

Arquiteturas Paralelas e Distribuídas

Arquitetura de hardware para comunicação entre processadores: memória compartilhada

Eduardo Furlan Miranda

Baseado em: LIMA, E. Fundamentos da Programação Paralela. 2018.

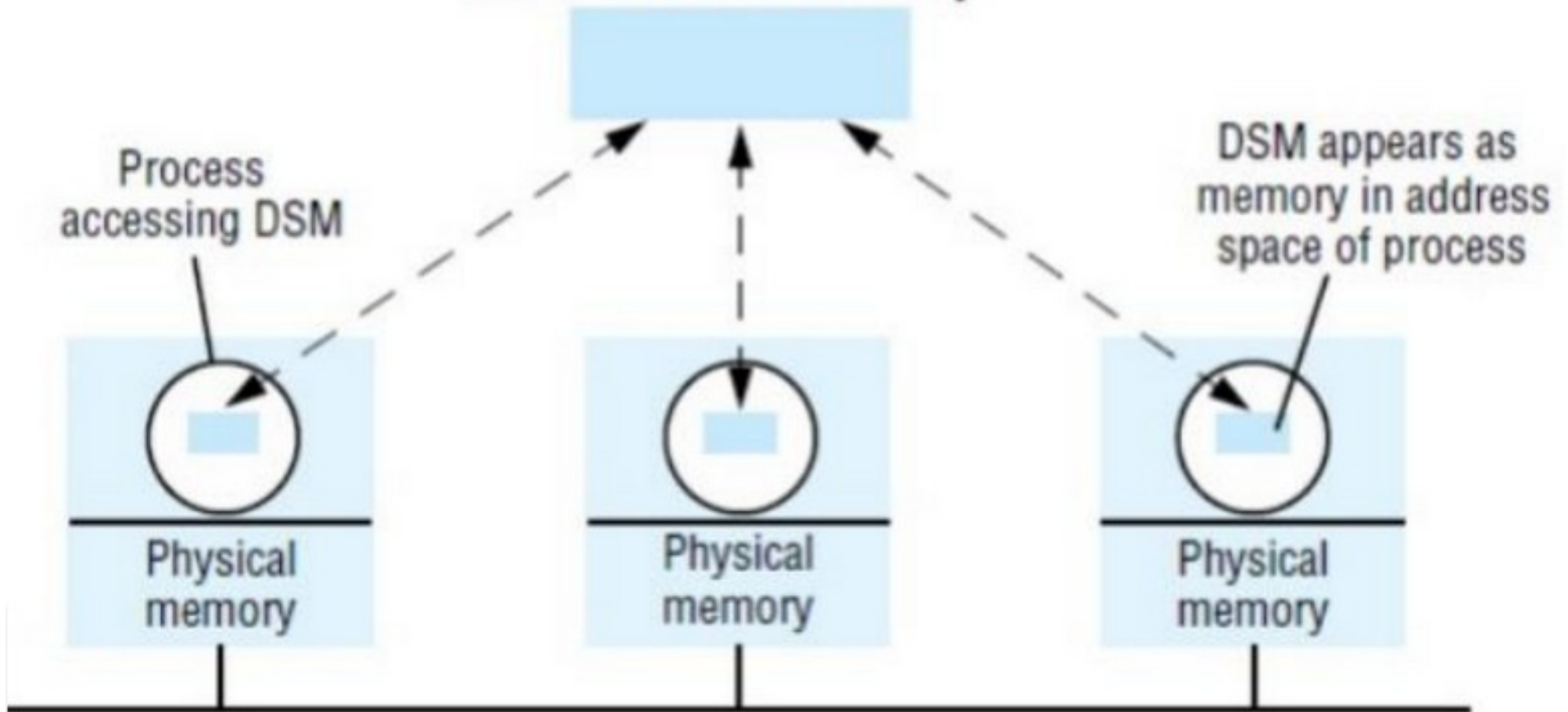
- Nó = PC, gabinete, máquina, conectado em rede
- Cluster = vários nós
- CPU = 1 núcleo do processador
- Processador = contém várias CPUs

