

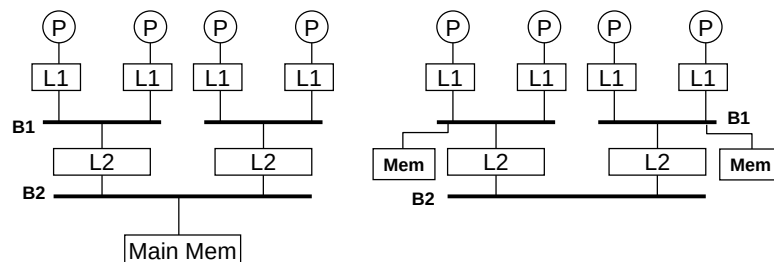
Large-Scale MPs MRs para Protocolos Snoopy vs. Diretório

Extendendo Coerência de Caches para Memórias Compartilhadas Distribuídas

- Ao contrário de memórias compartilhadas centralizadas, não possuímos neste caso um recurso centralizado que vê todas as transações (*bus*)
 - Não há meio de broadcast
 - Não há maneira de serializar transações
 - barramento funciona como meio de garantir serialização das operações
- Técnicas
 - Extender protocolos do tipo *snoopy*
 - Protocolos baseados em diretórios

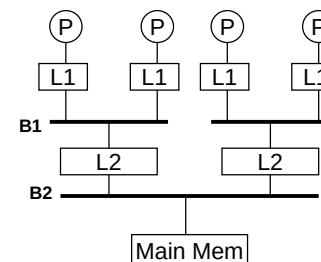
Snoopy Hierárquico

- Maneira mais simples: hierarquia de barramentos, coerência *snoopy* a cada nível
- Duas possibilidades



Hierarquias com Memória Global

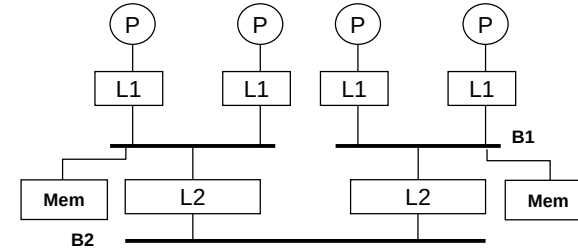
- Caches primárias:
 - Alta performance (SRAM)
 - B1 segue protocolo *snoopy* básico
- Caches secundárias:
 - Muito maiores que L1 (precisam manter propriedade de inclusão)
 - L2 funciona como filtro p/ B1 e L1
 - L2 pode ser baseada em tecnologia DRAM



Hierarquias com Memória Global

- Vantagens:
 - Misses na memória principal requerem tráfego único para a raiz da hierarquia
 - Utilização de dados compartilhados não é uma questão importante
- Desvantagens:
 - Misses para dados locais também trafegam na hierarquia
 - Memória em barramento global deve ser intercalada para aumentar BW

Hierarquias Baseadas em Clusters



- Memória principal é distribuída entre clusters
 - Alocação de dados locais pode reduzir tráfego em barramento global
 - Reduz latência (acessos locais mais rápidos)
 - Exemplos: Encore Gigamax
- L2 pode ser substituída por uma chave roteadora baseada em tags com coerência

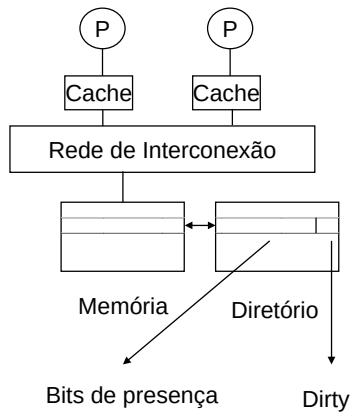
Resumo de Hierarquias

- Vantagens:
 - Conceitualmente simples de se construir (aplica-se snoopy recursivamente)
 - Pode-se reduzir hardware necessário ao se combinar as funcionalidades
- Desvantagens:
 - Gargalo de comunicação na direção da raiz
 - Latências podem ser grandes na direção da raiz

