

CS8625-2009

Bus-Contention & Experiments

CS8625 High Performance and Parallel Computing Dr. Ken Hoganson

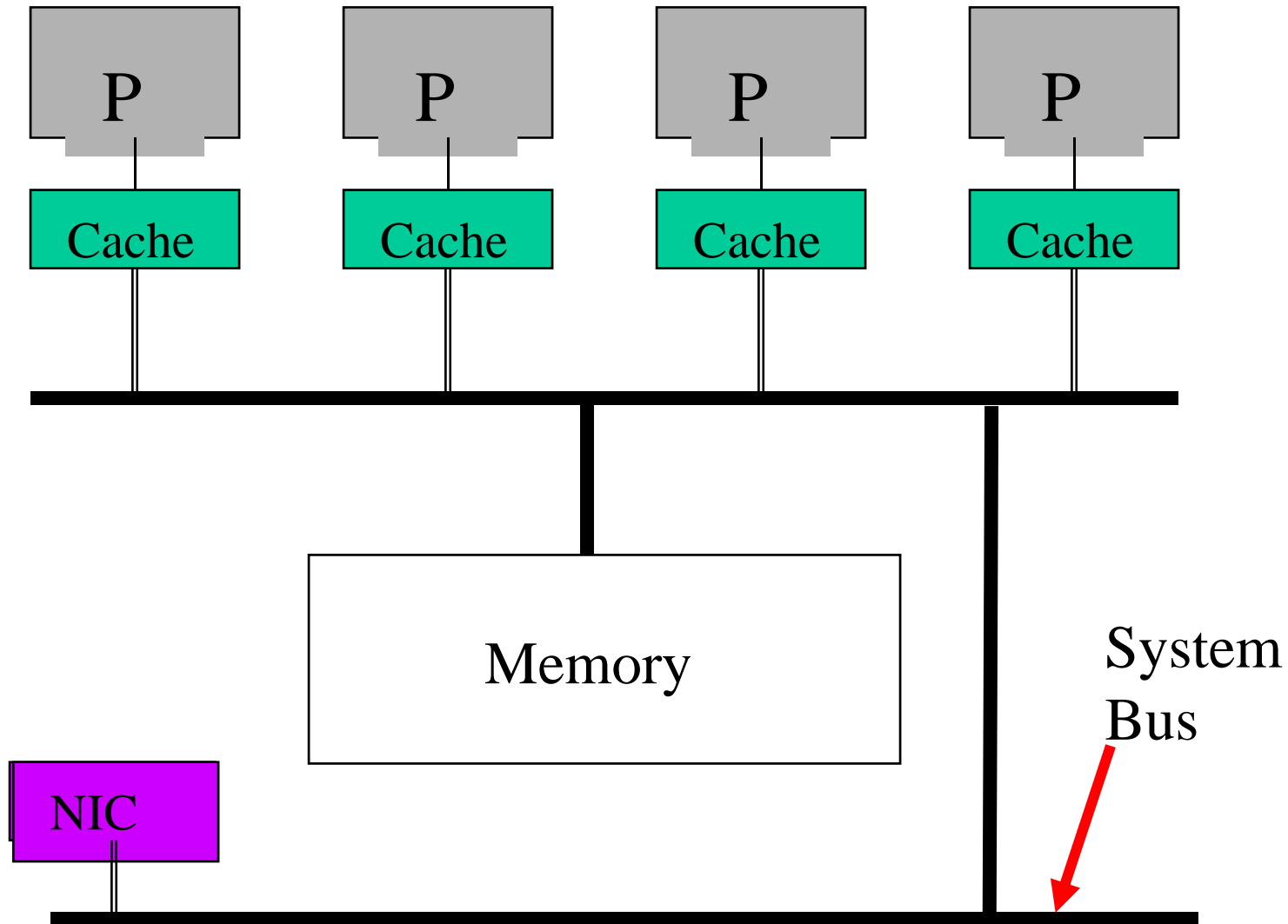
Class

Will

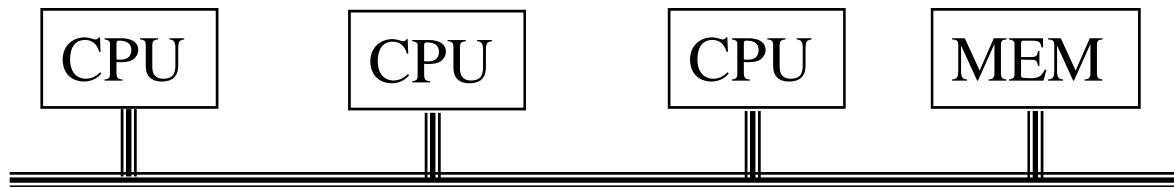
Start

Momentarily...

