CCNoC:

Specializing On-Chip Interconnects for Energy Efficiency in Cache-Coherent Servers

Stavros Volos, Ciprian Seiculescu,
Boris Grot, Naser Khosro Pour,
Babak Falsafi, and Giovanni De Micheli

PARALLEL SYSTEMS ARCHITECTURE LAB

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Toward Manycore Tiled Servers

Servers workloads

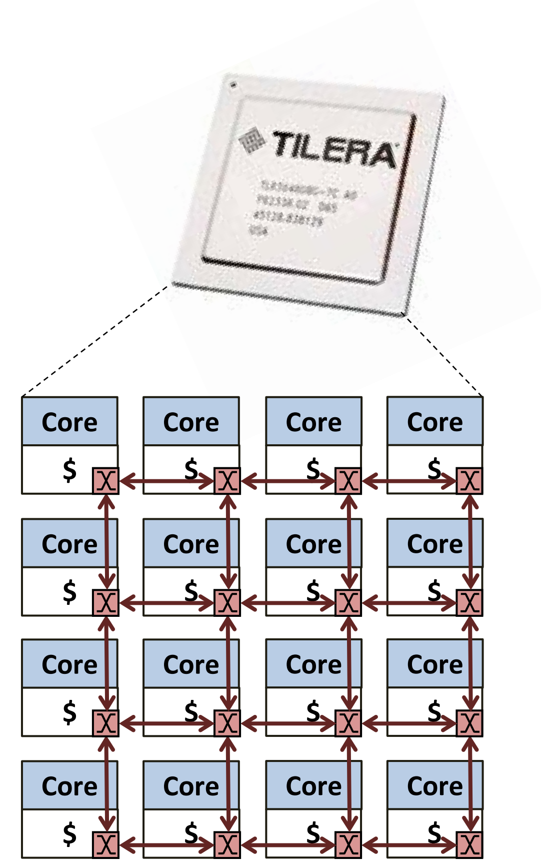
- Many clients using common service
- Manycore chips to maximize throughput

Tiled organizations inherently scalable

- Rely on NoCs for communication

NoCs play pivotal role

- Affect access latency of instructions & data
- Growing area & power footprints



Need efficient NoCs for Server Chips!

© 2012 Stavros Volos

Multi-Network NoCs: The Way to Specialization & Efficiency

Multi- superior to single-network NoCs [Balfour'06]

- Reduce crossbar area & power
- Improve wire utilization

But, multi-network NoC not simple for Servers:

- Cache coherence complicates NoC resource allocation
- Naïve division of networks across traffic is suboptimal

How do we build multi-network NoCs for Servers?

© 2012 Stavros Volos

Our proposal: CCNoC

Bimodal network traffic in server workloads

- Short requests & long responses dominate

CCNoC: dual-network NoC for servers

- Narrow request and wide response networks
- Specialization of router microarchitectures

Compared to homogenous dual-network NoC

- 15% less energy
- 31% less area
- No impact on performance

© 2012 Stavros Volos

Outline

- Overview
- Why Multi-Network NoCs?
- Multi-Network NoCs for Servers
- CCNoC
- Results
- Conclusion

© 2012 Stavros Volos

Why Multi-Network NoCs?

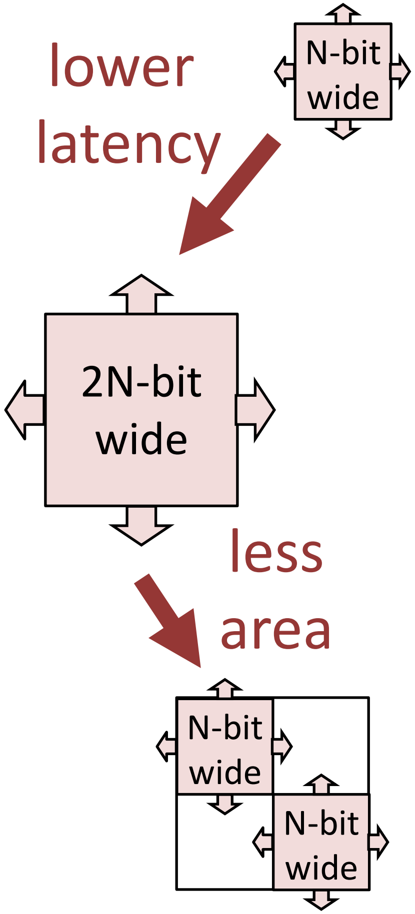
Wider networks reduce packet latency

But, crossbar costs can be prohibitive

- Area: quadratic in network width
- Power: linear in network width
- Utilization: poor on short packets

Multiple networks more efficient [Balfour'06]

- Reduce area & power for fixed NoC bandwidth
- Improve wire utilization



Build multi-network NoCs for Servers

© 2012 Stavros Volos

But, Servers Rely on Cache Coherence

Server software needs shared memory

- Software stacks are complex
- Shared memory facilitates programming
- Enables portability across platforms

Coherence complicates NoC design

- Control and data-carrying messages
- Multiple message classes to enhance protocol performance
  - Need to avoid protocol-level deadlocks

How to split messages across multiple networks?

© 2012 Stavros Volos

Cache Coherence 101: Message Class & Size Glossary

Protocol Message Class		Network Message Size
• Block fetch/evict requests		
– Read, write & upgrade	→	Short (~8 bytes)
– Evict dirty block	→	Long (~72 bytes)
– Evict clean	→	Short
• Coherence requests		
– Downgrade, invalidate	→	Short
• Responses		
– Response with data	→	Long
– Acknowledgements	→	Short

Divide by class, size, or hybrid?

