4/18

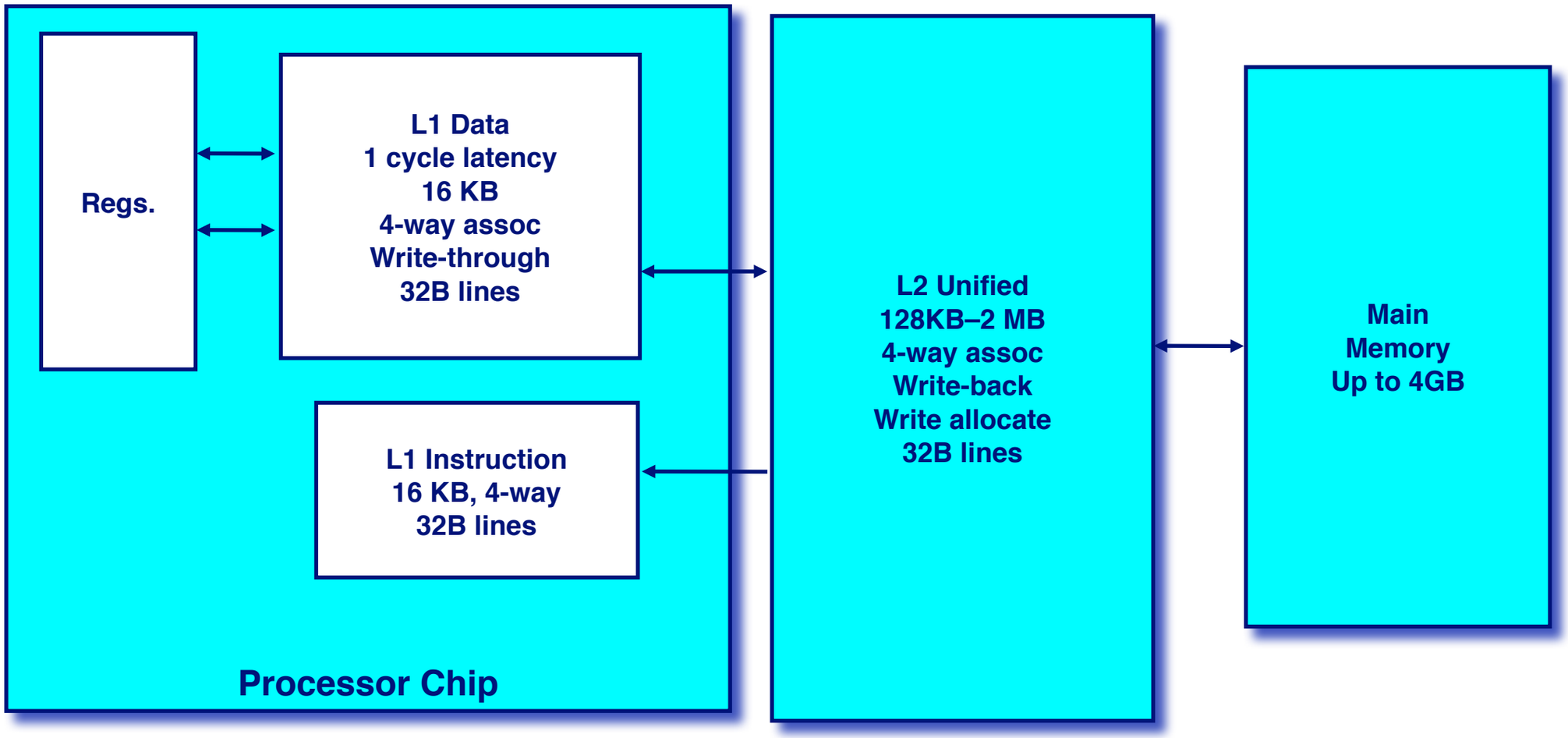
Admin

- Test script will be provided over the weekend.
- Sign up for demo slots on the ‘signup’ link on the calendar.

Consistency vs Coherence

- Coherence is micro-scale
  - Defines the behavior of a single data value.
  - All actors should have the same view of state.
- Consistency is macro-scale
  - Defines visible concurrency between data.
  - Contract for how shared resources behave.

x86 Cache


Regs.
L1 Data
1 cycle latency
16 KB
4-way assoc
Write-through
32B lines
L1 Instruction
16 KB, 4-way
32B lines
Processor Chip
L2 Unified
128KB–2 MB
4-way assoc
Write-back
Write allocate
32B lines
Main
Memory
Up to 4GB

