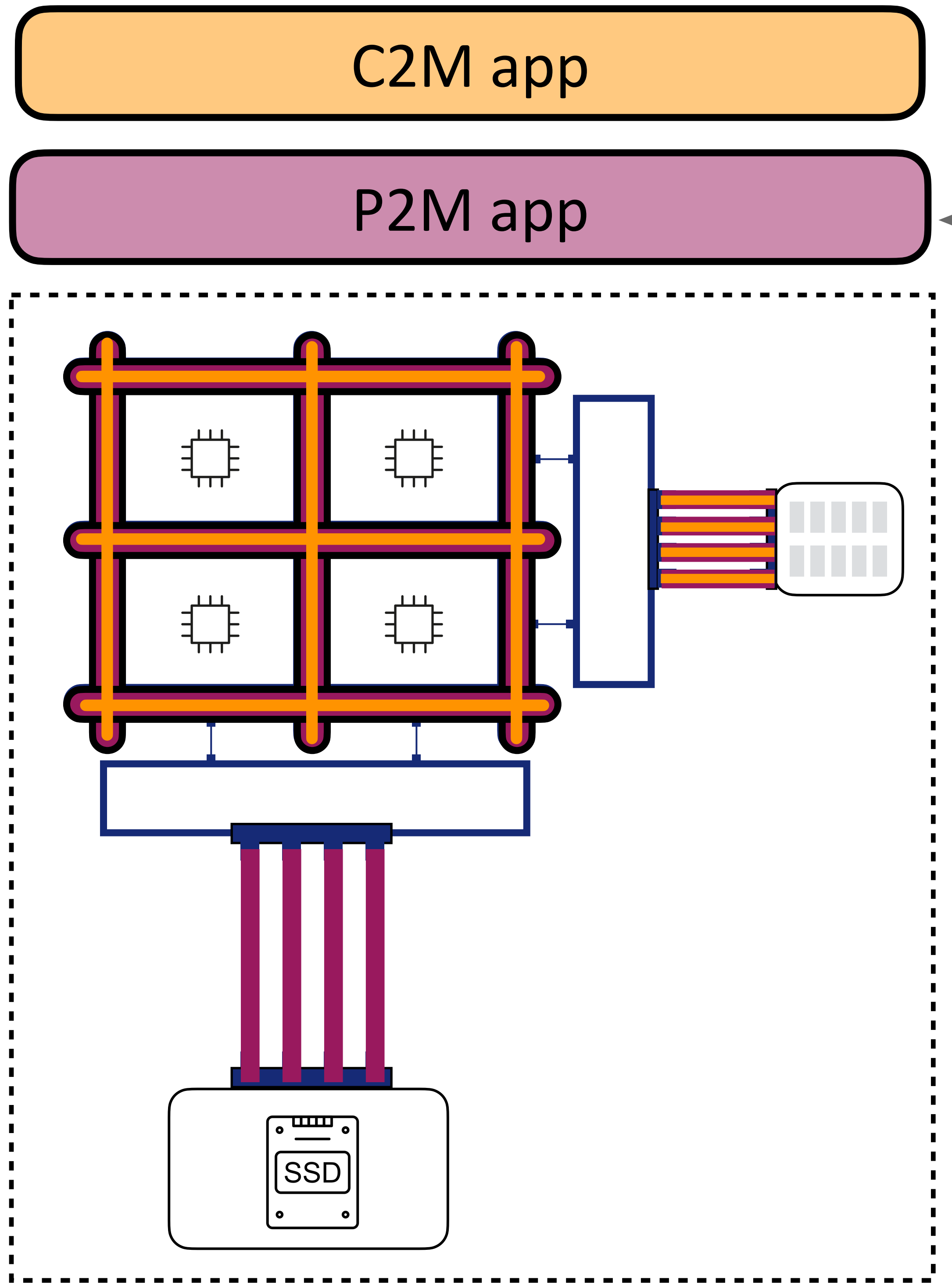
Domain-by-domain credit-based flow control

Captures the subtle interplay between different interconnects

Host network as a standalone network

All our results and observations apply even when all traffic is contained within a single host