© 2012 Stavros Volos

Divide Networks by Size?

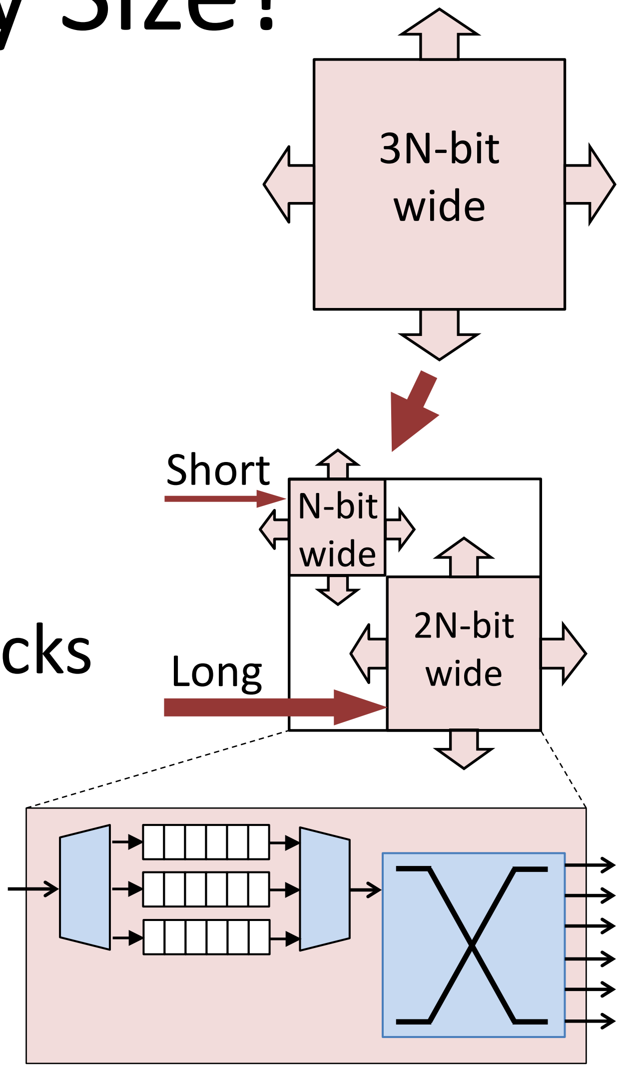
Can specialize network width

- Reduce crossbar area & power

Still need VCs for message classes

… to avoid protocol-induced deadlocks

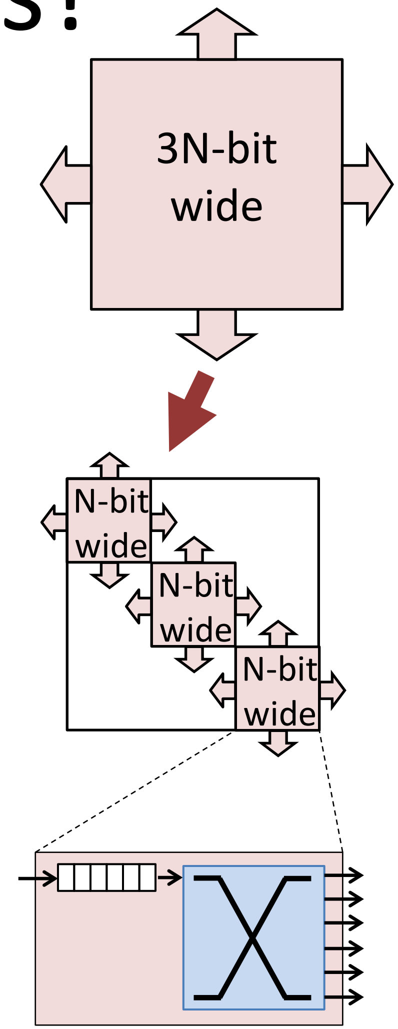
- Increase pipeline complexity
- Add to storage area & power



Network-wide VC overhead

© 2012 Stavros Volos

Divide Networks by Class?



Can eliminate VCs

- Lower complexity and router delay
- Lower buffer requirements

But, difficult to specialize network width

… different message sizes within class

Networks may be underutilized

- Variation in traffic across classes
- Suboptimal designs in cost or performance