Protocolos de Coerência Baseados em Diretórios

- Protocolos do tipo snoopy não escalam bem porque dependem de algum meio de broadcast
- Snoop hierarquico faz raiz se tornar um gargalo
- Mecanismos baseados em diretório escalam bem
 - Evitam broadcasts
 - Mantém informação de todas as caches que possuem cópia do bloco
 - Utilizam mensagens ponto-a-ponto para manter coerência

Protocolo Baseado em Diretório

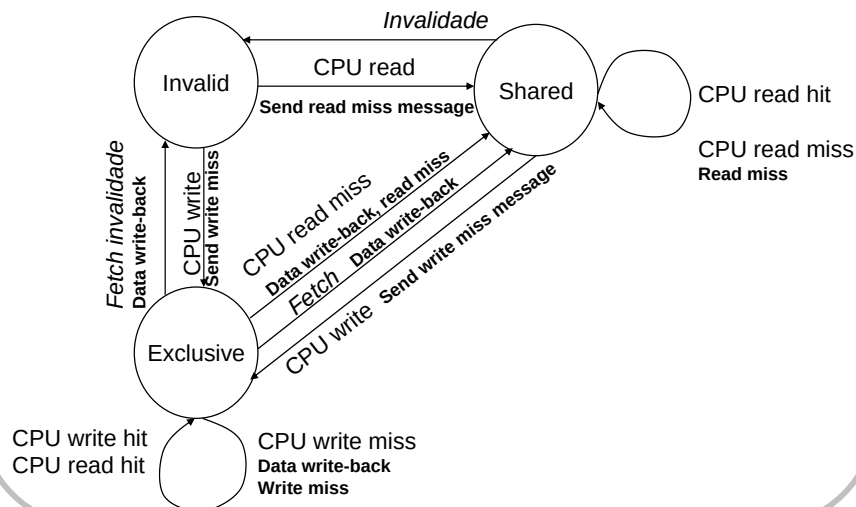


- K processadores
- Para cada bloco de cache na memória => k bits de presença, 1 *dirty-bit*
- Para cada bloco na cache
 - *Invalid*
 - *Exclusive*
 - *Shared*

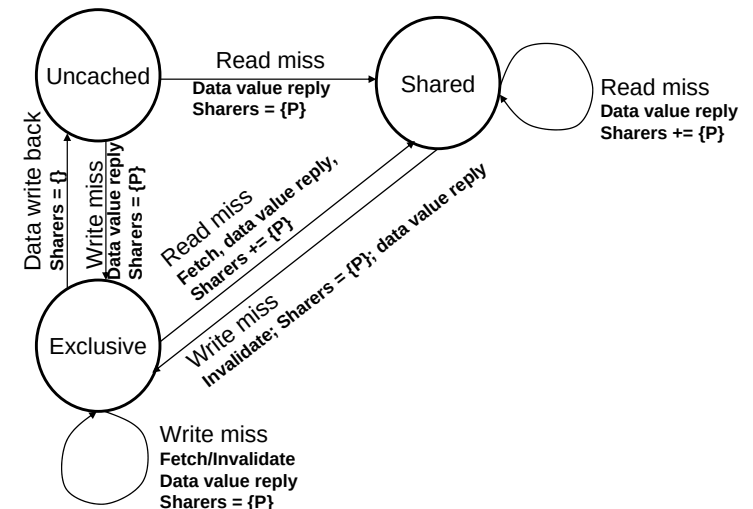
Protocolo Baseado em Diretório

Message	Source	Destination	Contents
Read miss	Cache	Dir	P,A
Write miss	Cache	Dir	P,A
Invalidate	Dir	R. Caches	A
Fetch	Dir	R. Caches	A
Fetch/Inv.	Dir	R. Caches	A
Data value reply	Dir	Cache	Data
Data write back	R.Cache	Dir	A,Data

Protocolo (Cache)



Protocolo (Diretório)



