

Características de Aplicações Técnicas vs. Comerciais Parte II

“Contrasting Characteristics and Cache Performance of
Technical and Multi-user Commercial Workloads”

By: Ann Marie Grizzaffi

Edward D Moreno

<http://www.lsi.usp.br/~edmoreno>

Resumo: Diferentes Atividades de Desempenho

Ambientes Comerciais e Técnicos (Engenharia) diferem nas suas respectivas características de predição de branch, atividade do sistema operacional, execução dinâmica, memória e cache, e atividade de I/O.

Os resultados dos caches de instruções mostraram que para aplicações comerciais o desempenho desse cache não pode ser desprezado, pois tais workloads possuem working sets (de código) muito grandes quando comparados aos técnicos.

Todos os resultados obtidos mostrando diferenças ajudam e ajudarão a projetistas e usuários a selecionar e aproveitar melhor certas características de algumas máquinas e de algumas aplicações.

Plataforma e Validação Experimental

Os resultados obtidos por Ann Marie et Al., foram obtidos através de simulação baseada em rastros (“trace driven simulation”).
Os “traces” foram capturados num IBM RISC System/6000 & AIX.

Usam-se workloads de aplicações técnicas e comerciais. Tanto para ambientes mono-usuários quanto multi-usuários. Aplicações variam de processamento de transações on-line a programas área química.

Os traces comerciais representam algumas medidas padrões mencionados na indústria (desempenho comercial) e os traces de técnicos representam trabalhos realizados em ambientes típicos de engenharia e de uso numérico intensivo.

A pesquisa também considerou alguns resultados com algumas aplicações do SPEC-92 (Standard Performance Evaluation Corp.).

O SPEC-92 não contém benchmark comparáveis aos benchmark comerciais usados. Além disso, não considera muitas áreas técnicas e muitos dos programas possuem código e requisição de dados muito menor a aqueles encontrados em típicos ambientes de usuário.

Um estudo famoso (Gee-93), concluiu que apesar do SPEC-92 tem se tornado um padrão industrial para predição de desempenho (1994) a maioria dos seus programas individuais são muito bem comportados e colocam pouco estresse no subsistema de memória.

Portanto, muito cuidado na hora de fazer comparações !!

TABLE 1. The Workloads Traced

Name	Application Type	Description
Commercial Environment:		
Kenbus	Multi-User/Software Development (Research/Development Environment)	From the Standard Performance Evaluation Corp. (SPEC) System Development Multi-Tasking (SDM) benchmark suite
Laddis	NFS TM File Server	The basis of SPEC System-level File Server (SFS) benchmark suite; server was traced
Netperf	System Communications	TCP/IP benchmark for system communications performance; single-user trace
TPC-A TM	Database/Banking	Transaction Processing Performance Council-A (TPC-A) benchmark; users connected in client-server configurations and database-server was traced; [McGrory 92]
TPC-C TM	Database/Warehouse Order Entry	Transaction Processing Performance Council-C (TPC-C) benchmark; users connected in client-server configurations and database-server was traced
SDET	Multi-User/Software Development (Commercial Environment)	From the SPEC SDM benchmark suite

Aplicações Comerciais

Kenbus e SDET: SDMT (System Development Multi-Tasking). Pesquisa e Desenvolvimento e Comercial (SDET)

Laddis: File Server.

Netperf: Comunicações de Sistemas.

TPC-A: Transaction Processing Council-A. Configurações Cliente/Servidor e Servidor de Banco de Dados.

TPC-C: Idem a TPC-A.

Technical/Numeric-Intensive Environment:

ABAQUS	Structural Dynamics analysis	ABAQUS 'T3-4-9' timing run from HKS, Inc.
Les	Aircraft Surface Turbulence Simulation	Large Eddy, owned by NASA Ames.
Fpppp	Computational Chemistry Code	From the SPECfp92 benchmark suite; kernel was not traced (see note)
G92	Computational Chemistry Code	Gaussian 92 "Test50" timing run from Gaussian, Inc.; kernel was not traced (see note)
MCNP	Monte Carlo Simulation	Monte Carlo simulation application from Los Alamos Labs.
MM4	Local-Weather model	Owned by National Oceanic and Atmospheric Administration (NOAA)
BRAT TM	Architecture Simulator	Simulates superscalar architectures [Poursepanj 94]; kernel was not traced (see note)
Spice	Electrical Simulation	From the SPECfp92 benchmark suite; kernel was not traced (see note)

Aplicações Engenharia

ABAQUS: Análise Dinâmica Estrutural.