Resource over-partitioning

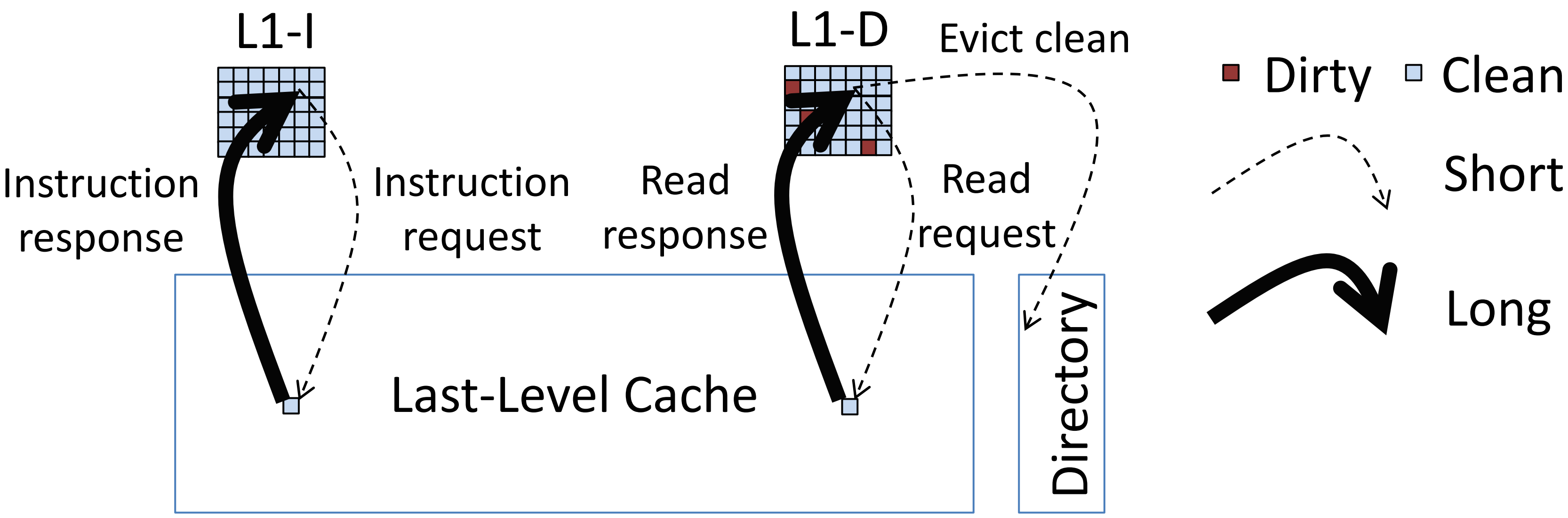
© 2012 Stavros Volos

Can Skewed Traffic Help Division?

Server workloads: [Hardavellas'09, Ferdman'12]

Most traffic → fetch clean blocks from last-level cache

- Instructions: high L1-I miss ratio, read-only
- Data: rarely modified (read mostly)



Short requests & long responses dominant

© 2012 Stavros Volos

Observation (1): Don't Care About Long Requests!

Long requests: dirty block writebacks

- 10% on average; less frequent in server workloads

Writeback latency:

- Not on critical path
- Hidden through buffers & relaxed models

Network efficiency for writebacks not important

© 2012 Stavros Volos

Observation (2): Short Responses are Rare in Servers!

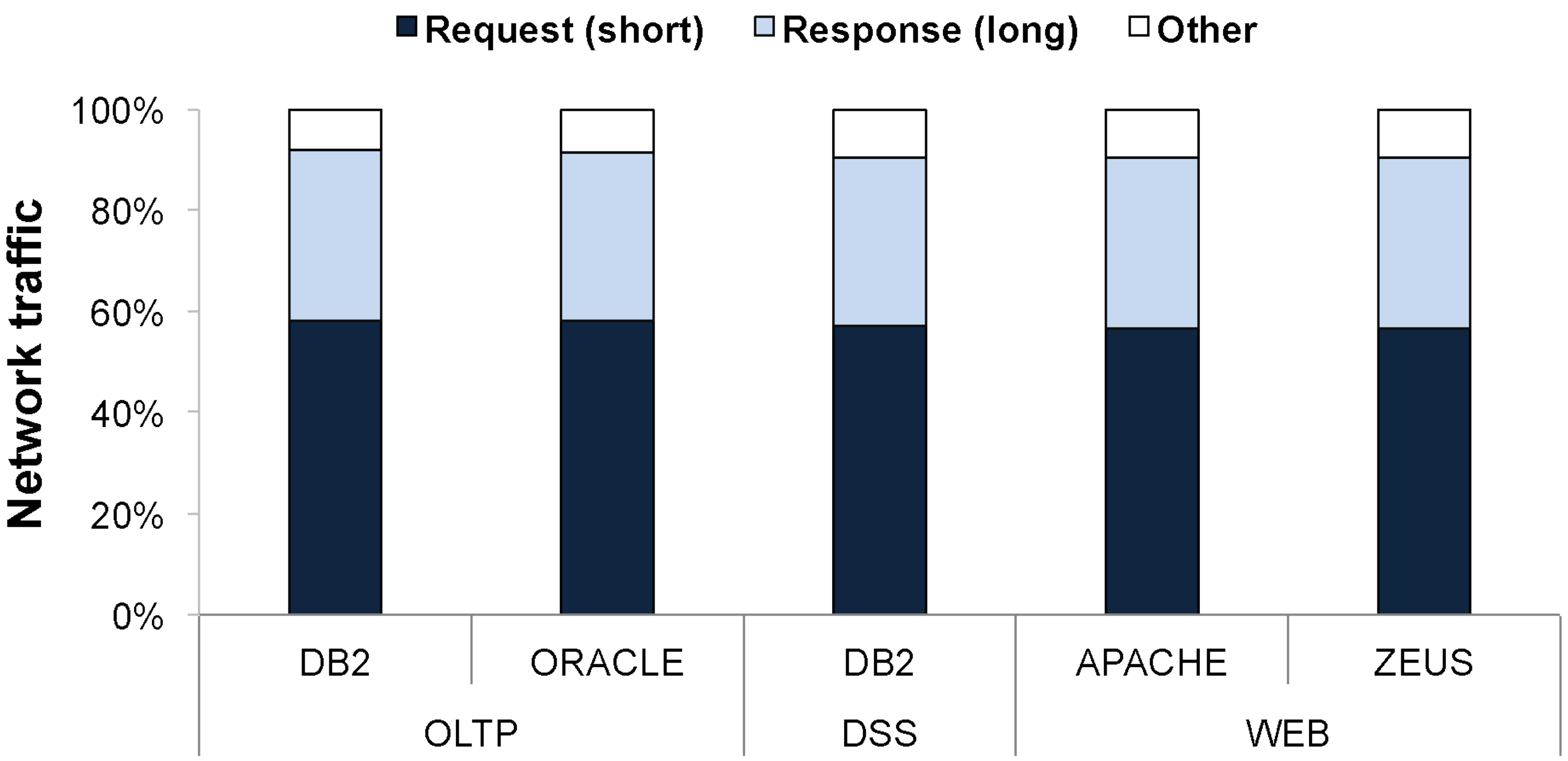
Short responses: Coherence ACK messages

- Instructions are read-only
  - No core-to-core coherence traffic
- Data sharing happens beyond L1 residency
  - Writers rarely modify shared data
  - Core-to-core coherence traffic infrequent

