Spandex: A Flexible Interface for Efficient Heterogeneous Coherence

Johnathan Alsop,* Sarita Adve, Matthew D. Sinclair[†]

Illinois, *AMD, [†] Wisconsin

sadve@Illinois.edu

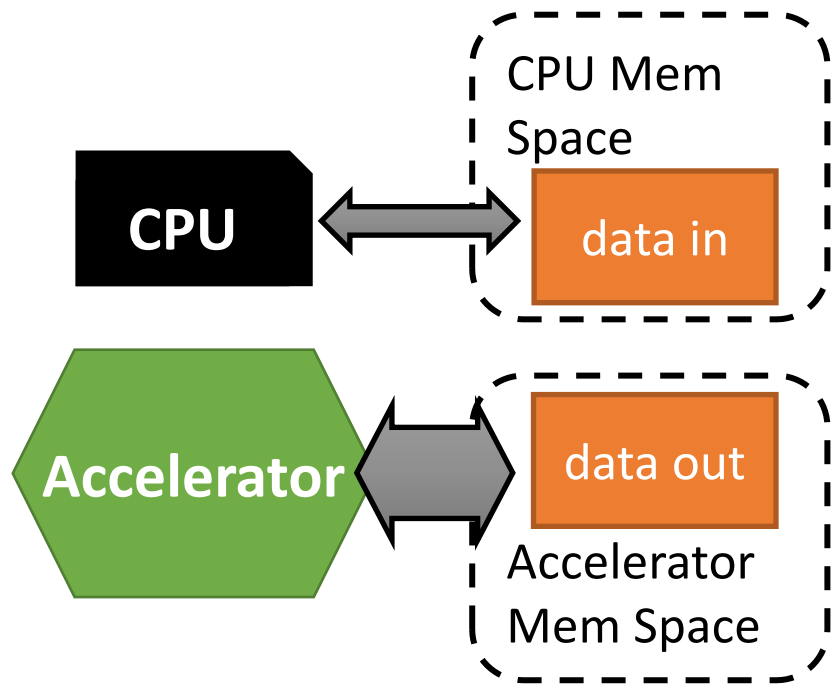# Specialized architectures are increasingly important in all compute domains

Specialization Requires Better Memory Systems

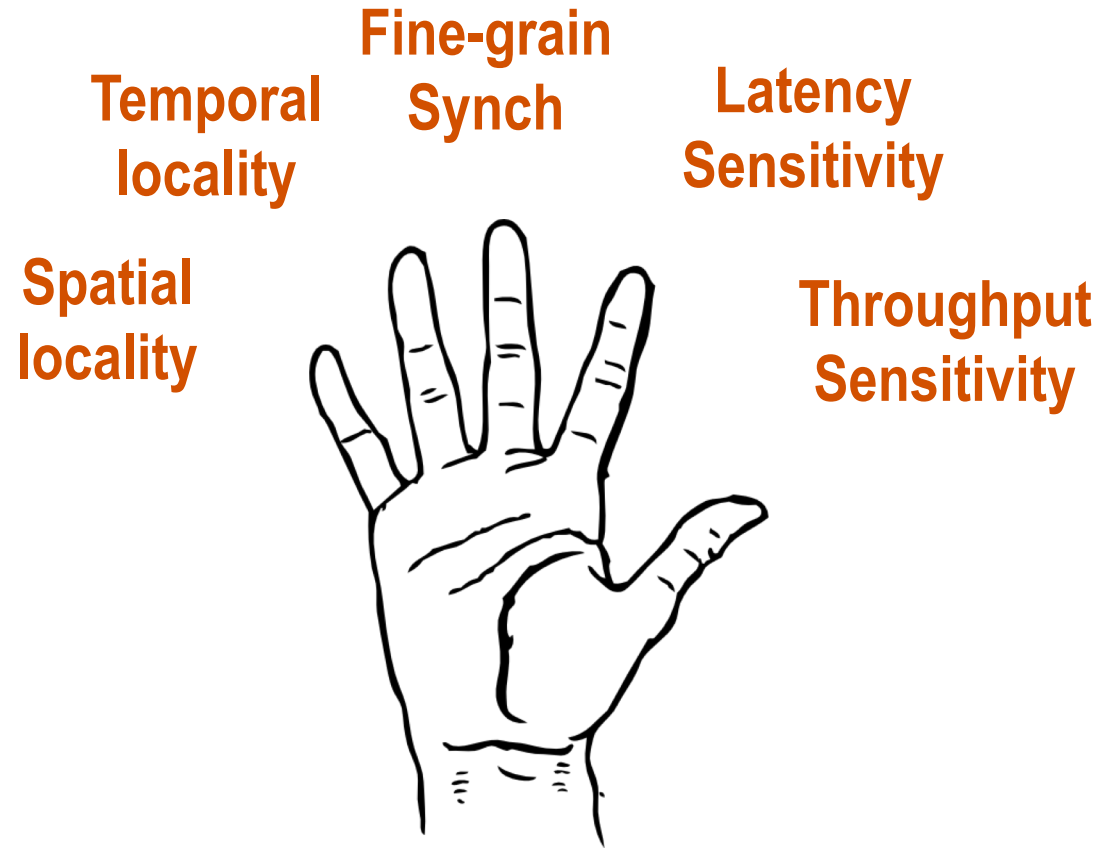## Traditional heterogeneity

CPU

CPU Mem Space

data in

Accelerator

data out

Accelerator Mem Space

✖ No fine-grain synchronization
✖ No irregular access patterns
✖ Wasteful data movement