Simulação de Turbulências.

Código de Química Computacional.

Simulação Monte Carlo.

Modelo Local de Tempo.

Simulador Arquitetural (Ambientes superescalares).

Simulação Elétrica

TABLE 2. Comparison of Technical Traces with SPEC92

Type of Application	SPEC92	Traces
Computational Chemistry	fpgrp (Gaussian 88) Instr Miss Rate - 3.57 Data Miss Rate - 0.07 mdljsp2 Instr Miss Rate - 0.08 Data Miss Rate - 0.70 mdljdp2 Instr Miss Rate - 0.08 Data Miss Rate - 0.53	Gaussian 92 Instr Miss Rate - 0 Data Miss Rate - 18.62
Monte Carlo Simulation	doduc Instr Miss Rate - 1.03 Data Miss Rate - 0.43	MCNP Instr Miss Rate - 2.36 Data Miss Rate - 0.92
Weather Related	swm256 Instr Miss Rate - 0.05 Data Miss Rate - 2.29	MM4 Instr Miss Rate - 0.30 Data Miss Rate - 3.07
Fluid Dynamics	tomeatv Instr Miss Rate - 0.09 Data Miss Rate - 2.97	Les Instr Miss Rate - 0.45 Data Miss Rate - 3.96
Structural Dynamics	BTRIX (from nasa7) Instr Miss Rate - 0.10 Data Miss Rate - 2.10	ABAQUS Instr Miss Rate - 0.58 Data Miss Rate - 2.62
Electrical Simulation (Floating Point Intensive)	spice Instr Miss Rate - 0.26 Data Miss Rate - 6.25	N/A
Architectural Simulation (Integer Intensive)	N/A	BRAT Instr Miss Rate - 2.82 Data Miss Rate - 14.25
Logic Design (Integer Intensive)	espresso Instr Miss Rate - 0.35 Data Miss Rate - 0.24 eqntott Instr Miss Rate - 0.19 Data Miss Rate - 2.24	N/A
Maxwell's Equations Solver	wave5 Instr Miss Rate - 0.19 Data Miss Rate - 0.42	N/A

Notes:

Os traces comerciais consistem aproximadamente, 31 milhões de instruções executadas.
TPC-C: 182 milhões.
Técnicos: 21-80 milhões

Os traces capturaram detalhes do kernel, bibliotecas com partilhadas, e código nível usuário.

Opções de Compilação (-O) e caches de:
I: 8K, 2-way, 64 bytes
D: 64K, 4-way, 128 bytes.

Características: Comerciais vs Técnicas

- Número de Processos
- Porcentagem de Sistema Operacional
- Comportamento dos Branchs
- Chaveamento de Processos
- Porcentagem de Operações Ponto-Flotante
- Características de I/O

Aplicações diferentes podem estimular difentemente.

As apps. Comerciais compartilham algumas características, que as distinguem de Apps Técnicas.

Características: Comerciais vs Técnicas -2

● Número de Processos

Ambientes comerciais, geralmente, são compostos de um grande número de usuários concorrentes num sistema único.

Não é estranho ter centenas ou miles de usuários concorrendo ativamente.

Ao contrário, é muito comum encontrar ambientes técnicos com um único processo e usuário.

Características: Comerciais vs Técnicas -3

● Sistema Operacional

Tipicamente, códigos comerciais apresentam uma parte substancial do seu tempo de execução gasto no Sistema Operacional.

Ao contrário, apps. Técnicas, geralmente, contém pouca atividade do S.O.

Características: Comerciais vs Técnicas -4

- **Comportamento Branchs** As COM possuem menos iterações de loops. Além disso, as instruções de branchs encontradas são mais do tipo non-looping.

As TEC, comumente, consistem de working sets de poucas instruções com loops do tipo tight.