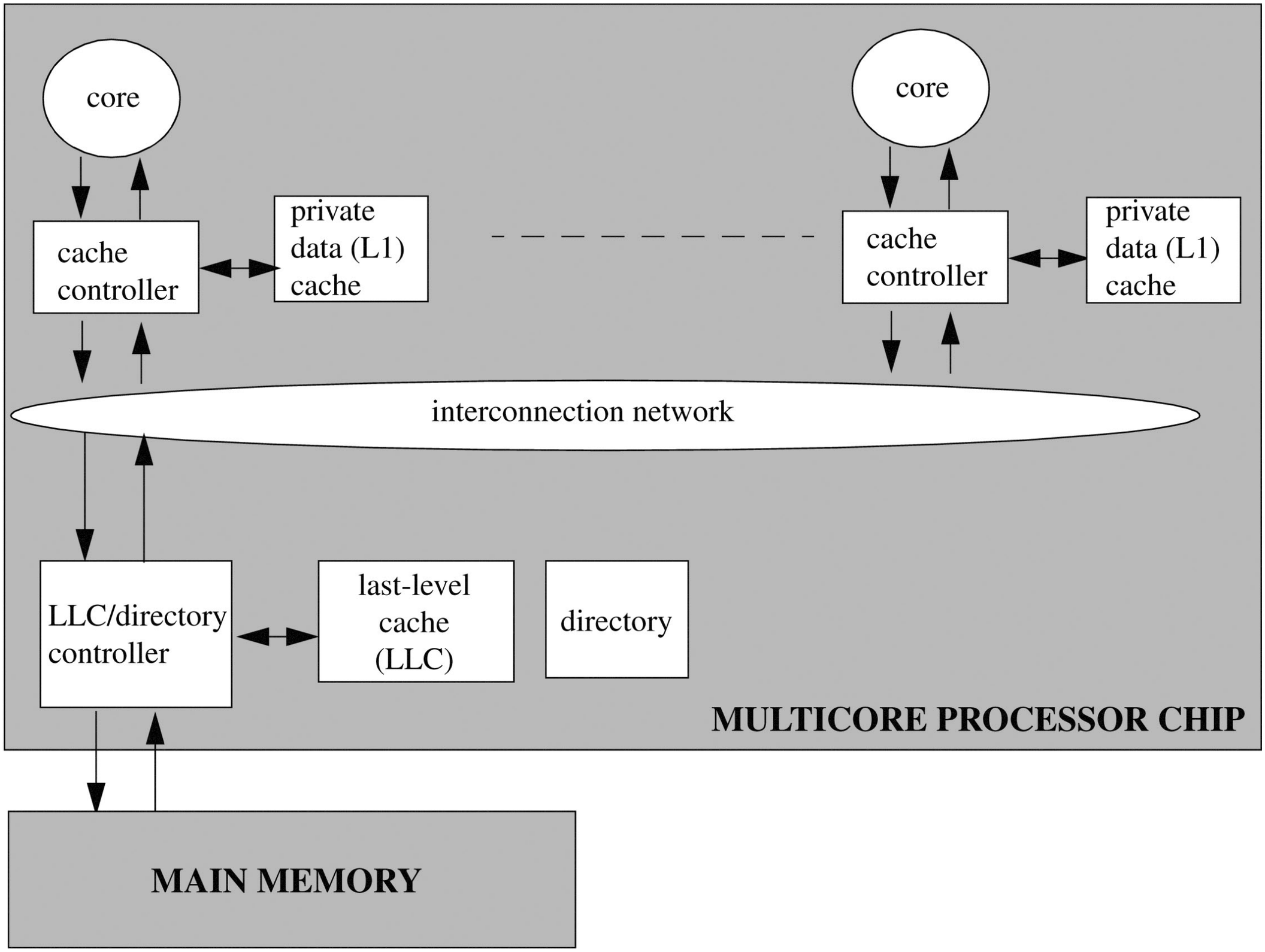
Coherence Strategies

(For write-back caches)