Coherent shared memory

CPU
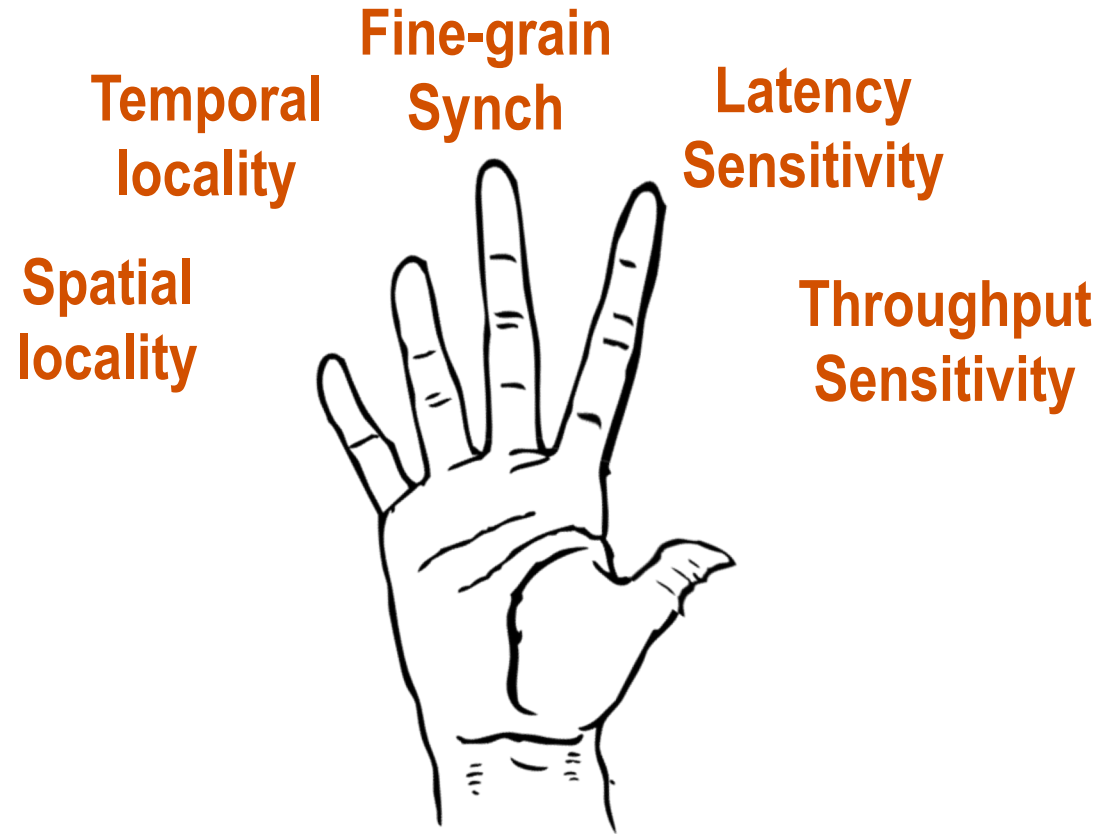
Shared Mem Space

Existing solutions: complex and inflexible

Accelerator

✓ Fine-grain synchronization
✓ Irregular access
✓ Implicit data reuse

3

Heterogeneous devices have diverse memory demands

Fine-grain Synch
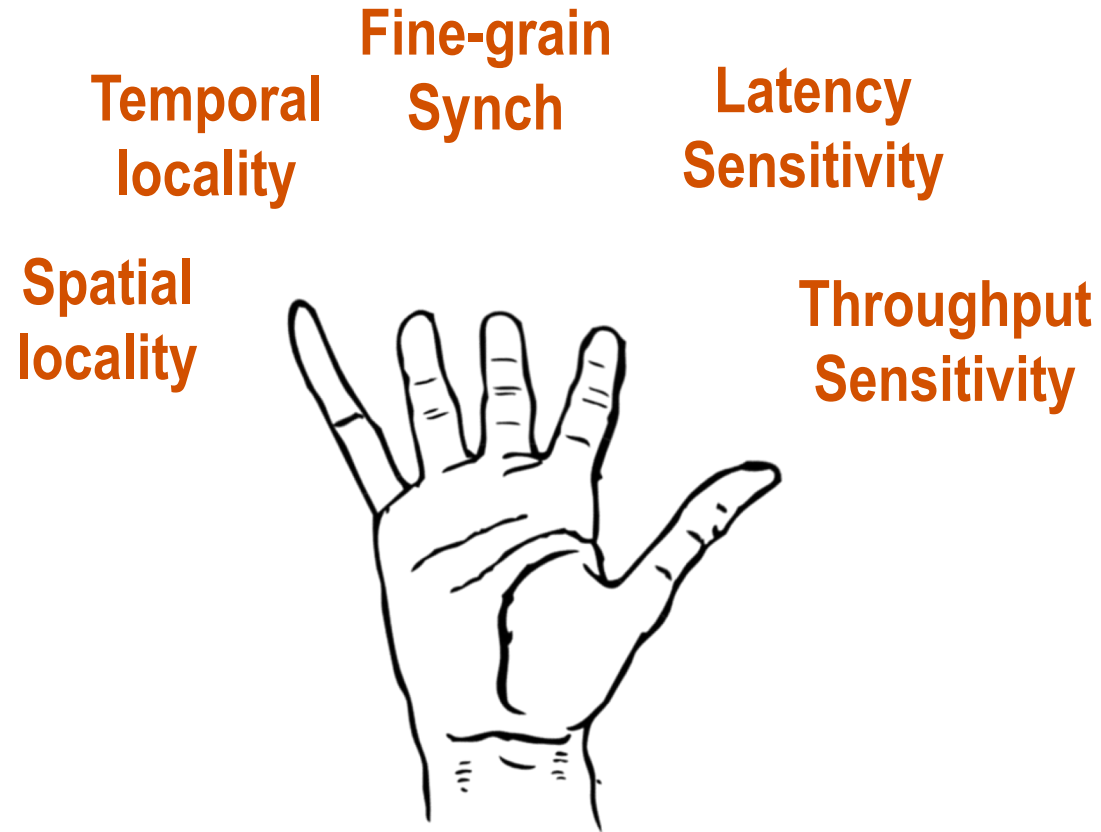
Temporal locality

Latency Sensitivity

Spatial locality

Throughput Sensitivity

Heterogeneous devices have diverse memory demands

Temporal locality

Fine-grain Synch

Latency Sensitivity

Spatial locality

Throughput Sensitivity

Typical **CPU** workloads:
fine-grain synch, latency sensitive

Heterogeneous devices have diverse memory demands

Fine-grain Synch

Temporal locality

Latency Sensitivity

Spatial locality

Throughput Sensitivity

Typical **GPU** workloads:
spatial locality, throughput sensitive

MESI protocol fits CPU workloads

Properties	MESI
Granularity	
Invalidation	
Updates	

Good for:

MESI

CPU

GPUs prefer simpler protocols

Properties	MESI	GPU coherence
Granularity	Line	
Invalidation	Writer-invalidate	
Updates	Ownership	

MESI

CPU

GPU coh.