Características: Comerciais vs Técnicas -5

- **Chaveamento Processos** Nas COM a tendência é ter uma alta taxa de chaveamento de processos e altos níveis de multiprogramação, devido à frequência de esperas de I/O e sincronização de processos.

As TEC, frequentemente, executam até o “time-slice” definida pelo “system-end” (finais do sistema).

Características: Comerciais vs Técnicas -6

● Operações PF

As COM, usam relativamente, pouco as operações aritméticas de PF.

A manipulação dos dados, consiste principalmente, de comparações ou inserções de inteiros e strings.

A maioria, mas não todas, as TEC são dominadas pelo uso intensivo de operações de ponto flotante.

Características: Comerciais vs Técnicas -7

● I/O

As COM, possuem altas e randômicas taxas de I/O, função da grande atividade multi-usuário e da alta interação com o mundo real.

Os dados, tipicamente, são espalhados em vários Mega-bytes e/ou Giga-bytes e/ou Tera-bytes de Disco. As operações de I/O (disco) são curtas (4 - 8 Kbytes) e geralmente são distribuídas randomicamente no espaço total do disco usado pela aplicação.

As TEC que usam quantidades significativas de I/O tendem a ser algoritmos de uso reduzido de dados com padrão de acesso seqüencial, principalmente .

Características: Comerciais vs Técnicas -8

TABLE 3. Comparison of Workload Characteristics

Workload	% of Instrs Executed in OS	% Branches	% Branches Taken	Avg. # Instrs Between Taken Branches
Commercial Environment:				
TPC-A	40%	16.9%	67.1%	8.8
TPC-C	43%	18.9%	65.1%	8.1
Netperf	97%	18.6%	66.2%	8.1
Laddis	100%	18.9%	68.7%	7.7
Kenbus	23%	16.3%	64.9%	9.5
Sdet	50%	17.8%	66.5%	8.4

Technical/Numeric Intensive Environment:

ABAQUS	7%	23.9%	97.3%	4.3
G92	0%	39.6%	97.6%	2.6
Les	7%	5.2%	90.6%	21.1
Fpppp	0%	2.4%	62.1%	66.4
Mcnp	1%	14.9%	59.4%	11.3
Mm4	4%	8.5%	74.4%	15.8
BRAT	0%	32.1%	75.4%	4.1
Spice	0%	16.8%	39.7%	15.0

Note: The number of sequential instructions executed

Sumário: Comerciais vs Técnicas (1)

Os resultados anteriores mostram que a maioria do trabalho realizado em ambientes comerciais, (hoje) é altamente dependente do sistema operacional. Razões: Movimento freqüente de pequenas quantidades de dados nos diferentes níveis do sistema e a alta dependência com a comunicação (código) no S.O. para iniciar e receber comunicação (ex. Requisições e Respostas em DB).

Aplicações técnicas, diferentemente, são escritas de maneira que um dado trazido ao espaço da aplicação do S.O., manipulação aritmética extensiva é realizada nele antes de um próximo retorno ao S.O. para operações de armazenamento.

Assim, a maioria das Apps COM realizam I/O (dicos) usando S.O.

Sumário: Comerciais vs Técnicas (2)

Outra observação fascinante é a influência na execução do S.O. e da construção do programa nas estatísticas de branches.

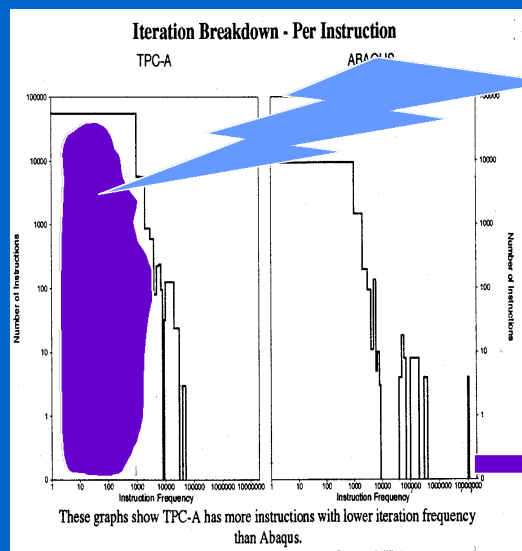
Workloads comerciais possuem características similares de branches.