Network efficiency for short responses not critical

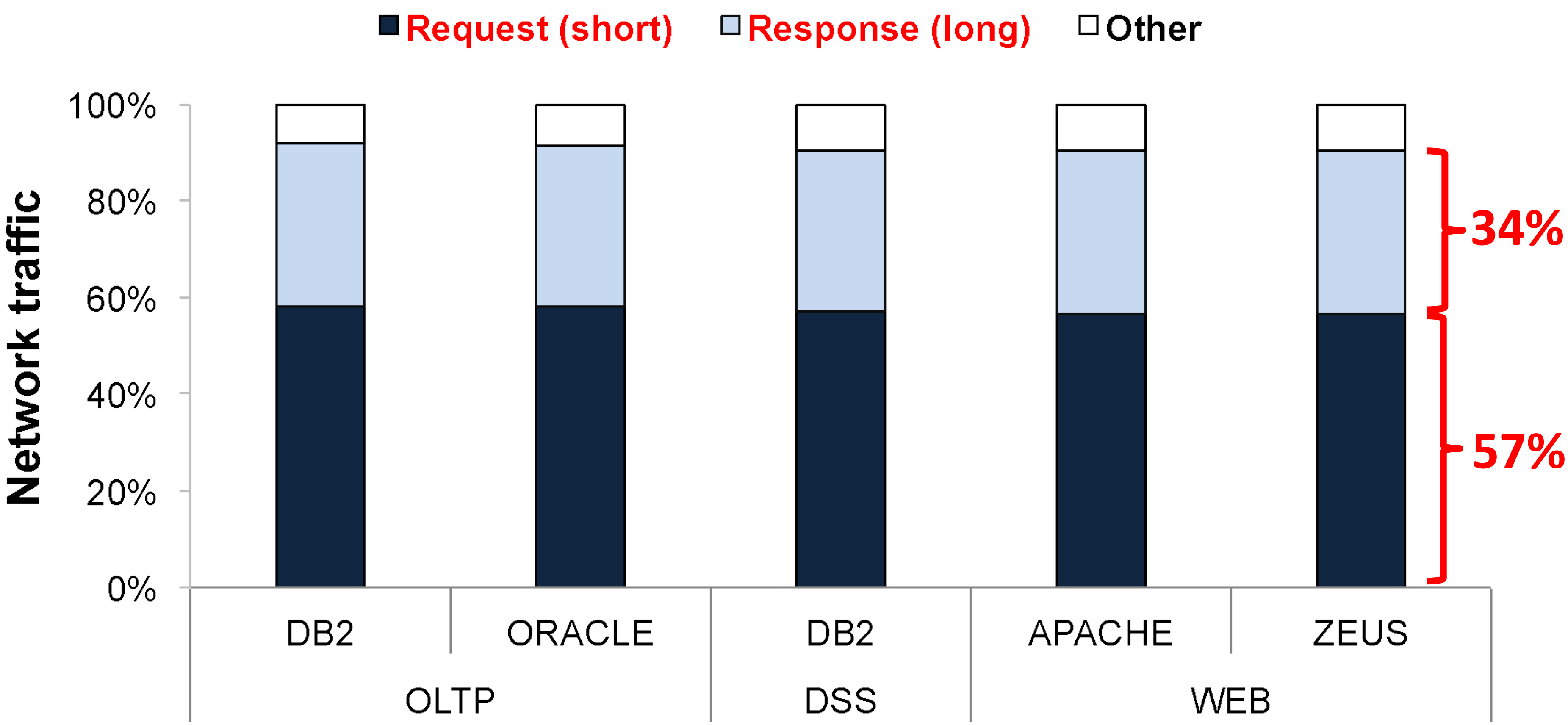
© 2012 Stavros Volos

Characterization of Network Traffic



© 2012 Stavros Volos

Characterization of Network Traffic



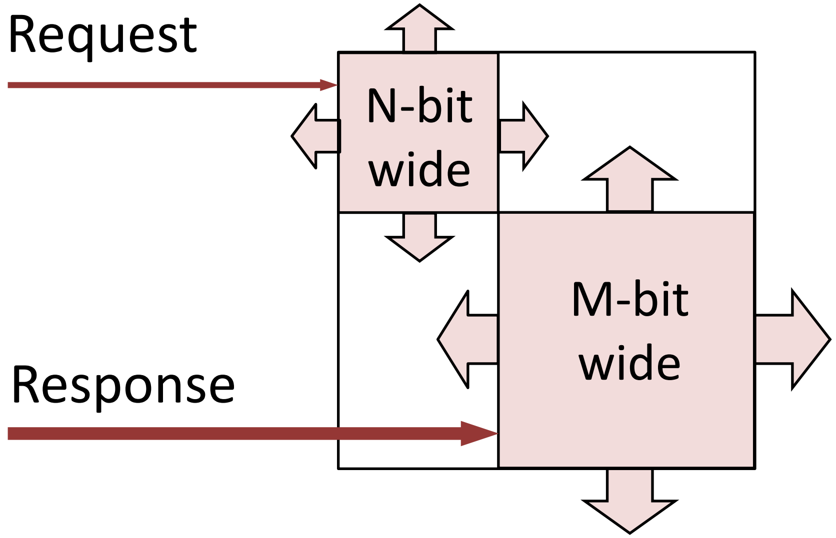
Servers exhibit bimodal network traffic

© 2012 Stavros Volos

Leveraging Bimodal Network Traffic

Recap: Bimodal network traffic in servers

- Short requests (57%), long responses (34%)
- Short responses, coherent requests, long requests (9%)



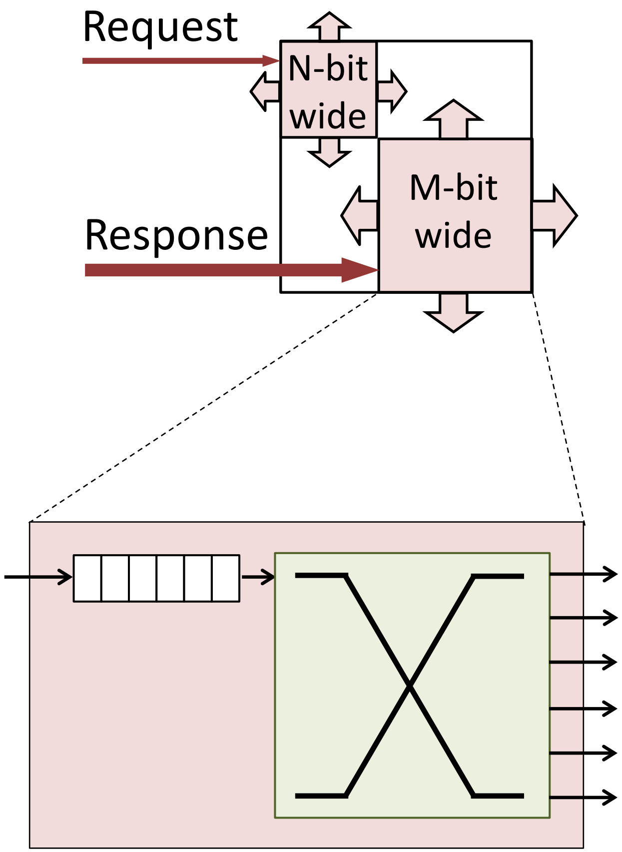
CCNoC: dual-network NoC

- Wide response network
- Narrow request network

© 2012 Stavros Volos

CCNoC Response Network

- Wide datapath