Host network contention: Impact broader than networked applications

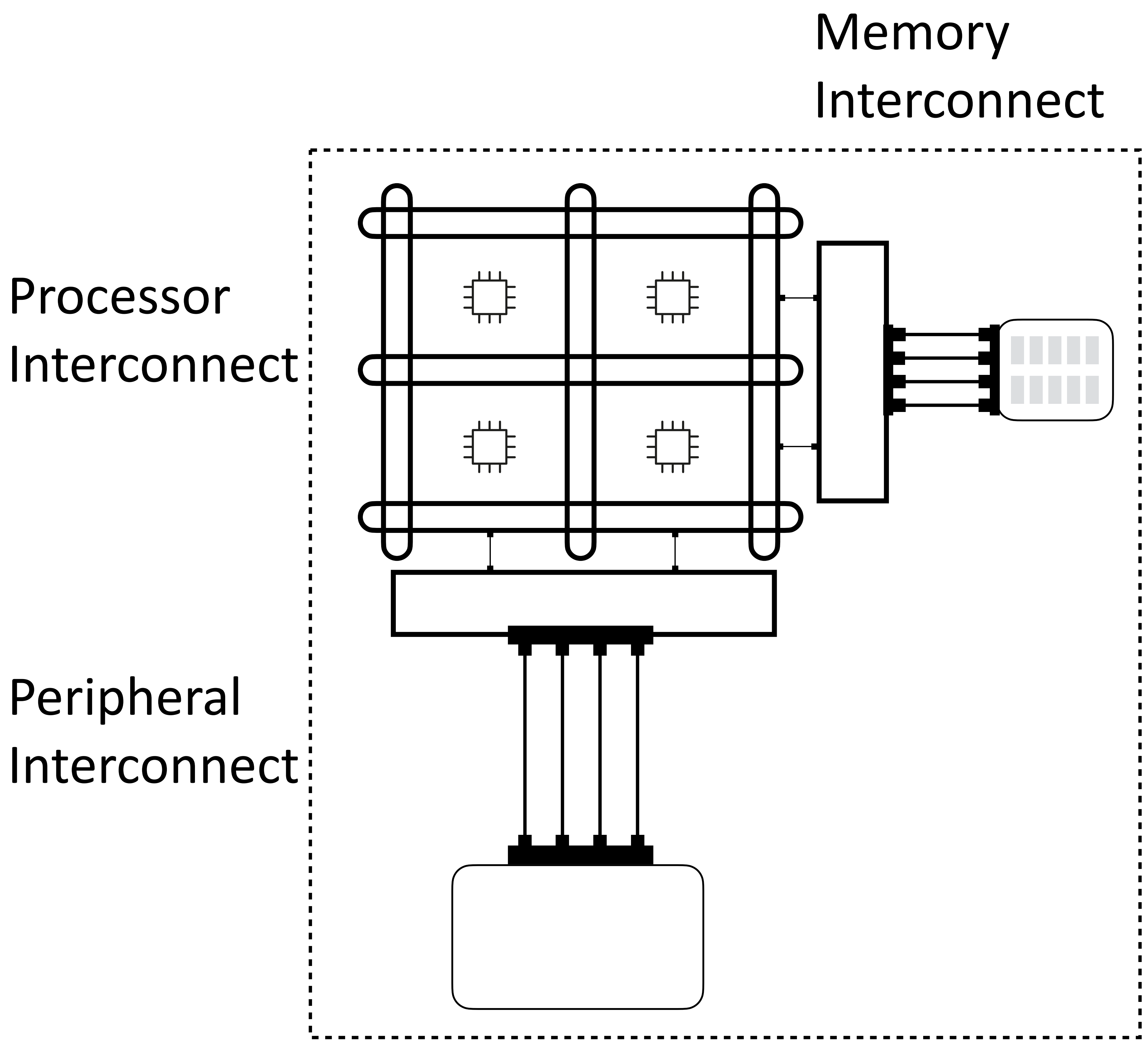


All our observations generalize even when all traffic within single host