Muitas aplicações típicas de engenharia e numéricas intensivas são dominadas por loops (laços), os quais têm uma conta muito alta de iteratividade (mais do que as aplicações comerciais).



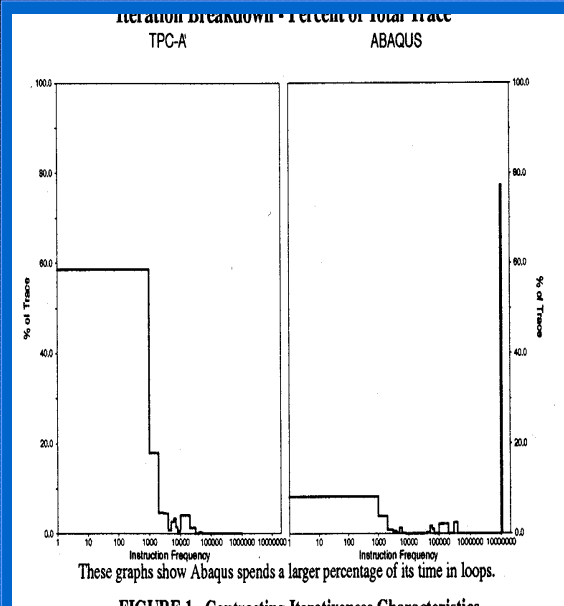
Ver figura seguinte !!

Alguns Resultados: Características de Iteração



TPC-A: Aprox. 55 mil instruções foram repetidas entre 1-1000 vezes Para 59% dos traces.

Abaqus: Quatro instruções executam aproximadamente 10 milhões de vezes, dominando os seus traces.



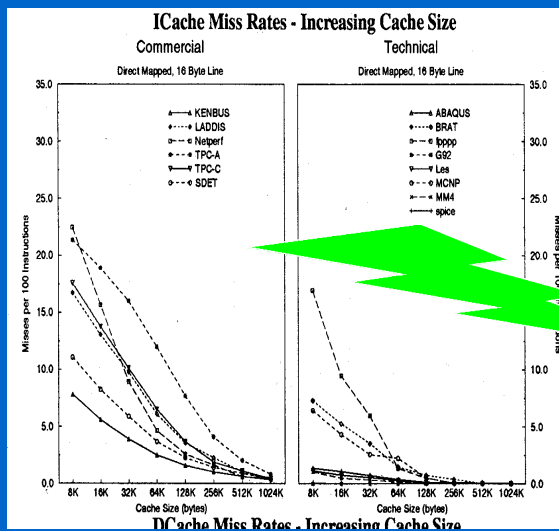
TPC-A

59% dos traces
(55 mil instruções)

Abaqus

80% dos traces
(4 instruções)

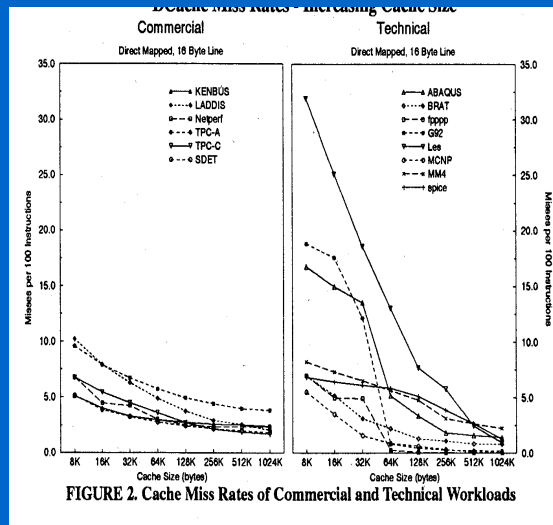
Desempenho de Cache (I): TEC vs COM



Caches I/D split:
8Kbytes - 1M
Linhas de 16-256 bytes
N-way: 1-8
Política LRU
Linha total (busca R/W)

Comportamento das
Instruções

Desempenho de Cache (D): TEC vs COM



COM experimentam altas taxas de misses de INST. Quando comparados aos misses de dados.

Situação contrária em TEC

Instruções (misses) maiores em COM que TEC

Análise do Cache: TEC vs COM

Aplicações comerciais experimentam uma taxa alta de instruções (quando comparados às Apps TEC) devido ao grande footprint de instruções e a atividade de chaveamento de processos.

