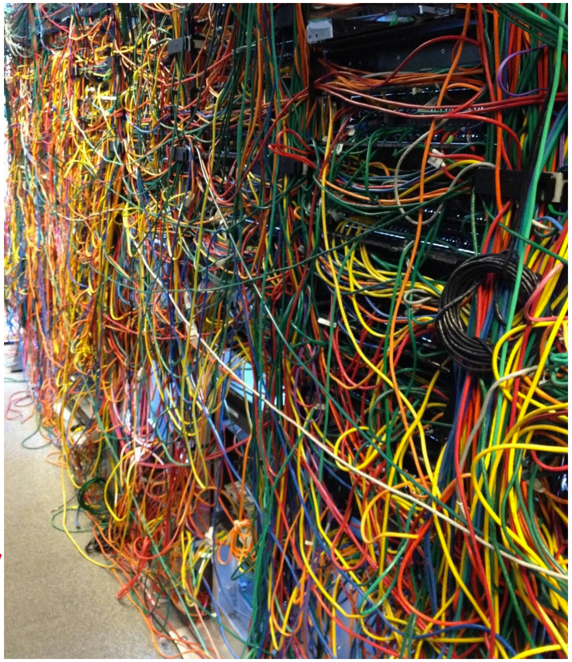
EECS 570

Interconnect - Intro

Winter 2025

Prof. Satish Narayanasamy

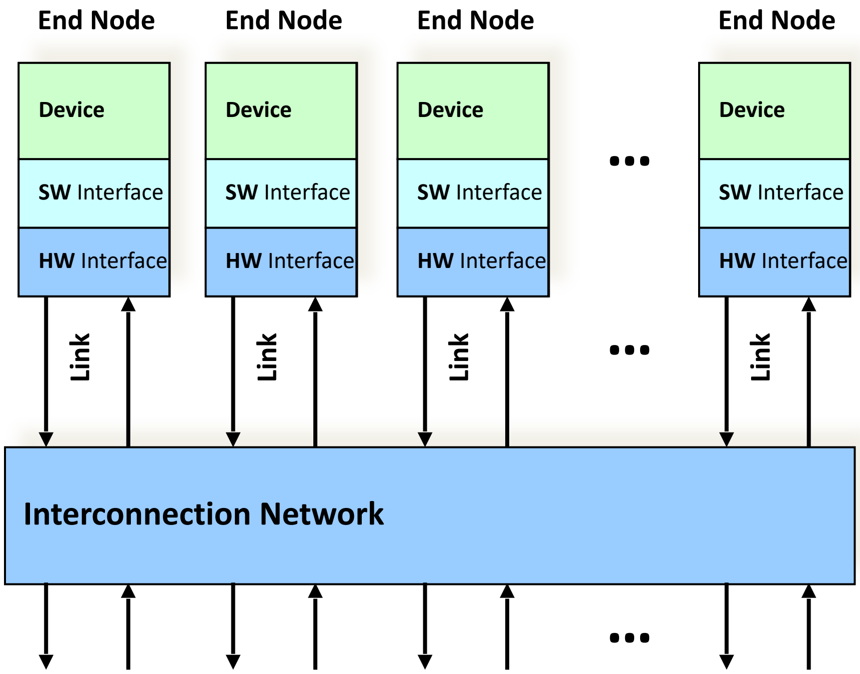
http://www.eecs.umich.edu/courses/eecs570/



Slides developed in part by Profs. Adve, Falsafi, Hill, Lebeck, Martin, Narayanasamy, Nowatzyk, Reinhardt, Singh, Smith, Torrellas and Wenisch. Special acknowledgement to Prof. Jerger of U. Toronto.

Interconnection Networks Introduction

- How to connect individual devices together into a group of communicating devices?
- Device:
  - Component within a computer
  - Single computer
  - System of computers



- Types of elements:
  - end nodes (device + interface)
  - links
  - interconnection network
- Internetworking: interconnection of multiple networks

Interconnection Networks Introduction

- ***Interconnection networks*** *should be designed*
  - *to transfer the maximum amount of information*
  - *within the least amount of time (and cost, power constraints)*
  - *so as not to bottleneck the system*

Types of Interconnection Networks

- Four different domains:
  - Depending on number & proximity of connected devices

- On-Chip networks (OCNs or NoCs)
  - Devices are microarchitectural elements (functional units, register files), caches, directories, processors
  - Latest systems: dozens, hundreds of devices
    - Ex: Intel TeraFLOPS research prototypes – 80 cores
    - Xeon Phi – 60 cores
  - Proximity: millimeters

System/Storage Area Networks (SANs)

- Multiprocessor and multicomputer systems
  - ❒ Interprocessor and processor-memory interconnections
- Server and data center environments
  - ❒ Storage and I/O components
- Hundreds to thousands of devices interconnected
  - ❒ IBM Blue Gene/L supercomputer
    (64K nodes, each with 2 processors)
- Maximum interconnect distance
  - ❒ tens of meters (typical)
  - ❒ a few hundred meters (some)
    - ❍ InfiniBand: 120 Gbps over a distance of 300m
- Examples (standards and proprietary)
  - ❒ InfiniBand, Myrinet, Quadrics, Advanced Switching Interconnect

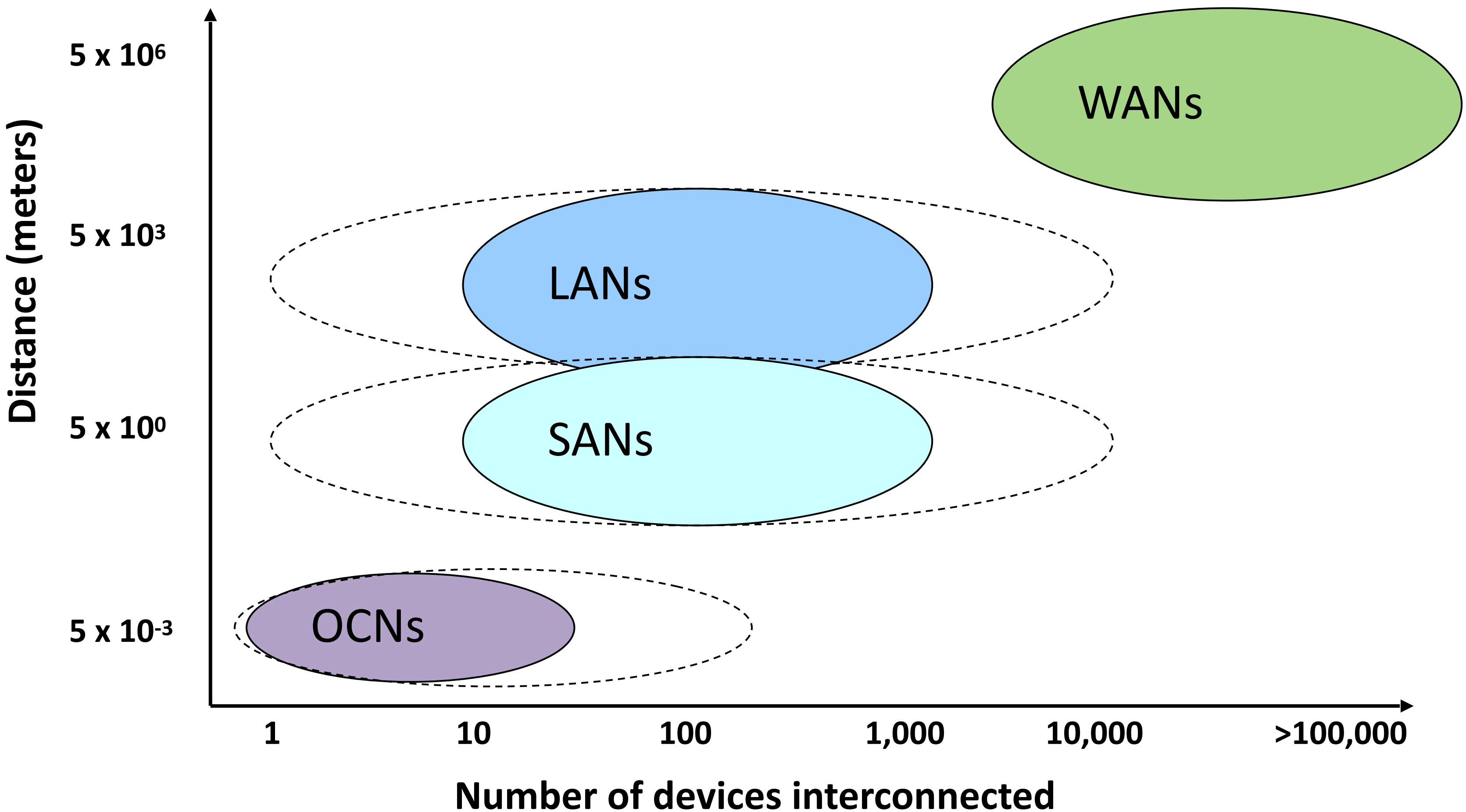
Local Area Network (LANs)

- Interconnect autonomous computer systems
- Machine room or throughout a building or campus
- Hundreds of devices interconnected (1,000s with bridging)
- Maximum interconnect distance
  - few kilometers
  - few tens of kilometers (some)
- Example (most popular): Ethernet, with 10 Gbps over 40Km

Wide Area Networks (WANs)