For example: using storage apps (P2M app) that read/write from local SSDs

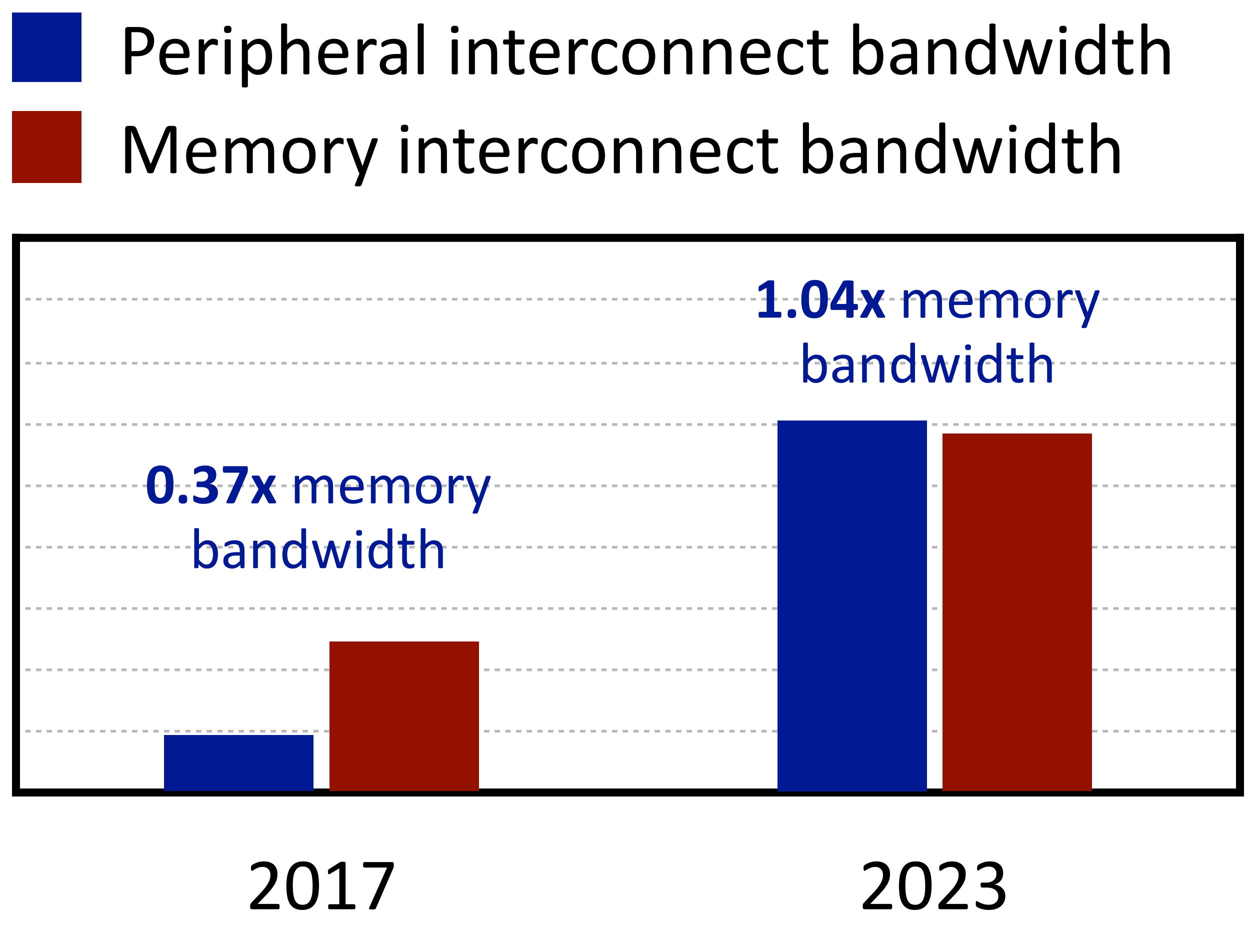
(Please see paper for details)