Para caches pequenos (< 32 Kbytes) o footprint de instruções domina o desempenho do cache.

Na medida em que o tamanho do cache aumenta, o efeito dos múltiplos processos e do intervalo de chaveamento de processos torna-se um fator.

Grandes caches possibilitam a existência de processos no cache, quando um novo chamado ao processo aconteça.

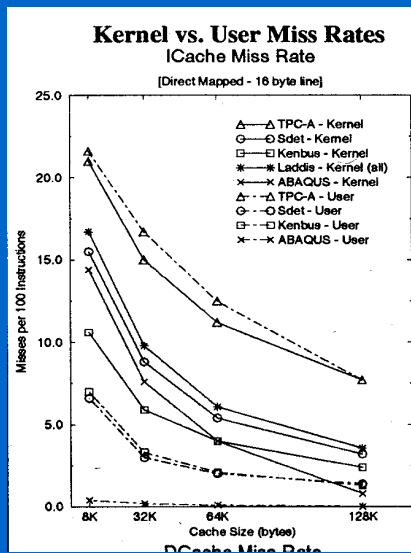
Análise do Cache: TEC vs COM (2)

Em Apps técnicas, a taxa de falhas no cache de instruções é determinada pelo (tipicamente) pequeno número de instruções necessárias para executar o trabalho.

Aplicações técnicas experimentam uma melhor re-utilização de instruções (instruction reuse) devido ao seu carácter de loops iterativos.

Um dado experimental, sugerido pelos autores, é que o footprints de dados são (aprox.) 10 vezes maiores que os seus footprints de instruções. Isso em Aplicações Comerciais.

Análise do Cache-I e S.O. (1)



A existência de um número grande de processos com grandes working sets locais produz um grande imbalance.

A maior parte do código é compartilhado entre diferentes processos mas os dados pertencem a um processo local (privado).

Ex. Servir uma transação a um usuário e depois não vê o usuário ou seus dados (novamente) por um longo período de tempo.

Análise do Cache-D e S.O. (2)

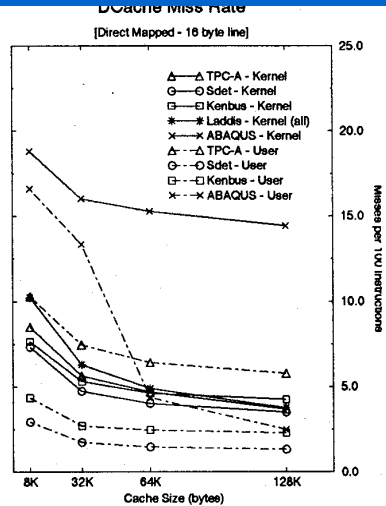


FIGURE 3. Miss Rate Breakdown

Taxa de Falhas de Kernel (linhas sólidas) vs Usuário (linhas traçadas).

Kernel >> (I/D) que Usuário

Pequenos caches, altos TDFs
Grandes caches, diminuição TDF

Análise do Cache e S.O. (3)

TABLE 4. Percentage of OS Activity (Reminder)

Workload	% of Instructions Executed in OS
ABRQUS	7%
Kenbus	23%
TPC-A	40%
Sdet	50%
Laddis	100%

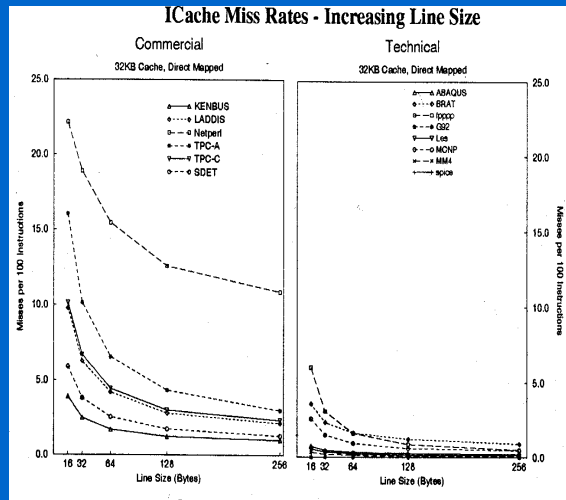
Diferentes Workloads possuem diferente atividade S.O.