  - Optimized for long responses
- Wormhole flow control
  - No virtual channels (only one class)
  - Reduce cost & complexity
- Two-stage pipeline: XA, XT

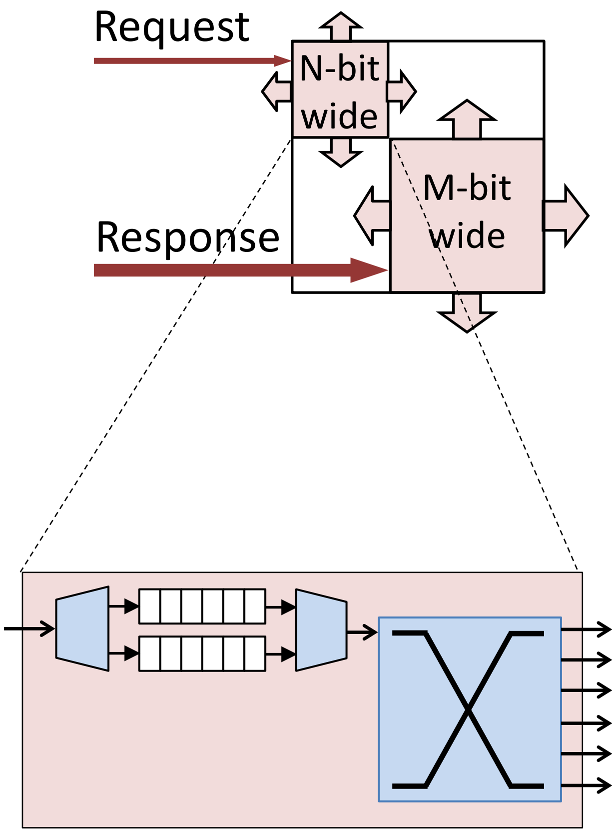


Wide response network: fast and low-cost

© 2012 Stavros Volos

CCNoC Request Network

- Narrow datapath
  - Optimized for short messages
  - Reduce crossbar area & power
- Virtual channel (VC) flow control
  - Avoid protocol-level deadlock among fetch block & coherence requests
- Standard VC-router pipeline
  - Three stages: VA, XA, XT



Narrow request network: VC cost is low

© 2012 Stavros Volos

Outline

- Overview
- Why Multi-Network NoCs?
- Multi-Network NoCs for Servers
- CCNoC
- Results
- Conclusion