GPU

Good for:

DeNovo is a good fit for CPU and GPU

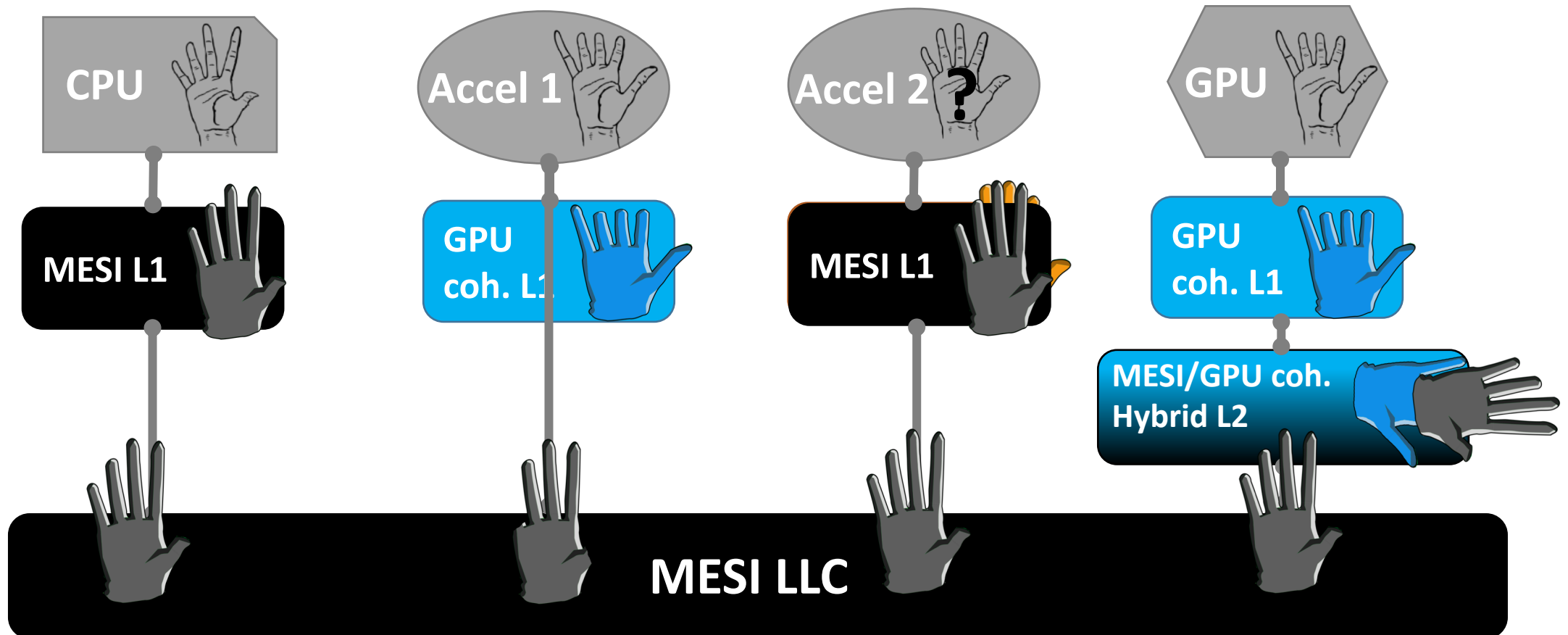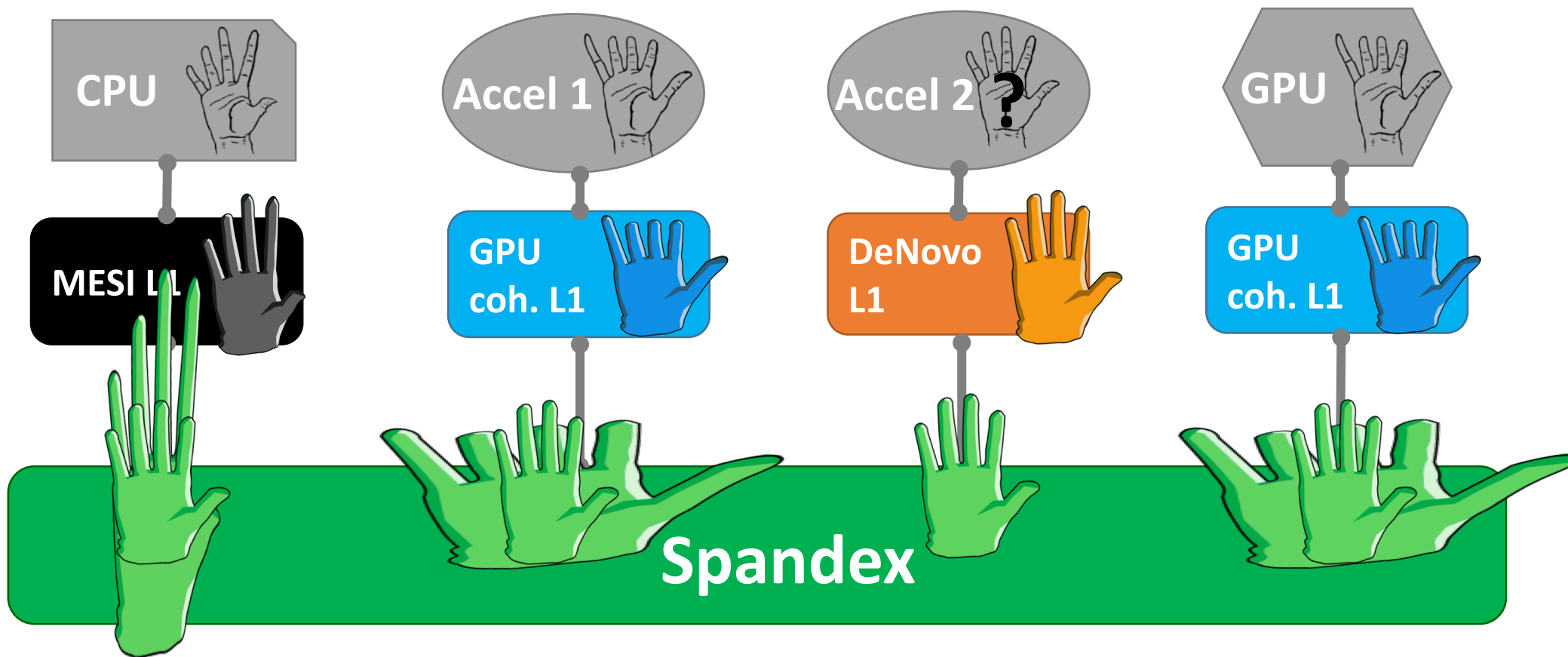| Properties	MESI	GPU coherence	DeNovo
Granularity	Line	Reads: Line Writes: Word	
Invalidation	Writer-invalidate	Self-invalidate	
Updates	Ownership	Write-through	