A atividade de S.O. em COM é significativa.

Duas observações:

- Código do sistema gera alta taxa de falhas de cache e,
- Uma pequena fração do código do sistema não necessariamente significa uma taxa pequena de falhas.

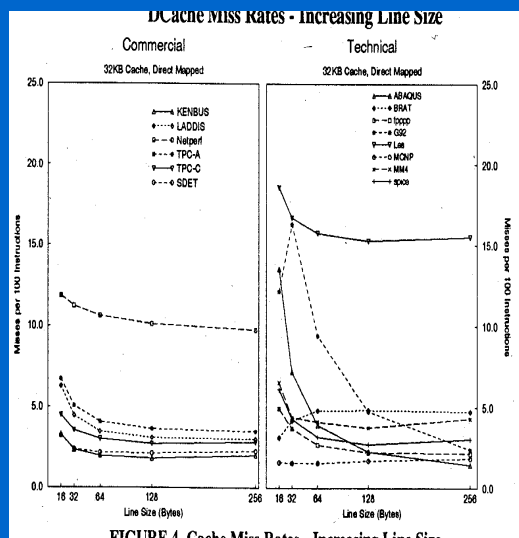
Tamanho de Linha Cache-I (1)



Maior impacto nas Apps COMERCIAIS

Apps COM usam EST-D mais árvores de decisão que realizar loops iterativos (“tight”)

Tamanho de Linha Cache-D (2)



A maioria das apps TEC computam através de arranjo de dados. Portanto, grandes linhas podem melhorar a TDF de dados trazendo adiantadamente o próximo conj. de dados.

Os TEC reação variada, e visualizam um padrão de referências muito dependente da forma de escrita (EST-D).

Tamanho de Linha Cache (3)

Exemplo:

A aplicação BRAT (simulador arquitetura), uso intensivo de inteiros que fazem uso extensivo de ponteiros.

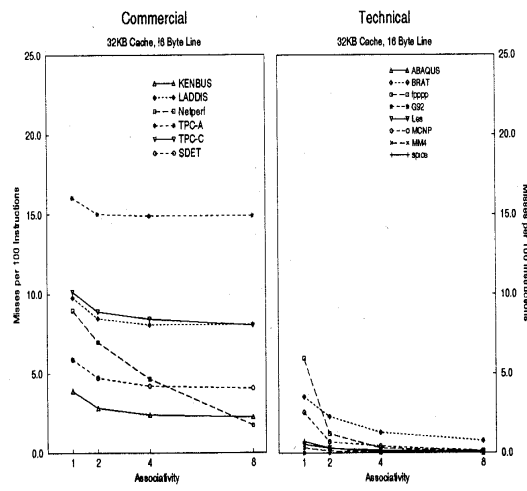
Assim, pequenas linhas (e muitas delas) são pre-referenciadas.

Aplicações TEC possuem comportamento diferente em relação aos benefícios relacionados com o tamanho de linha de cache.

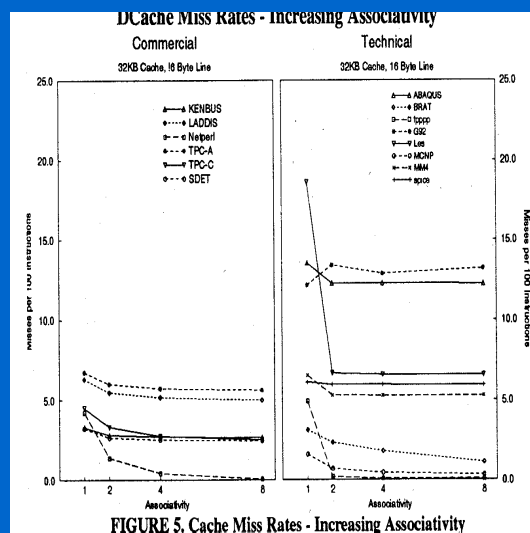
Grandes linhas em instruções podem obter grandes benefícios devido à natureza seqüencial dos acessos a instruções.