- Os primeiros sistemas assumiam que processos em sistemas distribuídos executavam em espaços de endereçamento separados
- Kai Li (1986): Distributed Shared Memory (DSM)
 - Nós compartilhando espaço de endereçamento virtual
- Cada página está em um nó
- Qualquer processador acessa diretamente qualquer posição do espaço de endereçamento virtual

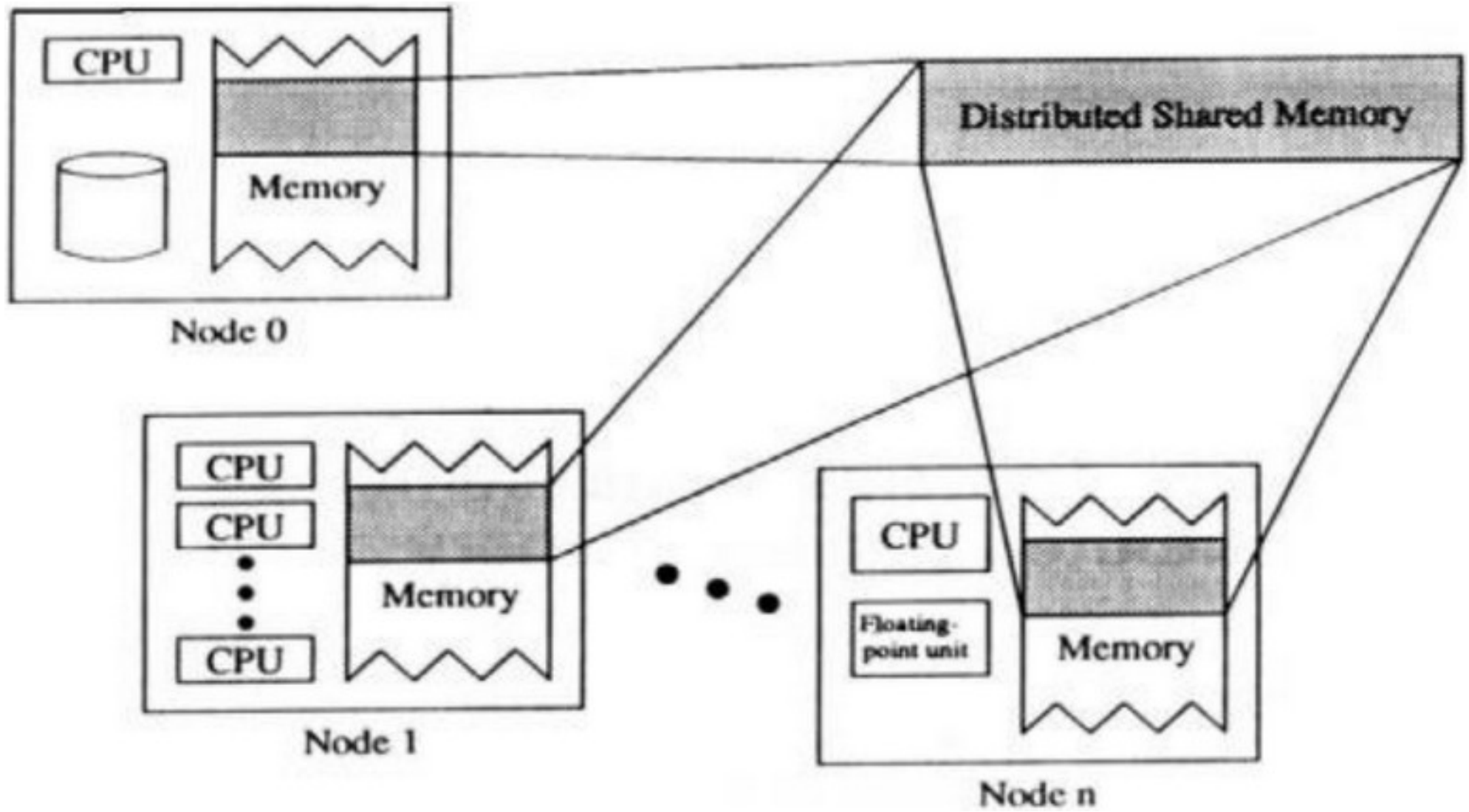
- O gerenciador de memória do DSM implementa o mapeamento entre a memória local e o espaço de endereçamento
 - memória local considerada uma cache do espaço de endereçamento compartilhado
- Referências à páginas locais são resolvidas por hardware, na velocidade da memória
- Referências à páginas presentes em outras máquinas causam falha de página
 - O gerenciador manda mensagem à máquina remota
 - Envia a página

