

# Avaliando Sistemas Operacionais para Máquinas de Alto Desempenho

Nicolas Kassalias, Edson T. Midorikawa

Departamento de Engenharia de Computação e Sistemas Digitais, Universidade de São Paulo  
Av. Professor Luciano Gualberto, travessa 3, 158, São Paulo, SP  
{kassalias@regulus.pcs.usp.br, edson@pcs.usp.br}

## Resumo—

Este trabalho destaca a importância dos sistemas operacionais em ambientes computacionais de alto desempenho e apresenta um estudo comparativo do desempenho de dois sistemas distintos. São comparados dois sistemas operacionais muito difundidos no mercado atual de servidores, o Windows e o Linux com a utilização de *benchmarks*.

Os resultados obtidos mostram que o Linux mantém seu desempenho por uma maior banda de memória e pode criar e remover arquivos com mais eficiência do que o Windows.

*Palavras-chave*— Análise de desempenho, *benchmarking*, sistemas operacionais, sistemas de alto desempenho.

## I. INTRODUÇÃO

Os sistemas operacionais exercem fundamental importância aos ambientes computacionais. Servem para gerenciar os computadores, controlar seus recursos e proporcionar aos usuários facilidades de operação.

Existe uma grande variedade de diferentes tipos de sistemas operacionais, gerando dificuldades na escolha do sistema mais adequado para um dado ambiente. Diversos aspectos devem ser considerados, entre os quais podem ser citados os seguintes: desempenho, segurança, portabilidade, estabilidade, suporte técnico, operacionalidade, tipo de equipamentos, suporte a rede, tolerância a falhas, custos com manutenção e contrato, limite de usuários, documentação, suporte a aplicativos, etc.

Além dos fatores mencionados anteriormente, deve-se considerar a existência da força exercida pelo mercado, sendo muito comum adotar-se como verdade algumas propagandas amplamente divulgadas e que muitas vezes acabam por exercer uma influência decisiva na escolha de um software.

Neste contexto a avaliação de sistemas operacionais assume grande importância. As comparações entre sistemas diferentes implica em estabelecer-se uma metodologia de avaliação, características e parâmetros de comparação, métricas, etc.

Para o mercado de sistemas operacionais e tratando-se de equipamentos baseados em plataforma PC  $\times 86$ , o Windows e o Linux são largamente discutidos atualmente. Em termos de mercado, o Windows apresenta-se como

líder. Já o Linux tem apresentado uma vigorosa dinâmica de crescimento. A tabela 1 a seguir apresenta dados fornecidos pela IDC – *International Data Corporation*, onde tem-se uma visão mundial do mercado de sistemas operacionais em servidores para 1997, 1998 e 1999.

Tabela 1. Mercado de sistemas operacionais em servidores (1997 a 1999).

Sistema Operacional	1997	1998	1999
Linux	6,8 %	17,2 %	24 %
Windows NT	36,0 %	36,0 %	36 %
Novell Netware	26,4 %	24,2 %	19 %
outros	30,8 %	22,6 %	21 %

A tabela 1 mostra a liderança de mercado exercida pelo Windows e também que o Linux aumentou sua participação em duas vezes e meia de 1997 para 1998, enquanto o Windows não apresentou crescimento algum. Um novo relatório da IDC mostra que o Linux assumiu o segundo posto em termos de novos licenciamentos no ano de 1999 [IDC00].

Este trabalho visa comparar dois sistemas operacionais distintos, o Windows e o Linux. Para comparar os sistemas realizaram-se experimentos utilizando alguns equipamentos e programas de *benchmarks*. As análises e conclusões partiram da pesquisa e dos testes com estes *benchmarks*.

Neste artigo, as seções II e III apresentam uma breve exposição de conceitos de *benchmarks* e uma discussão da problemática de efetuar-se comparações entre diferentes sistemas. A seção IV descreve o trabalho realizado, e os resultados dos *benchmarks* são apresentados na seção V. Por fim é realizada uma breve discussão e apresenta-se a conclusão do trabalho.