Associatividade do Cache-I (1)

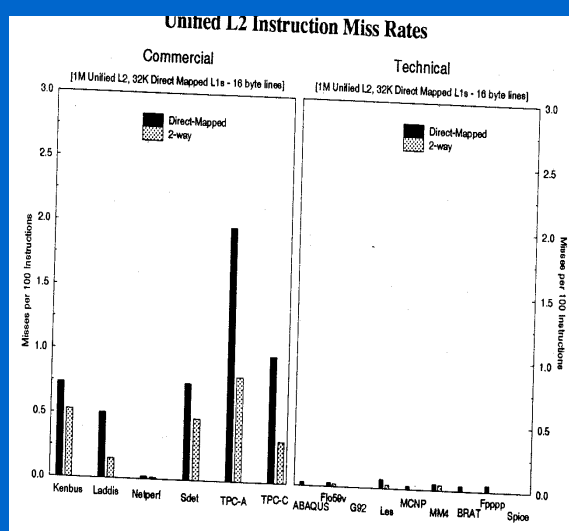
ICache Miss Rates - Increasing Associativity



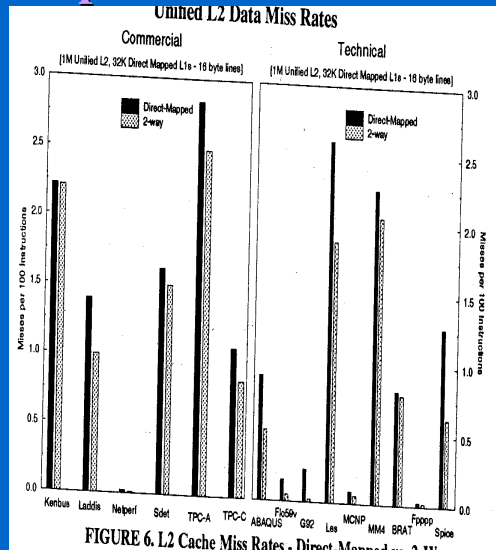
Associatividade do Cache-D (2)



Impacto do Cache-I L2 (1)



Impacto do Cache-D L2 (2)

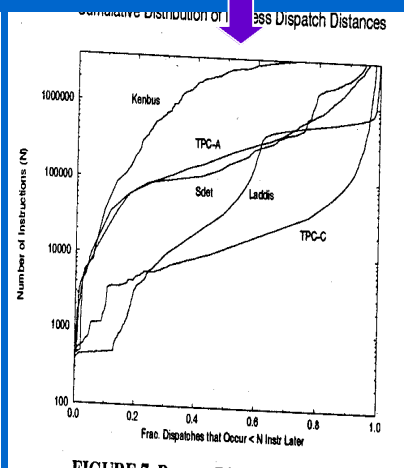
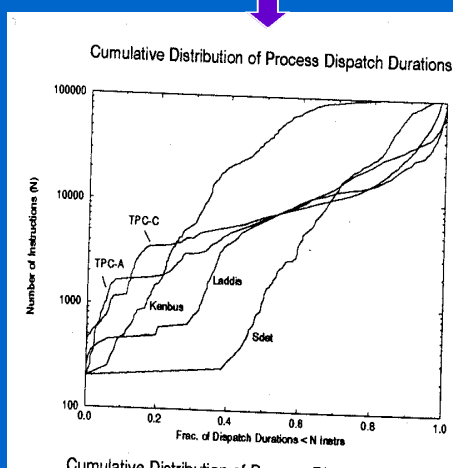


Efeitos do 'Dispatching' (1)

Duração

Distância

Chaveamento de Processos



Efeitos do 'Dispatching' (2)

TABLE 6. Commercial Dispatch Duration Statistics

Workload	Average # Instrs Per Dispatch	Median # Instrs Per Dispatch
Kenbus	49,341	37,692
Laddis	9,955	6,560
Sdet	17,348	1,206
TPC-A	12,745	6,971
TPC-C	14,465	7,178

Grandes Footprints de instruções são altamente sensíveis ao comportamento do dispatching.

Grandes caches L1 e L2 aumento da associatividade e tamanho de linhas pode tolerar esses efeitos negativos do chaveamento de processos.

**Próxima
Apresentação**