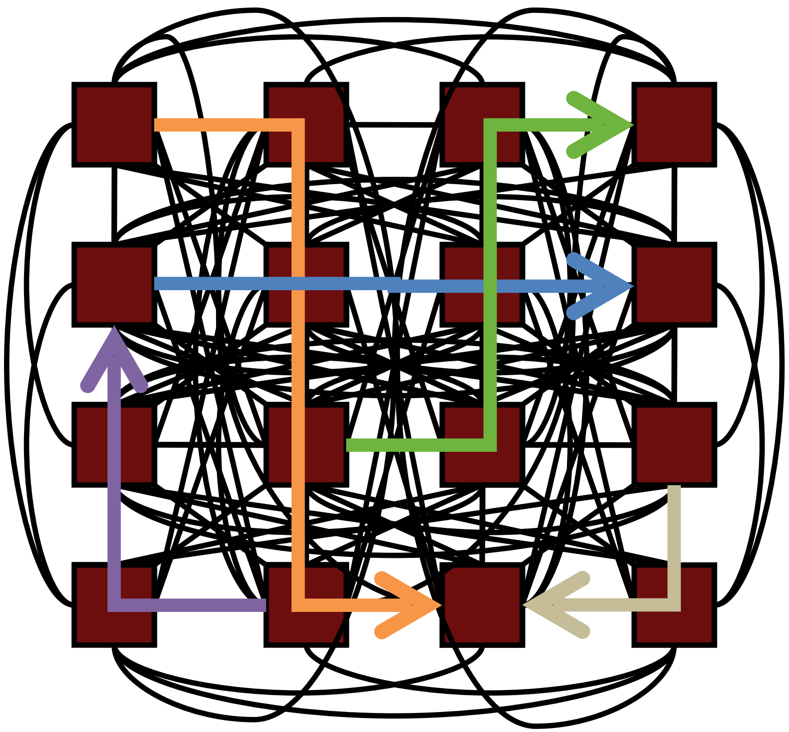
- Interconnect systems distributed across the globe
- Internetworking support is required
- Many millions of devices interconnected
- Maximum interconnect distance
  - many thousands of kilometers
- Example: ATM (asynchronous transfer mode)

Interconnection Network Domains



EECS 570 Focus:
On-Chip Networks

On-Chip Networks (OCN or NoCs)



- Why On-Chip Network?
  - Ad-hoc wiring does not scale beyond a small number of cores
    - Prohibitive area
    - Long latency
- OCN offers
  - scalability
  - efficient multiplexing of communication
  - often modular in nature (eases verification)

Differences between on-chip and off-chip networks

- Significant research in multi-chassis interconnection networks (off-chip)
  - Supercomputers
  - Clusters of workstations
  - Internet routers

- Leverage research and insight but…
  - Constraints are different

Off-chip vs. on-chip

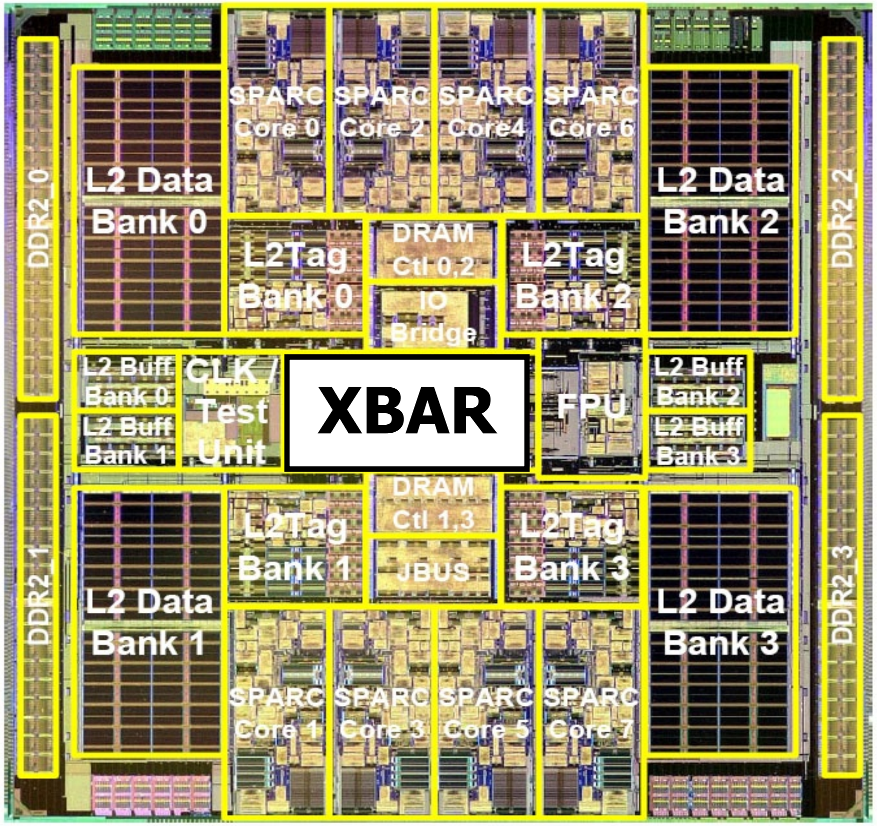
- Off-chip: I/O bottlenecks
  - Pin-limited bandwidth
  - Inherent overheads of off-chip I/O transmission
- On-chip
  - Wiring constraints
    - Metal layer limitations
    - Horizontal and vertical layout
    - Short, fixed length
    - Repeater insertion limits routing of wires
      - Avoid routing over dense logic
      - Impact wiring density
  - Power
    - Consume 10-15% or more of die power budget
  - Latency and bandwidth
    - Different order of magnitude
    - Routers consume significant fraction of latency

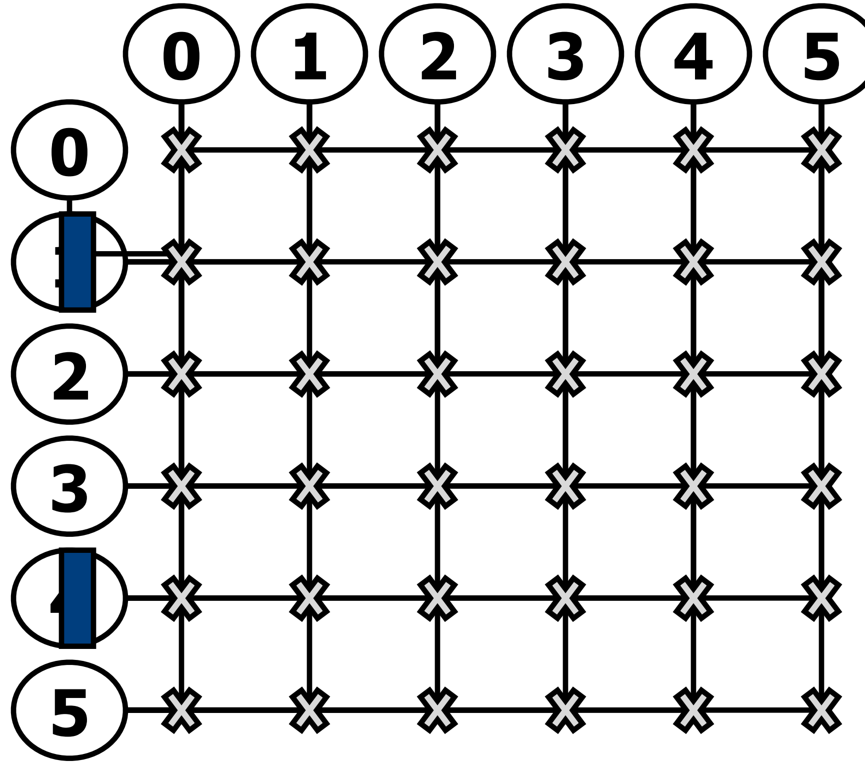
On-Chip Network Evolution

- Ad hoc wiring
  - Small number of nodes

- Buses and Crossbars
  - Simplest variant of on-chip networks
  - Low core counts
  - Like traditional multiprocessors
    - Bus traffic quickly saturates with a modest number of cores
  - Crossbars: higher bandwidth
    - Poor area and power scaling

Multicore Examples (1)