- Invalidation / Directory
- Snooping

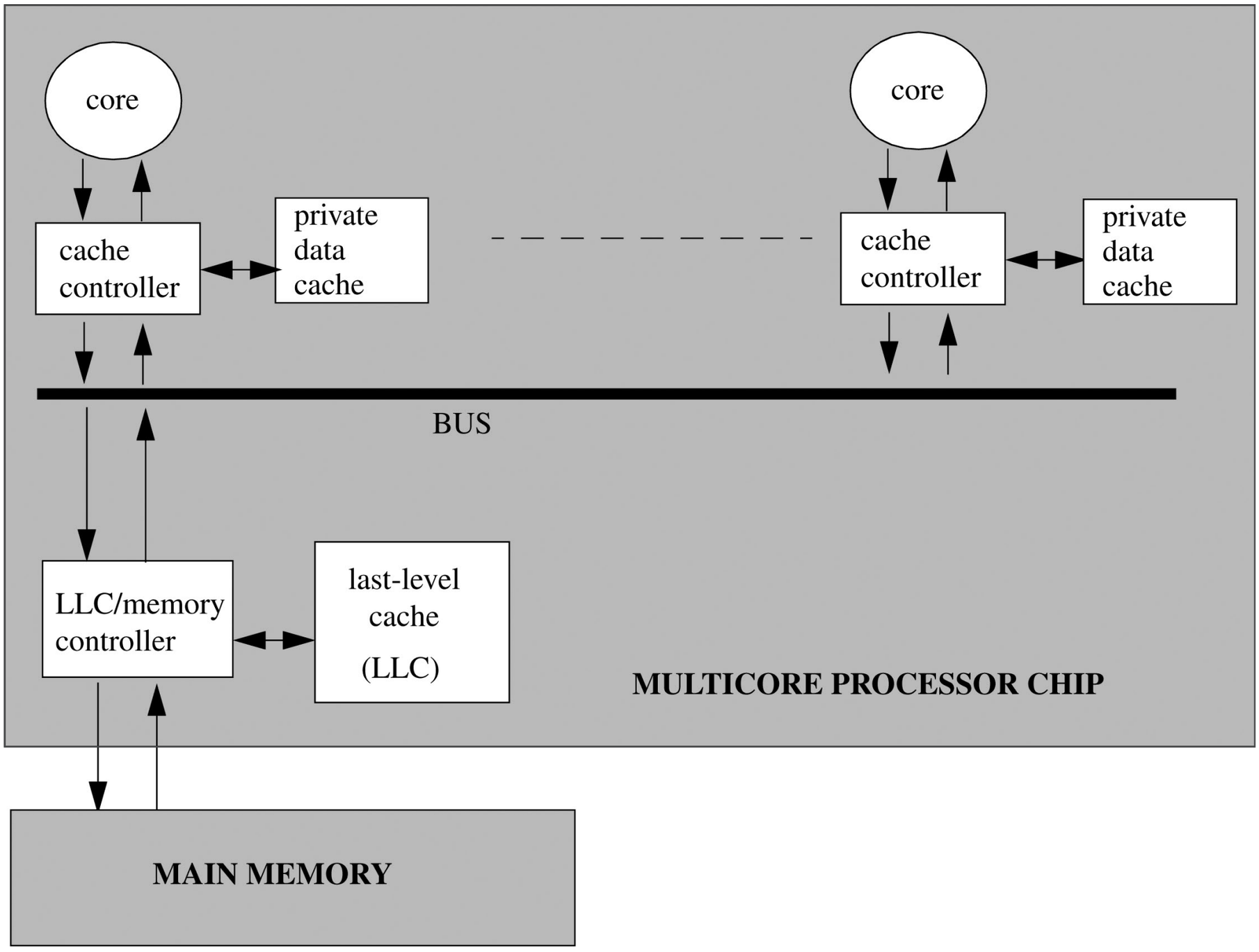
Directory Coherence

- Centralized manager for cache states.
- Manager/Directory sends invalidations on behalf of client.
- SGI Origin (mid 90s) had a directory with 1024 cores, Intel i7's use 'Home Snoop' , a directory protocol.


core
core
private
data (L1)
cache
private
data (L1)
cache
cache
controller
cache
controller
interconnection network
LLC/directory
controller
last-level
cache
(LLC)
directory
MULTICORE PROCESSOR CHIP
MAIN MEMORY

Snooping Coherence

- When one cache is the ‘owner’ in a write-back scenario, other caches can read by snooping from the owner.
- Also allows with minimal effort an ‘exclusive-read’ state, where data is not invalidated, but
- Implemented with a hardware bus by Sun Starfire (1991), IBM POWER5 (2005)


core
core
cache
controller
private
data
cache
cache
controller
private
data
cache
BUS
LLC/memory
controller
last-level
cache
(LLC)
MULTICORE PROCESSOR CHIP
MAIN MEMORY

Cycle	Core 1	Core 2	Memory	Request Bus	Data Bus
1	Load Miss, req S				
2				C1 req S	
3		Store Miss	C1 response		
4					data from mem
5	Perform Load	req M			
6				C2 req M	
7			C2 response		
8					data from mem
9		Perform store			
10	Load miss, req S				
11				C1 req S	
12		C1&mem response			
13					data from C2
14	Perform load		store data		

Dangers

- Violate single-writer-multiple-reader
- Deadlock

Consistency

- Eventual Consistency
- Total Store Order
- Sequential Consistency
- Linearizable

Consistency

x = 0, y = 0

Core 1	Core 2
x = 1 r1 = y	y = 1 r2 = x

Sequential:
(r1,r2)=(0,1)
or (1,0) or(1,1)

TSO:
(r1,r2)=(0,0)
or (1,0) or (0,1)
or (1,1)