

COMPARAÇÃO DE DESEMPENHO DE PARADIGMAS PARA ACESSO
DISTRIBUÍDO A BANCOS DE DADOS EM REDES LOCAIS

CARLOS A. M. BASTOS^{1,2} DANIEL A. MENASCÉ¹ LUIS F. G. SOARES¹
malcher@inf.puc-rio.br menasce@inf.puc-rio.br lfgs@inf.puc-rio.br

¹Departamento de Informática
PUC-Rio
Rua Mqs de São Vicente 225
CEP 22453 Rio de Janeiro RJ

²Departamento de Telecomunicações
Universidade Federal Fluminense
Rua Passo da Pátria 156
CEP 24210 Niterói RJ

RESUMO

O acesso a bancos de dados através de redes locais pode ser efetuado segundo dois paradigmas: o paradigma *Servidor de Arquivos (SA)* e o paradigma *Servidor de Banco de Dados (SBD)*. No primeiro, cada estação de usuário possui uma cópia completa do sistema gerenciador de banco de dados (SGBD), e o acesso a arquivos, gerado pela execução de transações, é realizado pelo servidor de arquivos da rede. No segundo, cada estação possui apenas uma interface entre a aplicação e a rede, e o SGBD é executado em um servidor de banco de dados responsável pelo processamento de toda a transação. Este trabalho compara o desempenho relativo de ambos paradigmas quando processando a mesma carga. Combinações diferentes de rede, servidor, estação usuário, e parâmetros da carga são analisados. Conclui-se que um fator relevante para o desempenho relativo dos paradigmas é a relação entre a capacidade de processamento das estações e do servidor.

ABSTRACT

Distributed Database access in LANs may be done according to two different paradigms: the *File Server* and the *Data Base Server* ones. In the former, a copy of the Database Management System (DBMS) runs at each user station and the file access requests generated by the execution of a transaction are performed by a File Server which offers file operations to the user stations. In the latter paradigm the DBMS runs at a Data Base Server which receives, from the stations, transactions to be completely processed at the server. This paper analyses, performance-wise, the relative speedup of both paradigms when processing the same workload. Different combinations of network and server capacity are evaluated. The results show that the server to station CPU speed ratio is a relevant factor for the relative speedup of both paradigm.

1. INTRODUÇÃO

O acesso a bancos de dados através de redes locais pode ser efetuado segundo dois paradigmas distintos: o paradigma *Servidor de Arquivos (SA)* e o paradigma *Servidor de Banco de Dados (SBD)*.

No primeiro caso, cada estação de usuário possui uma cópia completa do sistema gerenciador de banco de dados (SGBD), ficando os registros armazenados no servidor de arquivos da rede, que oferece para as estações de usuário operações a nível de arquivo (abrir, ler, armazenar, etc). Os registros são trazidos à estação do usuário e após as operações necessárias devolvidos ao servidor.

No segundo, cada estação possui apenas uma interface entre a aplicação e a rede. O SGBD é executado em um servidor de banco de dados, responsável também pelo armazenamento dos arquivos. Os comandos das transações submetidas da estação são encaminhados ao servidor que efetua seu processamento, e envia a estação usuário o resultado da transação.

É um resultado conhecido que o paradigma SBD possui um melhor desempenho do que o SA quando os analisamos apenas do ponto de vista da rede de comunicação. No SA todos os registros a serem acessados devem ser encaminhados pela rede para processamento na estação usuário, o que não ocorre no caso do SBD. No entanto, não podemos estabelecer a priori qual dos paradigmas apresenta melhor desempenho global, considerando o tempo de resposta ou a taxa de processamento de transações, já que o SBD deverá gerar uma maior carga de processamento na CPU do servidor.

Modelos analíticos tem sido extensivamente utilizados para comparar diferentes arquiteturas de sistemas [14,17,26]. Heidelbergger [14] comparou a performance de uma máquina de banco de dados baseada em múltiplos microprocessadores com um sistema baseado em *mainframe*. Lazowska [17] estudou o desempenho de estações de trabalho fazendo acesso a arquivos remotos e utilizando redes locais em um ambiente de desenvolvimento de software. Ramakrishnam [26] avaliou duas alternativas para armazenamento de dados em ambientes distribuídos: o servidor de disco e o servidor de arquivos. Os modelos analíticos tem sido utilizados também para modelar SGBDs específicos [9,10] e para avaliar os efeitos de mecanismos de controle de concorrência [1,12,18] no tempo de resposta de transações.

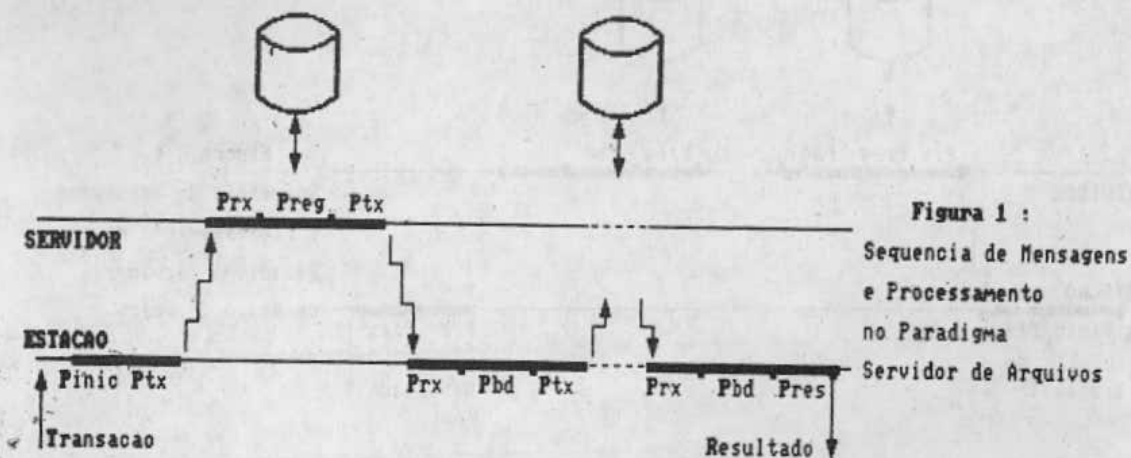
Este trabalho compara, através de modelos analíticos, o desempenho relativo de ambos paradigmas quando processando a mesma carga. Combinações diferentes de rede, servidor, estação usuário, e parâmetros da carga são analisados. Mostra-se que em alguns casos o SA possui melhor desempenho do que o SBD, e conclui-se que um fator relevante para o desempenho relativo dos paradigmas é a relação entre a capacidade de processamento das estações e do servidor, além do número de mensagens trafegando na rede.