Organização de Diretórios

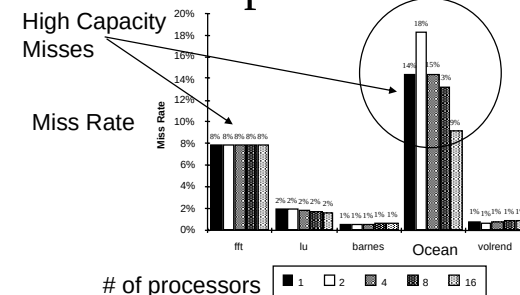
- Implementação dos diretórios na memória (como o apresentado anteriormente)
 - Apresenta problemas de escalabilidade
 - Pode apresentar problemas de BW
- Implementação dos diretórios na cache (SCI)
 - Listas encadeadas (simples ou duplo)
 - Mantém em cada cache “link” do próximo processador compartilhando bloco
 - SCI utiliza listas duplamente encadeadas
 - Precisamos agora da mensagem “desconectar da lista” e “conectar à lista” (a última equivalente a um MISS)

Avaliação de Desempenho dos Protocolos

Miss Rates para Protocolo Snooping

- 4o. C: Misses de Conflito, Capacidade, Compulsório e **Coerência**
- Mais processadores: aumenta misses de coerência enquanto reduz misses de capacidade (para um problema de tamanho fixo)
- Comportamento de caches para cinco programas:
 - **FFT** Fast Fourier Transform: Matrix transposition + computation
 - **LU** factorization of dense 2D matrix (linear algebra)
 - **Barnes-Hut** n-body algorithm solving galaxy evolution problem
 - **Ocean** simulates influence of eddy & boundary currents on large-scale flow in ocean: dynamic arrays per grid
 - **VolRend** is parallel volume rendering: scientific visualization

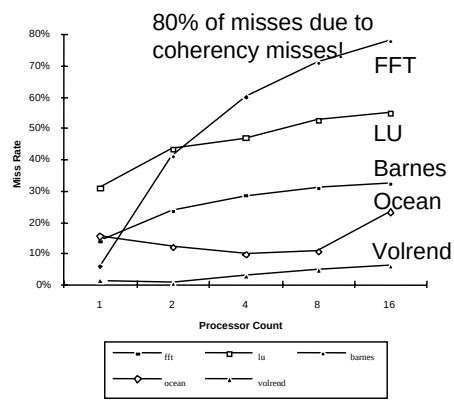
Miss Rates para Protocolo Snooping



Big differences in miss rates among the programs

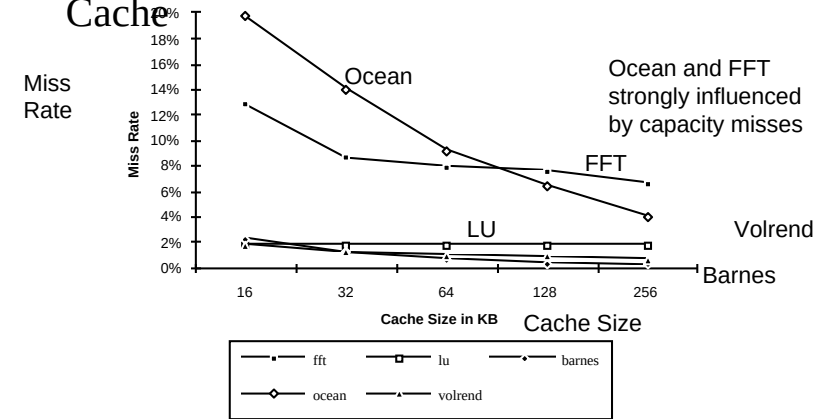
- Cache: 64KB, 2-way set associative, com blocos de 32B.
- Com exceção de Volrend, misses nessas aplicações são gerados por acessos a dados que são potencialmente compartilhados.
- Excetuando Ocean, dados são altamente compartilhados; em Ocean, somente os contornos dos sub-grids são compartilhados, embora o grid seja tratado como um objeto. Porque os contornos são modificados a medida que o # de procs aumenta, o aumento do MR para Ocean quando utilizamos 2 processadores ao invés de 1 é causado pelos acessos aos sub-grids.