- Similar ao tradicional sistema de memória virtual,
 - com a diferença de ao invés de buscar as páginas no disco,
 - busca em outra máquina da rede
- Comunicação e sincronização feita através da memória,
 - sem comunicação visível para os processos de usuário
- Memória compartilhada distribuída
 - intimamente relacionada com
 - arquitetura de computadores, sistemas operacionais, ambientes de execução e linguagens de programação

Distributed shared memory



- Organização básica similar a de um multicomputador que utiliza troca de mensagens
- Geralmente envolve clusters e redes escaláveis
- Cada cluster
 - contem um módulo local de memória física
 - pertencendo em parte ou inteiramente ao espaço de endereçamento global de DSM
 - controlador específico de interconexão
 - conecta ao resto do sistema



- Modelo tão importante para estes sistemas como é
 - para os que utilizam troca de mensagens
- Devem oferecer pequena latência, alta banda passante, custo razoável
- A relação de desempenho por custo depende da quantidade de nós no sistema
 - Para poucas dezenas: barramentos e anéis
 - Mais de 100 nós: outras soluções são necessárias

- Devido aos sistemas DSM permitirem centenas de nodos,
 - as redes de interconexão normalmente trocam a complexidade do modelo por maiores latências,
 - limitando a conectividade apenas a poucos nós adjacentes
- No caso de não usar nenhum esquema especial de modelo de interconexão,
 - alternativas atrativas com custo razoável também incluem tecnologias de rede, tais como Ethernet e ATM
 - ATM foi popular na década de 1990 e vem sendo substituída por Ethernet

- Geralmente os sistemas DSM são construídos aproveitando as redes de estações de trabalhos ou computadores pessoais em funcionamento
- Um dos maiores problemas dos sistemas DSM é o overhead associado à busca dos dados remotos (latência de memória)
- Latência de memória diz respeito ao intervalo de tempo compreendido entre
 - o momento no qual um processador iniciou um acesso a um dado compartilhado
 - até que este acesso seja satisfeito

- O problema da latência de memória é inerente aos sistemas DSM
 - por causa do grande overhead para localização e acesso aos dados
 - em sistemas com memória fisicamente distribuída
- A rede de interconexão tem influência direta no desempenho de sistemas DSM

- Representa a disposição global do espaço de endereçamento compartilhado,
 - Bem como a organização dos dados neste espaço
- Os dados podem estar organizados de diferentes maneiras
 - Na variação mais simples,
 - o espaço de endereçamento é dividido em páginas,
 - sendo que cada página está presente na memória de uma máquina

- Outra proposta é não compartilhar todo o espaço de endereçamento,
- mas somente uma parte dele,
 - apenas aquelas variáveis ou estruturas de dados que precisam ser usadas por mais de um processo
- Esta técnica reduz a quantidade de dados que devem ser compartilhados

- Outra proposta é ao invés de apenas compartilhar variáveis, pode-se encapsular tipos de dados, chamados objetos
- Cada objeto não possui apenas dados, mas também procedimentos (métodos), que atuam sobre os dados
- Processos podem manipular os dados do objeto apenas invocando seus métodos
- Não é permitido acessar diretamente os dados
- Restringindo o acesso desta forma, novas otimizações tornam-se possíveis

- O DSM divide o espaço global de endereçamento compartilhado em blocos
 - chamados de unidades de coerência
- para que os dados compartilhados possam ser manipulados pelo sistema de forma mais eficaz
- Dependendo da estrutura dos dados que fazem parte do espaço de endereçamento compartilhado, este espaço pode ser dividido de diferentes formas