A seção 2 discute ambos os paradigmas (SA e SBD) e descreve os modelos analíticos, assim como os parâmetros da carga e do sistema de computação utilizados nos resultados numéricos. Na seção 3 são apresentados os resultados e na seção 4 as conclusões.

2 O MODELO DE DESEMPENHO

2.1 HIPÓTESES DO SISTEMA

O sistema de computação é composto de estações de usuário (chamadas de estações) e uma estação servidora (chamada de servidor), interligadas por uma rede local de comunicação (LAN). O servidor é composto de CPU,



memória e disco. Existe, no servidor, uma área de memória reservada para "cache" de disco, de modo a evitar degradação de desempenho devido a acessos a disco [7,24].

Como mencionado, no paradigma SA, cada estação executa uma cópia do SGBD, que processa os comandos das transações submetidas localmente. Um pedido de serviço ao servidor de arquivos é gerado a cada operação de arquivo necessária para a execução de um comando de uma transação. A figura 1 apresenta, através de um eixo duplo de tempo, a seqüência de mensagens trocadas e de processamento executado entre uma dada estação e o servidor durante o processamento de comandos de uma transação. Cada seqüência de processamento é identificada pela letra P com um subscrito para indicar o tipo de processamento. Os subscritos possíveis são *inic*, *tx*, *rx*, *reg*, *bd*, *res* para indicar processamento inicial, processamento para transmissão de mensagem pela rede, recepção de mensagem pela rede, processamento para acesso a registro, operações sobre registros do banco de dados e processamento do resultado final respectivamente.

No paradigma SBD, o servidor atua como um servidor de banco de dados oferecendo operações a nível de banco de dados. Cada estação executa uma interface que é responsável por receber uma transação, executar um pré-processamento e transmitir a transação ao servidor. A figura 2 apresenta a seqüência de processamento e troca de mensagens entre o servidor e uma dada estação durante a execução de um comando. Os subscritos são idênticos ao do SA.

Algumas considerações qualitativas a respeito de ambos os paradigmas são relevantes :

1. O número de acessos a disco é idêntico em ambos os paradigmas: Os únicos acessos realizados são aqueles para armazenar ou obter registros que não estejam na memória cache.
2. O número de mensagens que trafegam na rede por transação é diferente para cada paradigma. O tamanho médio também é diferente. No paradigma SA depende do tamanho médio dos registros e da mensagem de pedido de registros. Já no SBD, depende do tamanho médio da mensagem de envio da transação submetida pelo usuário e do tamanho médio da resposta à essa transação. Considerou-se que o *overhead* de processamento devido a transmissão e recepção de mensagens é idêntico ($P_{tx} = P_{rx}$ nas figuras 1 e 2).
3. O processamento de operações sobre o banco de dados é idêntico em ambas as arquiteturas, exceto pelo local em que é realizado: estação

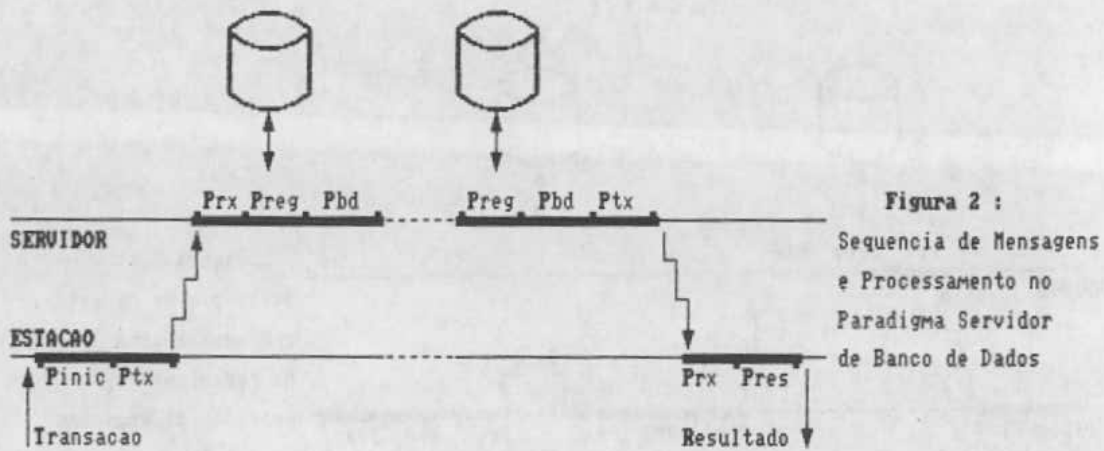


Figura 2 :
Sequencia de Mensagens
e Processamento no
Paradigma Servidor
de Banco de Dados

4. O pré-processamento de comandos, interface com usuário e processamento de resultados é idêntico em ambos os paradigmas, sendo sempre realizado pela estação (Pinic e Pres nas figuras 1 e 2).
5. O processamento executado para obter um registro é idêntico em ambas as arquiteturas. Realizado sempre pelo servidor, que controla as operações de acesso a disco (Preg nas figuras 1 e 2).

