



Arquitetura de Computadores

Capítulo 16

Processamento Paralelo

Profa. Gisele S. Craveiro

0 0 0

Tópicos

- o Organizações de múltiplos processadores
- o Multiprocessadores simétricos
- o Coerência de cache
- o Clusters
- o Acesso não-uniforme a memória (NUMA)
- o Computação Vetorial



Organizações de Múltiplos Processadores

- Única instrução, único dado (SISD - Single instruction, single data stream)
- Única instrução, múltiplos dados (SIMD - Single instruction, multiple data stream)
- Múltiplas instruções, único dado (MISD - Multiple instruction, single data stream)
- Múltiplas instruções, múltiplos dados (MIMD - Multiple instruction, multiple data stream)



Única instrução, único dado (SISD)

- Único processador.
- Única sequência de instruções.
- Dados armazenados em um única memória.
- Exemplos: sistema uniprocessador.



Única instrução, múltiplos dados (SIMD)

- Única instrução de máquina controla a execução simultânea de um certo número de elementos de processamento.
- Cada elemento de processamento tem uma memória de dados associada.
- Cada instrução é executada em um conjunto diferente de dados por um processador diferente.
- Exemplos: processadores vetoriais e matriciais.



Múltiplas instruções, único dado (MISD)

- Sequência de dados transmitida a um conjunto de processadores.
- Cada processador executa uma sequência diferente de instruções.
- Nunca foi implementado.



Múltiplas instruções, múltiplos dados (MIMD)

- o Conjunto de processadores.
- o Executa simultaneamente sequências diferentes de instruções.
- o Conjuntos de dados distintos.
- o Exemplos: SMPs, clusters e sistemas NUMA.



MIMD Visão Geral

- Processadores de propósito geral.
- Cada processador pode executar todas as instruções.
- Subdivididos de acordo com a forma de comunicação entre os processadores:
 - SMP (memória compartilhada);
 - NUMA: tempo de acesso não-uniforme;
 - Clusters: comunicação via rede.

0 0 0

Taxonomia de Arquiteturas de Processadores Paralelos

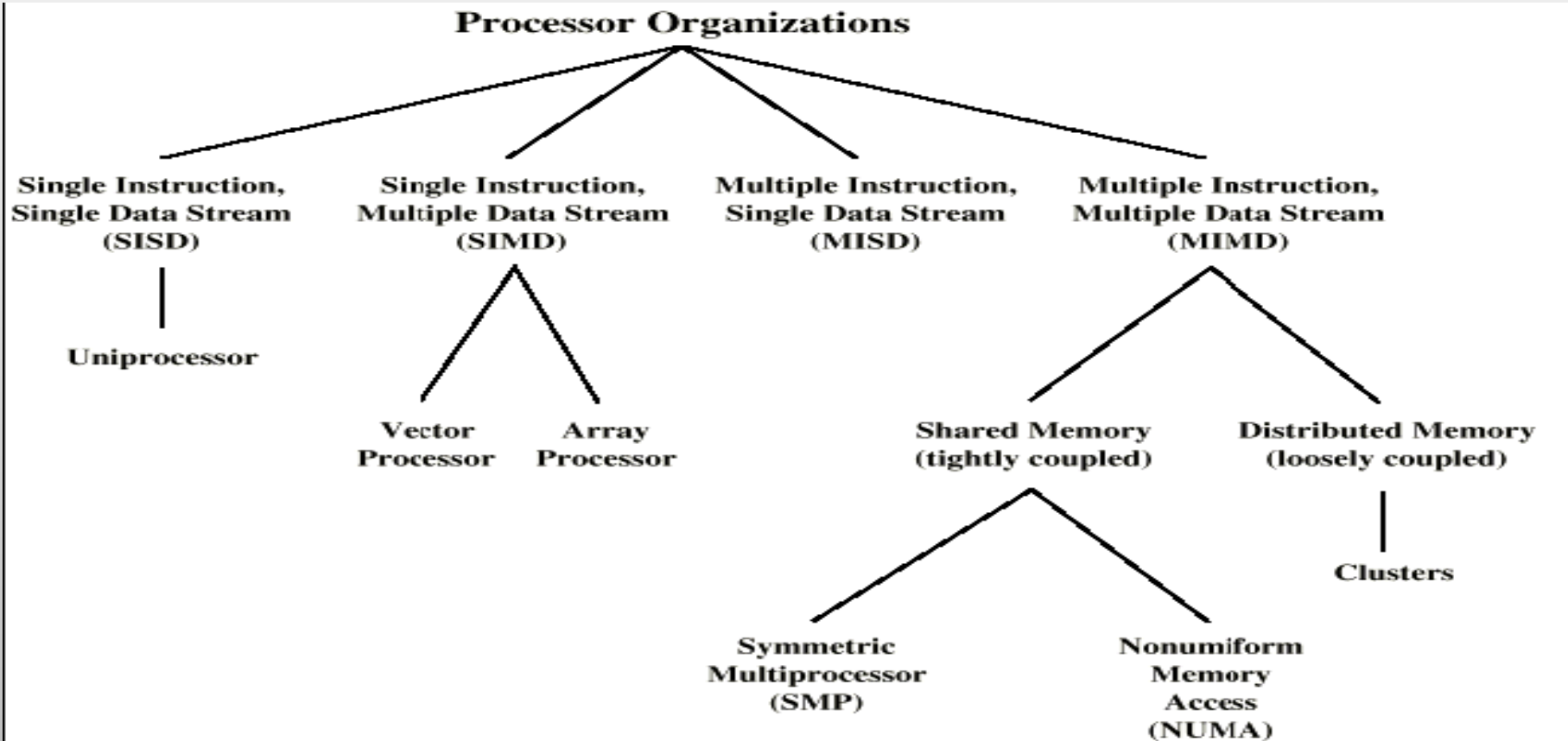
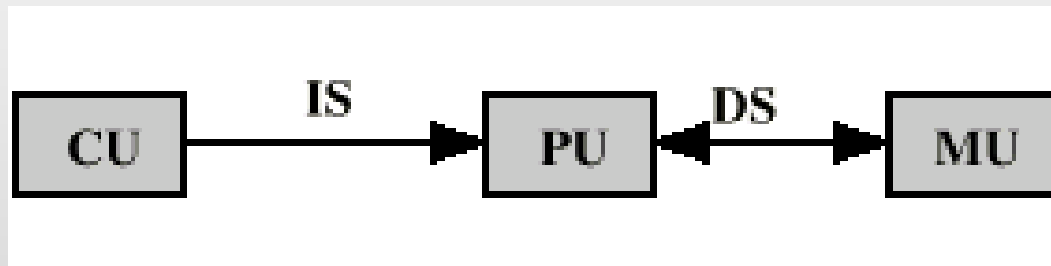


Figure 16.1 A Taxonomy of Parallel Processor Architectures

0 0 0

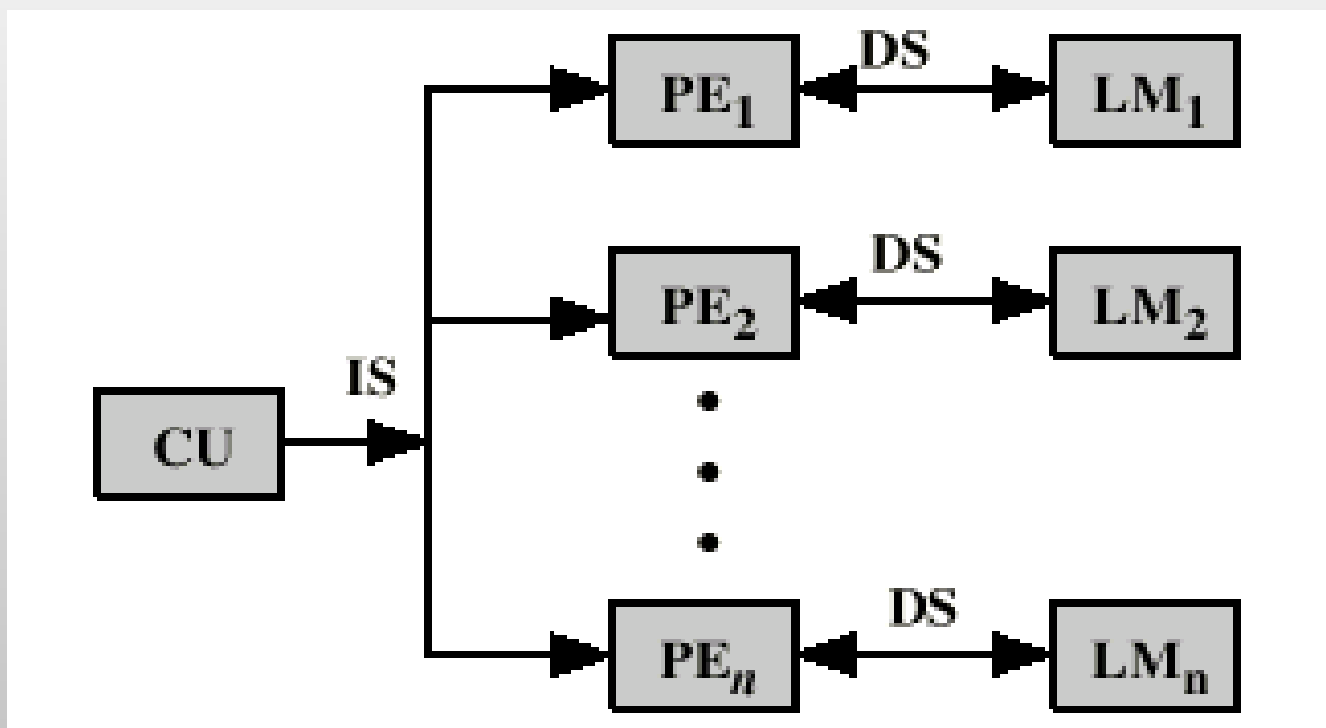
Organizações SISD



- o CU Unidade de Controle
- o IS Sequência de Instruções
- o PU Unidade de Processamento
- o DS Sequência de dados
- o MU Unidade de Memória

0 0 0

Organizações Paralelas: SIMD



- PE - Elemento de Processamento
- ML - Memória Local



Múltiplas instruções, múltiplos dados (MIMD)

- Conjunto de processadores.
- Executa simultaneamente sequências diferentes de instruções.
- Conjuntos de dados distintos.
- Exemplos: SMPs, clusters e sistemas NUMA.