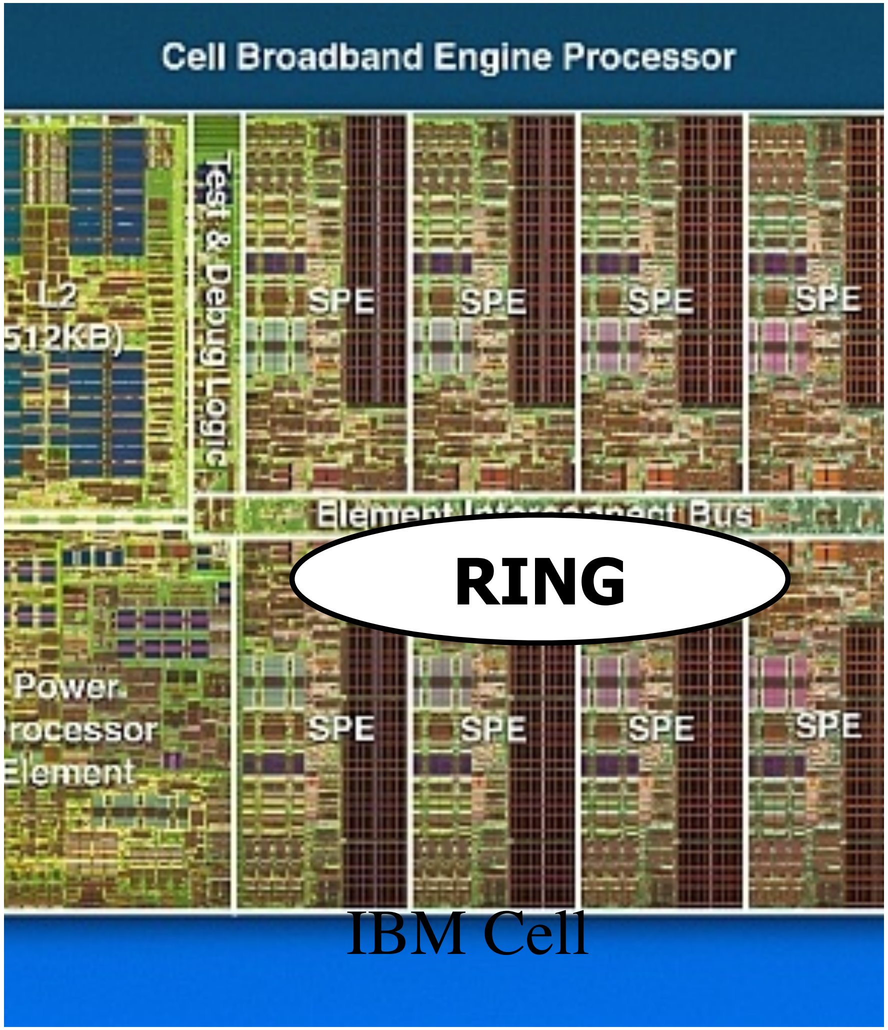




Sun Niagara

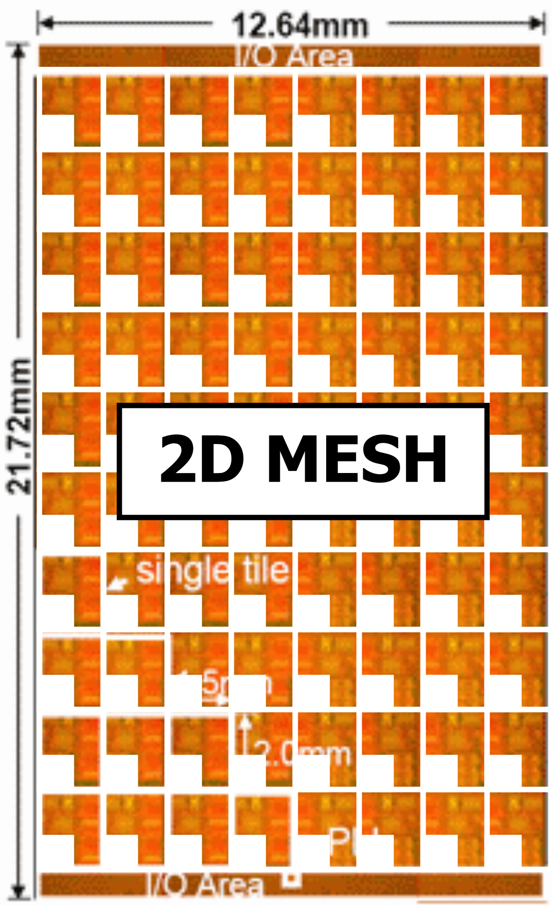
- Niagara 2: 8x9 crossbar (area ~= core)
- Rock: Hierarchical crossbar (5x5 crossbar connecting clusters of 4 cores)

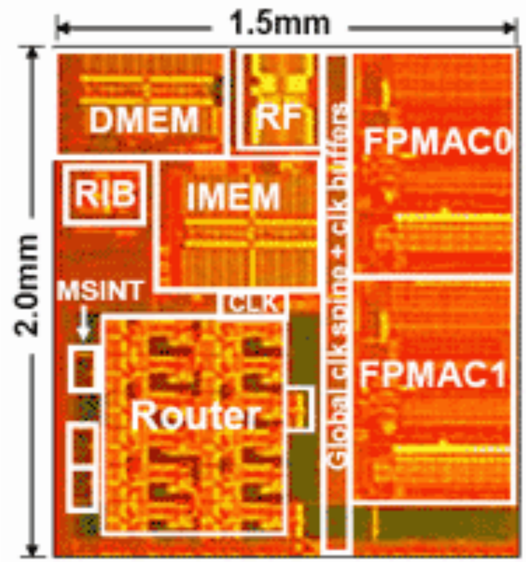
Multicore Examples (2)



- IBM Cell
- Element Interconnect Bus
  - 12 elements
  - 4 unidirectional rings
    - 16 Bytes wide
    - Operates at 1.6 GHz

Many Core Example

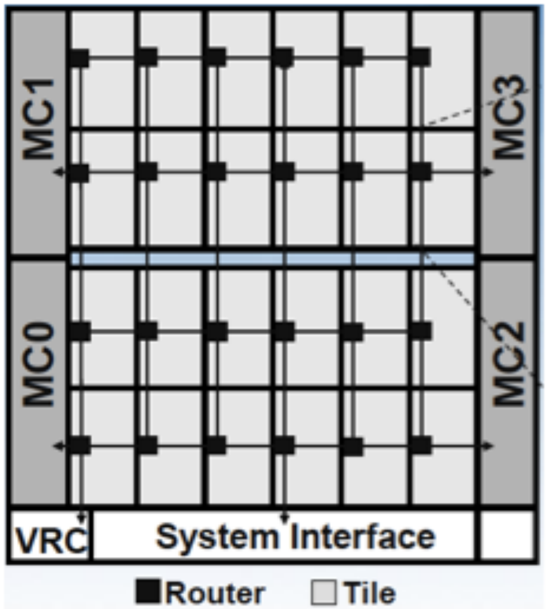


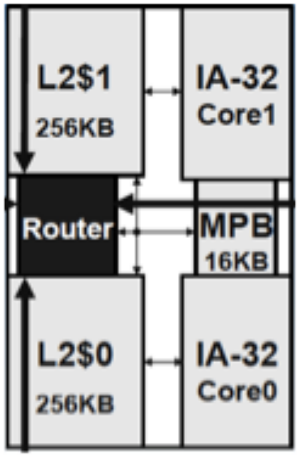


Technology	65nm CMOS Process
Interconnect	1 poly, 8 metal (Cu)
Transistors	100 Million
Die Area	275mm^2
Tile area	3mm^2
Package	1248 pin LGA, 14 layers, 343 signal pins

- Intel TeraFLOPS
  - 80 core prototype
  - 5 GHz
  - Each tile:
    - Processing engine + on-chip network router

Many-Core Example (2): Intel SCC

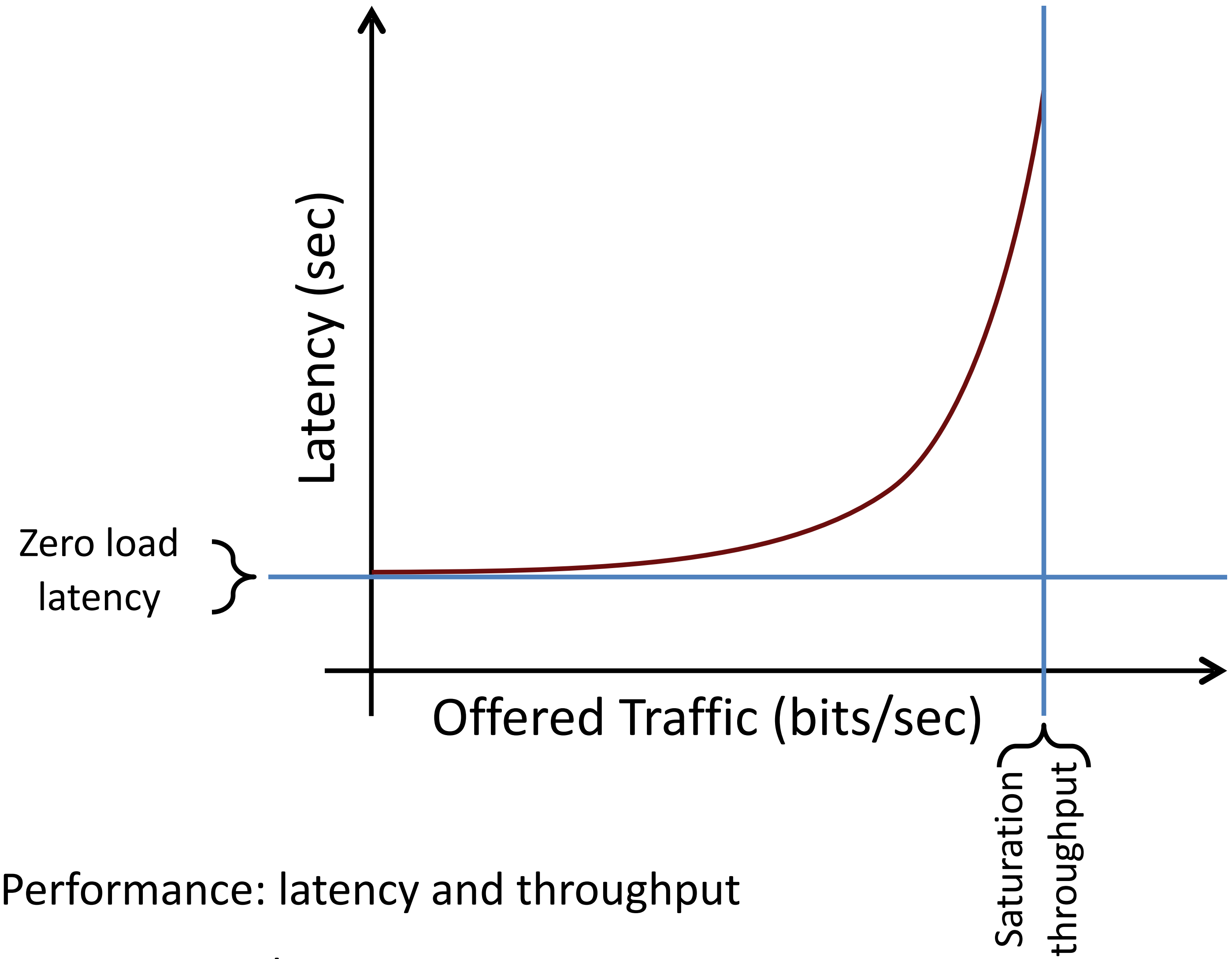




Courtesy: Jason Howard, Intel

- Intel's Single-chip Cloud Computer (SCC) uses a 2D mesh with state of the art routers