Os parâmetros que caracterizam o modelo são então definidos:

Parâmetros da aplicação:

- N - número médio de estações ativas
- Z - Tempo médio de pensar, em Seg
- Treg - Tamanho médio de um registro em bytes
- Nar - Número médio de acessos a registros por transação
- Ibd - Número médio de instruções executadas processando operações do SGBD em cada acesso a registro
- Iint - Número médio de instruções executadas devido a interface com o usuário
- Ttrs - Tamanho médio em bytes de uma transação submetida ao banco de dados
- Trsl - Tamanho médio em bytes da resposta a uma transação
- Tpar - Tamanho médio em bytes de um pedido de acesso a registro remoto trafegando pela rede de comunicação

Parâmetros do sistema de computação :

- csev - capacidade de processamento do servidor em Mips
- cest - capacidade de processamento das estações em Mips
- td - tempo médio de serviço do disco em ms/registo
- Irede - número médio de instruções executadas pelo sistema operacional para transmissão ou recepção de mensagem
- Iar - número médio de instruções executadas pelo sistema operacional por acesso a um registro
- crede - Taxa de transmissão de dados da rede em Mbps
- Δrede - retardo de transmissão de mensagem pela rede em ms, incluído espera em fila.

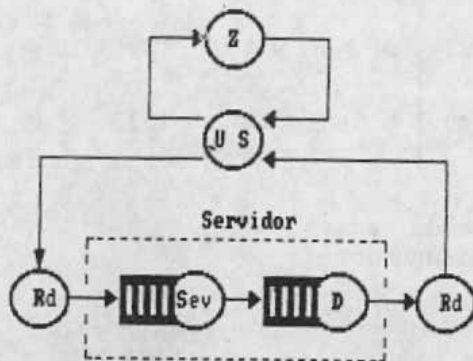


Figura 3A: Rede de Filas para o Paradigma SA

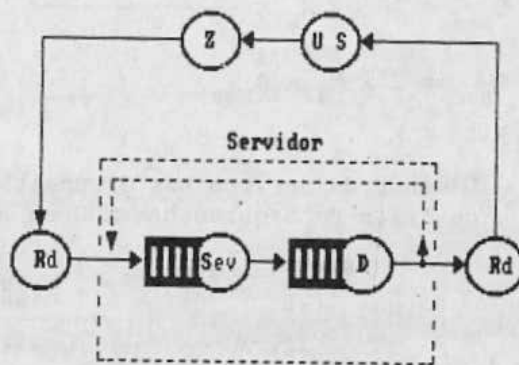


Figura 3B: Rede de Filas para o Paradigma SBD

2.2 DESCRIÇÃO DO MODELO PARA AMBOS OS PARADIGMAS

O modelo, utilizado na análise dos paradigmas, é em rede de filas fechada, conforme apresentado na figura 3. Como a cada estação corresponde apenas um usuário e, por hipótese, cada usuário submete apenas uma transação de cada vez, esperando sua resposta para submeter uma outra, não se forma fila na CPU das estações e podemos modelá-las como um servidor de retardo (US) para todos os usuários. O servidor Z representa o tempo de pensar de cada usuário. O disco (D) e a CPU do servidor (Sev) são modelados por servidores com fila, já que são recursos compartilhados.

Os servidores (Rd) na figura representam o retardo da rede de comunicação. O retardo depende do tráfego [5], que por sua vez depende da taxa de processamento de transações no sistema e do número de acessos a registro efetuado por cada transação. Devido a essa interdependência, a rede é representada por servidores de retardo cuja capacidade depende do tráfego, e resolvida utilizando uma técnica conhecida como aproximação hierárquica [16].

2.3 CÁLCULO DA DEMANDA DE SERVIÇO EM CADA DISPOSITIVO

Para a solução do modelo em rede de filas fechada, é necessário calcular a chamada *demanda de serviço*, que é o tempo médio total de serviço dispendido por um usuário em um centro de serviço (dispositivo da rede de filas) [16].

Com os parâmetros do sistema e da aplicação definidos em 2.1, e reportando-se às figuras 1 e 2, podem ser calculadas as demandas de serviço para os dispositivos da rede em cada paradigma.

Demanda de serviço nos dispositivos da rede de filas na arquitetura SA em milissegundos

$$D_{US} = 10^3 \times \frac{I_{int}}{c_{est}} + 10^3 \times N_{ar} \times \frac{I_{bd} + 2 \times I_{rede}}{c_{est}} \quad (1)$$

$$D_{sev} = 10^3 \times N_{ar} \times \frac{I_{ar} + 2 \times I_{rede}}{c_{sev}} \quad (2) \quad \left| \quad D_d = N_{ar} \times t_d \quad (3) \right.$$

$$D_{Rd} = 2 \times N_{ar} \times \Delta_{rede} \quad (4) \quad \left| \quad D_z = Z \times 10^3 \quad (5) \right.$$

Demanda de serviço nos dispositivos da rede de filas na arquitetura SBD em milissegundos

$$D_{US} = 10^3 \times \frac{I_{int}}{c_{est}} + 10^3 \times \frac{2 \times I_{rede}}{c_{est}} \quad (6)$$

$$D_{sev} = 10^3 \times \frac{N_{ar} \times (I_{bd} + I_{ar}) + 2 \times I_{rede}}{c_{sev}} \quad (7) \quad \left| \quad D_d = N_{ar} \times t_d \quad (8) \right.$$

$$D_{Rd} = 2 \times \Delta_{rede} \quad (9) \quad \left| \quad D_z = Z \times 10^3 \quad (10) \right.$$

2.4 PARAMETRIZAÇÃO DO SISTEMA DE COMPUTAÇÃO

As CPUs consideradas possuem capacidades entre 1 e 10 Mips, correspondendo a computadores pessoais ou estações de trabalho baseados em processadores das famílias 68000 e 286, que cobrem a faixa de 1 (286/8Mhz) a 5 Mips (386/25Mhz); ou baseados em processadores RISC que atingem facilmente 10 Mips [25,28,27,21,13].

As redes consideradas foram as do tipo anel com passagem de permissão a 4 e 20 Mbps [15]. Essas taxas de transmissão correspondem a limites comumente encontrados na prática.

Conforme [11,9,14,17,19,20,29], o custo de processamento para transmitir uma mensagem pela rede se situa entre 744 e 10000 instruções. Considerou-se $I_{rede} = 2000$ instruções como o custo por mensagem de uma boa interface de rede. Variou-se I_{rede} entre 500 e 4000 instruções para estudar a sensibilidade do SA com relação a esse parâmetro.