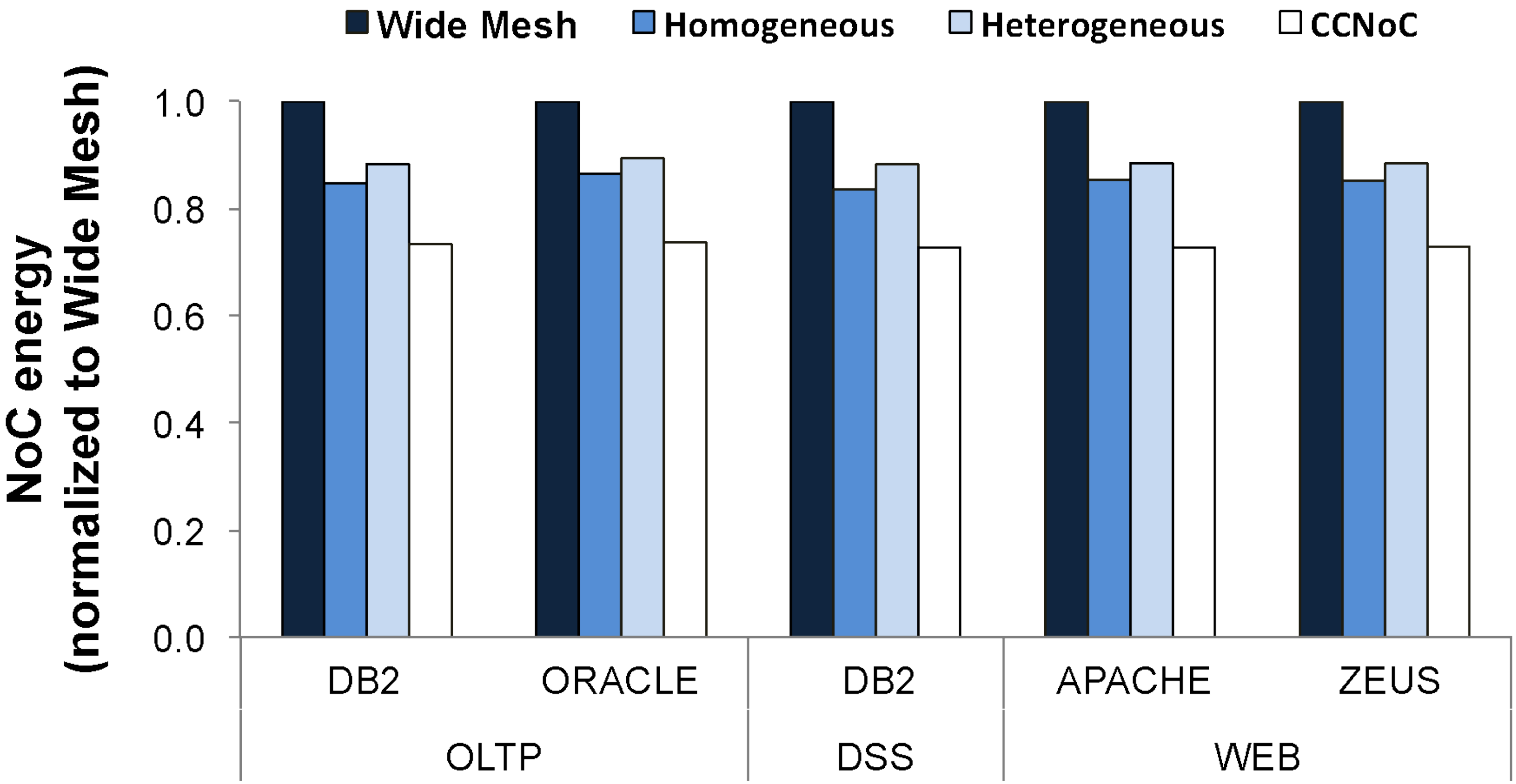
© 2012 Stavros Volos

Methodology

- Flexus [Wenisch'06]
  - Full system simulation
  - 16-core tiled CMP
  - MESI protocol
- Server workloads
  - OLTP, DSS, Web
- Custom power models
- Wide Mesh
  - 176 bits, 3 VCs
- Homogeneous
  - 2x 88 bits, 3 VCs/network
- Heterogeneous
  - Short: 64 bits, 3 VCs
  - Long: 112 bits, 3 VCs
- CCNoC
  - Request: 64 bits, 2 VCs
  - Response: 112 bits, WH