Performance and Cost



- Performance: latency and throughput
- Cost: area and power

Topics to be covered

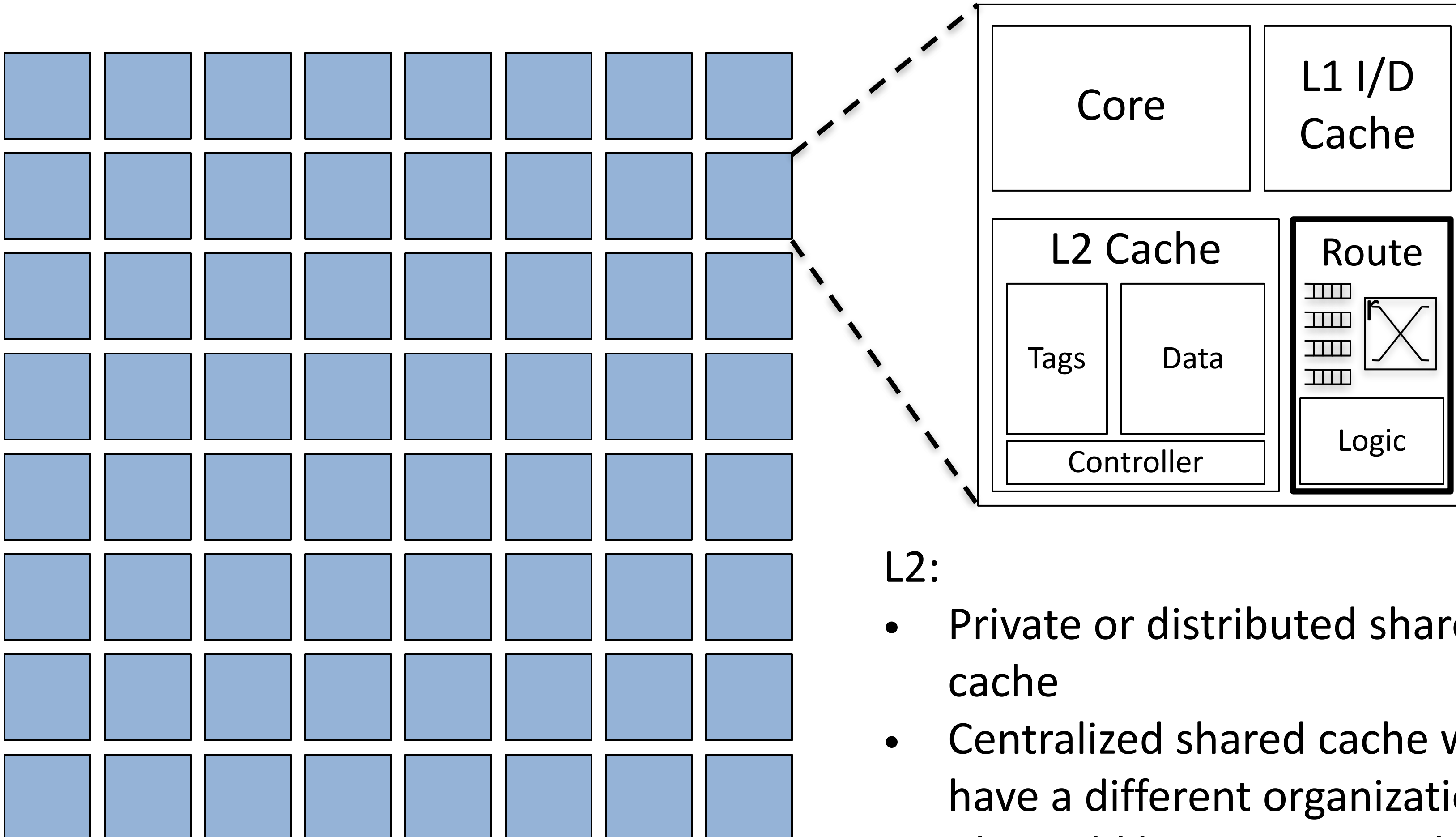
- Interfaces
- Topology
- Routing
- Flow Control
- Router Microarchitecture

System Interfaces

Systems and Interfaces

- Look at how systems interact and interface with network
- Types of multi-processors
  - Shared-memory
    - From high end servers to embedded products
  - Message passing
    - Multiprocessor System on Chip (MPSoC)
      - Mobile consumer market
    - Clusters
- We focus on on-chip networks for shared-memory multi-core

Shared Memory CMP Architecture



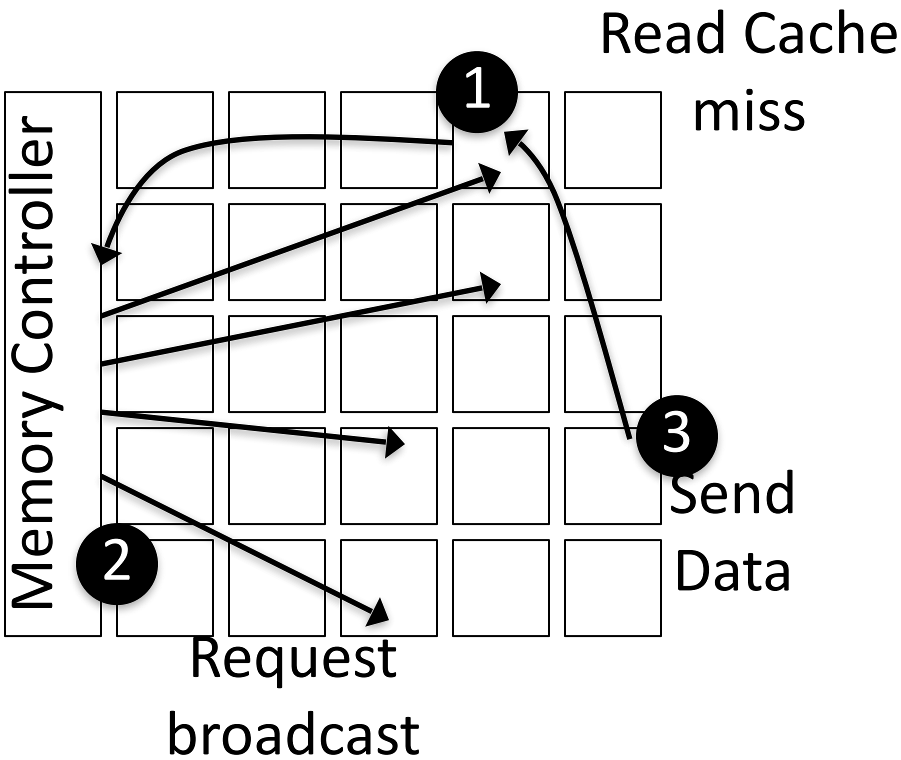
L2:

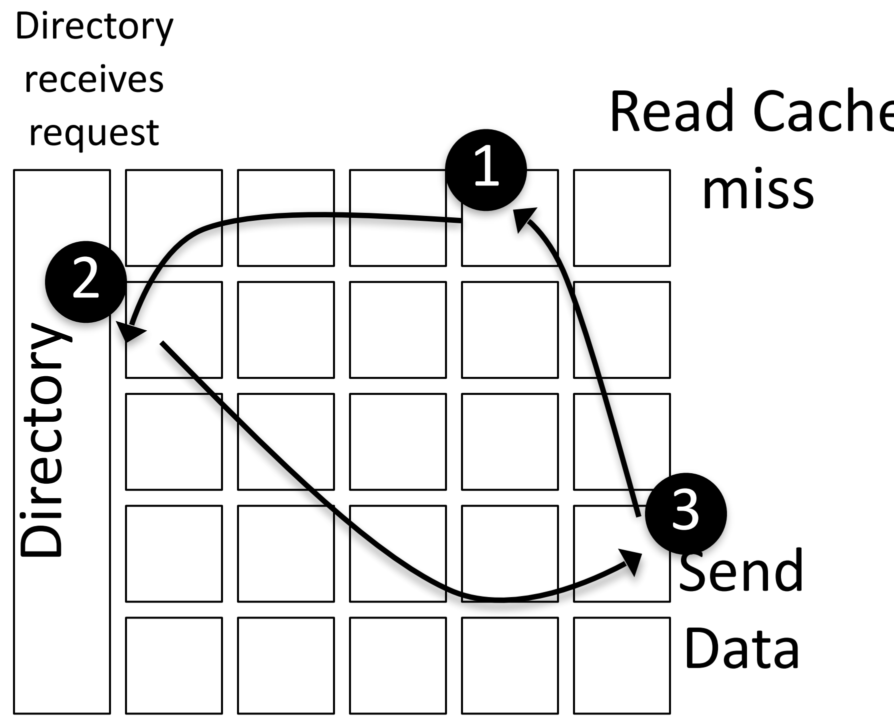
- Private or distributed shared cache
- Centralized shared cache will have a different organization; a tile could be a core or L2 bank