Heidelberger [14] utilizou 20 ms por acesso a disco e supôs um cache hit de 0.33 para acessos a páginas de índices ou dados, o que dá uma média de 13.3 ms por acesso. Nos sistemas avaliados por ele pode-se sempre acrescentar um número suficiente de discos, de modo a manter o tempo gasto em filas sempre pequeno quando comparado com a demanda de serviço. Já Lazowska [17], para o V kernel [8] utilizando blocos de 8 Kbytes, obteve 30 ms como tempo de acesso a disco (média de 1.88 ms por 512 bytes). Para o sistema SUN utilizando blocos de 4 Kbytes obteve 18.8 ms (2.35 ms por 512 bytes). Suas simulações e outros estudos [2,22,23,7] sugerem tempos de serviço ainda menores com o uso de técnicas específicas. Devido às considerações acima, e ao tamanho do registro

utilizado (512 bytes, ver próxima seção), foram escolhidos três valores para t_d : 1.2 ms, 2.5 ms e 15 ms.

Como custo de processamento para acesso a um registro de 512 bytes foi utilizado $I_{ar} = 2500$ instruções, valor esse consistente com as referências [17,26,12,10,1].

2.5 PARAMETRIZAÇÃO DA CARGA

Dos dados de Heidelberg [14] pode-se calcular um valor de 13462 instruções executadas pelo SGBD por acesso a registro para a transação débito-crédito [3]. Utilizou-se este valor como um caso base para I_{bd} e variou-se esse parâmetro de 2000 a 20000 instruções, consistentemente com outros estudos [17,26,12,10,1].

Heidelberg usou 100000 instruções para um parâmetro equivalente a I_{int} , porém incluindo custos de comunicação. Considerou-se $I_{int} = 50000$.

Heidelberg considerou também que uma transação débito-crédito executa um total de 5 comandos SQL [6] e 13 operações de E/S.

Anon et al [3] utilizou a transação débito-crédito para classificar SGBDs em uso comercial. Definiu como um bom sistema gerenciador de banco de dados aquele que executa 100000 instruções e 10 operações de E/S por transação débito-crédito, e que é capaz de processar 50 transações por segundo. Classificou como sistemas comumente encontrados na prática, aqueles que executam 300000 instruções e 20 operações de E/S, e que são capazes de processar 20 transações por segundo. Assumindo os parâmetros deste trabalho com os valores base e 13 acessos a registro por transação, obtêm-se 309000 instruções por transação, o que representa um sistema entre aqueles considerados bons e aqueles considerados comuns.

Finalmente, os valores utilizados para T_{trs} , T_{par} , T_{rsl} , Tempo de pensar e T_{reg} foram 128 bytes, 128 bytes, 4 kbytes, 20 segundos e 512 bytes respectivamente, também consistente com os estudos mencionados. A tabela 1 resume valores utilizados para parâmetros do sistema e aplicação

Parâmetros da Carga		Parâmetros do Sistema	
M	25 estações	c_{sev}	5 e 7 MIPS
Z	20 segundos	c_{est}	1 e 5 MIPS
N_{com}	5 p/ trans db_cr	t_d	1.2, 2.5 e 15 ms
T_{reg}	512 bytes	I_{rede}	500, 2K e 4K
N_{reg}	13/5 p/trans db_cr	I_{ar}	2500
I_{bd}	13462, 2K e 20K	c_{rede}	4 e 20 Mbps
I_{int}	50000	T_{rsl}	4K bytes
T_{com}	128 bytes	T_{par}	128 bytes

Tabela 1 : Resumo dos parâmetros do sistema e da carga

3 ANÁLISE DOS RESULTADOS NUMÉRICOS

Na análise realizada foi estudado o desempenho relativo de ambos os paradigmas através de uma métrica, chamada fator de aceleração (equação 11), definida como a razão entre os tempos de resposta de uma transação para cada arquitetura. O fator de aceleração fornece uma medida do desempenho relativo entre elas. Foi feita uma análise desse fator em função de parâmetros da aplicação e do sistema de computação, definidos

conforme as equações 1 a 10. Foi também analisada uma transação padrão, a transação débito-crédito.

$$F_a = \text{fator de aceleração} = T_{sa}/T_{sbd} \quad (11)$$

3.1 ANÁLISE DO FATOR DE ACELERAÇÃO

As figuras 4 e 5 apresentam o comportamento de F_a em função do número de acessos a registro por transação N_{ar} . Todas as curvas utilizam os valores base para os parâmetros, exceto para aqueles parâmetros indicados na própria figura. Na figura 4 a CPU do servidor é mais rápida do que a da estação. Em todas as curvas desta figura, exceto a curva 5, a razão $c_{sev}/c_{est} = 5$. Na curva 5, é igual a 7. A figura 5 considera o caso de um sistema homogêneo ($c_{sev} = c_{est}$). Para os objetivos desta análise dividiu-se as transações em três tipos:

Transações Triviais : aquelas que acessam menos de dez registros

Transações Médias: as que acessam um número moderado de registros (de dez a cem registros). A transação débito-crédito pertence a esta categoria.

Transações Complexas: aquelas que executam um grande número de acessos a registro (mais do que cem). Exemplos de transações desta categoria são aquelas que manipulam objetos complexos em aplicações de CAD/CAM.