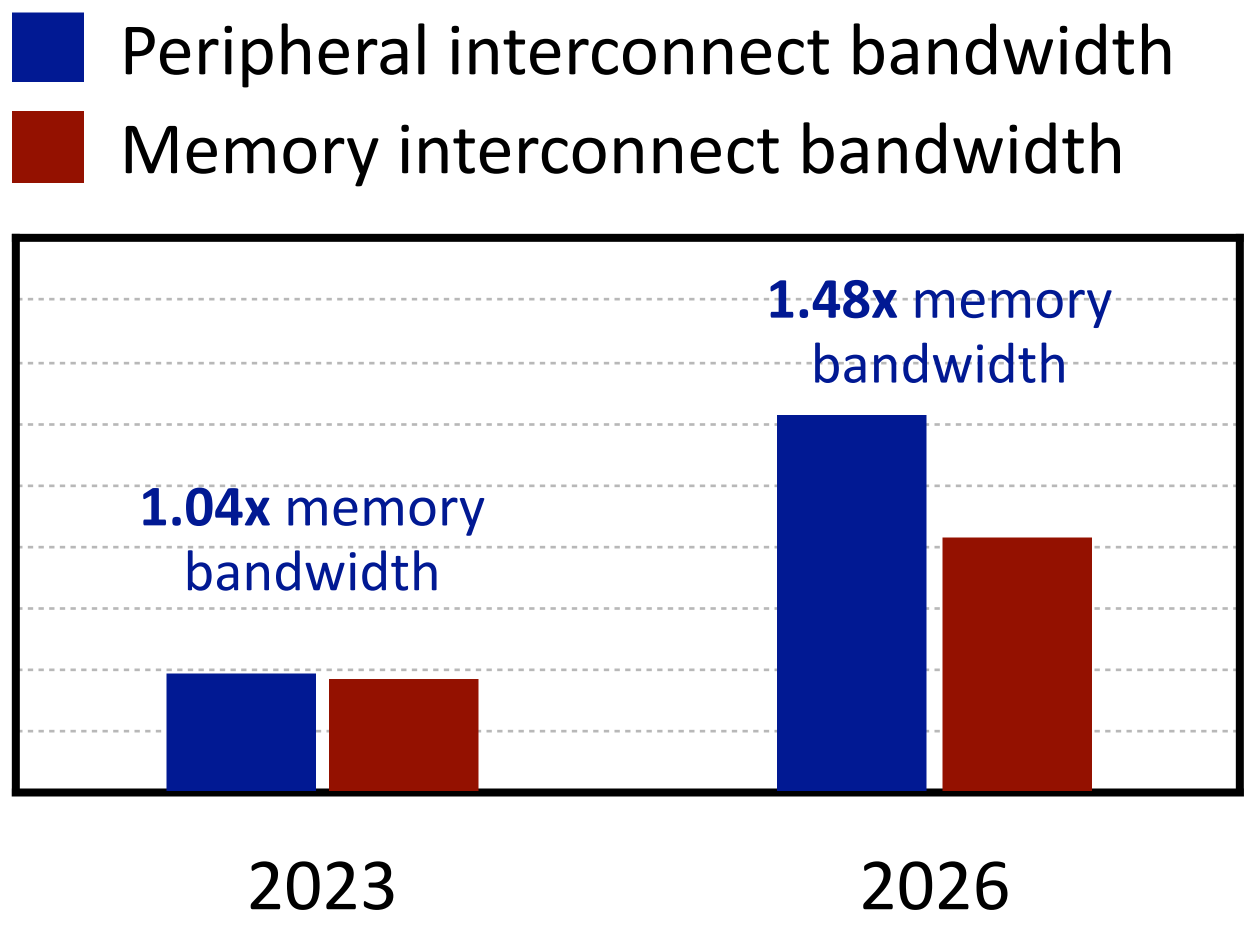
Increasing Importance of Host Network: Technology Trends



Processor interconnect bandwidth has always