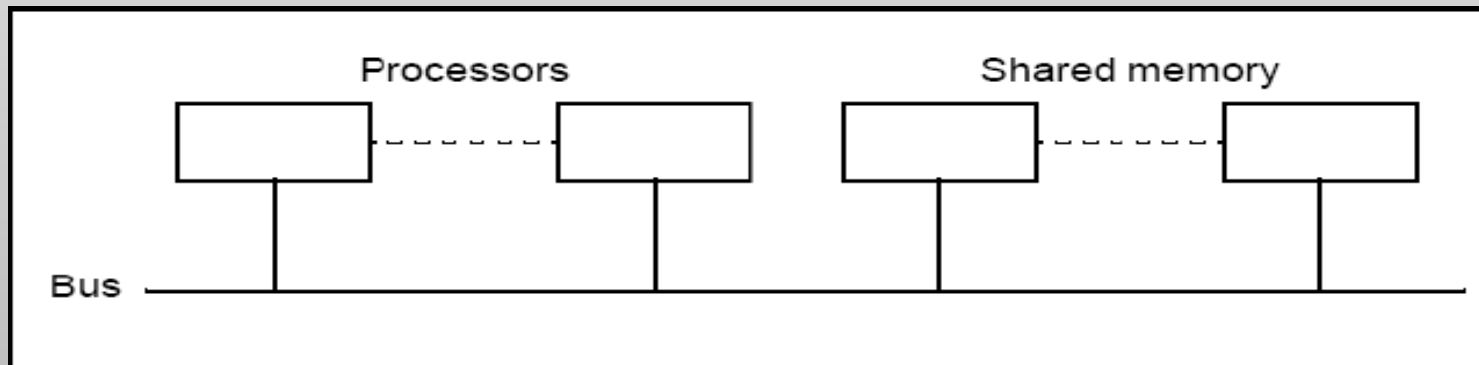
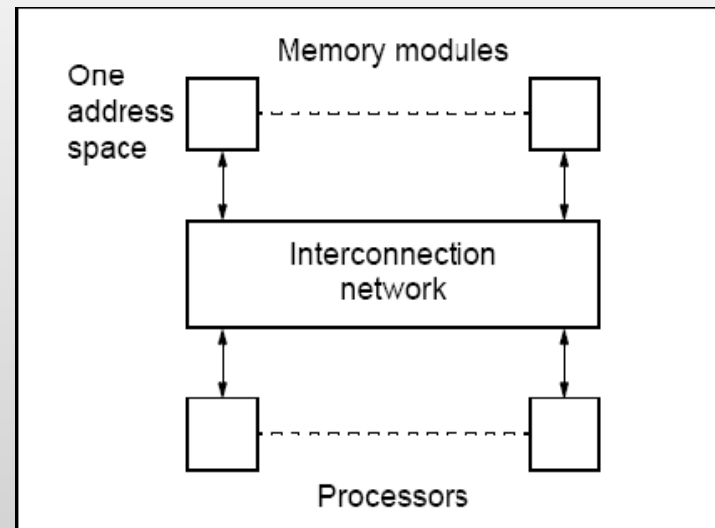
0 0 0

MIMD - Multiprocessador Simétrico

- Processadores compartilham memória.
- Comunicação via memória compartilhada.
- SMP - Symmetric Multiprocessor
 - Compartilham uma memória única ou conjunto
 - Compartilham barramento de acesso à memória
 - Tempo de acesso à memória para uma dada área é aproximadamente o mesmo para qualquer processador.

o o o

Vários processadores, memória compartilhada





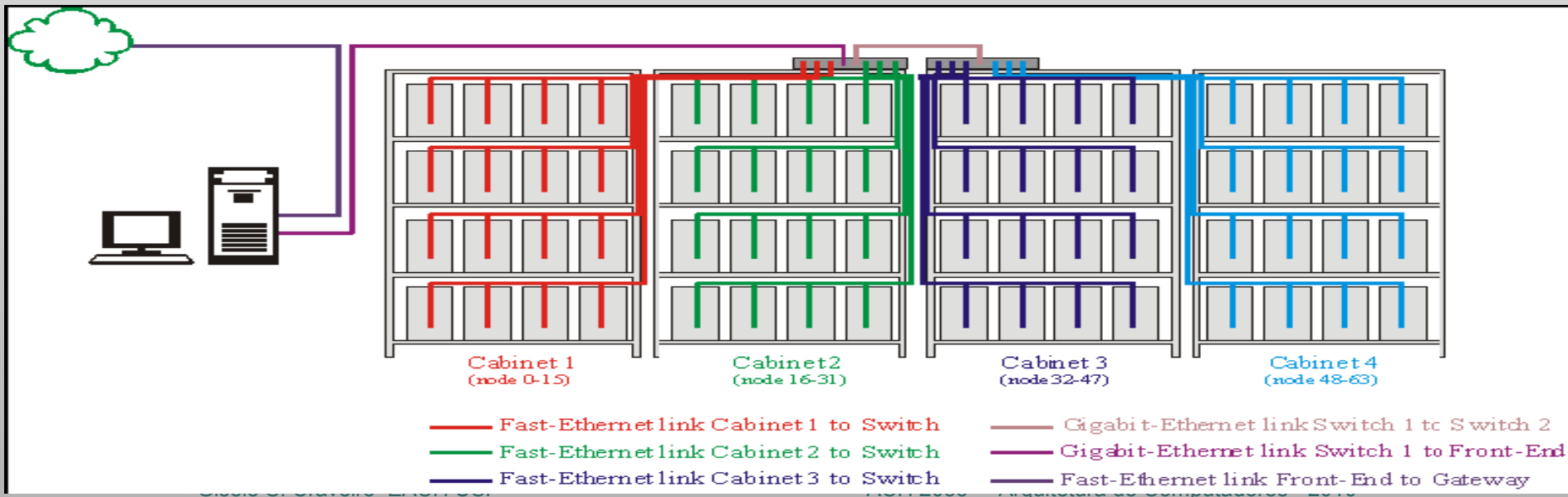
MIMD - NUMA

- o Nonuniform memory access ou Acesso não-uniforme à memória
- o Tempo de acesso a diferentes regiões da memória pode ser diferente para o processador.

0 0 0

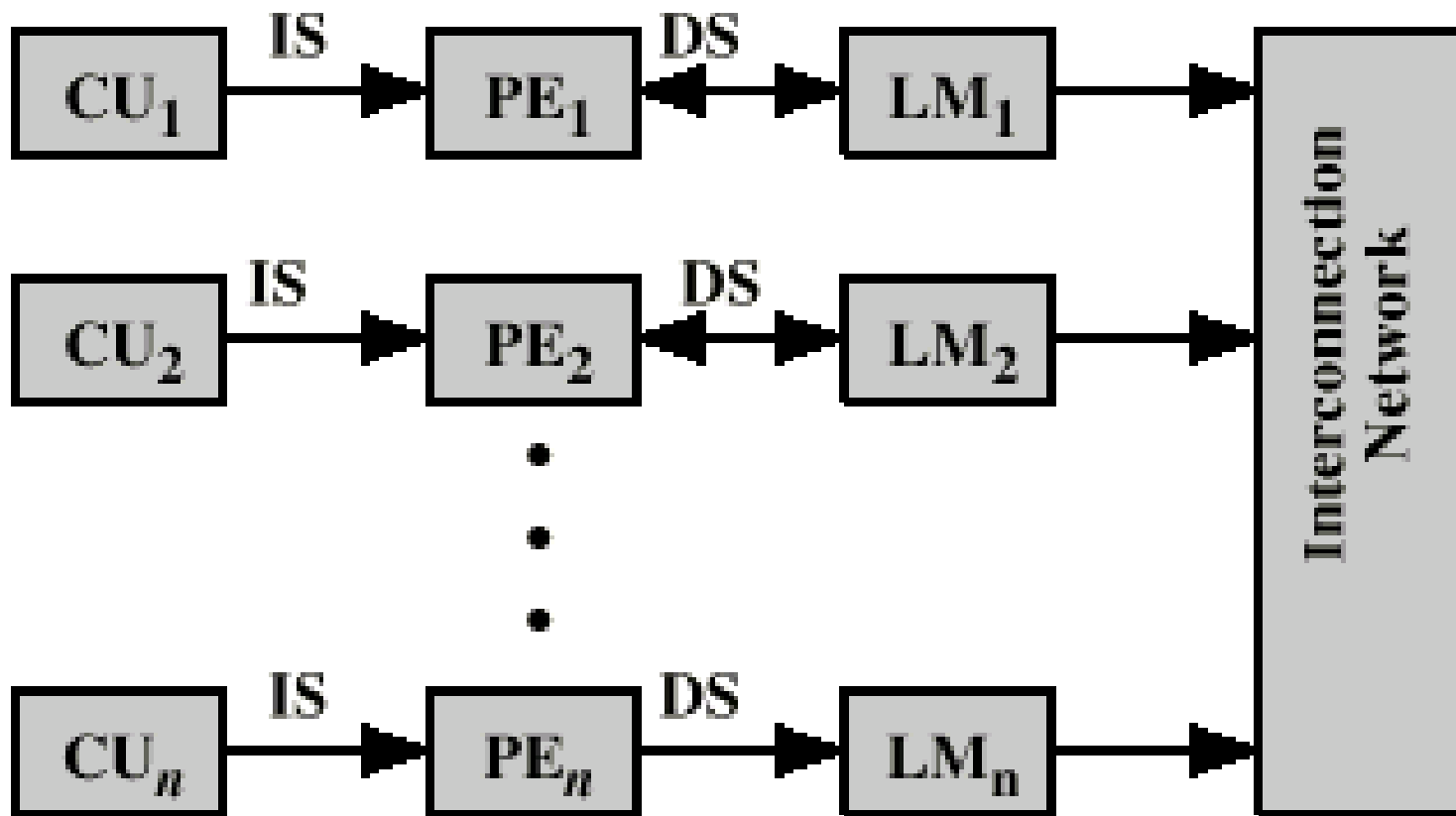
MIMD - Clusters

- o Coleção de uniprocessadores ou de SMPs independentes interconectados.
- o Comunicação via caminho fixo ou conexões de rede.



o o o

Organizações Paralelas: MIMD - Memória Distribuída





Multiprocessadores simétricos

- Refere-se tanto a arquitetura de hardware, quanto ao comportamento do sistema operacional.
- Sistema de computador independente.