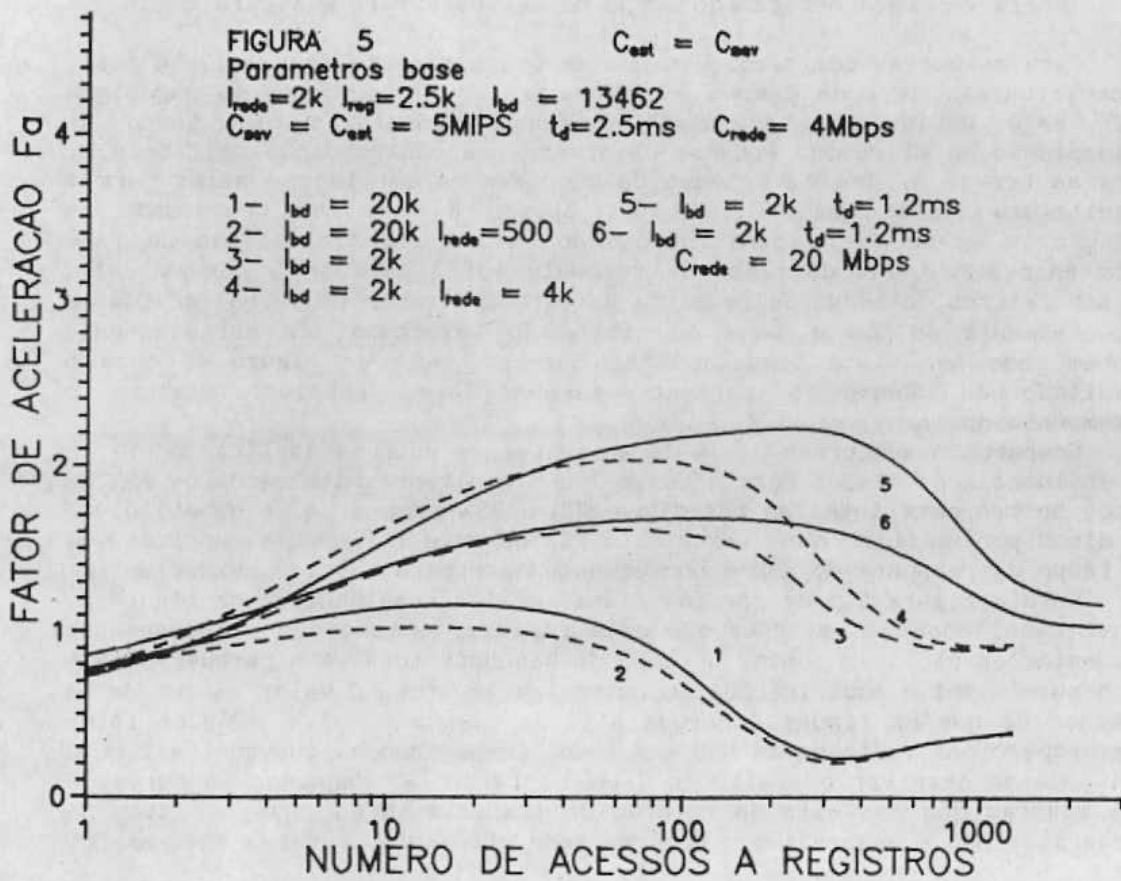
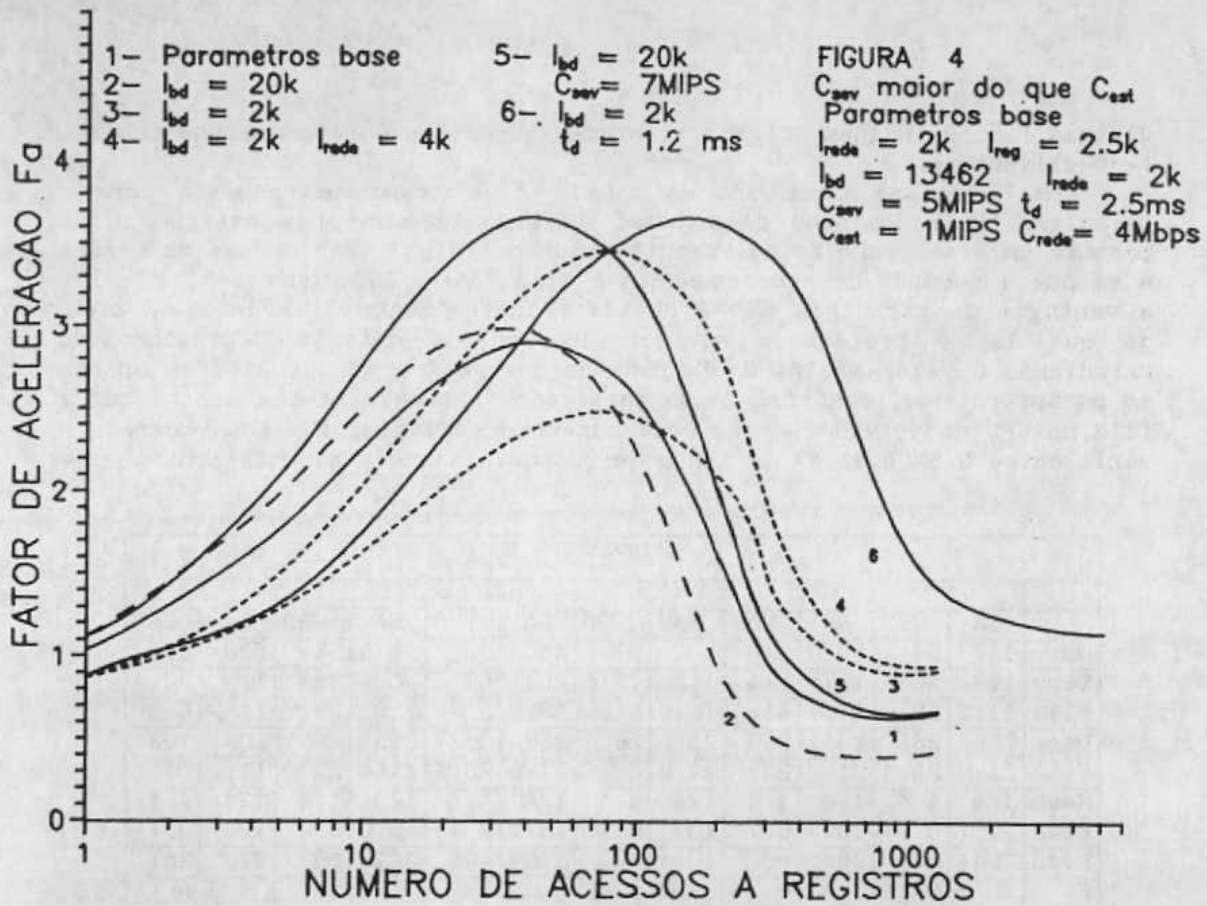
O comportamento de todas as curvas é similar. O fator F_a é próximo de um para transações triviais porque para esse tipo de transação o fator dominante no tempo de resposta dos paradigmas é o parâmetro I_{int} , e ele afeta igualmente a ambos. O valor exato de F_a no caso de transações triviais depende dos seguintes fatores:

1. Relação entre a capacidade do servidor e da estação, que altera o tempo de processamento de $N_{ar} \times I_{bd}$ em cada paradigma
2. Relação entre I_{bd} e I_{int} , que pode tornar desprezível o primeiro fator caso $I_{int} \gg (N_{ar} \times I_{bd})$
3. Relação entre T_{rs1} e T_{reg} . O primeiro afeta o tamanho médio da mensagem que trafega na rede no paradigma SBD enquanto o segundo afeta o tamanho médio no SA.