% Misses Causados por Tráfego de Coerência vs. # de Processadores



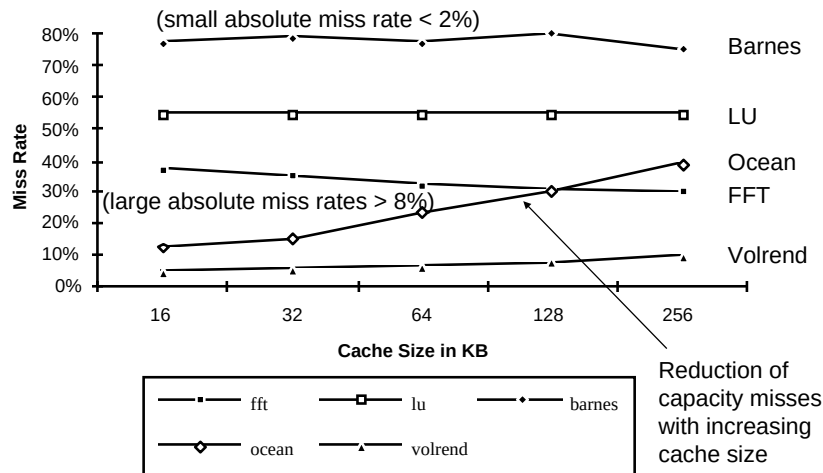
- % misses causados por coerência tipicamente aumentam quando um problema é executado em mais processadores.
- O número absoluto de misses de coerência aumenta em todos os benchmarks, incluindo o Ocean. Entretanto, no Ocean, é difícil separar esses misses de outros, porque a quantidade de compartilhamento varia com o número de procs.
- Invalidações aumentam substancialmente; No FFT, o MR devido a misses de coerência aumenta em 7%.

Miss Rates vs. Aumento do Tamanho da Cache



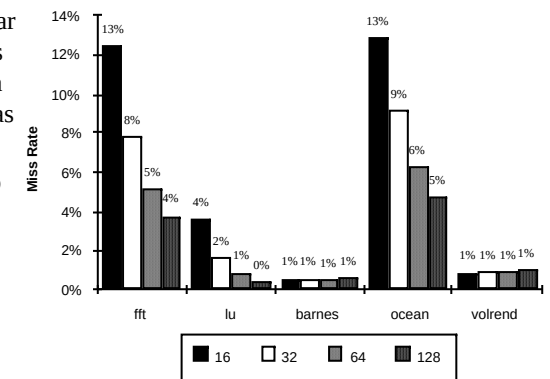
- MR cai quando o tamanho da cache aumenta, a não ser que o MR seja dominado por misses de coerência.
- Tamanho de bloco é de 32B & a cache é 2-way set-associative. # procs = 16.

% Misses Causados por Tráfego de Coerência vs. Tam. Cache



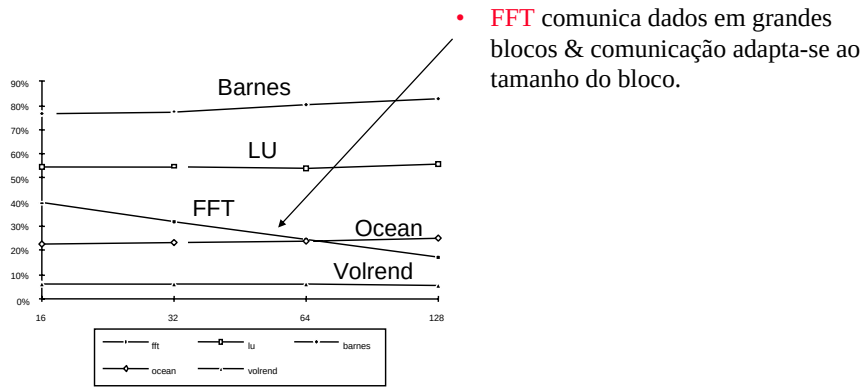
Miss Rate vs. Tamanho de Bloco

- Já que blocos podem suportar múltiplas palavras, podemos ter tráfego de coerência para variáveis não correlacionadas no mesmo bloco
- **False sharing** aparece com o uso de algoritmos para coerência baseados em invalidação. É causado quando um bloco é invalidado e é subsequentemente acessado para leitura