Impact of Coherence Protocol on Network Performance

- Coherence protocol shapes communication needed by system
- Single writer, multiple reader invariant
- Requires:
  - Data requests
  - Data responses
  - Coherence permissions

Broadcast vs. Directory

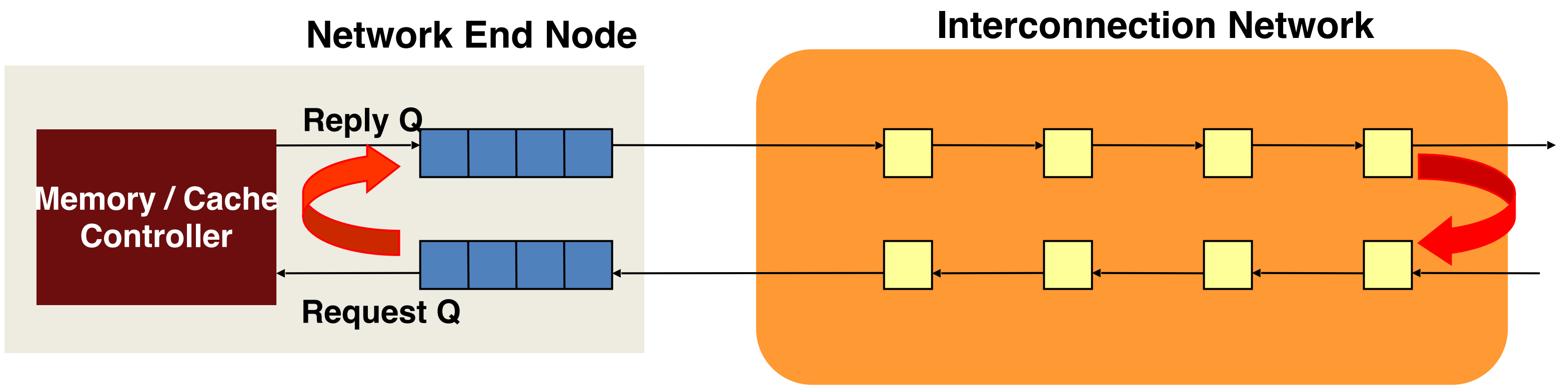




Coherence Protocol Requirements

- Different message types
  - Unicast, multicast, broadcast
- Directory protocol
  - Majority of requests: Unicast
    - **Lower bandwidth** demands on network
  - More scalable due to point-to-point communication
- Broadcast protocol
  - Majority of requests: Broadcast
    - **Higher bandwidth** demands
  - Often rely on **network ordering**

Protocol Level Deadlock



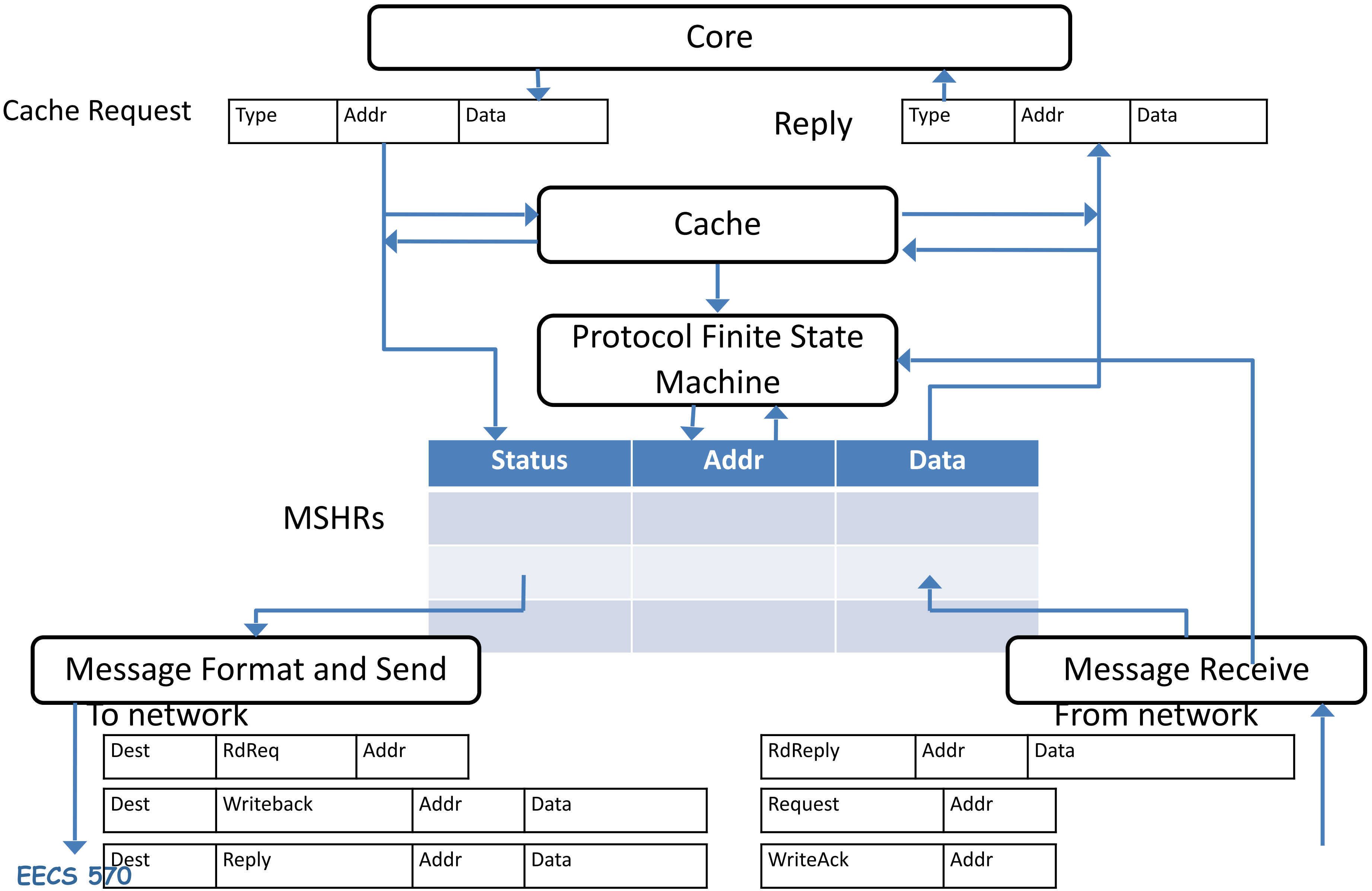
- Request-Reply Dependency
  - Network becomes flooded with requests that cannot be consumed until the network interface has generated a reply
- Deadlock dependency between multiple message classes
- Virtual channels can prevent protocol level deadlock (to be discussed later)

Home Node/Memory Controller Issues

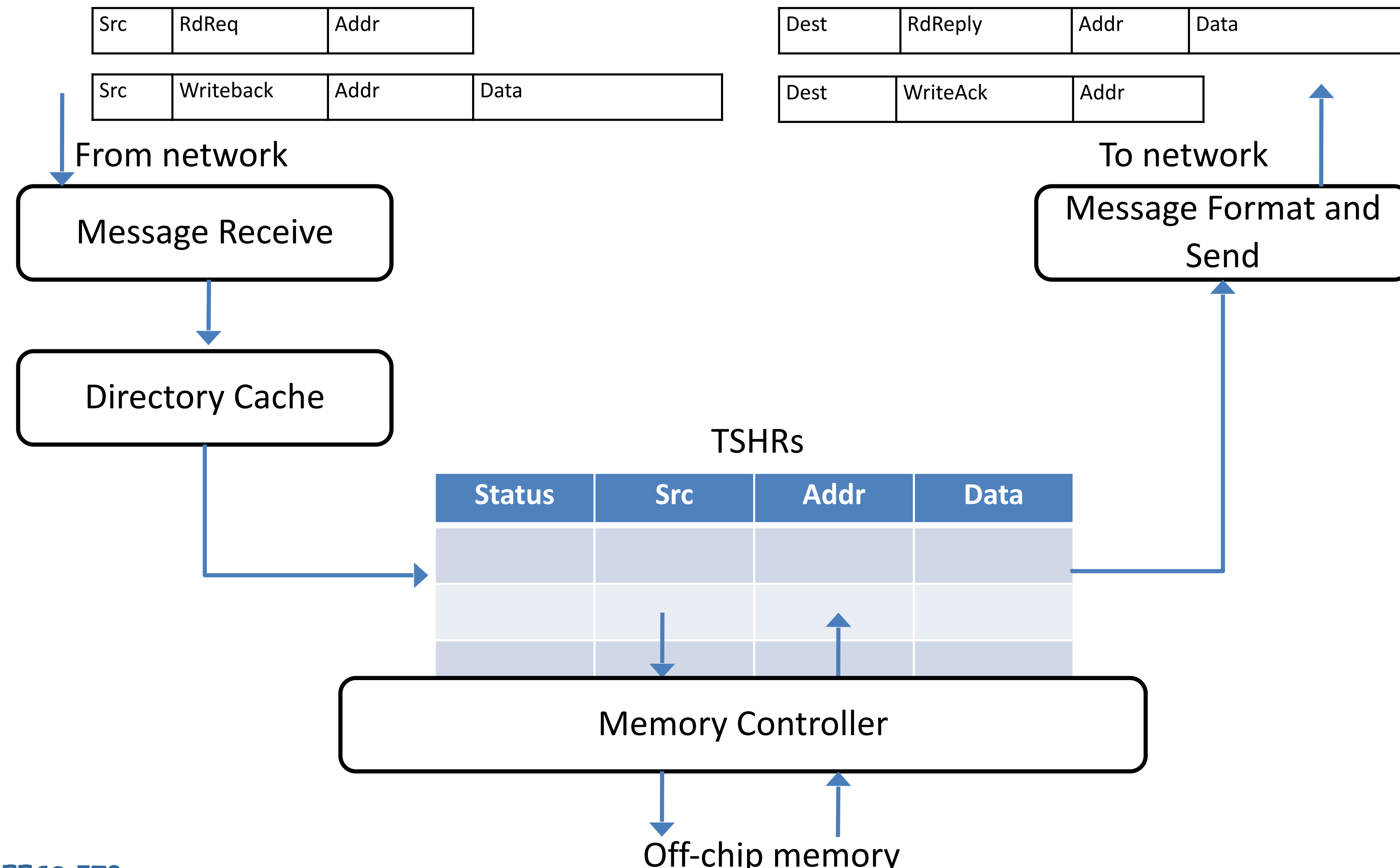
- Heterogeneity in network
  - Some tiles are memory controllers
    - **Co-located** with processor/cache or **separate** tile
    - Share injection/ejection bandwidth?
- Home node
  - Directory coherence information
  - <= number of tiles
- Potential hot spots in network?