À medida que o número de acessos cresce, passando de transações triviais para transações médias, F_a cresce porque o número de mensagens que trafegam na rede, consumindo tempo em transmissão e processamento de mensagens, é proporcional ao número de acessos a registros no SA e constante no SBD; e porque quando $c_{sev} > c_{est}$ o processamento de instruções relativas a $N_{ar} \times I_{bd}$ é efetuado mais rapidamente no paradigma SBD. A influência de I_{int} se reduz quando comparada a dos parâmetros que dependem de N_{ar} .

A vantagem do paradigma SBD começa a decrescer quando a carga aumenta, aumentando as filas no disco e no servidor. Para transações complexas, F_a decresce ainda mais e em alguns casos chega a ser menor do que um.

A tabela 2 apresenta a decomposição do tempo de resposta em cada dispositivo do sistema para ambos os paradigmas e para cada curva da figura 4, quando $N_{ar} = 80$ (transação média). Tempos de resposta em cada dispositivo são dados em milissegundos e incluem tempo de serviço e o tempo de fila, que é dado como o percentual do tempo de resposta do dispositivo para aqueles modelados por servidores com fila. As duas



últimas linhas indicam o tempo total de resposta e o fator de aceleração, respectivamente.

Como pode ser observado da tabela 2, o tempo de resposta para o paradigma SA é dominado pelo tempo de processamento nas estações, que possuem um desempenho relativamente pequeno (1Mips). Nas curvas da figura 4 em que a demanda de processamento é alta ($I_{bd} > 10K$, curvas 1, 2 e 5), a vantagem do paradigma SBD é devida principalmente à razão C_{sev}/C_{est} , já que I_{bd} é processado no servidor ou na estação dependendo do paradigma. O valor máximo de F_a para as curvas 2 e 5, que diferem apenas no parâmetro C_{sev} , confirmam esse resultado. A tabela também mostra que a fila na CPU do servidor, para este número de acessos, é ainda moderada e varia entre 8.5% e 41.4% do tempo de resposta naquele dispositivo.

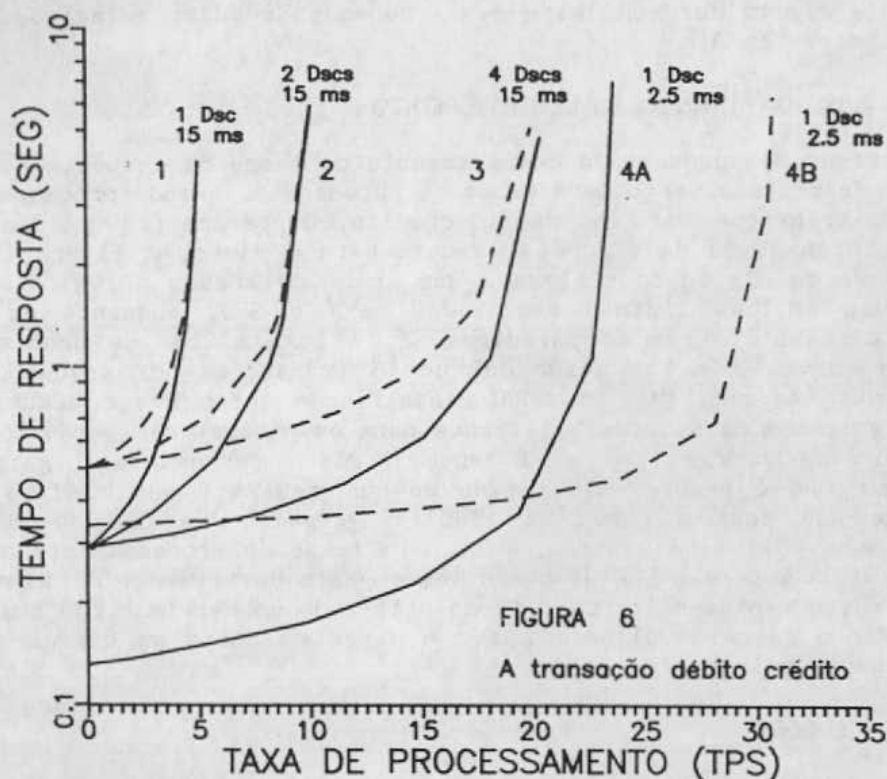
Curvas												
	1		2		3		4		5		6	
	SA	SBD	SA	SBD	SA	SBD	SA	SBD	SA	SBD	SA	SBD
US	1447	54	1970	54	530	54	850	58	1970	54	530	54
Serv	117	365	117	616	118	79.6	207	80.6	80.7	368	118	80
Fila	11.5	29.6	11.1	41.4	11.9	8.5	18.8	8.7	7.9	29.9	11.9	8.6
Dsc	256	261	254	260	259	262	258	262	255	260	108	108
Fila	21.9	23.4	21.2	23	22.8	22.5	21.6	23.6	21.6	23	11.1	11.1
Rede	124	4.4	124	4.4	124	4.4	124	4.4	123	4.4	124	4.4
Fila	13.7	0.7	13.7	0.7	13.7	0.7	13.7	0.7	13.0	0.7	13.7	0.7
Resp	1944	684	2440	932	1031	400	1438	405	2428	687	880	247
F_a	2.8		2.6		2.6		3.6		3.5		3.6	

Tabela 2 : Decomposição do Tempo de Resposta para a Figura 4

Para as curvas com menor demanda de processamento (curvas 3, 4 e 6), a contribuição da rede também é relevante para a vantagem do paradigma SBD, seja devido ao "overhead" de processamento ou ao tempo de transmissão em si mesmo. Pode-se observar essa contribuição na tabela 2: para as curvas 3, 4 e 6, o tempo de resposta no servidor é maior para a arquitetura SA do que para o SBD, apesar de I_{bd} ser processado na estação, e vê-se que a contribuição do retardo de transmissão da rede está entre 8% e 14% do tempo de resposta total para as 3 curvas. Além desses fatores, o tempo de resposta da estação também contribui devido ao processamento de I_{bd} e I_{rede} . O efeito do "overhead" de processamento também pode ser visto comparando as curvas 3 e 4 da figura 4, onde o resultado de dobrar o parâmetro I_{rede} piorou consideravelmente o desempenho do SA.