- No caso de um sistema DSM no qual o espaço de endereçamento compartilhado é organizado em páginas,
 - a unidade de coerência é a página
- No caso de sistemas DSM baseados em objetos,
 - a unidade de coerência é o objeto

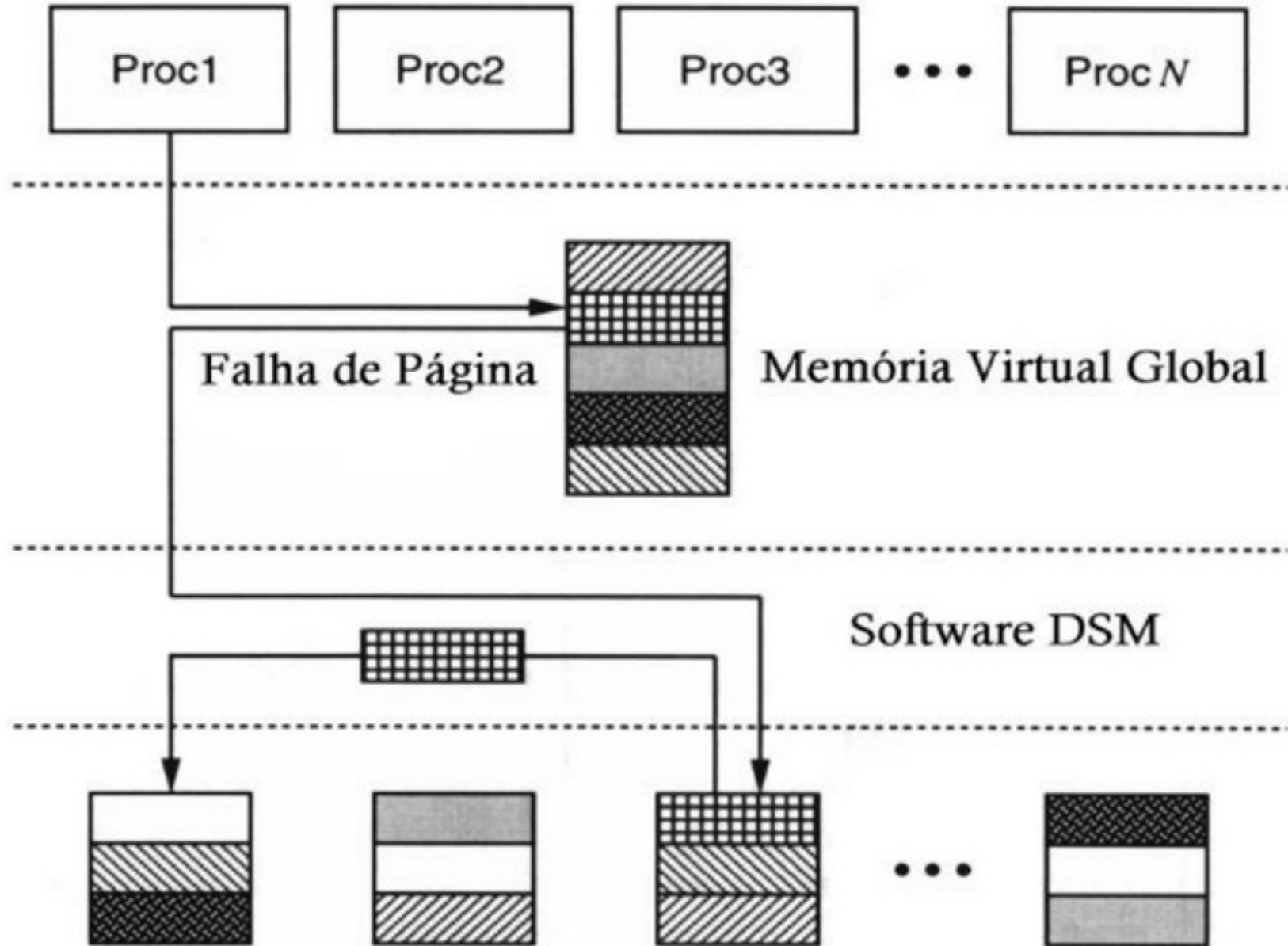
- A granulosidade da unidade de coerência determina o tamanho do bloco de dados manipulado pelo mecanismo de DSM
- Em geral, sistemas de hardware utilizam unidades pequenas (tipicamente blocos de cache),
 - enquanto que soluções de software, baseados em mecanismos de memória virtual, organizam os dados em grandes blocos físicos (páginas),
 - o que acarreta em compartilhamento de grandes grãos