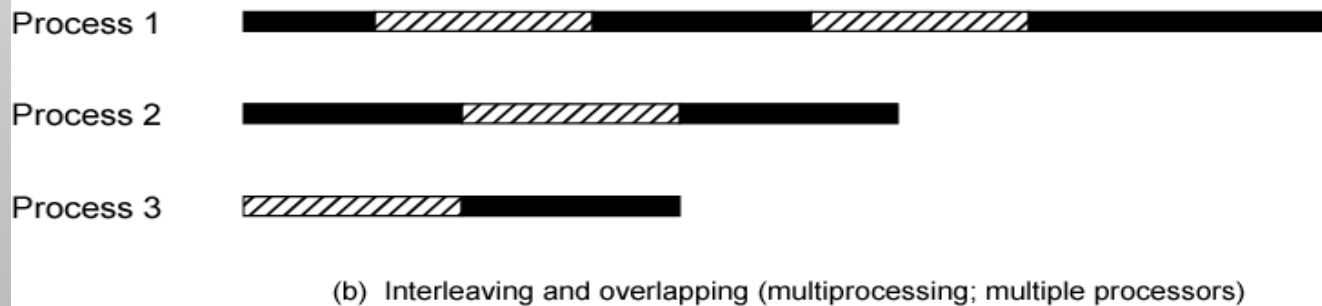
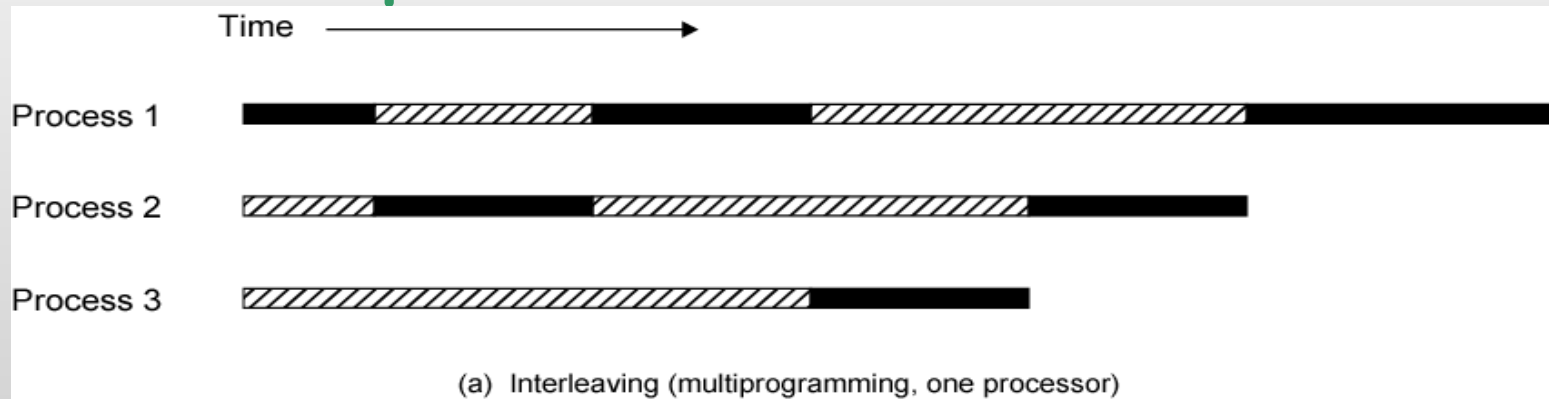
Multiprocessadores simétricos

Características:

- Todos os processadores compartilha acesso a dispositivos de I/O.
- Todos os processadores podem desempenhar as mesmas funções (daí o termo *simétrico*)
- Controlado por SO integrado:
 - Provê interação entre processadores.
 - Interação a nível de tarefas, arquivos e dados.

0 0 0

Multiprogramação e Multiprocessamento



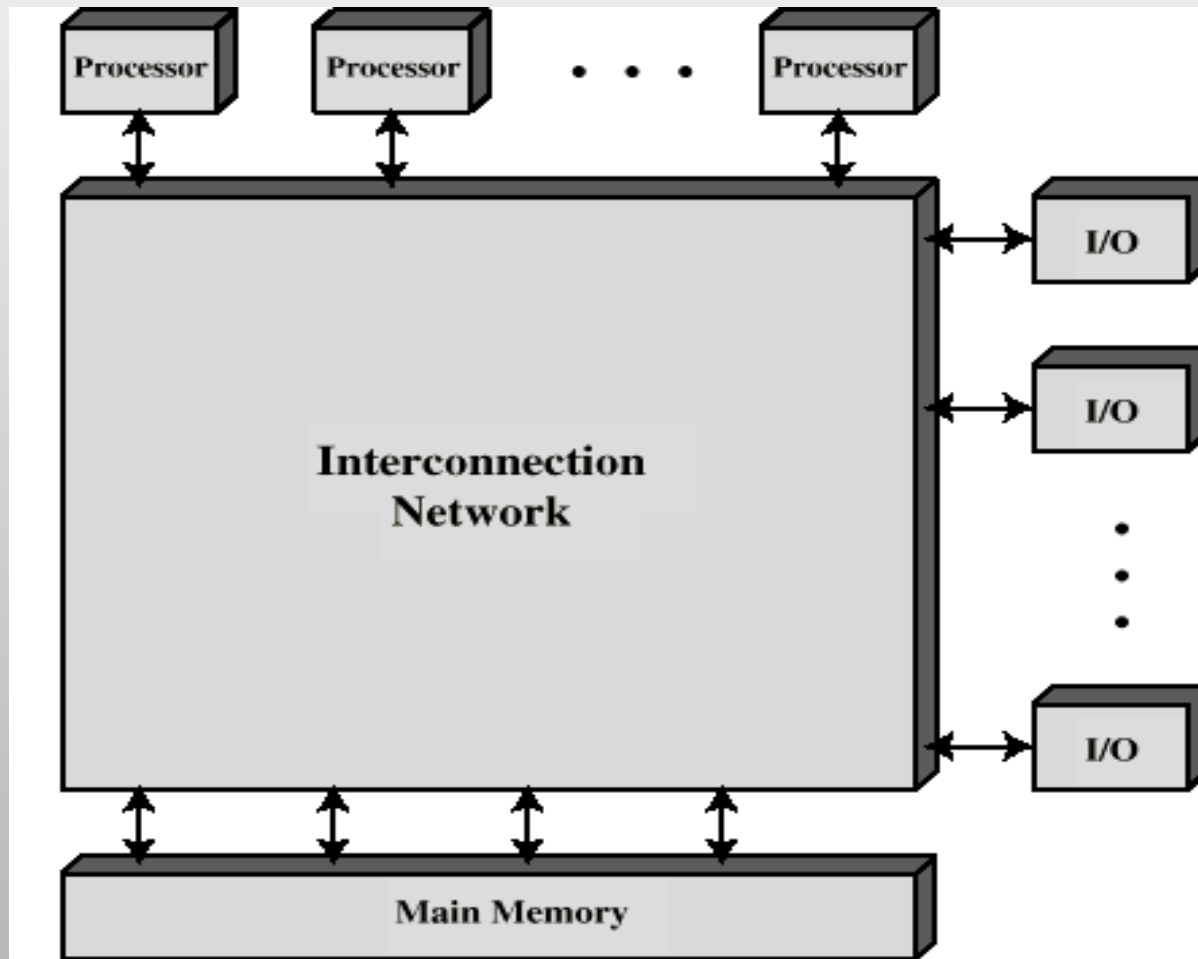


Vantagens de Arquitetura SMP

- o Desempenho
- o Tolerância a defeito
- o Crescimento incremental:
 - o usuário pode aumentar o desempenho do sistema adicionando novos processadores.
- o Escalabilidade:
 - o fabricantes podem oferecer variedade de produtos com características de desempenho e custo diferentes (número de processadores).

o o o

Organização de Sistema Multiprocessador





Classificação da Organização

- o Tempo compartilhado ou barramento comum
- o Memória com múltiplas portas
- o Unidade de controle central