Network Interface

Network Interface: Miss Status Handling Registers



Transaction Status Handling Registers
(for centralized directory)



MPSoCs

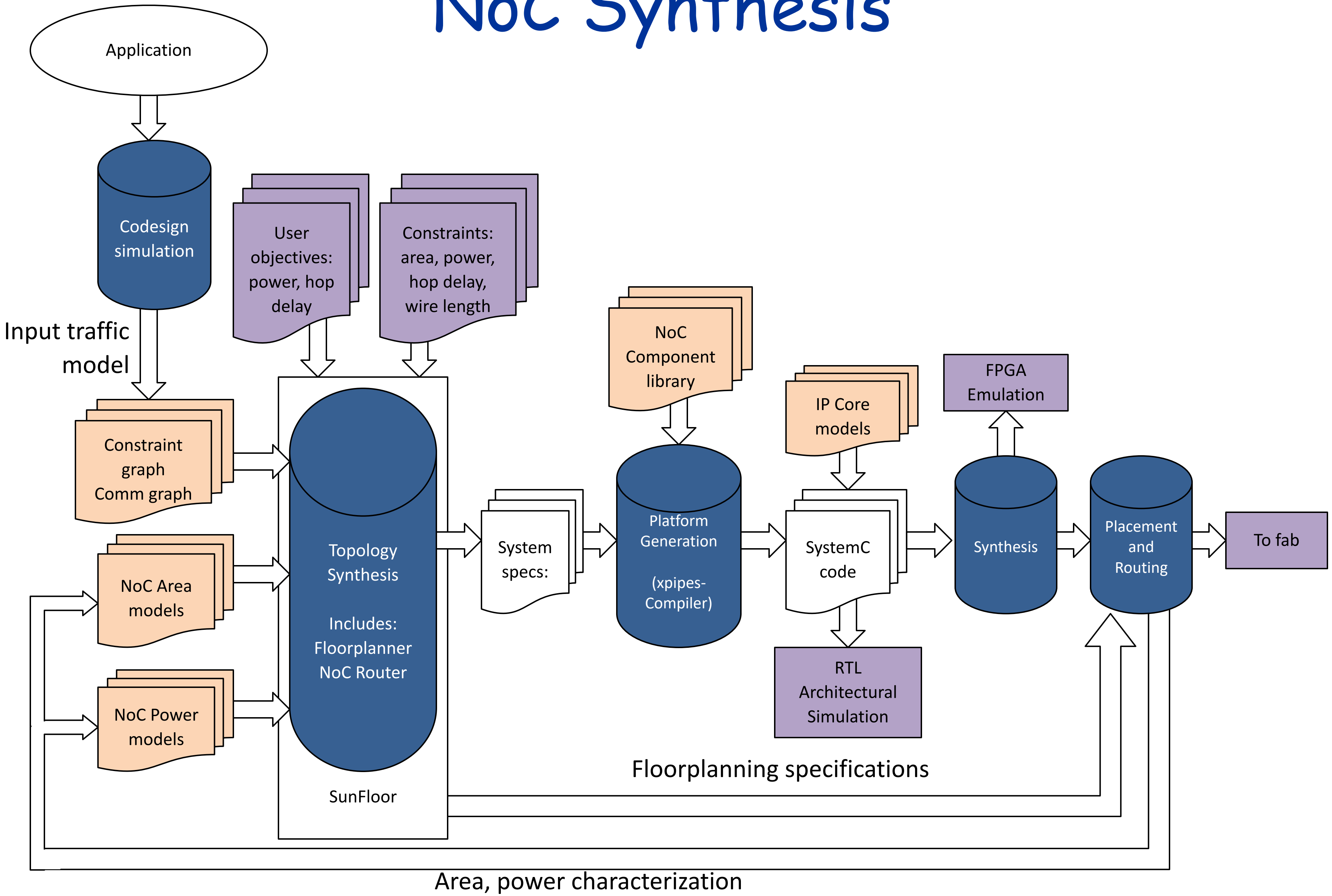
Synthesized NoCs for MPSoCs

- System-on-Chip (SoC)
  - Chips tailored to specific applications or domains
  - Designed quickly through composition of IP blocks
- Fundamental NoC concepts applicable to both CMP and MPSoC
- Key characteristics
  - Applications known **a priori**
  - **Automated** design process
  - **Standardized** interfaces
  - Area/power constraints tighter

Design Requirements

- Less aggressive
  - CMPs: GHz clock frequencies
  - MPSoCs: MHz clock frequencies
  - Pipelining may not be necessary
  - Standardizes interfaces add significant delay
- Area and power
  - CMPs: 100W for server
  - MPSoC: several watts only
- Time to market
  - Automatic composition and generation

NoC Synthesis



NoC Synthesis

- Tool chain
  - Requires accurate power and area models
  - Quickly iterate through many designs
  - Library of soft macros for all NoC building blocks
  - Floorplanner
    - Determine router locations
    - Determine link lengths (delay)

NoC Network Interface Standards

- Standardized protocols
  - Plug and play with different IP blocks
- Bus-based semantics
  - Widely used
- Out of order transactions
  - Relax strict bus ordering semantics
  - Migrating MPSoCs from buses to NoCs.

Summary

- Architecture
  - Impacts communication requirements
  - Coherence protocol: Broadcast vs. Directory
  - Shared vs. Private Caches
- CMP vs. MPSoC
  - General vs. Application specific
  - Custom interfaces vs. standardized interfaces

Topics to be covered

- Interfaces
- Topology
- Routing
- Flow Control
- Router Microarchitecture

Types of Topologies

Types of Topologies

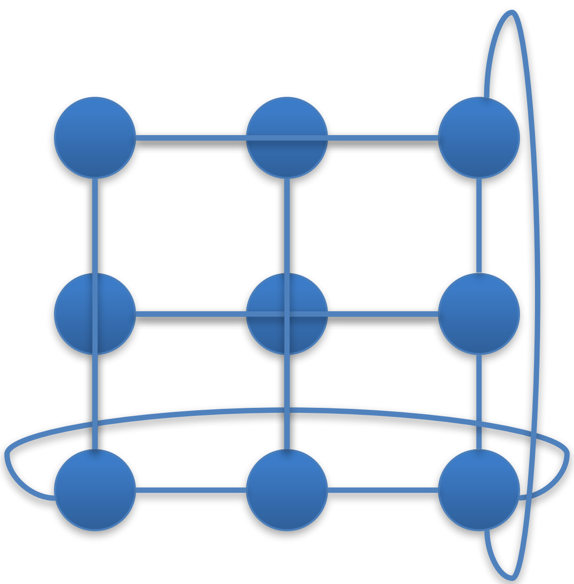
- Focus on switched topologies
  - Alternatives: bus and crossbar
  - Bus
    - Connects a set of components to a single shared channel
    - Effective broadcast medium
  - Crossbar
    - Directly connects ***n*** inputs to ***m*** outputs without intermediate stages
    - Fully connected, single hop network
    - Component of routers

Types of Topologies

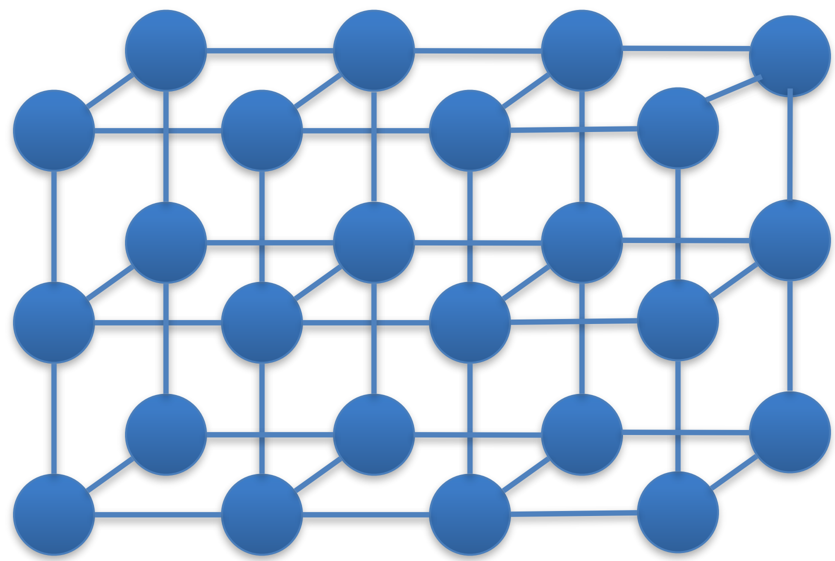
- **Direct**
  - Each router is associated with a terminal node
  - *All* routers are sources and destinations of traffic
- **Indirect**
  - Routers are distinct from terminal nodes
  - Terminal nodes can source/sink traffic
  - Intermediate nodes switch traffic between terminal nodes
- Most on-chip network use direct topologies