been large O(10 Tbps)

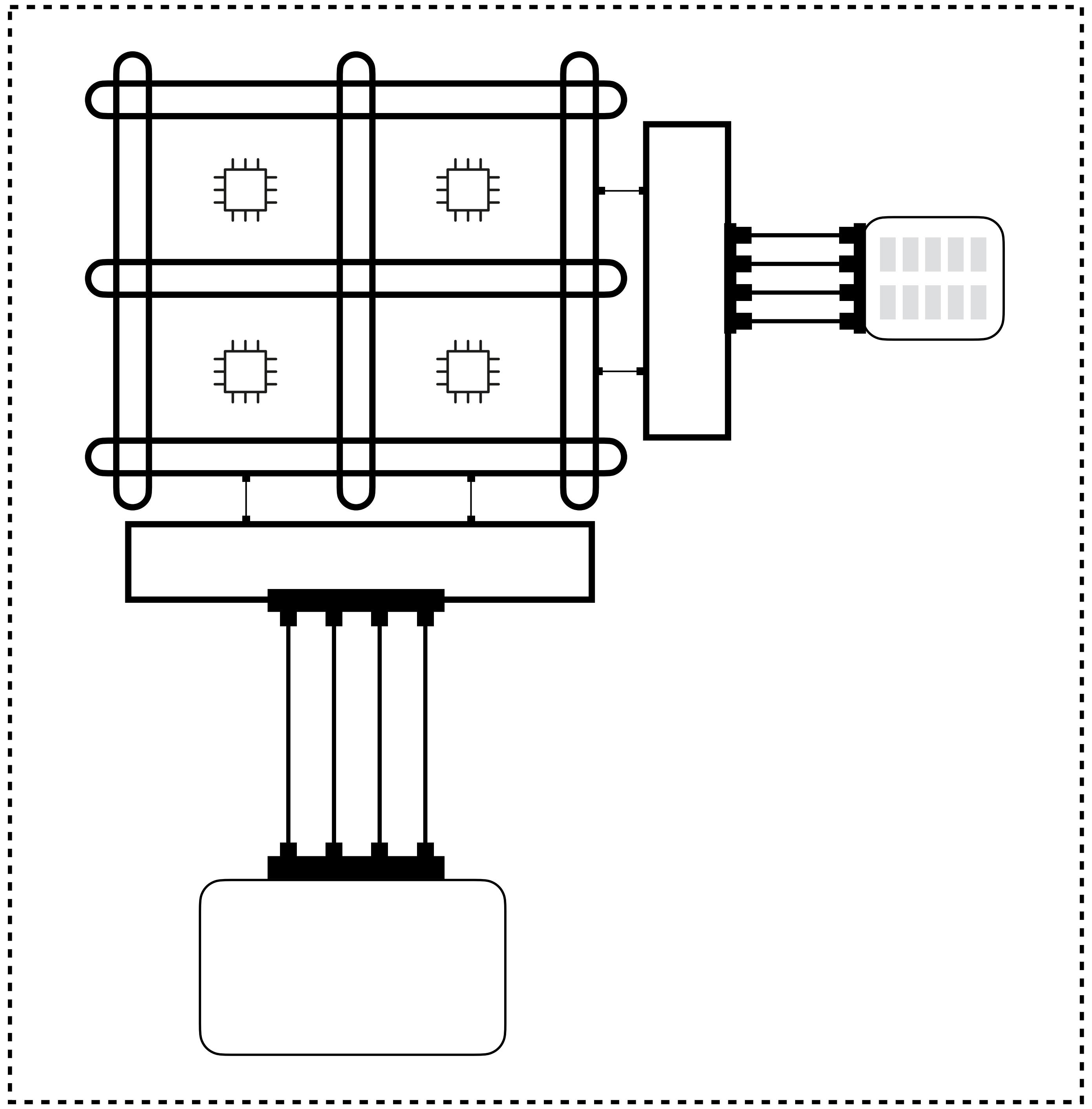
Peripheral interconnect bandwidth growing faster than memory interconnect