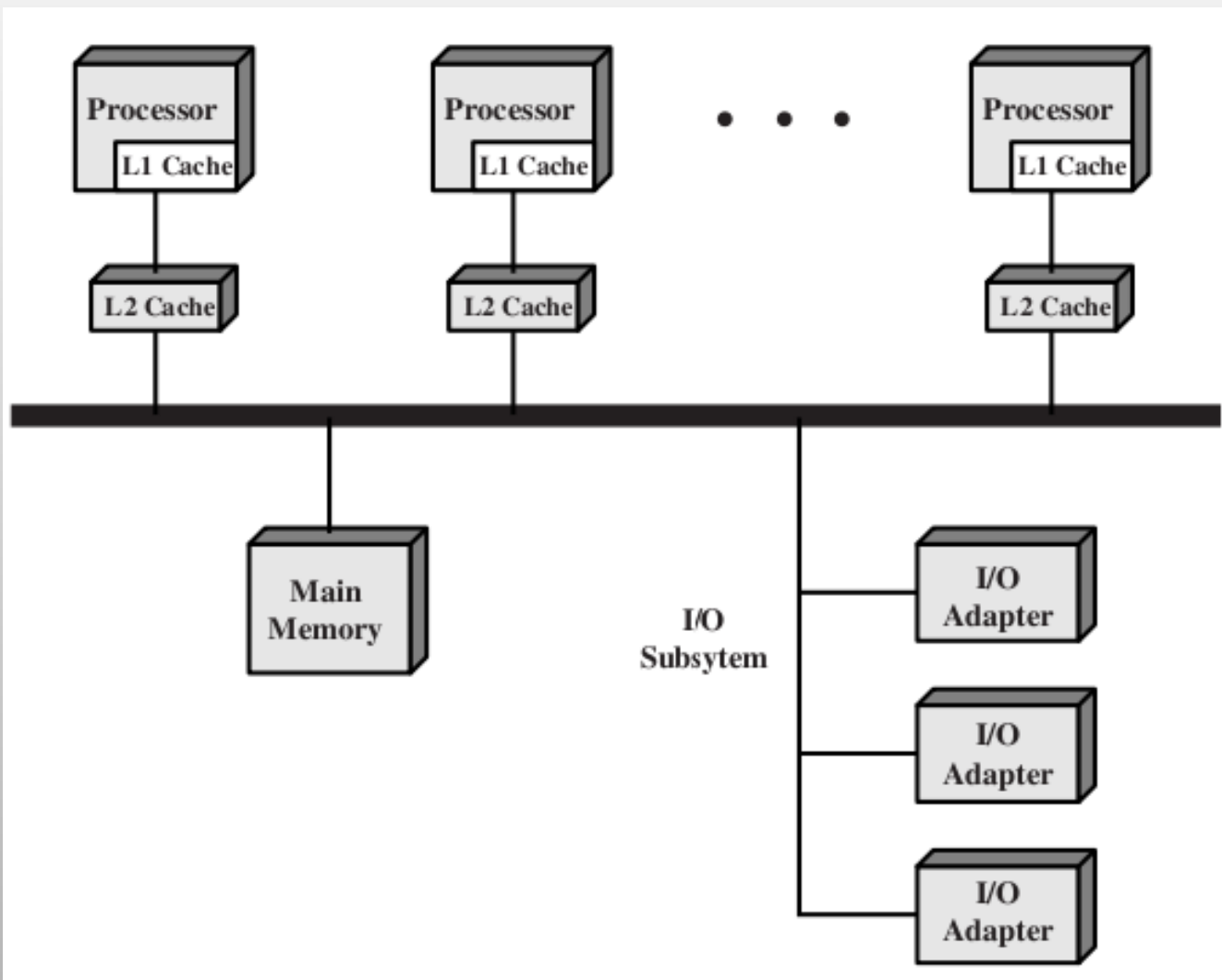
0 0 0

Barramento de tempo compartilhado

- Forma mais simples de construção SMP
- Estrutura e interface similar a sistema de único processador.
- Provê os seguintes recursos:
 - endereçamento: distinguir módulos conectados ao barramento.
 - arbitração: qualquer módulo de I/O pode funcionar temporariamente como mestre.
 - compartilhamento de tempo: se um módulo controla o barramento, os demais devem suspender a operação e esperar a liberação.

0 0 0

Organização de SMP





Barramento de tempo

compartilhado - Vantagens

- o Simplicidade
- o Flexibilidade:
 - o expansão
- o Confiabilidade:
 - o barramento é um meio passivo, e a falha em um dispositivo a ele conectado não deve causar a falha no sistema como um todo.

o o o

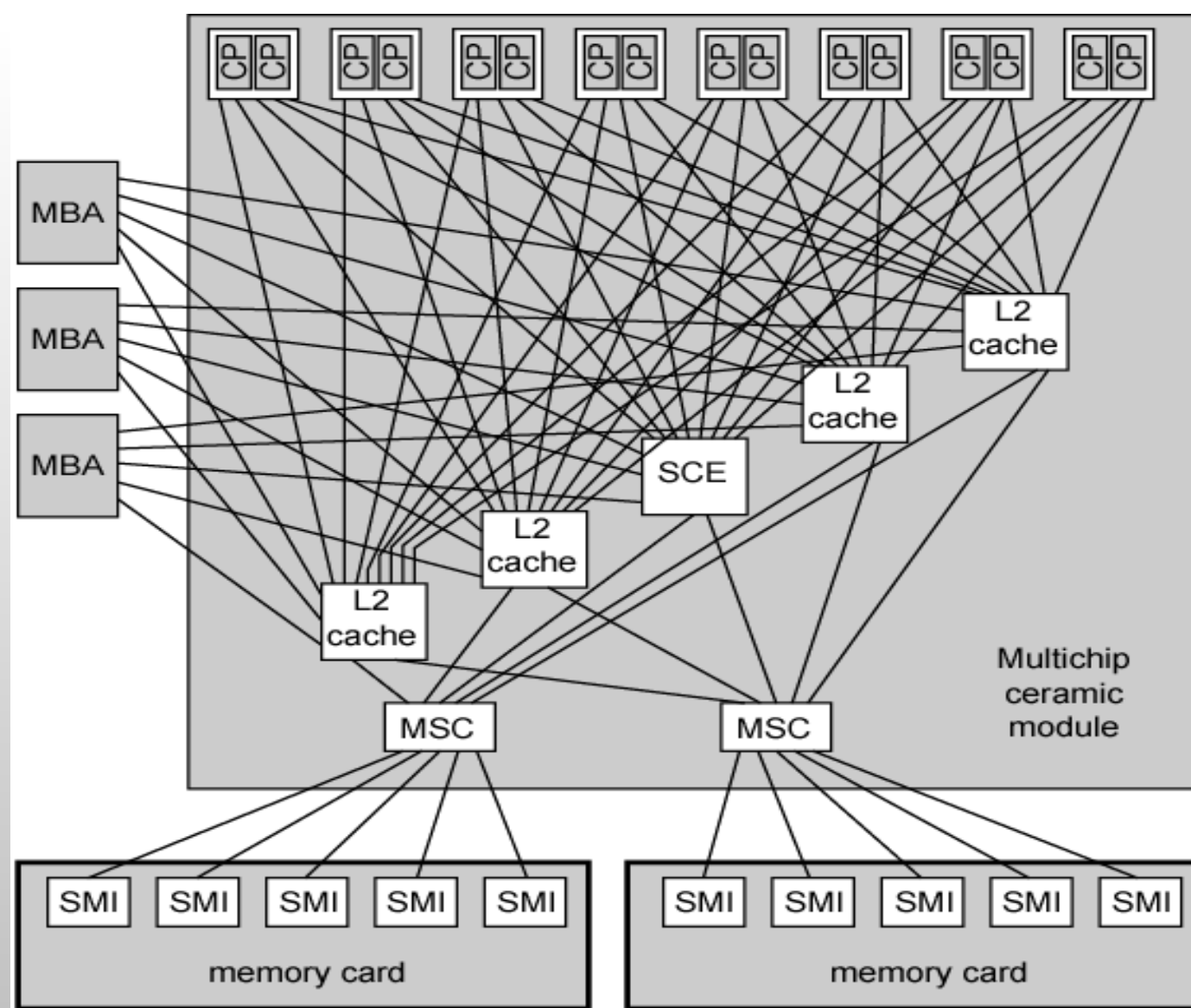
Barramento de tempo compartilhado - Desvantagens

- o Desempenho:
 - o Limitada pelo tempo de ciclo do barramento.
- o Cada processador deveria ter cache local
 - o Reduziria número de acessos ao barramento.
 - o Introduziria problemas de coerência de cache

000

IBM

zSeries



CP = central processor
MBA = memory bus adapter
MSC = main store control
SCE = system control element
SMI = synchronous memory interface



Memória com múltiplas portas

- o Cada processador e módulos de I/O possui acesso direto e independente aos módulos de memória.
- o Conflitos de acesso
 - o Lógica para resolução de conflitos.
 - o Normalmente: prioridade permante associada a cada porta.
- o Pouca ou nenhuma modificação necessária nos processadores e módulos de I/O.

o o o | Memória com múltiplas portas - Vantagens e Desvantagens

- o Mais complexa:
 - o lógica extra no sistema de memória.
- o Melhor desempenho:
 - o caminho dedicado.
- o Possível configurar partes da memória como dedicada a um processador ou I/O
 - o aumenta segurança (acessos)
- o Deve utilizar política de escrita direta no cache (write through).



Unidade de controle central

- Comanda fluxos de dados distintos de e para módulos diferentes (processador, memória, I/O).
- Pode armazenar requisições temporariamente.
- Executa funções de arbitração e temporização.



Unidade de controle central

- Passa mensagens de controle e de estado entre processadores.
- Alerta quando cache é atualizado.
- Utilizada em sistemas como IBM S/370, porém raramente utilizada atualmente.



Unidade de controle central - Vantagens e Desvantagens

- o Flexível e simples:
 - o Interfaces dos módulos são as mesmas.
- o UCC complexa!
 - o Potencial gargalo no sistema.



Questões de Sistemas Operacionais

- o Processos concorrentes simultâneos
- o Escalonamento (scheduling)
- o Sincronização
- o Gerenciamento de memória
- o Confiabilidade e tolerância a falhas



Coerência de cache

- o Problema: múltiplas cópias do mesmo dado em caches diferentes!
- o Pode resultar em inconsistência do ponto de vista da memória!
- o Política *write back* (escrita de volta): pode levar diferenças nos caches.
- o Política *write through*: precisa monitorar tráfego cache-memória.



Soluções por Software

- Compilador e SO tratam o problema.
- Detecção de potenciais problemas em tempo de compilação.
- Complexidade de projeto transferida do hardware para software.
- Porém as decisões por software tendem a decisões conservadoras:
 - utilização ineficiente do cache.



Soluções por Software

- Abordagem mais simples:
- evita que variáveis compartilhadas sejam armazenadas em cache
- Abordagem mais eficiente:
- analisa o código para determinar períodos em que variáveis compartilhadas possam ser armazenadas em cache de forma segura.