Comparando as curvas 3 e 6 da mesma figura pode-se analisar o efeito da influência do disco. Para a curva 3 o disco representa cerca de 65% do tempo de resposta total no paradigma SBD e 25% para o SA. A substituição do disco por um outro duas vezes mais rápido teve influência considerável no tempo de resposta do SBD e conseqüentemente para o valor máximo de F_a .

Para a figura 5 pode ser feita uma análise semelhante a da figura 4. Nesse caso, como as estações são mais rápidas, o tempo de processamento nas estações não mais domina o tempo de resposta total e o paradigma SA é mais suscetível a modificações em outros parâmetros. O valor máximo de F_a é menor do que na figura 4 porque a razão $C_{sev}/C_{est} = 1$ e o único fator que proporciona vantagem ao SBD é a rede. Comparando as curvas 1 e 2 ou 3 e 4 pode-se observar o efeito de I_{rede} no fator F_a , enquanto as curvas 5 e 6 mostram que o efeito do retardo de transmissão da rede no fator de aceleração não é desprezível. Com uma rede eficiente, e carga alta na CPU



do servidor (curvas 1 e 2), provocando tempo gasto em filas, o SA terá um desempenho superior ($F_a < 1$) ao SBD.

Comparando as curvas 1-fig.5 e 2-5-fig.4 ou então 3-fig.5 e 3-fig.4, que possuem valores idênticos para os parâmetros de carga, verificamos que, quando o servidor possui capacidade maior do que as estações (sistema heterogêneo), a variação do desempenho de um paradigma para o outro é considerável, o mesmo não ocorrendo para o caso homogêneo. Isso mostra que não é suficiente escolher um SGBD baseado no SBD, de modo a reduzir o tráfego na rede e melhorar o desempenho. É desejável também utilizar um servidor de maior capacidade do que as estações.

Quando o número de acessos a registros aumenta, a fila na CPU do servidor aumenta, assim como a fila nos outros dispositivos compartilhados. Por exemplo, para as curvas da figura 4 com $N_{ar} = 320$, o tempo gasto na fila da CPU do servidor varia de 28% a 91% do tempo total gasto nesse dispositivo. Então, para transações complexas, o fator de aceleração depende dos dispositivos compartilhados. As seguintes observações são relevantes:

1. A rede afeta negativamente o desempenho do paradigma SA
2. A demanda de serviço no disco é a mesma em ambos os casos
3. O paradigma que possui maior carga de processamento no servidor depende da razão ($2 \times I_{rede} / I_{bd}$).

Para transações complexas, o valor limite de F_a quando N_{ar} tende ao infinito depende do dispositivo de maior demanda de serviço. Se o gargalo é a rede, F_a tende para um valor maior do que 1 (observação 1). Se o

gargalo é o disco, então F_a tende para 1 (observação 2). Finalmente se o gargalo é o servidor, o limite de F_a depende do valor relativo de I_{bd} e I_{rede} (observação 3).

3.2 ANÁLISE DA TRANSAÇÃO DÉBITO CRÉDITO

A figura 6 apresenta o comportamento do tempo de resposta em função da taxa de processamento para ambos os paradigmas, quando processando uma carga de transações do tipo débito-crédito. Os parâmetros utilizados são os do sistema base da figura 4, exceto para o disco. A figura indica a quantidade de discos do sistema e seu tempo de acesso a registro (t_d). As curvas sólidas indicam resultados para o SBD, enquanto as curvas pontilhadas se referem ao paradigma SA. A taxa máxima de processamento para as curvas 3 e 4 é maior do que 15 transações por segundo, o que representa, segundo [3], um SGBD classificado entre bom e comum e está consistente com os valores utilizados para os parâmetros (seção 2.5).

Para as curvas 1, 2 e 3, onde o disco representa o gargalo, o paradigma SBD é sempre mais rápido do que o SA até que o disco esteja saturado. Já para a curva 4A (SBD) o gargalo é a CPU do servidor, enquanto na 4B (SA) é o disco. Daí, para taxas de processamento acima de 20 Tps, o SA supera o SBD já que o tempo gasto em fila na CPU do servidor cresce muito rapidamente. Caso fosse utilizado um servidor com uma CPU de 6.5 MIPS, o gargalo voltaria a ser o disco em ambas as curvas e o SBD seria sempre mais rápido do que o SA.

4 CONCLUSÕES

Os resultados mostram que, para transações triviais e médias, em sistemas heterogêneos onde a CPU do servidor seja mais rápida do que a das estações, o paradigma SBD supera o SA. Mesmo para transações complexas, o paradigma SBD pode ter um desempenho superior ao SA, desde que seja feita uma escolha adequada dos componentes do sistema. A análise foi feita para um sistema com 25 estações ativas, com todas elas realizando transações do mesmo tipo. Isso, no caso de transações complexas, representa uma extrema carga para o servidor. Caso o sistema tenha poucos usuários efetuando transações complexas, o SBD poderá superar o SA para um número bem maior de acessos a registros. Esse é o caso, por exemplo, de transações de atualização global em um banco de dados, ou procedimentos de manutenção.

O tempo de serviço do disco é um fator importante para o desempenho do sistema, e o uso de múltiplos discos e/ou altas taxas de hit é fundamental para obter um bom desempenho, caso se deseje acessar mais do que poucas dezenas de registros.