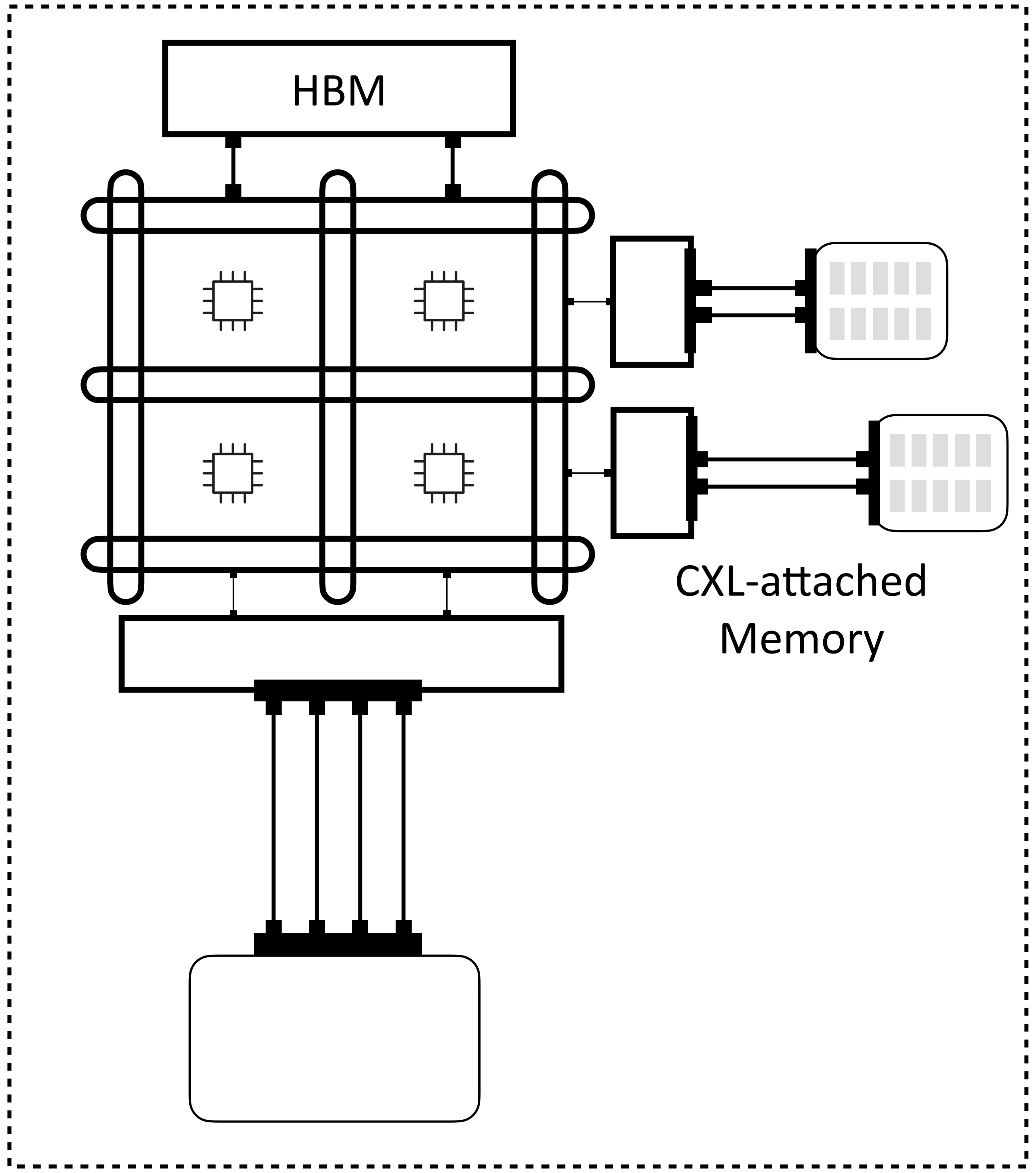


Different technology trends for different interconnects: Resource imbalances in the host network

Host network is becoming increasingly complex



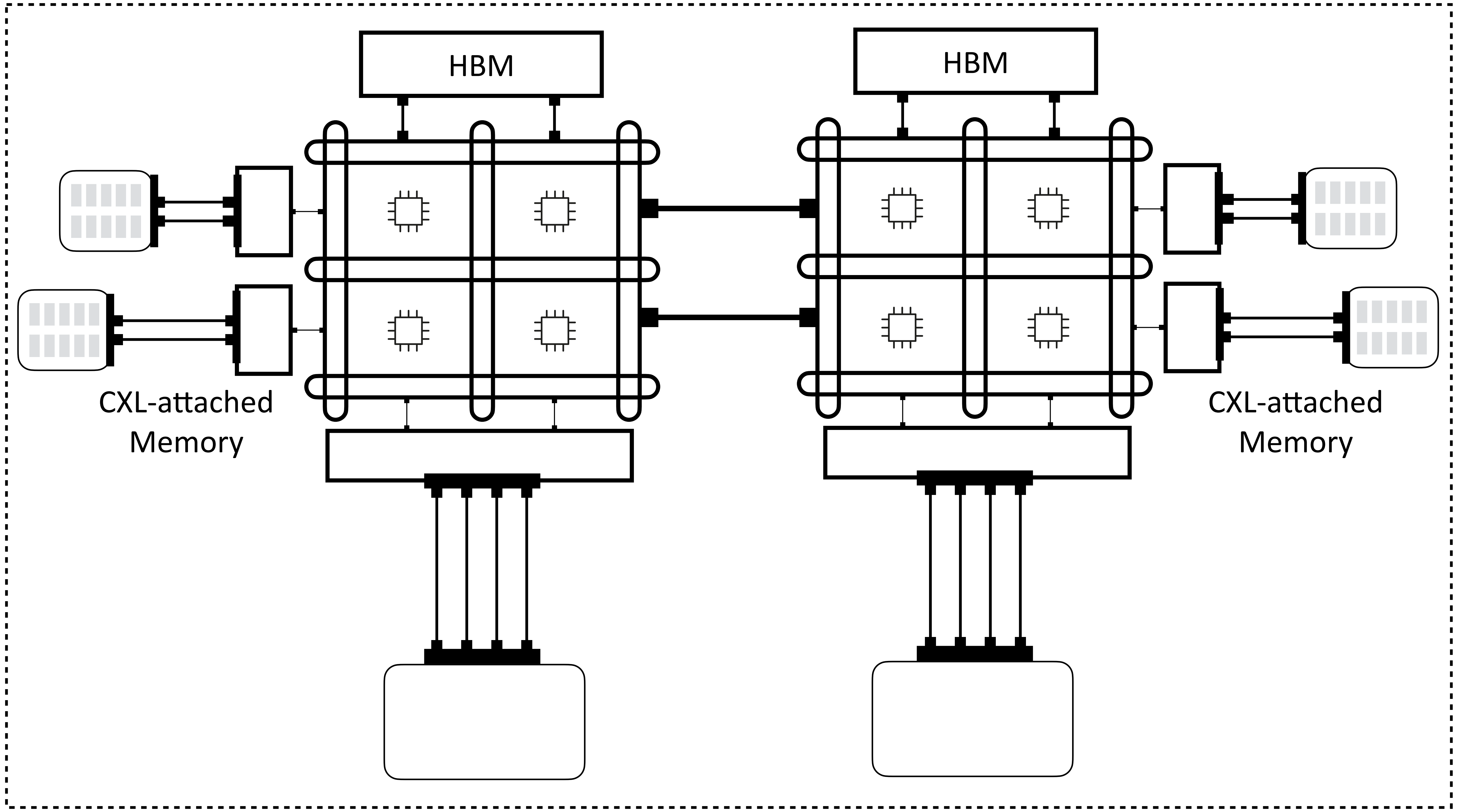
Host network is becoming increasingly complex