## II. BENCHMARK DE SISTEMAS OPERACIONAIS

Para realizar a comparação entre os sistemas, adotaram-se *benchmarks* de sistemas operacionais que são programas utilizados para medir o desempenho de determinadas características dos sistemas operacionais.

Uma visão mais ampla sobre *benchmark* é apresentada por André D. Balsa [BALS97]. Para o autor, *benchmarking* significa medir a velocidade com a qual um sistema computacional irá executar uma tarefa de modo a permitir a

comparação entre diferentes combinações de *hardware* e *software*. Pode-se utilizá-lo para auxiliar processos decisórios através da obtenção de dados referentes à relação desempenho / custo, velocidade de execução, desempenho de funções, estabilidade de sistema, etc. Para utilizar o *benchmark* deve-se identificar com clareza qual seu objetivo, quais os processos estão sendo medidos e envolvidos e quais os recursos serão necessários.

Os *benchmarks* podem ser classificados como:

- *Benchmarks de baixo nível*: servem para medir diretamente o desempenho de componentes isolados, tais como: ciclos de máquina, tempo médio para acessar disco rígido, etc;
- *Benchmarks de alto nível*: servem para medir o desempenho da combinação entre *hardware/driver*/sistema operacional/aplicação para execução de uma determinada tarefa, como por exemplo, medir o desempenho de E/S de um sistema de arquivos.

Existem outras classificações para os *benchmarks*, tais como *benchmarks* sintéticos, de aplicações, etc.

Para atender o objetivo de comparar os serviços básicos de ambos os sistemas operacionais analisados neste trabalho, foram adotados *benchmarks* classificados como de baixo nível que pertencem ao pacote *lmbench* [MCVO96].

### III. COMPARAÇÃO DE DESEMPENHO

Os sistemas operacionais avaliados apresentam arquiteturas distintas. A arquitetura do *Linux* é monolítica, onde o sistema operacional é um conjunto de programas ligados em um único arquivo objeto. Nele, cada processo pode chamar qualquer outro, desde que este último realize algo de que o primeiro necessite [BECK98] [CARD98] [TANE92].

A arquitetura do *Windows* é do tipo cliente-servidor, cuja característica principal é a presença de processos clientes enviando requisições a um processo servidor, o qual realiza o trabalho solicitado e envia a resposta de volta ao processo cliente solicitante. Além disso, essa arquitetura apresenta um núcleo reduzido rodando em modo kernel; os demais módulos são independentes e são executados no modo usuário, sem acesso direto ao hardware [SOLO98] [GALL00].

Para comparar-se o *Windows* com o *Linux* é necessário testar aspectos comuns a ambos sistemas. Seria difícil testar-se qual a eficiência na execução de um aplicativo baseado na API Win32, pois esse tipo de aplicativo é suportado pelo *Windows*, mas não pelo núcleo do *Linux*. Assim, para efetuar uma comparação entre dois sistemas operacionais distintos há a necessidade de definir-se o que será comparado entre ambos. Por essa razão foram utilizados *benchmarks* que pudessem ser executados em ambos sistemas operacionais.

Além da influência do sistema operacional estudado, existe também a influência da máquina e dos compiladores no desempenho dos programas. Para isso seria necessário manter constantes a máquina e os programas executáveis, e submeter-se os programas de teste aos diferentes sistemas operacionais com parâmetros comuns, com mesmos equipamentos, configurações e procedimentos. Cabe-se fazer a ressalva de que não se pode executar os mesmos programas executáveis no *Windows* e no *Linux*. Para tanto, deve-se compilá-los em compiladores compatíveis com os respectivos sistemas operacionais. Aqui já decorre um problema, pois haverá a interferência exercida pelo compilador sobre o desempenho dos programas. Portanto para um mesmo equipamento, onde os testes são realizados, e para um dado programa fonte, duas variáveis interferem nos resultados:

- o próprio sistema operacional, no caso o *Linux* e o *Windows*;
- o programa executável, gerado por um dado compilador.