O paradigma SBD reduz a carga na rede de comunicação mas em contrapartida aumenta a carga no servidor. Para se explorar totalmente o seu potencial é necessário utilizar um servidor de alto desempenho, de modo a processar mais rapidamente as operações sobre o banco de dados e manter a fila na CPU pequena.

Para obtermos, com a arquitetura SA, um desempenho semelhante ao da arquitetura SBD, precisamos lançar mão de um sistema de alto custo, com uma rede de alta velocidade e estações de grande desempenho. Por outro lado, o desempenho da arquitetura SBD pode ser otimizado simplesmente com o uso de um servidor de maior capacidade de processamento.

Os resultados mostram também que a arquitetura SA é fortemente influenciada pelo "overhead" de processamento devido ao tráfego da rede (I_{rede}). Para otimizar o desempenho da arquitetura SA, não basta utilizar

uma rede com alta taxa de transmissão. É necessário que o impacto gerado pela comunicação na CPU das estações seja pequeno ($I_{rede} < 2000$).

Os resultados obtidos no processamento de uma transação típica, a transação débito-crédito, mostram vantagem do servidor de banco de dados sobre o servidor de arquivos até um valor limite da taxa de processamento de transações. Quando for necessário processar um número de transações maior do que esse limite, pode ser vantajoso utilizar a arquitetura SA, ou então é necessário aumentar a capacidade do servidor.

5 AGRADECIMENTOS

Ao professor A. R. Sapienza e a FAPERJ pelo uso de recursos computacionais que viabilizaram a execução deste trabalho, que também foi suportado pela Secretaria de Ciência e Tecnologia.

6 REFERÊNCIAS

[1] Agrawal R, De Witt D. J. Integrated concurrency control and recovery mechanisms : Design and performance evaluation. *ACM Transactions on database systems*. Vol 10 no 4 dez. 1985.

[2] Andrews, W. Intelligent caching controllers drop effective disk access times. *Computer Design*, Vol 28 no 17 set 1989.

[3] Anon et al. A measure of transaction processing power. *Datamation*, abril 1985.

[4] Braunstain, A., Rilley M., Wilkes, J., Improving the efficiency of UNIX file buffer caches. *Proceedings of the twelfth ACM symposium on operating systems principles*, Dezembro 1989.

[5] Bux, W. Local-Area subnetworks: A performance comparison. *IEEE Trans. on Comm.* Vol. Com-29, No 10, Outubro 1981.

[6] Chamberlin, D.D., Boyce, R.F. *SEQUEL: A Structured English Query Language*. *Proceedings of the 1974 ACM SIGMOD Workshop on Data Description, Access and Control*.

[7] Chaney, R, Johnson, B Maximizing Hard-Disk Performance. *Byte* Maio 84.

[8] Cheriton, D.R. The V kernel: A Software Base for Distributed Systems. *IEEE Software*, 1,2, abril 1984.

[9] Copeland G et all. *ACM Sigmod Conference*, Chicago 1988.

[10] Demurjian, S.A., Hsiao, D.K. Benchmarking data base systems in multiple backend configurations. *Database Engineering*, Vol 8 no 1 março 1985.

[11] Duchamp, D. Analysis of transaction management performance. *Proceedings of the twelfth ACM symposium on operating systems principles*, Dezembro 1989.

[12] Eich, M.H., Wells, D.L. Database concurrency control using data flow graphs. *ACM Transactions on database systems*, Vol 13 no 2 junho 1988.

[13] Grehan, R. Introd the new BYTE benchmarks. *Byte*, Jun. 1988

[14] Heidelberger, P., Lakshmi, M. S. A performance comparison of multimicro and mainframe database architectures. *IEEE Trans. on Soft Eng.*, Vol 14 no 4 abril 1988.

[15] The Institute of Electrical and Electronics Engineers, Token Ring Access Methods and Physical Layer Specifications. ANSI/IEEE Std. 802.5 1985.

[16] Lazowska, E. D. et all. Quantitative System Performance. Computer System Analysis Using Queuing Network Models. Prentice-Hall, Inc 1984.

[17] Lazowska E. et all. File access performance of diskless workstations. *ACM Trans. on Comput. Syst.*, Vol 4 no 3 Agosto 1986.

[18] Menascé, D.A., Nakanishi, T. Performance Evaluation of Two Phase Commit Based Protocol for DDBs. *Proceedings of the First ACM SIGACT-SIGMOD Symposium on Princ. of Database Systems*, L. A. 1982

[19] Mitchel, J. G., Dion, J. A comparison of two network-based file servers. *Communications of ACM*, Vol 25 no 4 abril 1982.

[20] Mogul, J. C. et all. The packet filter: an efficient mechanism for user-level network code. *Proceedings of the eleventh ACM symposium on operating systems principles*, Novembro 1987.

[21] Nicholls, B. The current crop. *Byte*, Fevereiro 1989.

[22] Ousterhout, J. K. et all. A trace-driven analysis of the UNIX 4.2 BSD file system. *Proceedings of the tenth ACM symposium on operating systems principles*, Dezembro 1985.

[23] Ousterhout, J. K. Beating the I/O Bottleneck: a case for log-structured file systems. *Operating Systems Review. ACM Press*, Vol 23 no 1 Janeiro 1989.

[24] Paans, R. A Close Look At MVS Systems : Mechanisms, Performance and Security. Elsevier, 1986

[25] Price, W. A benchmark tutorial. *IEEE Micro*, Vol 9 no 5 Out. 1989.

[26] Ramakrishnan, K. K., Emer, J. S. Performance Analysis of Mass Storage Service Alternatives For Distributed Systems. *IEEE Trans. Soft. Eng.* Vol. 15 No 2 Fevereiro 1989.

[27] Robinson, P. A world of workstations. *Byte*, Novembro 1987.

[28] Rood, A. L. et all. A comparison of MC 68000 family processors. *Byte*, setembro 1986.

[29] Schroeder, M. D., Furrows, M. Performance of the Firefly RPC. *Proceedings of the twelfth ACM symposium on operating systems principles*, Dezembro 1989.