Good for:

MESI
CPU

GPU coh.
GPU

DeNovo
CPU or GPU

Existing Solutions: Inflexible and Inefficient

Examples: ARM ACE, IBM CAPI, AMD APU

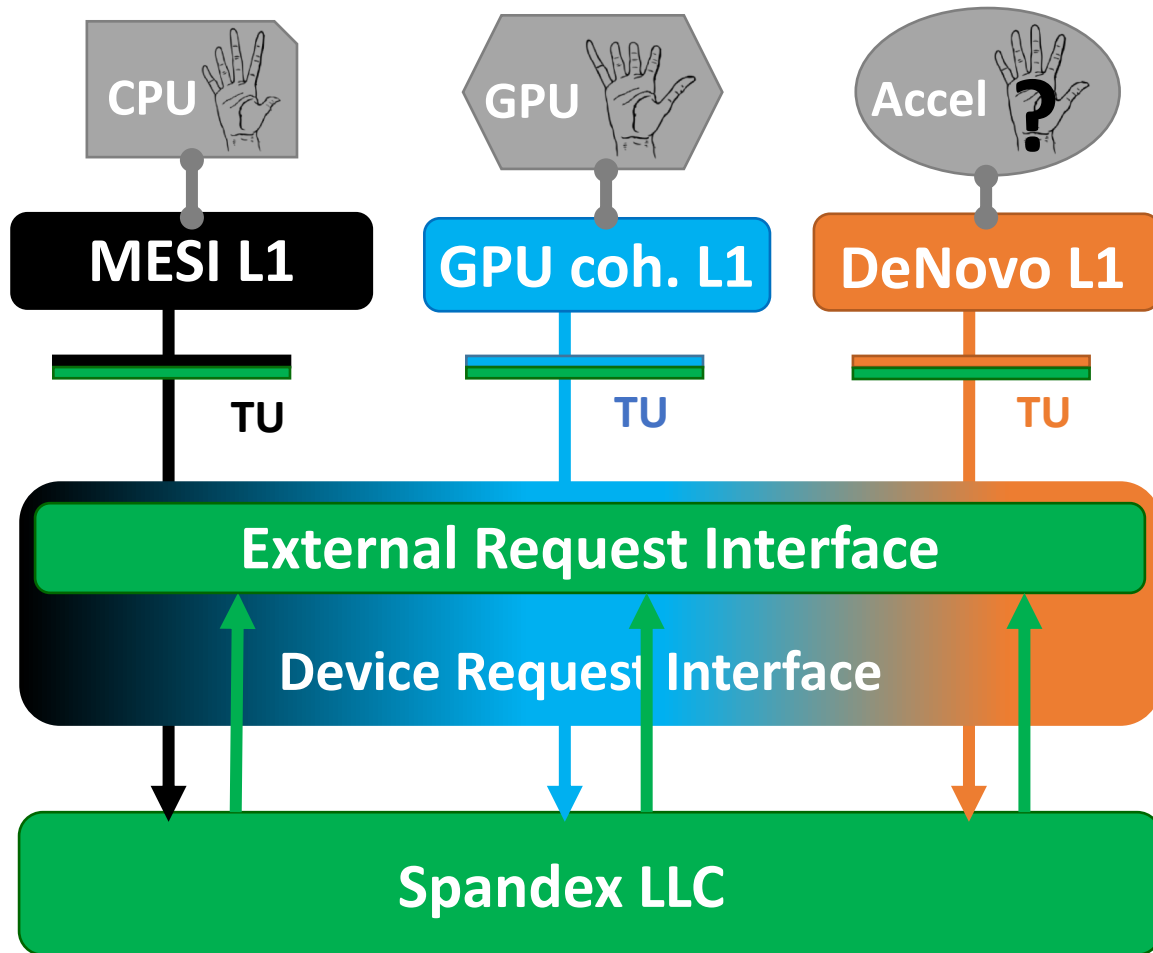# Spandex: Flexible Heterogeneous Coherence Interface

Adapts to exploit individual device's workload attributes

Better performance, lower complexity
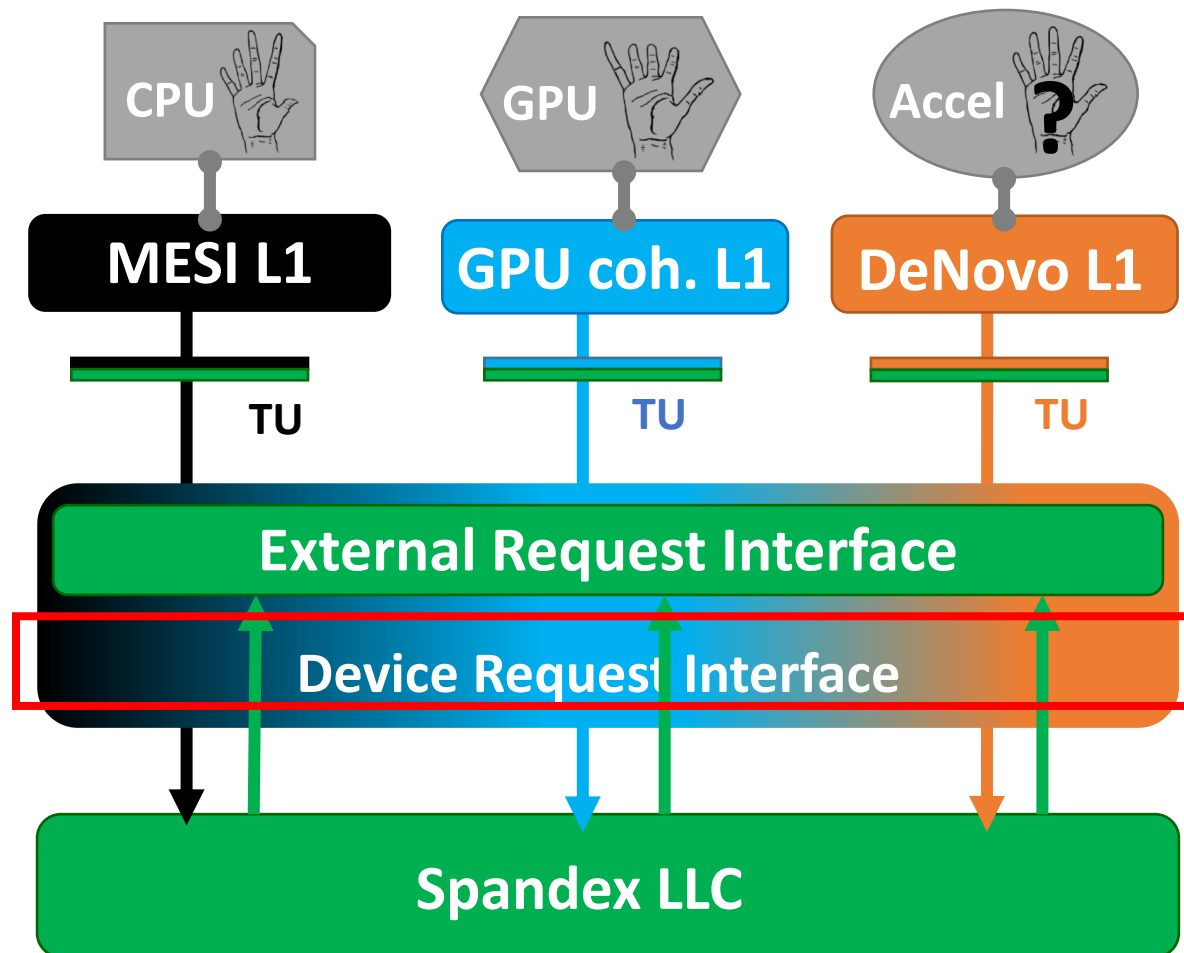⇒ Fits like a glove for any heterogeneous system!

