REACTIVE NUMA: A DESIGN FOR UNIFYING S-COMA AND CC-NUMA

Babak Falsafi and David A. Wood
University of Wisconsin-Madison
Presented for ECE 259 by Ralph Nathan
CC-NUMA

- Cache Coherent Non-Uniform Memory Access
  - Remote data is cached in CPU’s cache or in shared per-node cluster cache (block cache)
  - Block cache only holds remote data
  - If there’s a miss, need to go to page’s home node

+ Block cache reduces number of remote misses
- Read/Write shared pages perform poorly
S-COMA

- Simple Cache Only Memory Architecture
- Remote data may reside in both node’s cache and main memory
- Remote pages live in main memory in the page cache
  + Can use the node’s main memory to cache remote data dynamically
  + Dense sharing patterns amortize overhead of replacement
  - Fragmentation due to a lack of spatial locality
  - Cost of page replacement is high
PROPOSAL

- R-NUMA combines best of both protocols by choosing to implement one dynamically.
- Default protocol is CC-NUMA.
- Occurs at a page granularity and can switch during runtime.
- Pages containing data structures (Reuse pages) that are accessed many times within a node are treated as S-COMA.
- Pages used to transfer data (communication pages) are treated as CC-NUMA.
- Pages are distinguished through the use of counters that count the number of times blocks are refetched.
IMPLEMENTATION

- Eight SMP nodes with 4 processors in each node
- Snooping is used inside nodes
- R-NUMA is used for coherence between nodes
THOUGHTS AND QUESTIONS

- Do we buy the results that R-NUMA is better than either CC-NUMA/S-COMA?
- How large can R-NUMA scale?
- Since each protocol has its own advantages/disadvantages, should we merge multiple protocols to create single protocols or just have a single independent protocol?