Soluções por hardware

- o Protocolos de Coerência de Cache. Características:
- o Tempo de execução.
- o Reconhecimento dinâmico de potenciais problemas.
- o Transparente para programador e compilador.
- o Uso mais eficiente do cache.

0 0 0

Soluções por hardware

- o Dois tipos:
- o Protocolos de Diretório.
- o Protocolos de Monitoração (Snoopy protocols).



Protocolos de Diretório

- Coleta e mantém informação sobre as cópias de dado em cache.
- Controlador central: Diretório armazenado em memória principal. Requisições são verificadas no diretório.
- Cria gargalo central.
- Eficazes em sistemas de grande escala.



Protocolos de Monitoração

- o Distribui responsabilidade de coerência de cache entre os controladores de cache.
- o Cache reconhece que a linha é compartilhada.
- o Anuncia atualizações a demais caches.
- o Adequada a multiprocessadores baseados em barramento.
- o Aumenta tráfego no barramento

0 0 0

Protocolos de Monitoração

- o Duas abordagens:
- o escrita com invalidação (write-invalidate);
- o escrita com atualização (write-update).

o o o

Escrita com atualização (write-update)

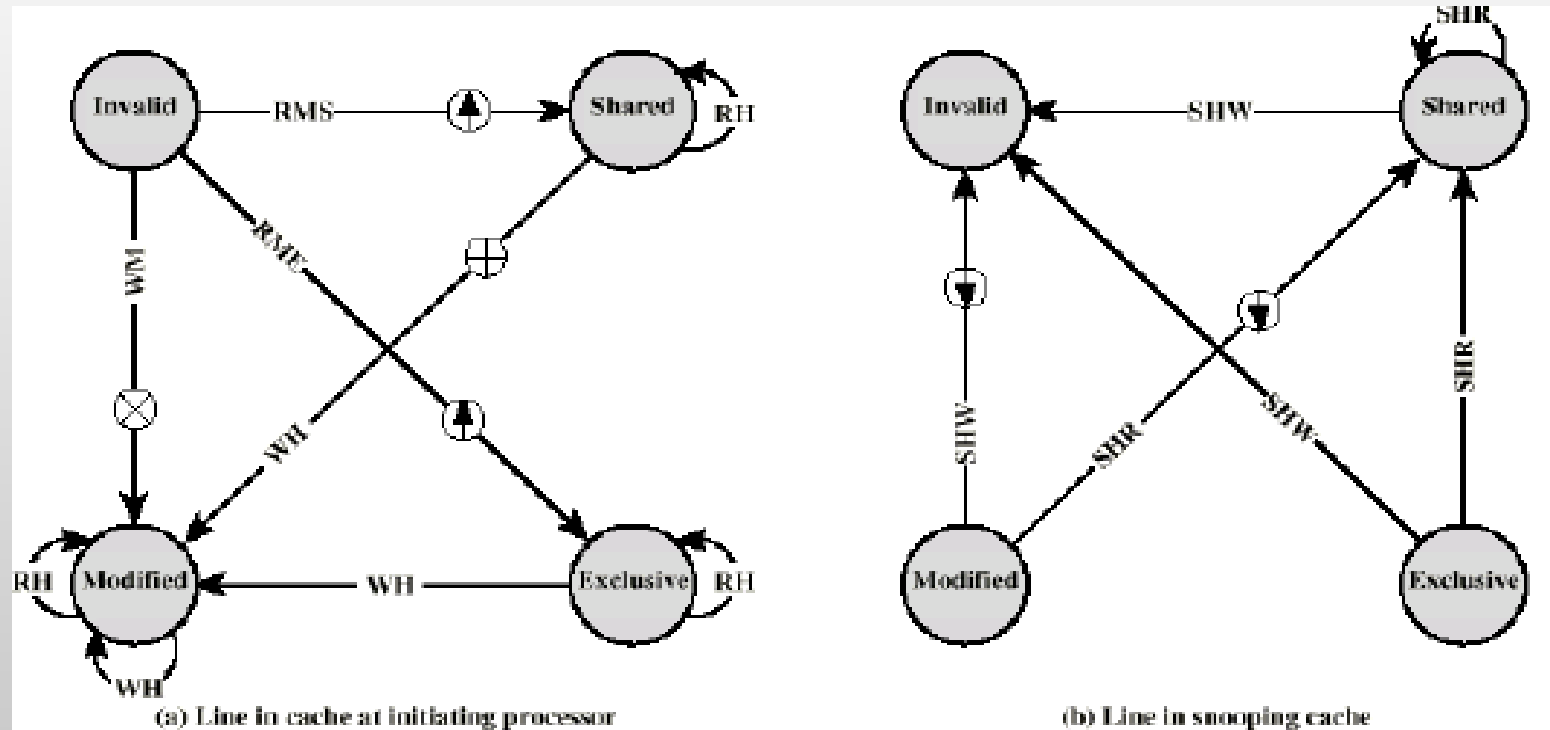
- o Múltiplos leitores e escritores.
- o Linha atualizada é distribuída a todos os demais processadores.

Escrita com invalidação (write-invalidate)

- o Múltiplos leitores, apenas um escritor.
- o Quando uma escrita é requisitada, a linha nos demais caches é invalidada.
- o Processador escritor terá acesso exclusivo, até que a linha seja requisitada por outro processador.
- o Usado no Pentium II e PowerPC.
- o Estado de cada linha é marcado como: modified, exclusive, shared or invalid (MESI).

0 0 0

Diagrama de Estados



RH Read hit
RMS Read miss, shared
RME Read miss, exclusive
WH Write hit
WM Write miss
SHR Snoop hit on read
SHW Snoop hit on write or read-with-intent-to-modify

⬇️ Dirty line copyback
 ⊕ Invalidate transaction
 ⊗ Read-with-intent-to-modify
 🌲 Cache line fill

0 0 0

Aumentando o desempenho

- Desempenho do processor pode ser medido pela taxa de execução de instruções

$$\text{MIPS rate} = f * \text{IPC}$$

- f freq. de clock frequency, em MHz
- IPC média de instruções por ciclo

0 0 0

Aumentando o desempenho

- Aumento de desempenho pode ser dado através do aumento da freq de clock e o aumento do número de instruções completas por ciclo
- Pode estar alcançado o limite
- Complexidade
- Consumo de energia



Definições de threads e processos

- Thread em processadores multithreaded pode ser ou não a mesma coisa que threads de software
- Processo:
 - Uma instancia da execução de programa
 - Posse de recurso
 - Espaço de endereçamento virtual para manter a imagem do processo
 - Escalonamento/execução
 - Troca de processo



Definições de threads e processos

- Thread: unidade despachável de trabalho dentro de um processo
 - Inclui o contexto (contador de programa e ponteiro de pilha) e área de dados para a pilha
 - Thread executa sequencialmente
 - Interrompível: processador pode trocar para outra thread
- Troca de thread
 - Processador troca entre threads do mesmo processo
 - Tipicamente custa menos que a troca de processo



Abordagens de Multithreading

- o Intercalada
 - o Granularidade fina
 - o Processador lida com dois ou mais contextos de threads ao mesmo tempo
 - o Troca de thread a cada ciclo de clock
 - o Se a thread é bloqueada ela é pulada
- o Bloqueada
 - o Granularidade grossa
 - o Thread executada até que ocorre um evento que cause demora
 - o Ex .Cache miss
 - o Efetiva em processador in-order
 - o Evita pipeline stall

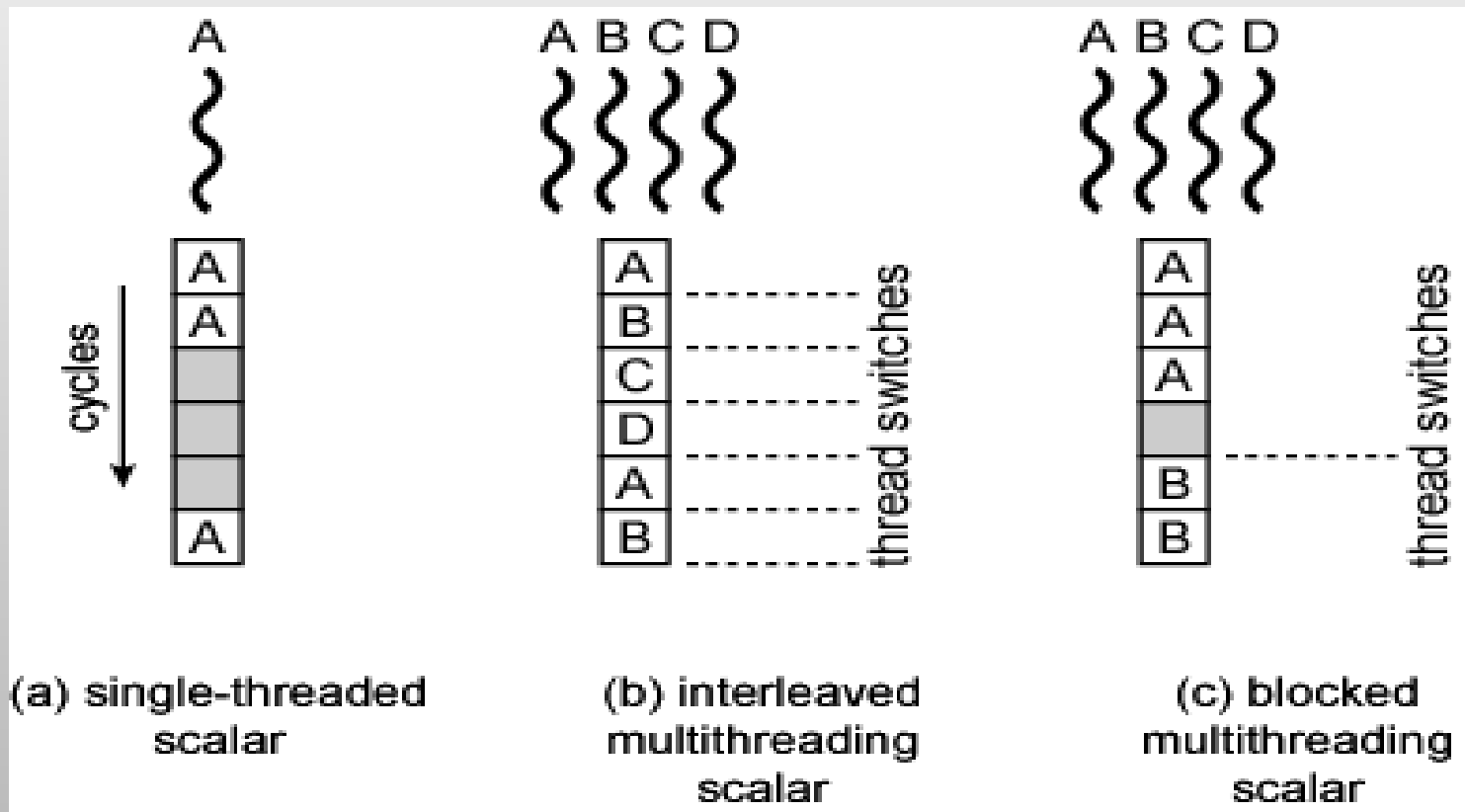


Abordagens de Multithreading

- Simultânea (SMT)
 - As instruções são simultaneamente enviadas a partir de múltiplas threads para unidades de execução de processador superscalar
- Multiprocessamento de Chip
 - Processador é replicado num único chip
 - Cada processador lida com threads separadas

