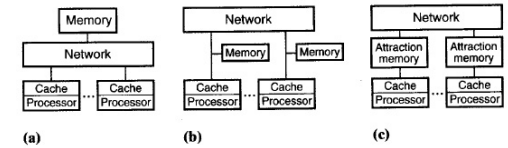


DDM – A Cache-Only Memory Architecture

Erik Hagersten et al.
 Summary by Victor Wen
 4/3/2002

Glossary

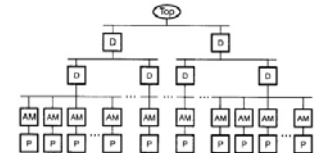


- UMA
 - single bus based, shared memory organization
 - uniform access time to memory
 - example: Sequent and Encore
- NUMA
 - basically distributed shared memory
 - arbitrary interconnect network
 - example: BBN Butterfly, IBM RP3
- COMA
 - non-static partition of memory address
 - distributed memory acts like cache, called *attraction memory*
 - DDM -- COMA machine with a hierarchy

Motivation

- Why COMA?
 - good locality
 - adapts to dynamic scheduling
 - run NUMA and UMA optimized program well
 - useful when communication latency long
- Why the hierarchy?
 - for scalability and use of snoopy protocol

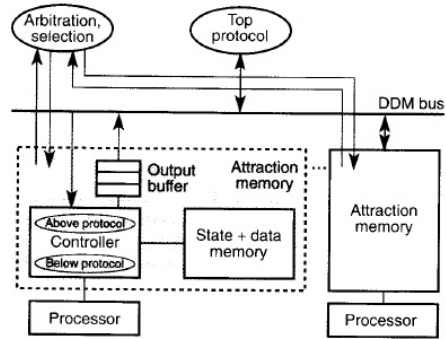
Now, details...



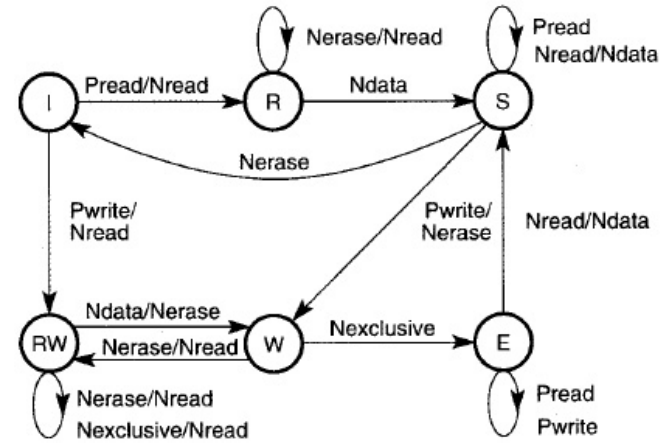
- processor+attraction memory at leaves
- directory in interior nodes
- use asynchronous split Xaction bus to increase throughput
 - outstanding requests encoded in coherence protocol states
- Use fat tree to alleviate bandwidth contention up stream

More details...

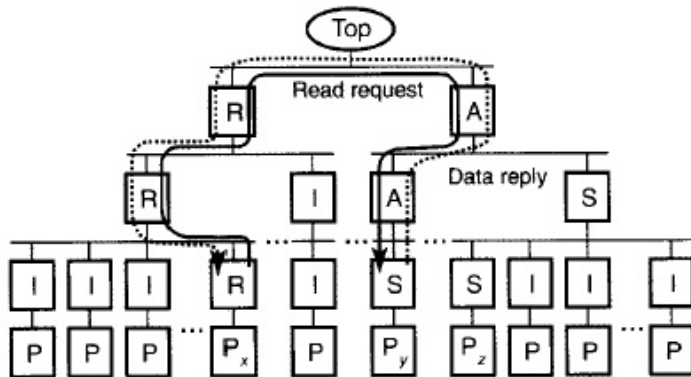
- Minimal data unit: *item*
 - VM address translated to item identifier
 - coherence state associated with each item
 - home bus for item statically allocated, used for Replacement
- State info includes tag and coherence state
 - higher overhead for smaller item size (6-16% for 32-256 Ps)
 - but high false sharing if item size gets large



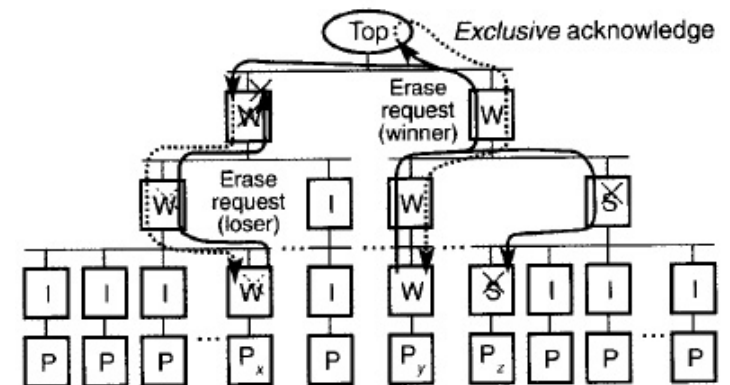
Coherence Protocol



Multilevel Read



ML Write Races



Replacement

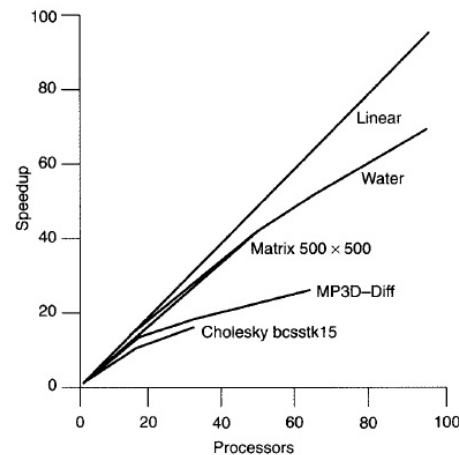
- happens when attraction memory is full and need space
- results in an Out transaction (if in S)
 - terminates when finding other items in S, R, W or A state
 - turns to Inject if last copy
- results in Inject transaction (if in E)
- inject tries local DDM bus first, then home bus
 - if home bus also full, evict foreigners

Prototype

- TP881V system
 - 4 Motorola MC88100 20 MHz proc.
 - 2 16 KB caches and MMU
 - 8 or 32 MB DRAM
- Custom designed DDM node controller
- 4 phases in DDM bus, 20 MHz
 - Xaction code
 - Snoop
 - Selection
 - Data
- Memory system 16 percent slower than original TP881V system

Performance

- Benchmarks written for UMA
- Speed up normalized to single DDM node with 100% hit in attrac. mem



Summary and Discussion

- 6-16% memory overhead with 16% slower access time to memory
- COMA works well for programs with high locality (ie Water)
- Introduced transient states in coherence protocol to handle replacement and split Xaction bus
- Introduced hierarchy to allow scalability
- Simulated performance sketchy. Why not compare against UMA or NUMA machines using same hardware parameters?
- Is network+cache access latencies longer than memory?