Torus (1)

- K-ary n-cube: k^n network nodes
- n-Dimensional grid with k nodes in each dimension



3-ary 2-mesh
3-ary 2-cube



2,3,4-ary 3-mesh

Torus (2)

- 1D or 2D torus map well to planar substrate for on-chip
- Topologies in Torus Family
  - Ex: Ring -- k-ary 1-cube
- Edge Symmetric
  - Good for load balancing
  - Removing wrap-around links for mesh loses edge symmetry
    - More traffic concentrated on center channels
- Good path diversity
- Exploit locality for near-neighbor traffic

Torus (3)

- Degree = 2n, 2 channels per dimension
  - All nodes have same degree
- Total channels = 2nN
  - N is total number of nodes

Mesh

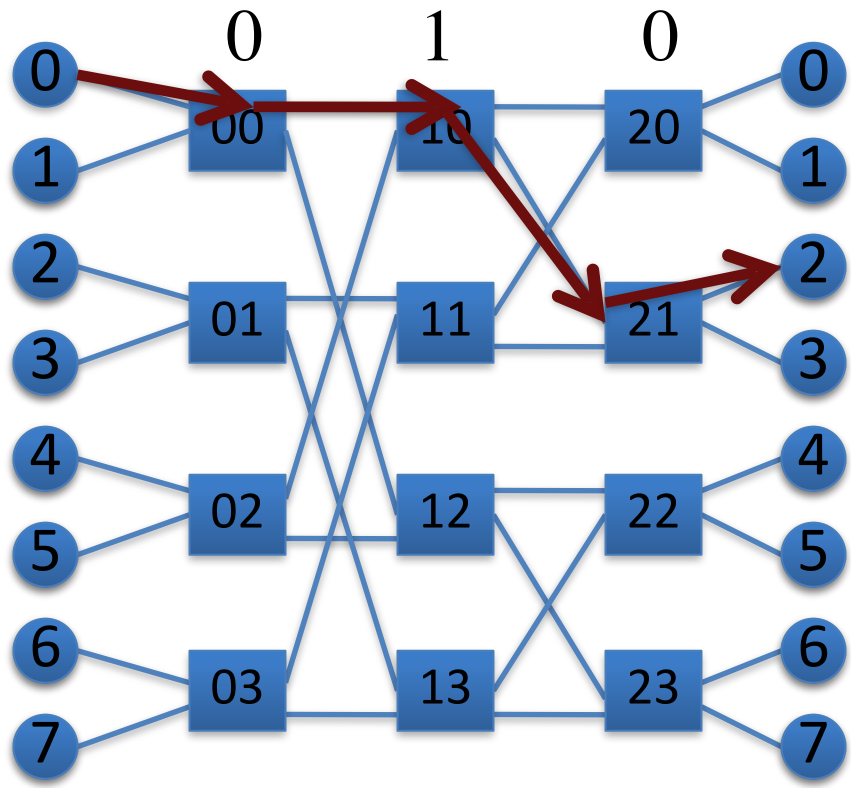
- A torus with end-around connection removed

- Same max node degree

- Higher demand for central channels
  - Load imbalance

Butterfly

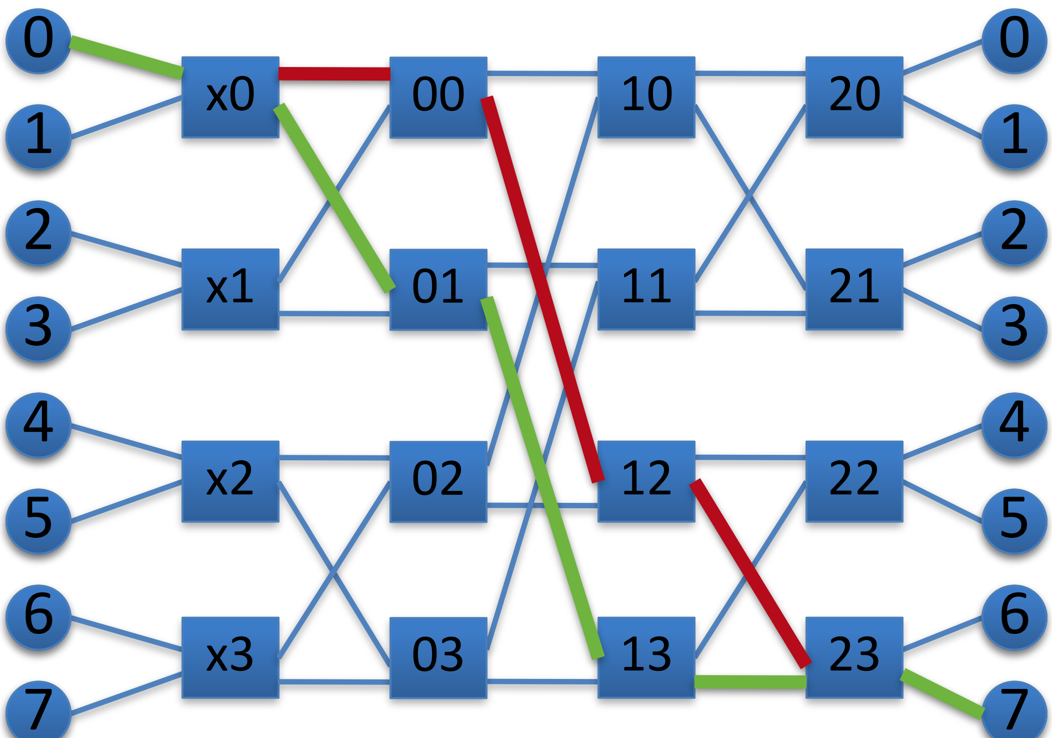
- Indirect network
- K-ary n-fly: k^n network nodes
- Every source-dest pair has the same hop count
- Routing from 000 to 010
  - Dest address used to directly route packet
  - Bit *n* used to select output port at stage *n*



2-ary 3-fly
2 input switch, 3 stages

Butterfly (2)

- No path diversity
  - Can add extra stages for diversity
    - Increase network diameter



Butterfly (3)

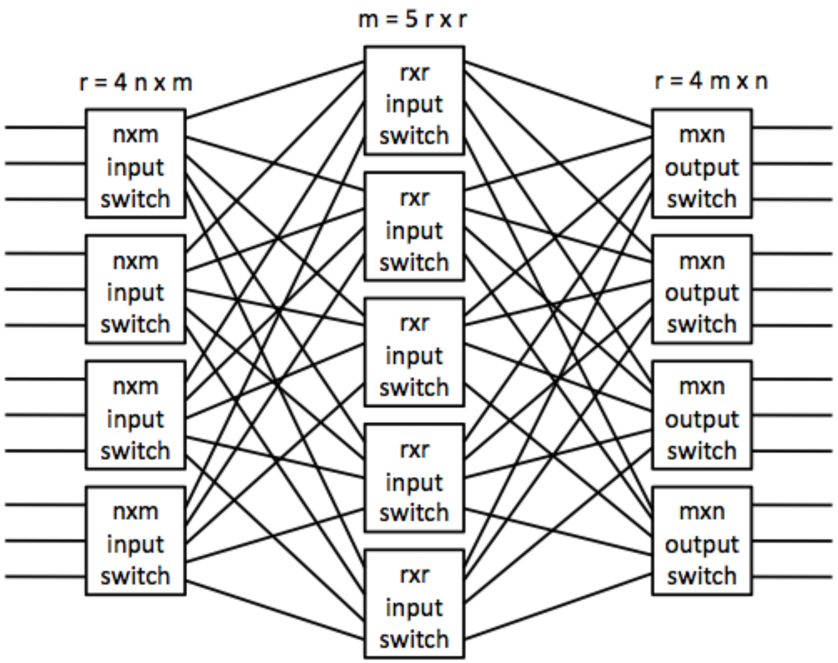
- Hop Count
  - $\text{Log}_k N + 1 = n + 1$ ($N = k^n$ = total number of terminal nodes)
  - Does **not** exploit **locality**
    - Hop count same regardless of location