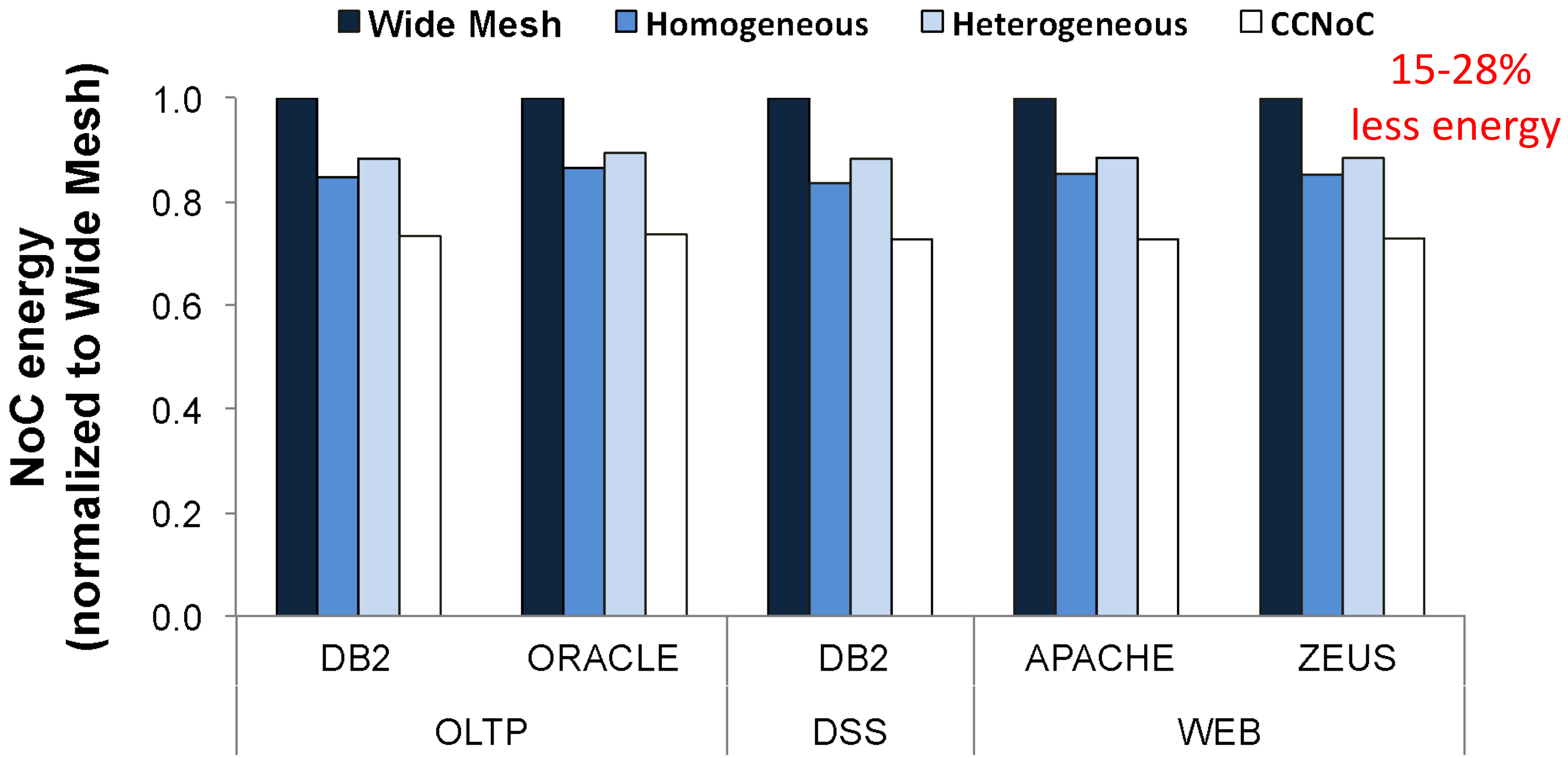
© 2012 Stavros Volos

CCNoC Energy Efficiency



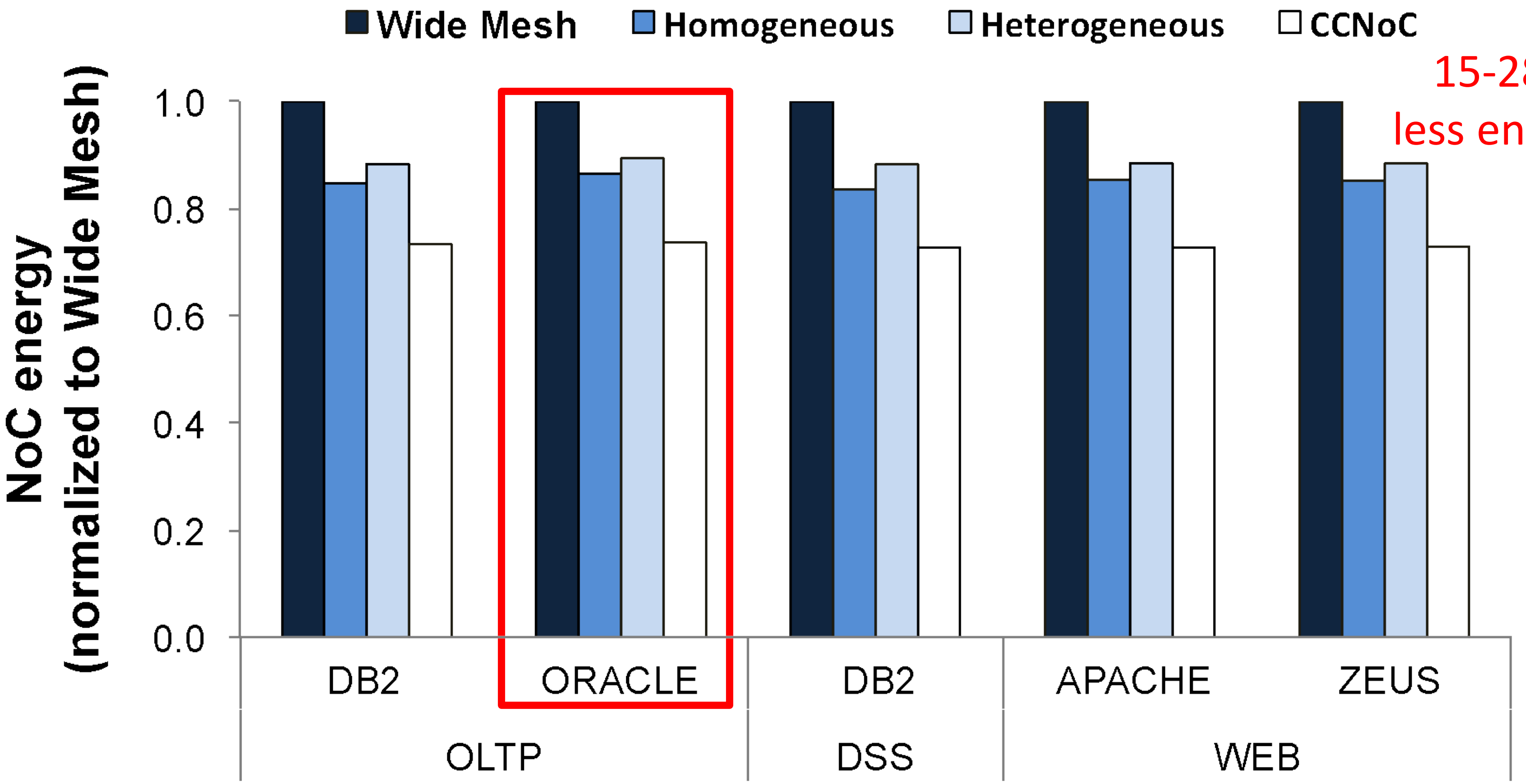
© 2012 Stavros Volos

CCNoC Energy Efficiency



© 2012 Stavros Volos

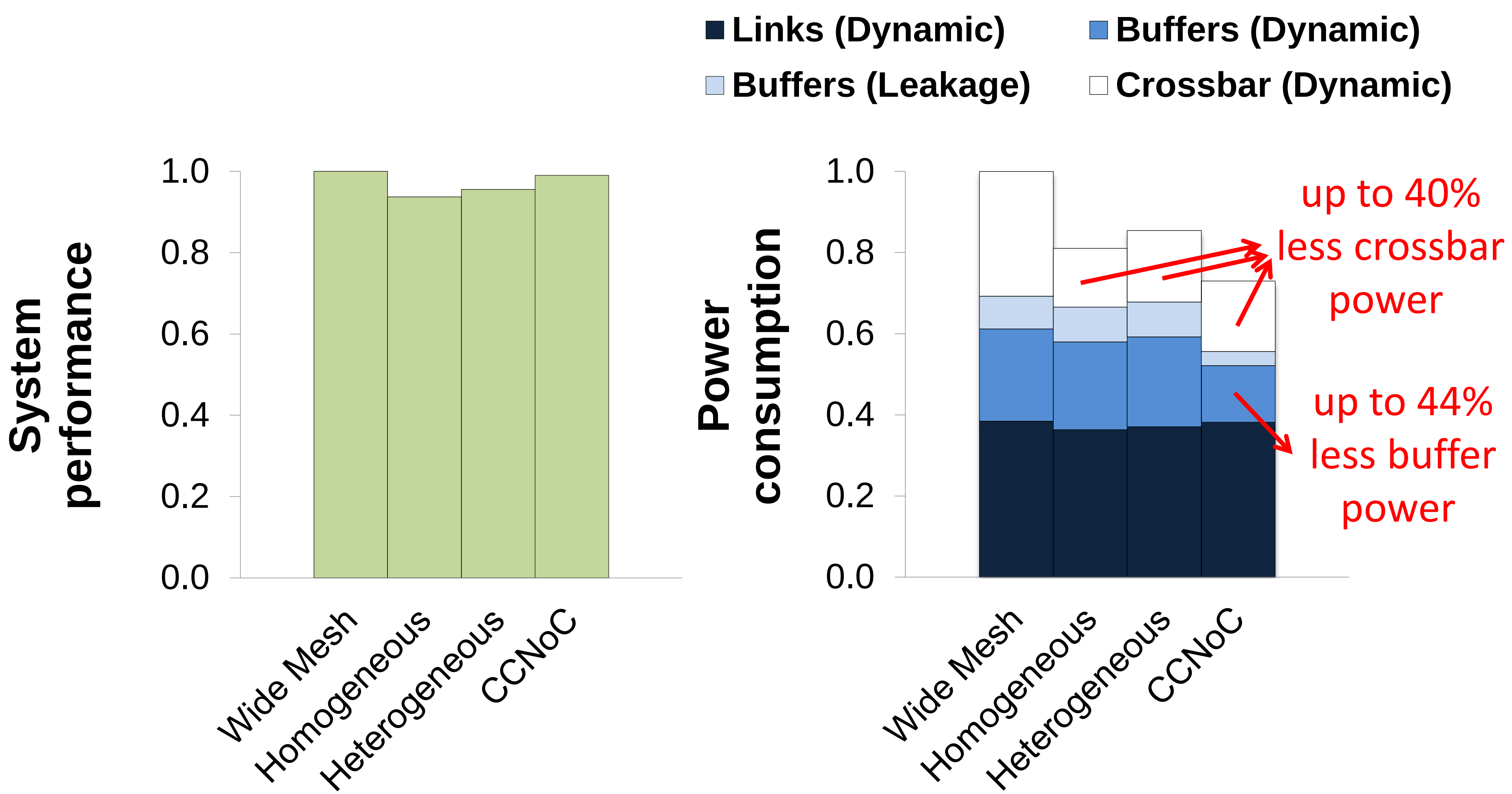
CCNoC Energy Efficiency



15-28% less energy

© 2012 Stavros Volos

CCNoC Efficiency



Significant power savings w/o performance loss

Conclusion

Bimodal network traffic in server workloads

- Short requests & long responses dominate

CCNoC: dual-network NoC for servers

- Narrow request and wide response networks
- Specialization of router microarchitectures

Compared to homogenous dual-network NoC

- 15% less energy
- No impact on performance

© 2012 Stavros Volos

Thanks!

Questions?

For more information,

http://parsa.epfl.ch/visa