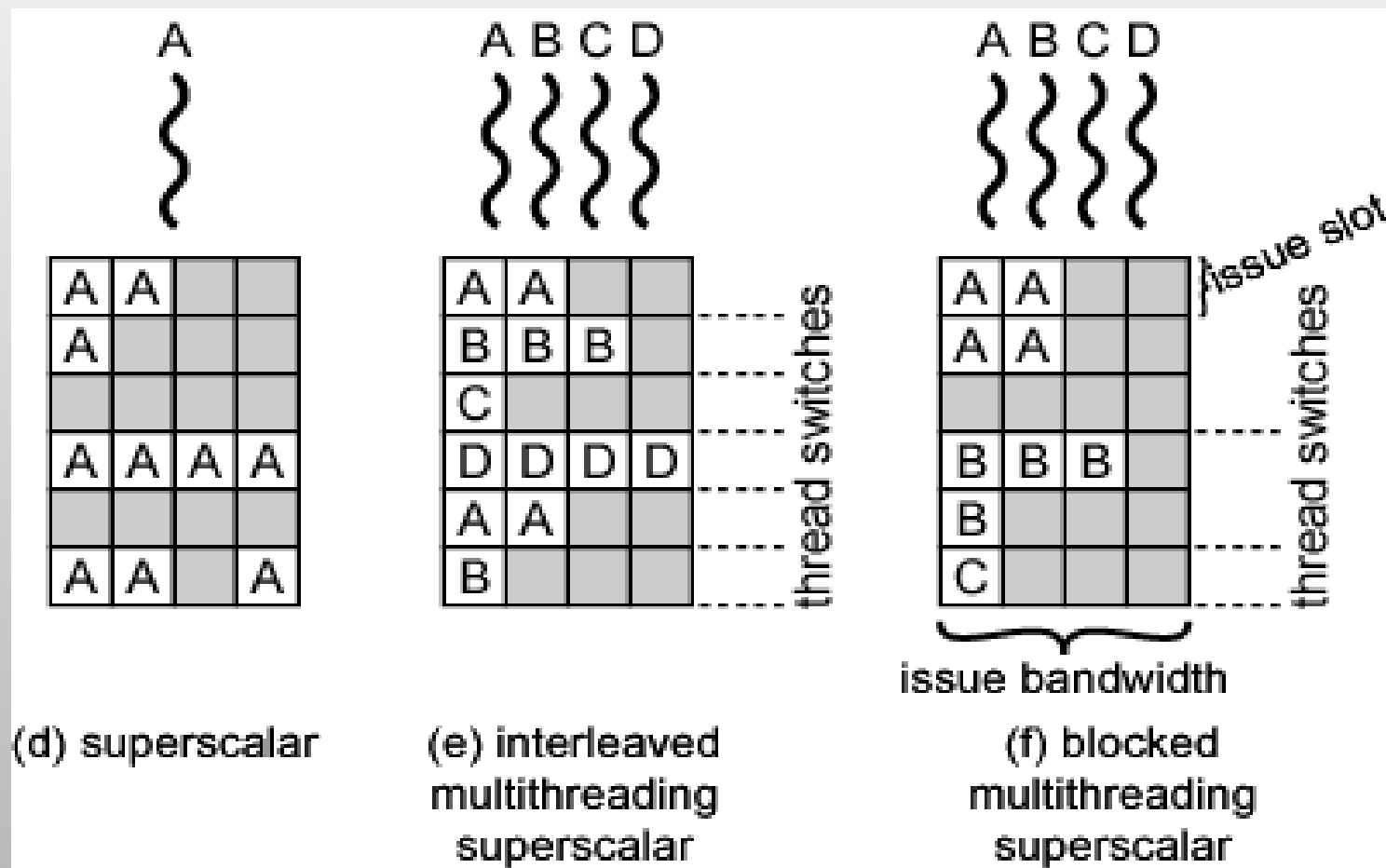
0 0 0

Processador Escalar - Abordagens



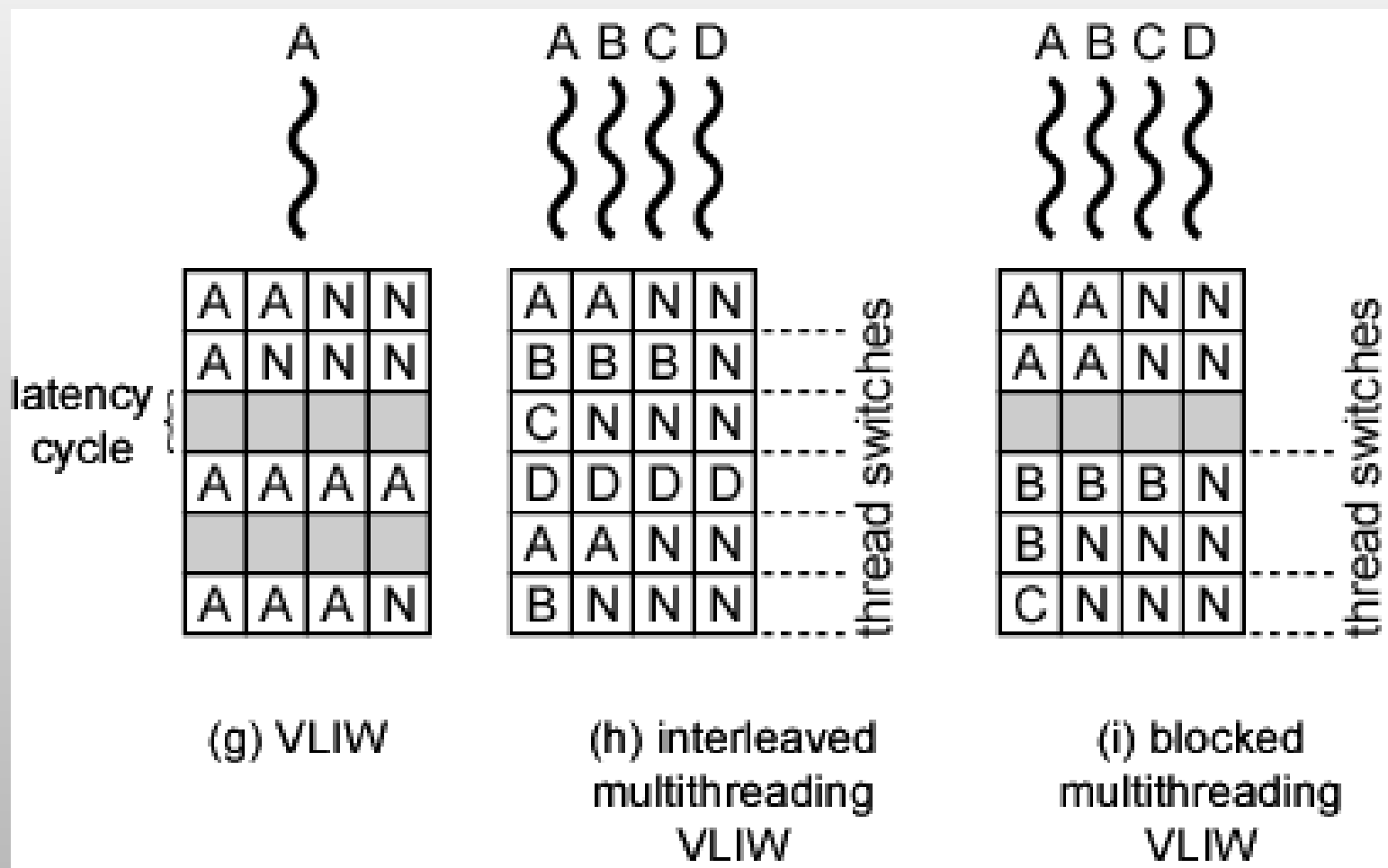
0 0 0

Envio de múltiplas instruções (1)



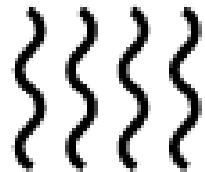
0 0 0

Envio de múltiplas instruções (2)




0 0 0

Threads Simultâneas

A B C D


A	A	A	A	B	B	B	C
D	D	D	A	A	A	B	D
D	D	D	A	A	A	B	C
B	D	A	A	A	A	B	B
C	D	D	A	A	A	A	A
A	B	B	D	D	D	D	D

(j) simultaneous
multithreading
(SMT)

A B C D


A	A	B	B	C			
A		B	B				D
		B					D
A	A			C			D
		B	B	C	C		D
A	A	B		C	C		D

(k) chip multiprocessor



Clusters

- o Alternativa a SMP. Bom desempenho. Alta disponibilidade.
- o Atrativo para aplicações baseadas em servidores.
- o Melhor relação custo/desempenho.