- Multiple processors
- Multiple programs or threads of execution
- Each processor has access to memory, either:
 1. Shared Memory
 - Processors have access to a common memory
 - Easy to exchange data, through accessing memory
 - Very fast access (low latency)
 - Contention for memory
 - Clustered (grouped) shared-memory multiprocessors.



- Shared-Memory Multiprocessors are very fast
 - Low latency to memory on bus
 - Low communication overhead through shared-memory
- Scalability problems
 - Length of bus slows signals (.75 SOL)
 - Contention for the bus reduces performance
 - Requires Cache to reduce contention



Multiple devices – processors, etc, compete for access to a bus

Only one device can use a bus at a time

r = probability of a processor requesting a bus

n = number of processors

$1 - r$ = probability of not requesting a bus

probability that none will request the bus $(1 - r)^n$

probability that at least one will request the bus $1 - (1 - r)^n$

probability that exactly one processor will request the bus =

$$\frac{n!}{(n-1)! \times 1!} \times r(1-r)^{(n-1)} = nr(1-r)^{(n-1)}$$

probability of more than zero or one request (at least one request is blocked)

$$1 - (1 - r)^n - nr(1 - r)^{(n-1)}$$

	N=4	N=8	N=16
R	0.1	0.1	0.2
1-r	0.9	0.9	0.8
$(1-r)^n$	0.6561	0.430	0.028
$1-(1-r)^n$	0.3439	0.57	0.972
$Nr(1-r)^{(n-1)}$	0.2916	0.3826	0.1126
Blocked	0.0523	0.1873	0.8594

- Performance degrades as requests are blocked
- Resubmitted blocked requests degrades performance even further than that shown above

Clearly, the probability that a processor's access to a shared bus increases, with both:

- The number of processors sharing a bus
 - The probability a processor will need access to the bus.
-
- What can be done? What is the "universal band-aid" for performance problems?

	N=4	N=8	N=16	N=16
R	0.1	0.1	0.2	0.01
1-r	0.9	0.9	0.8	0.99
$(1-r)^n$	0.6561	0.430	0.028	0.8515
$1-(1-r)^n$	0.3439	0.57	0.972	0.1485
$Nr(1-r)^{(n-1)}$	0.2916	0.3826	0.1126	0.1376
Blocked	0.0523	0.1873	0.8594	0.0109

- If cache greatly reduces access to mem, then
- Blocking rate on the bus is much lower.

Two approaches to improving shared memory machine performance:

- Invest in large amounts, and multiple levels of, cache,
 - and a connection network to allow caches to synchronize contents.
- Invest in multiple buses and independently accessible blocks of memory

- Your project is to explore the effect on the performance of a shared-memory bus-based multiprocessor, of interconnection network contention.
- You will do some calculations, build a simulation system, and write a couple-page report to turn in.

- For a machine with processors that include on-chip cache that yield a cache hit rate of 99.0%, determine the maximum number of processors that can go on a single shared-bus, and still maintain at least a 97% acceptance of requests.
- Use the calculations shown in the lecture to zero in on the correct answer, recording your calculations in a table for your report. Show each step of the calculation as was done in the lecture/ppt.
- Your results should “bracket” the maximum.

- Use the maximum number of processors (Task 1) and Amdahl's law at the balance point, to figure out what workload parallel fraction yields a balance in the denominator.
- Determine the theoretical speedup that will be obtained.

- Use the data values developed so far, to run the your simulation system. Record the speedup obtained from this system.
- If it differs markedly from the theoretical value, check all the settings, and rerun the simulation, and explain any variation from the theoretical expected value.
- Record your results in your report, showing each step of the calculation as was done in the lecture/ppt.

**End
Of
Today's
Lecture.**

- This slide intentionally blank