miss rates mostly fall with increasing block size

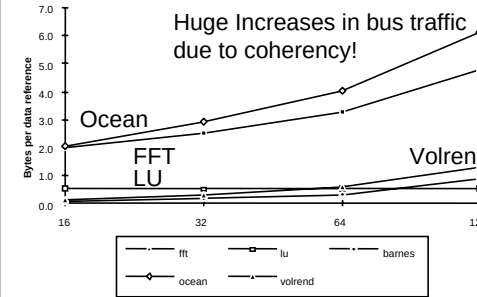
% Misses Causados por Tráfego de Coerência vs. Tam. Bloco



Behavior tracks cache size behavior
FFT: Coherence misses reduced faster than capacity misses!

Tráfego no Barramento com o Aumento do Tamanho do Bloco

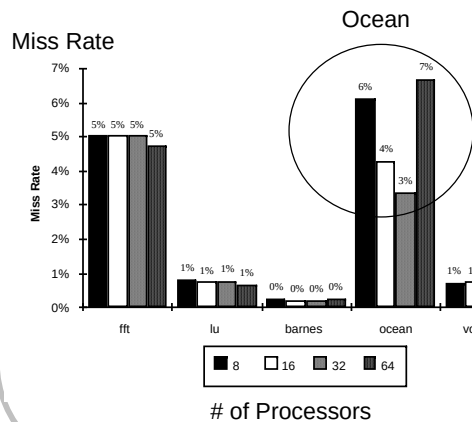
Bytes per data ref



- Tráfego no barramento aumenta a medida que aumentamos o tamanho do bloco.
- Volrend**: $\geq 10x$, embora o baixo MR mantenha valor absoluto baixo.
- O aumento de 3x para o Ocean é o melhor argumento contra blocos maiores.

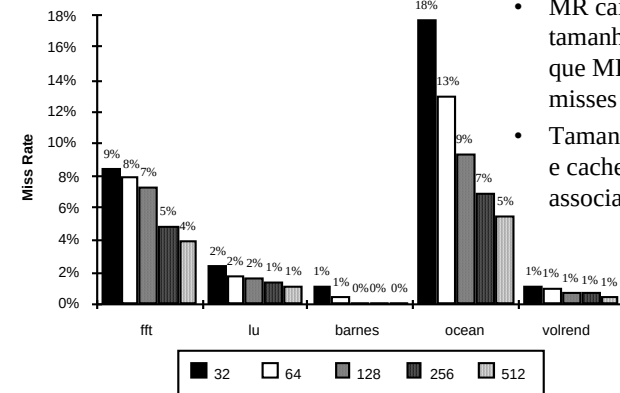
Miss Rates para Diretório

Use caches maiores para evitar latências longas dos diretórios



- Tamanho da cache 128 KB, 2-way set associative, com blocos de 64B.
- Ocean**: somente os contornos dos sub-grids são compartilhados. O aumento do MR de 32 para 64 processadores vem da redução do tamanho dos subgrids em 64 processadores.

Miss Rates com Aumento do Tamanho da Cache para Mecanismo de Diretório



- MR cai com aumento do tamanho da cache, a não ser que MR seja dominado por misses de coerência.
- Tamanho do bloco é de 64B e cache é 2-way set-associative. # procs = 16

Tamanho de Bloco para Diretório

- Assumir cache de 128 KB & 64 processadores
 - Tamanho do bloco maior é utilizado para evitar altas latências de memória