Clusters

- o Grupo de computadores completos interconectados.
- o Trabalham juntos, como um recurso de computação unificado.
- o Cria a ilusão de ser uma máquina única. Cada computador é um nó do cluster.



Configurações de Clusters

- o Classificação simples:
- o Servidor Independente, sem compartilhamento de discos.
- o Discos compartilhados.

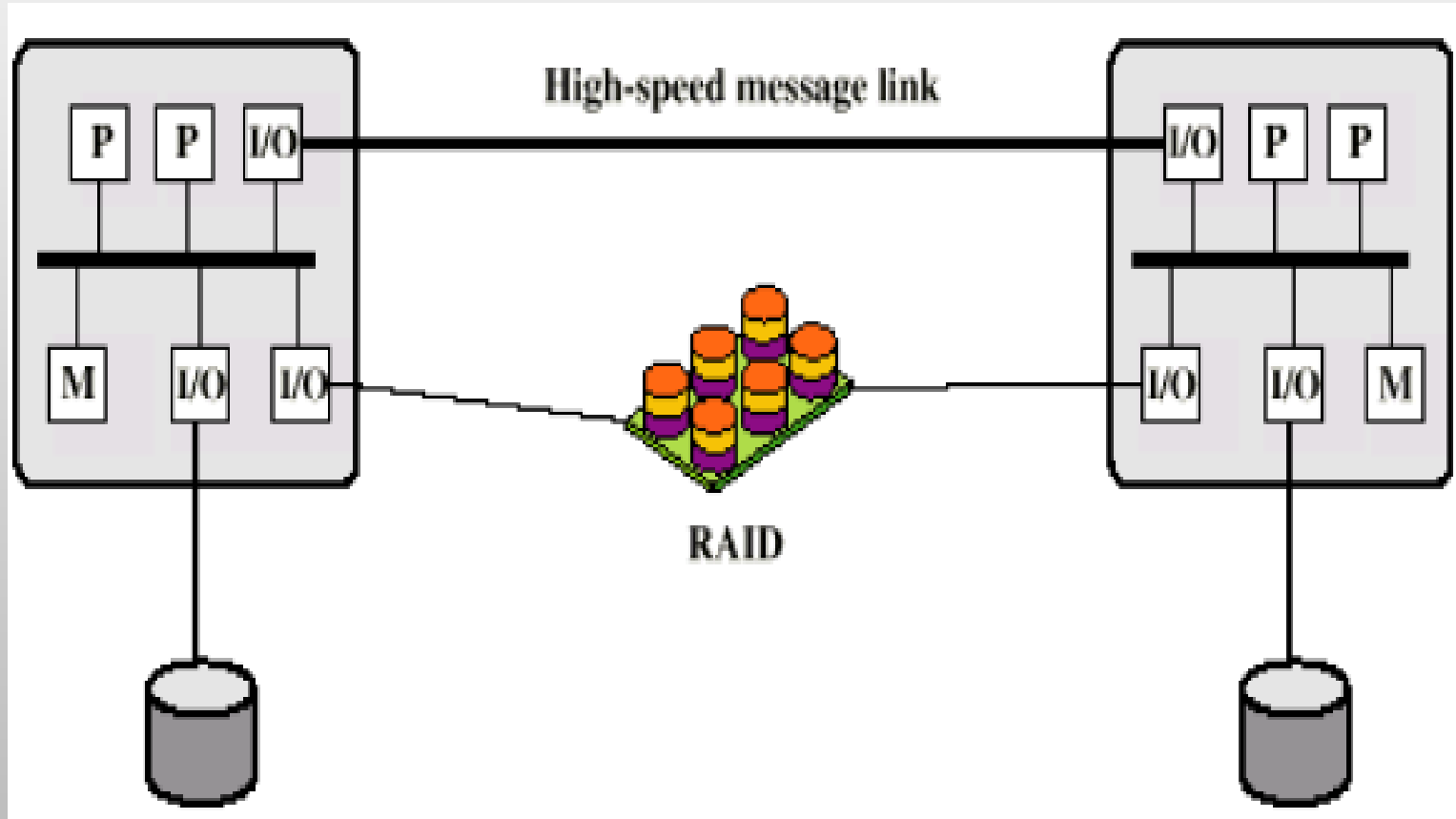
0 0 0

Servidor Independente, sem compartilhamento de discos



0 0 0

Discos Compartilhados





Sistemas Operacionais

- o Gerenciamento de falhas:
 - o Cluster tolerante a falha:
 - o todos os recursos sempre disponíveis;
 - o uso de discos compartilhados redundantes; mecanismos para manter cópias de transações ainda não confirmadas e para confirmar transações completadas.
- o Balanceamento de carga:
 - o mecanismos para incluir um computador adicionado ao cluster no escalonamento.

Computação Paralela

- Compilação Paralela

- Determina em tempo de compilação que partes podem ser executadas em paralelo
- Divisão em diferentes computadores

- Aplicação

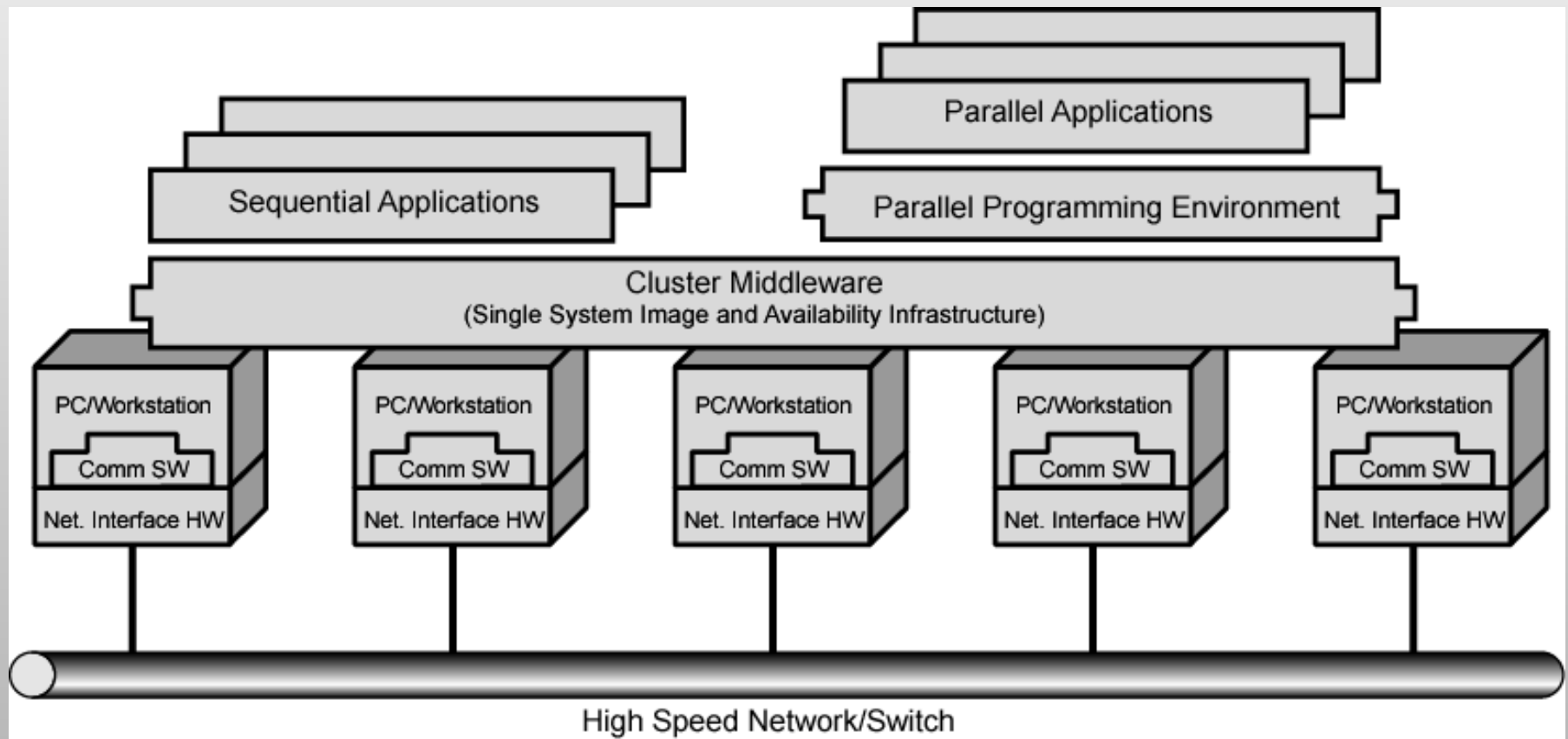
- Aplicação escrita desde o início para ser paralela
- Passagem de mensagem para mover dados entre os nós
- Difícil de programar, porém obtém melhores resultados

- Computação Paramétrica

- Problema é a execução de um algoritmo sobre diferentes conjuntos de dados. Ex. Simulação usando diferentes cenários
- Necessita de ferramentas para organizar a execução

0 0 0

Arquitetura de um cluster computacional





Cluster X SMP

SMP:

- o configuração mais fácil;
- o mais próximo do uniprocessador; principal mudança de uniprocessador: função de escalonamento;
- o menor espaço físico e menos energia;
- o produtos estáveis.

0 0 0

Cluster X SMP

- o Cluster:
 - o maior escalabilidade absoluta e incremental;
 - o melhor disponibilidade.
- o Tendência: clusters.



Acesso não-uniforme a memória (NUMA)

- o todos os processadores acessam a memória principal via "lw e sw".
- o Tempo de acesso difere conforme a região da memória sendo usada e processador.
"Equivalente" a cluster.
- o NUMA com coerência de cache:
 - o sistema NUMA com coerência de cache dos vários processadores.