Spandex Overview

Key Components

- Flexible device request interface

- DeNovo-based LLC

- External request interface

Device may need a translation unit (TU)

Spandex Overview

Key Components

- Flexible device request interface

- DeNovo-based LLC
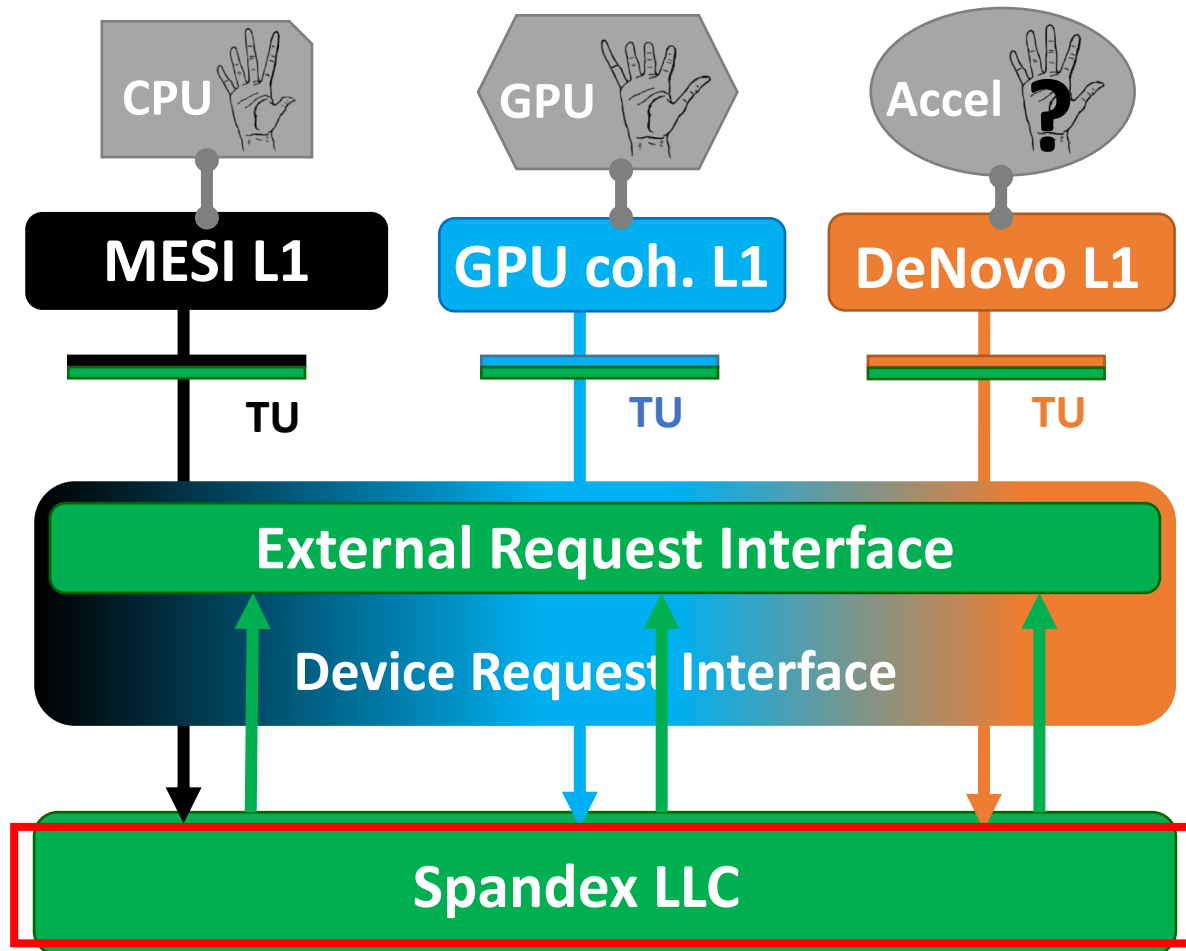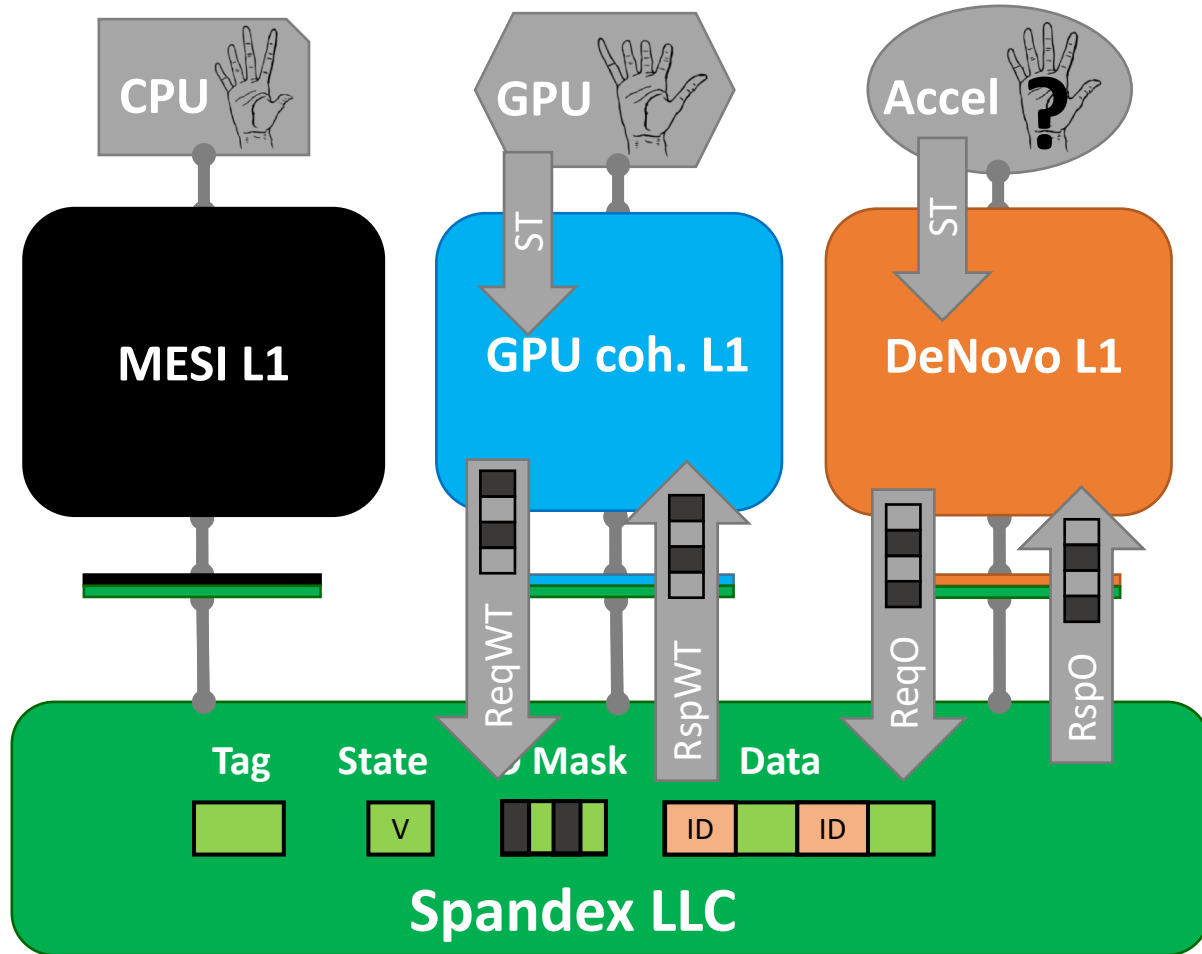
- External request interface