Different kinds of memory

e.g., CXL, HBM

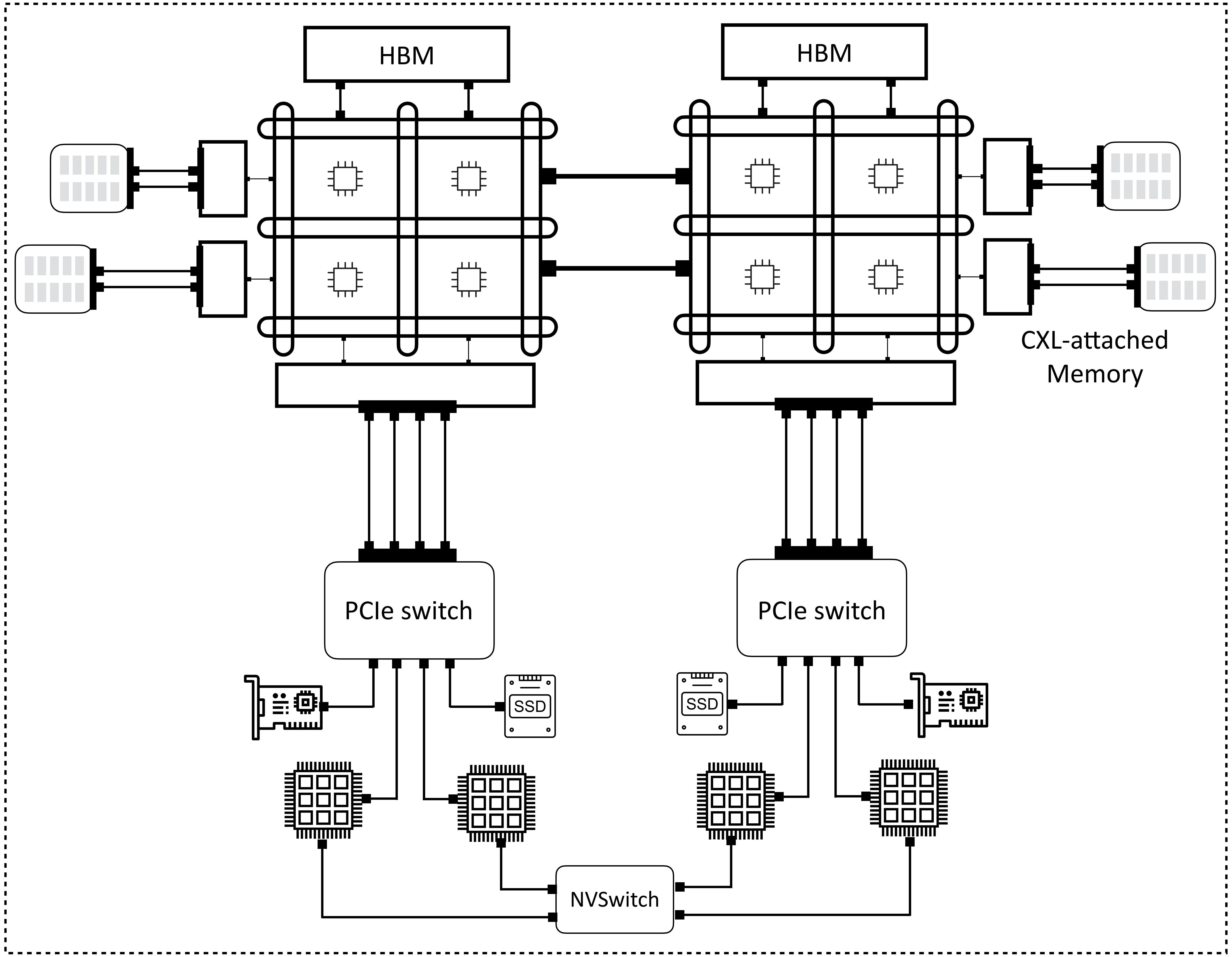
Host network is becoming increasingly complex