NUMA: Motivação

- o SMP praticamente alcançou o limite de processadores
 - o Tráfego no barramento limita entre 16-64 processadores
- o Nos clusters cada nó tem sua memória
 - o Aplicações não veem uma grande memória global
 - o Coerência mantida por software e não por hardware
- o NUMA procura manter características do SMP aliadas ao multiprocessamento em larga escala
 - o e.x. Silicon Graphics Origin NUMA com 1024 processadores MIPS

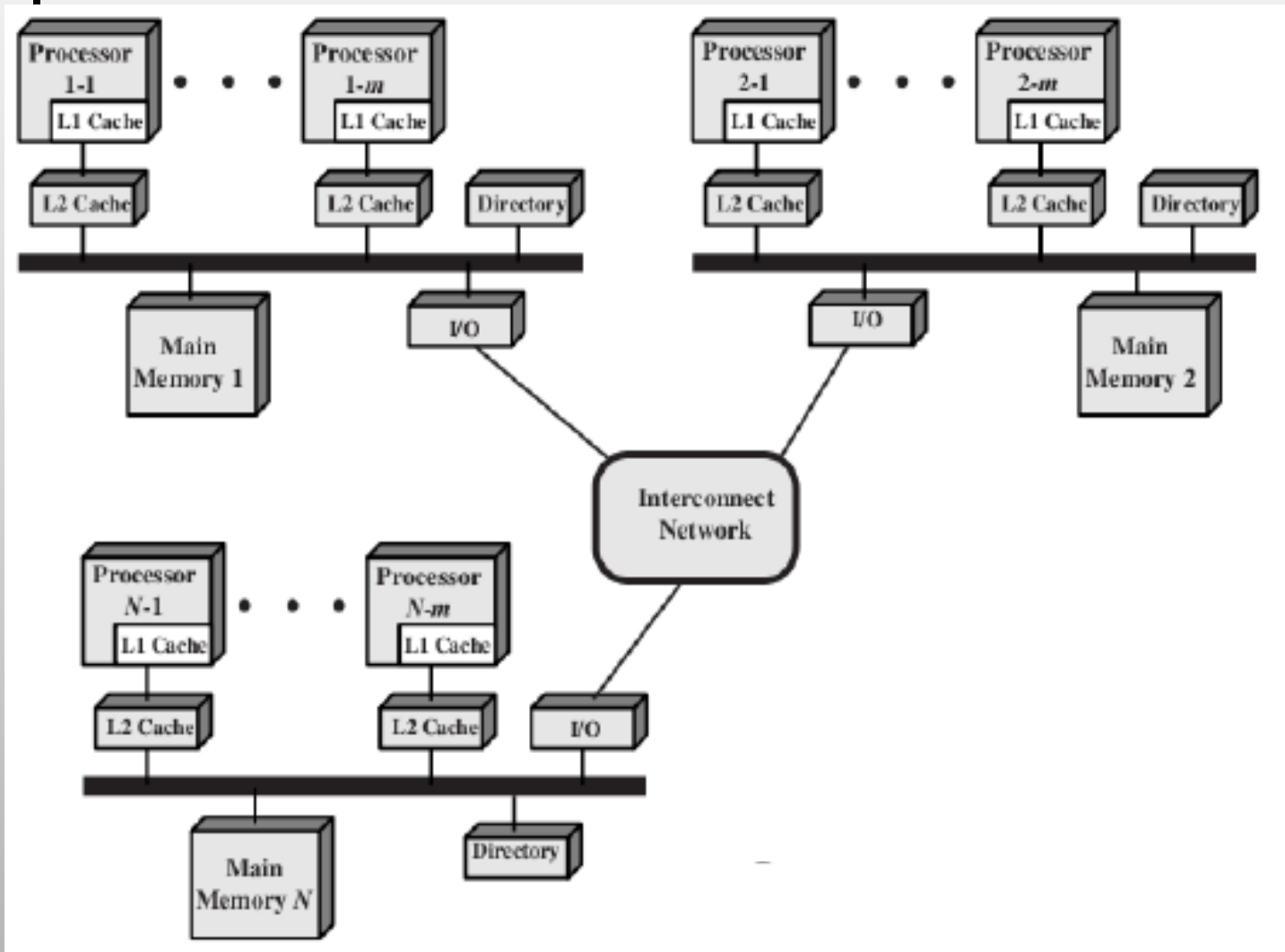


NUMA: Objetivo

- o Manter, de forma transparente, uma visão de uma grande e única área de memória do sistema, permitindo ao mesmo tempo vários nós multiprocessadores, cada qual com seu próprio barramento ou outro sistema interno de interconexão.

0 0 0

Organização CC-NUMA





CC-NUMA: Vantagens e Desvantagens

- Desempenho efetivo em níveis de paralelismo maior que SMP.
- Desempenho diminui de possui muitos acessos a memória localizada em nós remotos.
- Sistema não é transparente como SMP: modificações de SO e programa. Disponibilidade depende da implementação do sistema.



Computação Vetorial

- o Necessidade de resolver problemas matemáticos relativos a processos reais, como em aerodinâmica, sismologia, meteorologia e física atômica, nuclear e de plasma.
- o Requerem:
 - o alta precisão numérica;
 - o operações aritméticas de ponto flutuante em grandes vetores de números.



Processador Matricial

- Supercomputador: otimizado para computação vetorial, porém suporta operações de propósito geral (escalares e tarefas de processamento de dados genéricas).
- Processador Matricial: não inclui processamento escalar!
 - configurados como periféricos para executar partes vetorizadas de programas.



Abordagens para

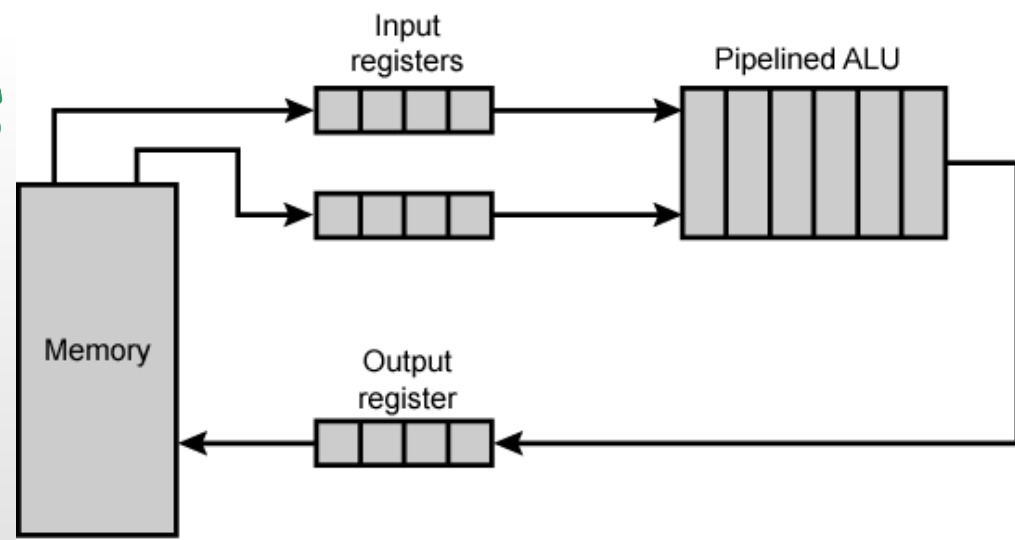
processamento vetorial

- Tarefa: efetuar operações aritméticas sobre vetores ou matrizes de números de ponto flutuante.
- Em computador de propósito geral:
- requer iteração sobre cada elemento do vetor ou da matriz.
- Processamento vetorial: operações manipulam vetor de dados unidimensional.

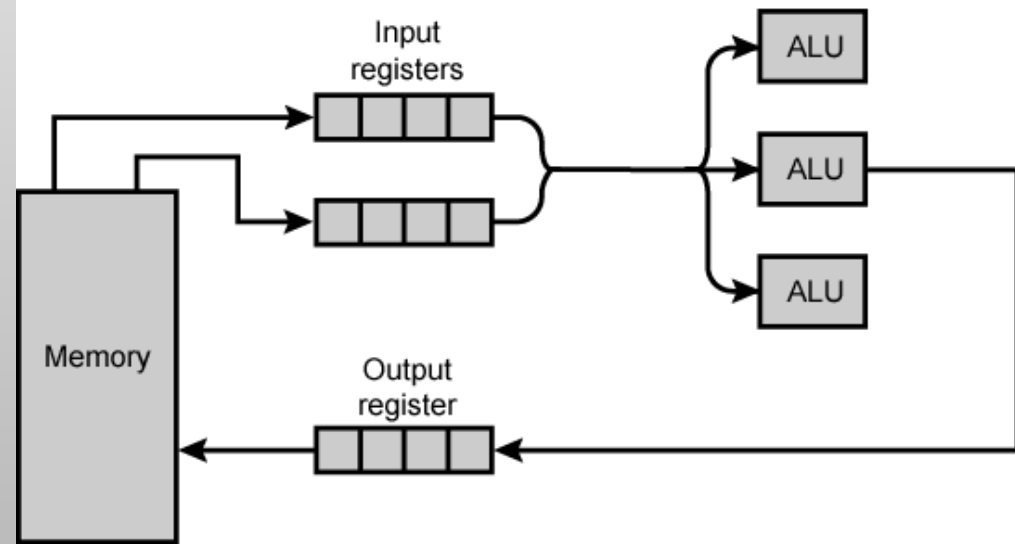
0 0 0

Organizações do processador

- ALUs com pipeline
- ALUs Paralelas
- Processadores Paralelos



(a) Pipelined ALU

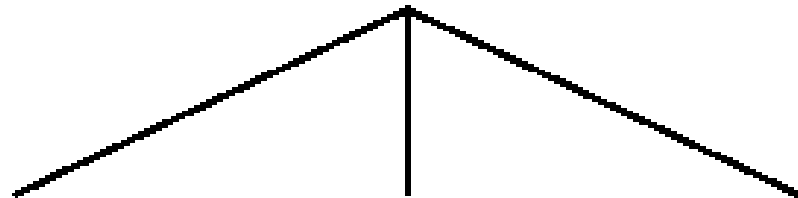


(b) Parallel ALUs

0 0 0

Organizações do processador

Single Control Unit



Multiple Control Units