- O uso de grandes blocos aumenta a probabilidade de que vários processadores irão requisitar acesso ao mesmo bloco simultaneamente,
 - mesmo que eles realmente acessem partes independentes do bloco de dados
- Este fenômeno é conhecido como falso compartilhamento (false sharing)

- Duas estratégias utilizadas para distribuição dos dados compartilhados são replicação e migração
- Replicação permite que várias cópias do mesmo item de dado residam em diferentes memórias locais
 - Utilizado principalmente para habilitar acessos simultâneos de diferentes nodos ao mesmo dado,
 - predominantemente quando compartilhamento de leitura prevalece
- Migração implica que somente uma cópia de um item de dado existe no sistema
 - os itens de dados são movidos sob demanda para uso exclusivo
 - Com o objetivo de diminuir o overhead de gerenciamento de coerência, esta estratégia é utilizada quando padrões sequenciais de compartilhamentos de escrita prevalecem

- A maioria dos sistemas DSM optam por replicar os dados,
 - pois esta proposta alcança melhor desempenho para uma grande quantidade de aplicações
- DSM fazem uso de replicação de dados para permitir acesso concorrente
 - No entanto, é fundamental manter estas cópias com a mesma informação,
 - impedindo que um nó acesse dados desatualizados

- Duas estratégias básicas para manutenção da coerência de réplicas
 - Protocolo de invalidação
 - uma escrita em um dado compartilhado causa a invalidação de todas as suas cópias, que passam a inacessíveis
 - Protocolo de atualização
 - uma escrita em um dado compartilhado causa a atualização de todas as suas cópias
- A figura do próximo slide mostra a atividade que ocorre como resultado de uma falha de página no processador 1, o qual resulta em uma cópia da página necessária sendo enviada da memória local do processador 3



- Os algoritmos para a implementação de DSM tratam com dois problemas básicos
 - distribuir estática ou dinamicamente os dados compartilhados através do sistema,
 - para minimizar a latência de acesso
 - preservar a coerência dos dados compartilhados,
 - enquanto minimiza o overhead gerado com o gerenciamento de coerência
- Algoritmos Único Leitor / Único Escritor (SRSW)
 - proíbe replicação, enquanto permite, mas não requer migração

- Algoritmos Vários Leitores / Único Escritor (MRSW)
 - reduzir o custo médio das operações de leitura,
 - contando que compartilhamento de leitura é o padrão que prevalece em aplicações paralelas
- Algoritmos Vários Leitores / Vários Escritores (MRMW)
 - permitem replicação dos blocos de dados com permissão de leitura e gravação
- Gerenciamento de DSM
 - Centralizado - pode tornar-se um gargalo
 - Distribuído - pode causar overhead

- Política que determina como e quando mudanças na memória feitas por um processador são vistas pelos outros processadores do sistema
 - Escrita mais recente
 - A leitura sempre retorna o valor escrito mais recentemente
 - Consistência sequencial
 - processadores executam em alguma ordem sequencial
 - Consistência de processador
 - escritas de um processador serão vistas por todos os outros processadores na mesma ordem

- Consistência Fraca
 - Usa barreiras
- Consistência de Liberação
 - Usa lock
- Consistência de Liberação Preguiçosa
 - Propagação apenas quando tenta adquirir o mesmo lock
- Consistência de Entrada
 - Atualizações quando um processador adquire um lock