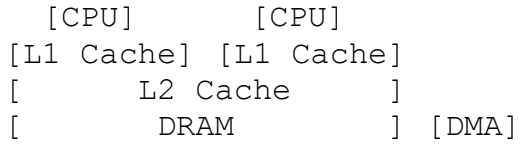
Notes for 11/11/09

Hardware	Software
Consistency Coherence	Thread safety

“A system is distributed if the [uncertainty of] message transmission delay is not negligible compared to the time between events in a single process.” – Leslie Lamport

Even with a single processor, we still have multiple agents writing to memory, such as DMA controllers.



How will the first CPU's writes to L1 cache be seen by the other's?

Memory coherence – how multiple agents share an area of memory with respect to writing to a single location to allow you to write correct programs
Assumptions of memory coherence:

1. A processor sees its own writes immediately
2. Sees another processor's writes to a location if there are no other writes after sufficient time
3. Writes to a location are serialized (but generally not writes among all locations)

MESI protocol and cache coherence:
Use two bits to represent four cache line states:

- Modified – the cache line is only in present cache but is dirty; this must be written back before other agents try to read it from RAM
- Exclusive – the cache line is only in present cache but is clean; on write, this will go to the modified state
- Shared – the cache line is in other caches and is clean; may become invalid if another agent writes to this cache line
- Invalid – contents are not valid

While you are doing a write to an address, other agents cannot read from the cache line until you put it back in a shared state after a writeback.

Possible states if address is shared between two processors.

	M	E	S	I
M	N	N	N	Y
E	N	N	N	Y
S	N	N	Y	Y
I	Y	Y	Y	Y

Race conditions

Thread 1	Thread 2

`a = 1`
`b = 1`
`flag = 1`
	`while (flag == 0)`
	`    print a`
	`    print b`

What will this print? This depends on the memory consistency model.

Memory consistency – how memory operations are synchronized across multiple locations

Strict consistency – a read has to return the latest write; this is impossible to implement on a distributed system because there is no way to synchronize clocks, i.e., to absolutely know the order of operations

Processor consistency: if the processor writes 1, 2, 3, in that order, then other agents see 1, 2, 3 written in that order