Different kinds of memory
e.g., CXL, HBM

Increasing scale of processors
e.g., mutli-socket or chiplet-based designs

Host network is becoming increasingly complex

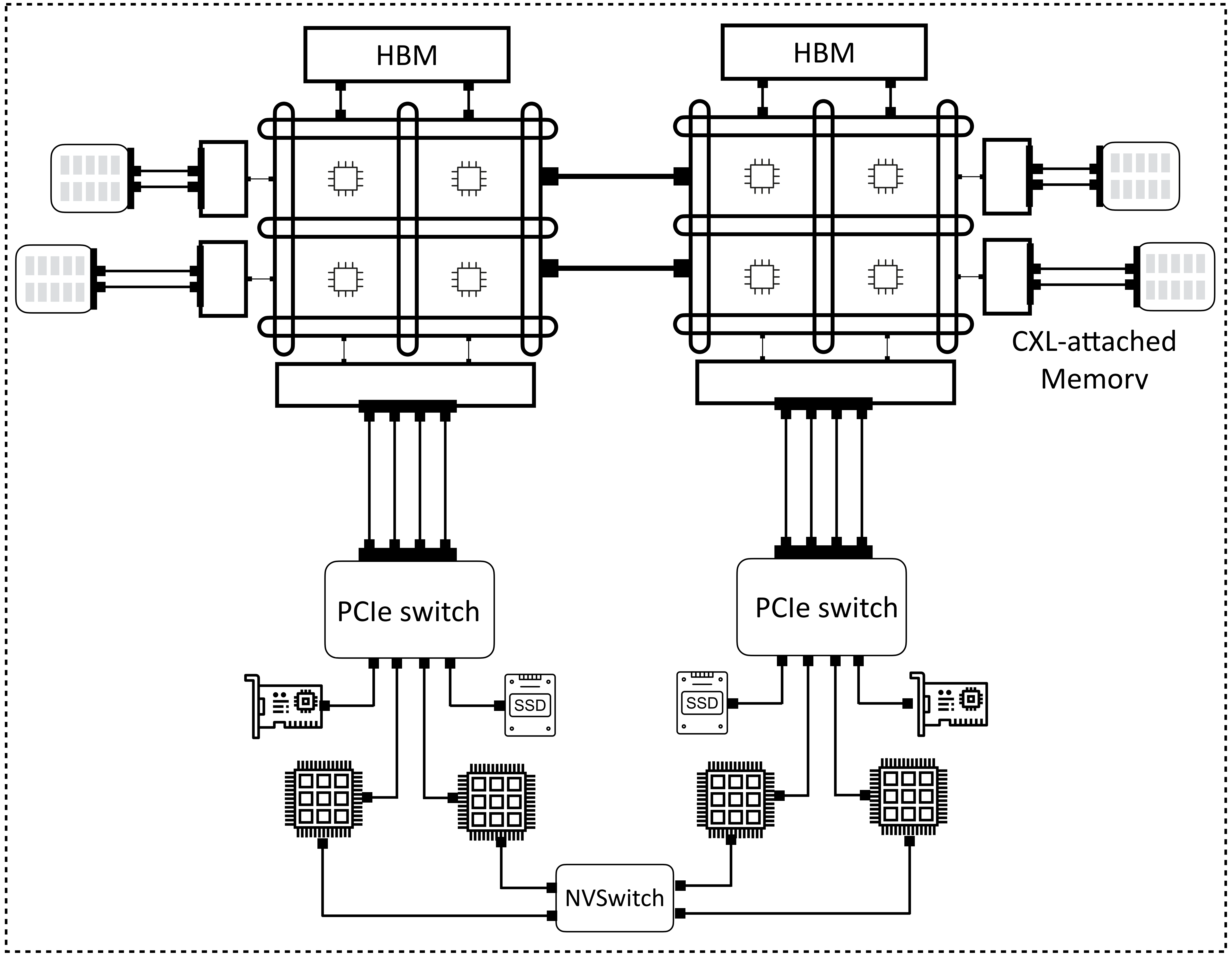


Different kinds of memory
e.g., CXL, HBM

Increasing scale of processors
e.g., mutli-socket or chiplet-based designs

Deeper topologies
e.g., PCIe lanes/switches and NVlinks/switches

Future directions



Building even deeper understanding of host network

Extending to more complex host networks

Analytical modeling to predict performance

Rearchitecting protocols, OS, host hardware

New mechanisms for host network resource allocation

Better mechanisms for load balancing host network traffic

https://github.com/host-architecture/understanding-the-host-network