Device may need a translation unit (TU)

Device Request Interface

Action	Request	Indicates
Read	ReqV	Self-invalidation
	ReqS	Writer-invalidation

Requests also specify granularity and (optionally) a bitmask

Spandex Overview

Key Components

- Flexible device request interface

- DeNovo-based LLC
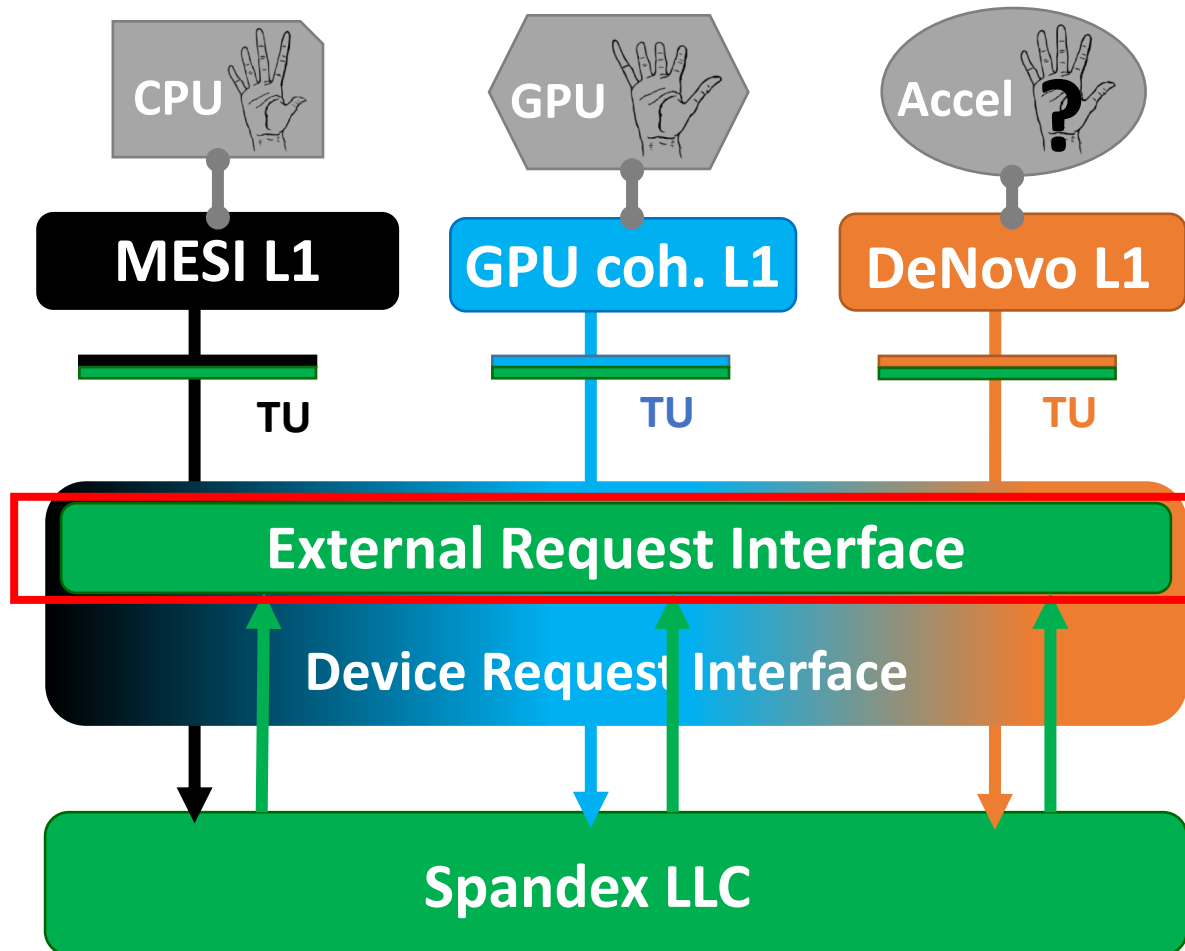
- External request interface