- Switch Degree = 2k

- Requires long wires to implement

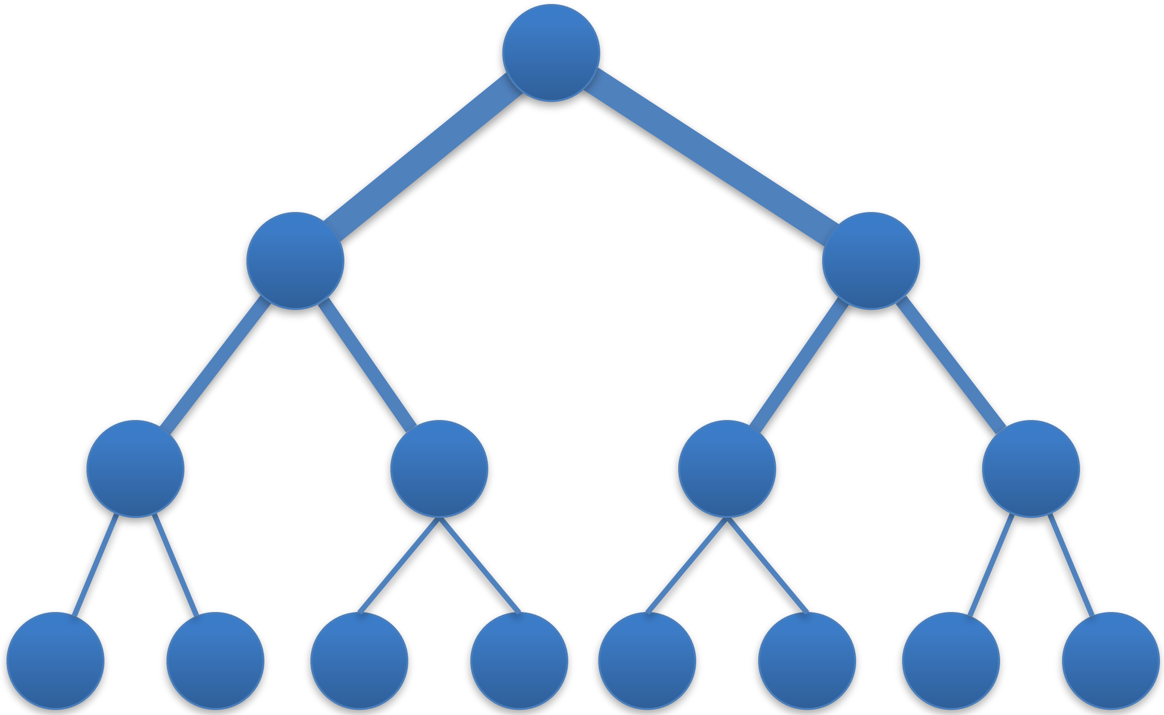
Clos network

- 3-stage networks where all input/output nodes are connected to all middle routers
- Key attribute: path diversity
  - Input node can select any middle router
  - Can enable non-blocking routing algorithms
- (m, n, r)
  - m = Number of middle stage switches
  - n = input/output ports per input/output switch
  - r = number of input/output switches



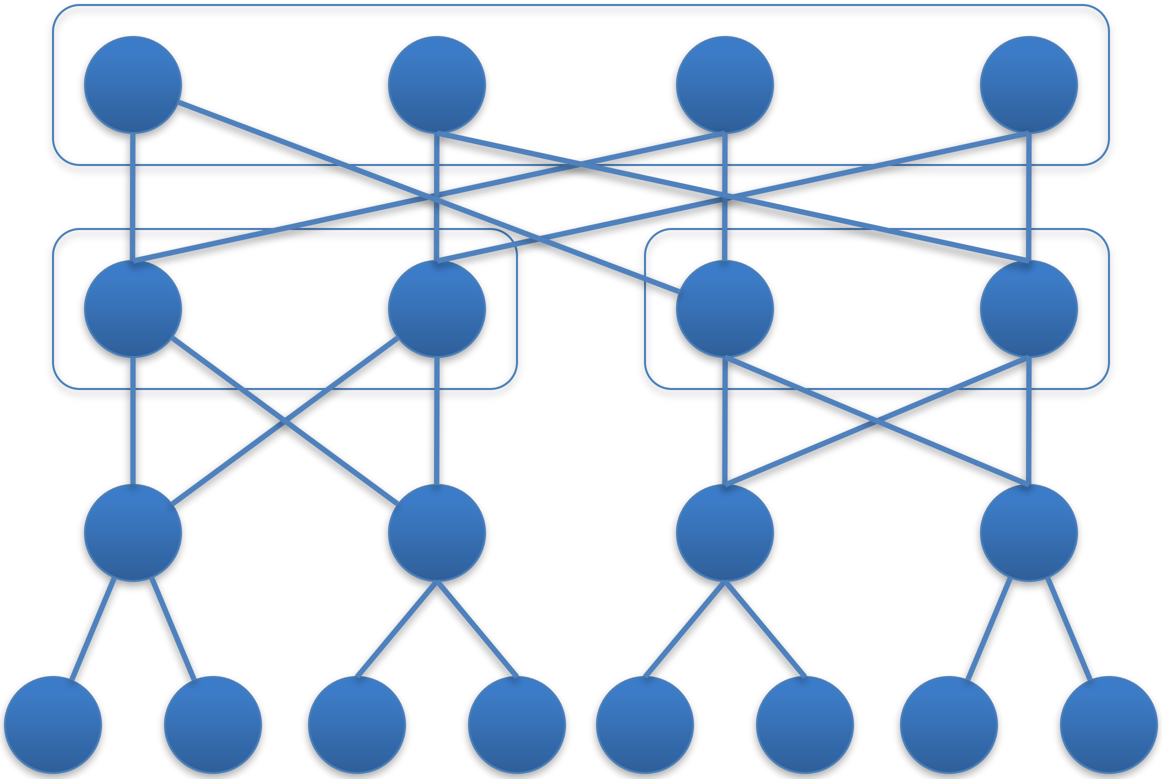
(5,3,4) Clos network

Fat Tree



- Bandwidth remains constant at each level
- Regular Tree: Bandwidth decreases closer to root

Fat Tree (2)



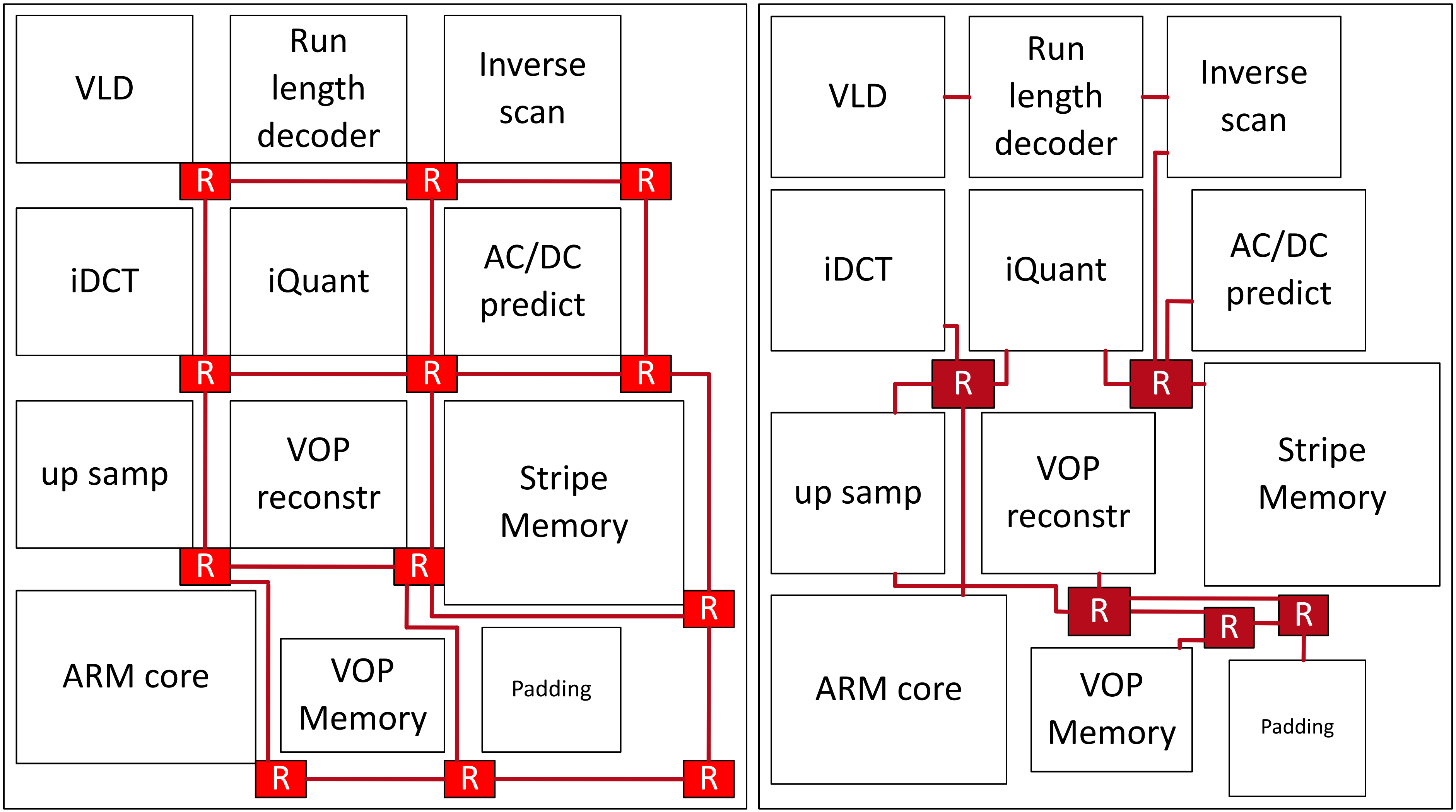
- Can be constructed from folded Clos
- Provides path diversity

Irregular Topologies

Irregular Topologies

- MPSoC design leverages wide variety of IP blocks
  - Regular topologies may not be appropriate given heterogeneity
  - Customized topology
    - Often more power efficient and deliver better performance
- Customize based on traffic characterization

Irregular Topology Example



Topology Customization

- Merging
  - Start with large number of switches
  - Merge adjacent routers to reduce area and power

- Splitting
  - Large crossbar connecting all nodes
  - Iteratively split into multiple small switches
    - Accommodate design constraints