Device may need a translation unit (TU)

Spandex LLC

- States: I, V, O, S
- Allocation at line granularity
- Ownership at word granularity
- Data field tracks owner ID
- May generate requests to owner/sharer

✓No false sharing
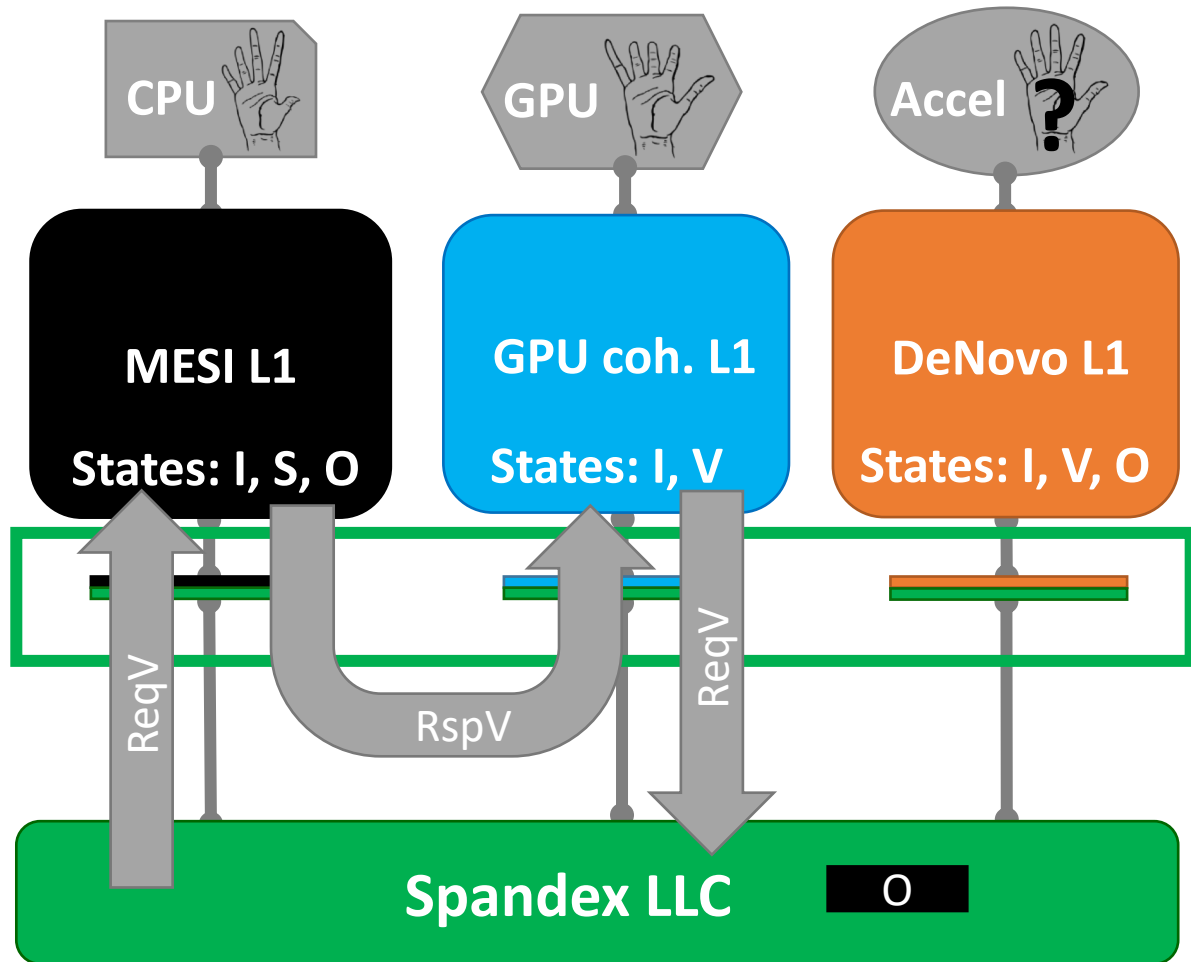✓Non-blocking ownership transfer

Spandex Overview

Key Components

- Flexible device request interface

- DeNovo-based LLC

- External request interface
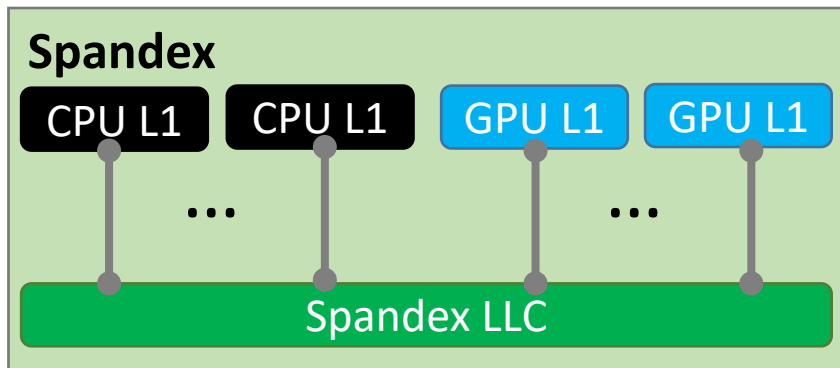
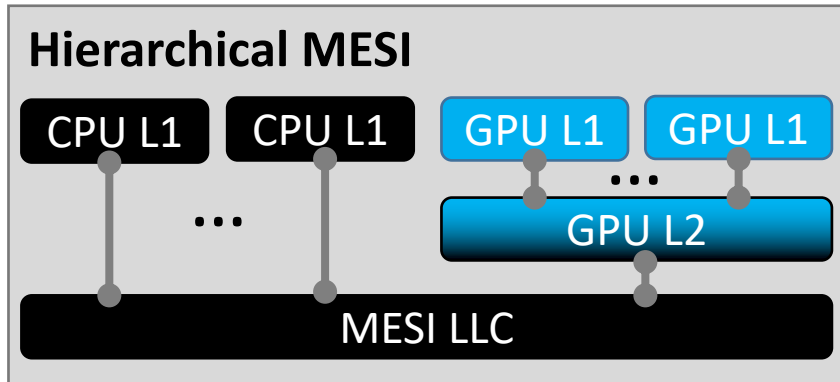Device may need a translation unit (TU)

External Request Interface

External Request	Must handle if supports state
ReqV	O
ReqO	O
ReqO+data	O
RvkO	O
Inv	S
ReqS	S and O

- **Translation Unit** may implement functionality if not supported by device

Evaluation: Configurations

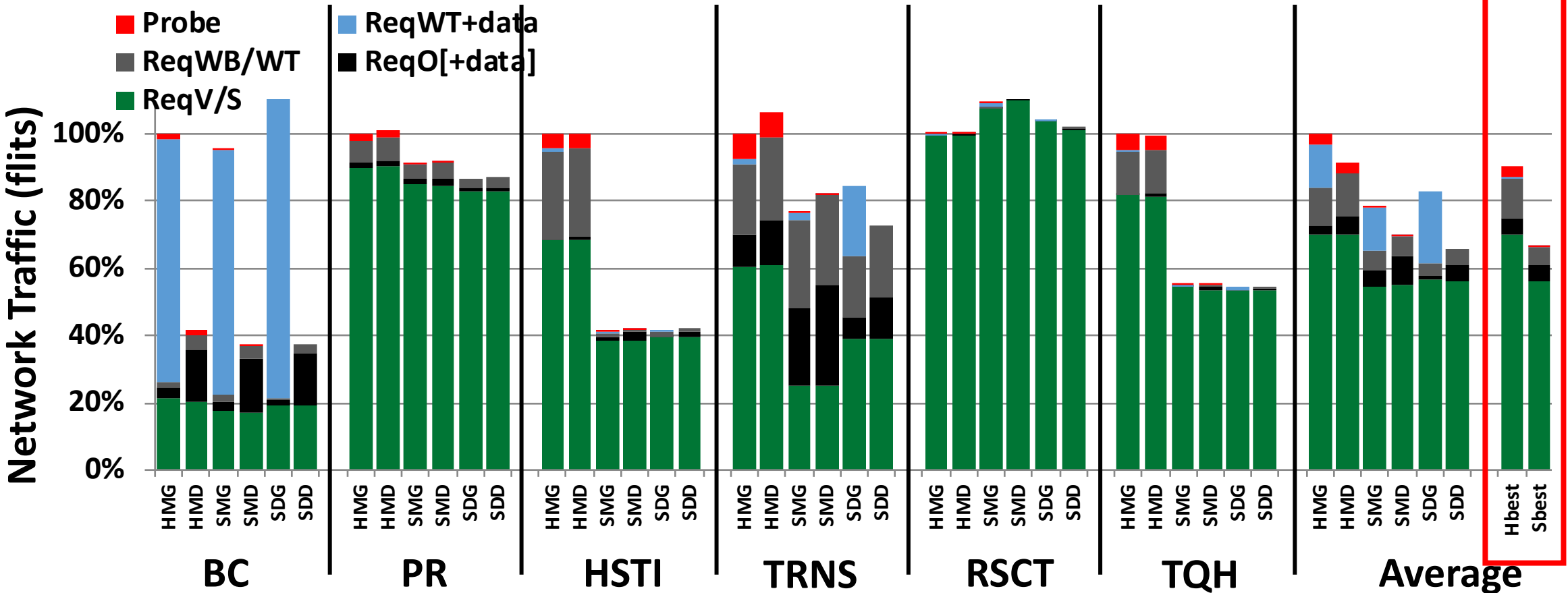| Configuration	LLC protocol	CPU protocol	GPU protocol
HMG	Hierarchical MESI	MESI	GPU coherence
HMD	Hierarchical MESI	MESI	DeNovo

CPU-GPU workloads from **Pannotia** and **Chai** benchmark suites
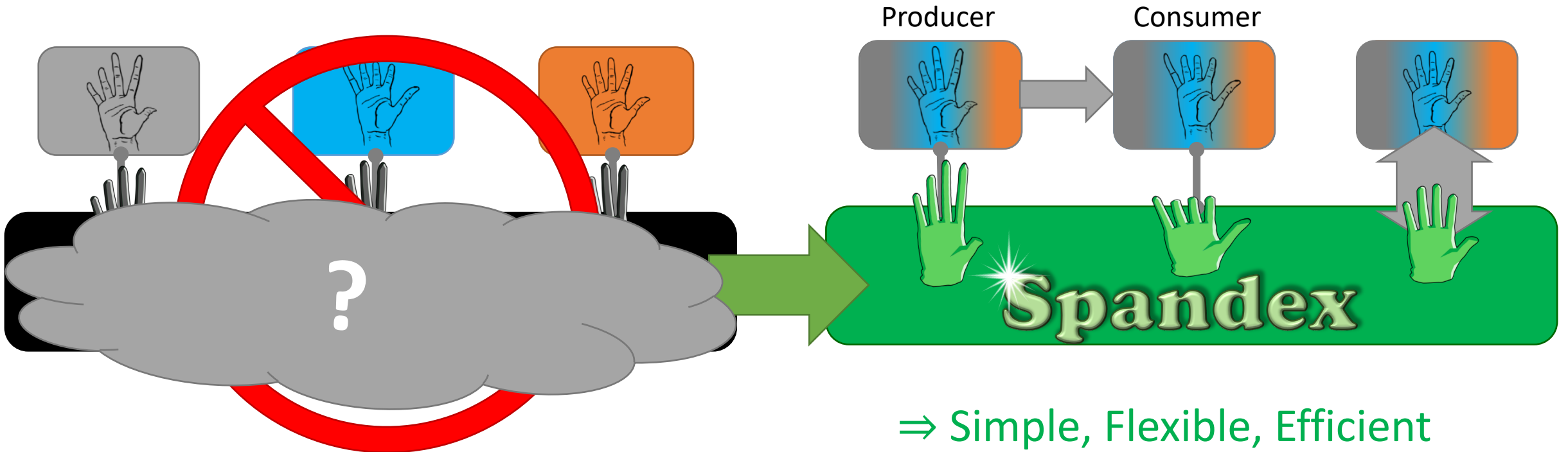
Evaluation: CPU-GPU Applications

- Different workloads prefer different protocols

- Spandex flexibility ⇒ consistently better execution time (avg 16% lower)

Evaluation: CPU-GPU Applications

- **Spandex flexibility ⇒ consistently better NW traffic (avg 27% lower)**

Conclusions and Ongoing Work

Producer   Consumer

⇒ Simple, Flexible, Efficient

Next: Exploit SW or HW hints about data access patterns (coherence specialization)

- Dynamic Spandex request selection
- Producer-consumer forwarding
- Extended granularity flexibility

- With different accelerators and memory structures (e.g., scratchpads, FIFOs)