É necessário destacar que os resultados obtidos em um equipamento não serão necessariamente os mesmos quando realizados em algum outro.

### IV. TRABALHO REALIZADO

O presente trabalho adotou os *benchmarks* de baixo nível, pertencentes ao pacote *lmbench*. O *lmbench* foi projetado para medir operações básicas de sistemas operacionais e de hardware [MCVO96] [LMBE96].

O conjunto de programas do *lmbench* medem o desempenho isolado de tempos de latência e largura de banda para a transferência de dados entre o processador, cache, memória, rede, sistema de arquivos, discos, etc. Os *benchmarks* adotados neste trabalho concentram seus testes nas larguras de banda porque geralmente neles residem os problemas de desempenho. O pacote também serve para identificar e avaliar “gargalos” de desempenho. Os programas são escritos na linguagem ANSI C.

Estudar o comportamento do sistema de memória é de importância relevante devido à sua influência no desempenho dos sistemas operacionais. As operações de leitura, escrita e movimentação em memória são altamente frequentes nas aplicações. Essas operações envolvem instruções de transferência de dados, sendo que a maioria dos programas contém mais instruções de transferência de dados que qualquer outro tipo de operação [PATT96] [TOKH85]. Além disso, seria muito interessante testar comparativamente o sistema de memória do *Linux* com o do *Windows* para se determinar em quais circunstâncias iniciam-se operações de paginação e substituição de páginas. Por tudo isso, os *benchmarks* *bw\_mem\_rd*, *bw\_mem\_wr* e *bw\_mem\_cp* do pacote *lmbench* foram os adotados, pois estes servem para testar o sistema de memória quanto às operações de leitura, gravação e

transferência de dados em memória.

Outro aspecto interessante para ser investigado é o comportamento dos sistemas com relação ao acesso aos arquivos em disco. O tempo de acesso a disco é alto comparado ao ciclo do processador, impactando o desempenho de processamento, especialmente de aplicações que necessitam ler ou gravar frequentemente em disco rígido. Seria interessante saber se o tempo necessário para criar e remover arquivos é o mesmo para o sistema de arquivos *ext2* (*Second Extended File System*) que é o formato padrão do Linux [BECK98] e o NTFS (*NT File System*) que é o padrão do Windows [SOLO98][MICR98]. Por essas razões, o *lat\_fs* foi o programa selecionado para testar o comportamento dos sistemas quanto à criação e remoção de arquivos em disco. A tabela 2 apresenta uma breve descrição de cada um dos *benchmarks*.

Tabela 2. Benchmarks adotados do pacote *Lmbench*.

Programa	Descrição
<i>bw_mem_rd</i>	mede a largura de banda de leitura em memória, medida em MB/s, para uma área de memória reservada
<i>bw_mem_wr</i>	mede a largura de banda de gravação em memória, medida em MB/s, para uma área de memória reservada
<i>bw_mem_cp</i>	mede a taxa em MB/s para copiar de uma área de memória reservada para uma outra área de memória de mesmo tamanho
<i>lat_fs</i>	o programa mede a taxa de criação e remoção de arquivos (o número de arquivos criados / removidos por segundo de tamanhos de 0 KB, 1 KB, 4 KB e 10 KB)

Os *benchmarks* *bw\_mem\_rd*, *bw\_mem\_wr* e *bw\_mem\_cp* recebem como parâmetro de entrada o tamanho da área de memória em que são realizadas as operações de leitura, gravação e transferência de dados, respectivamente. Cada um dos programas irá gerar como resposta uma taxa que é calculada pela relação entre o tamanho da área reservada dividida pelo tempo necessário para executar a operação.

O programa *lat\_fs* não tem nenhum parâmetro de entrada e executa sucessivamente a criação de múltiplos arquivos com tamanhos de 0 KB, 1 KB, 4 KB e 10 KB e, posteriormente, remove-os. Ao final reporta o número de arquivos criados e removidos por segundo.

Para cada uma das análises apresentadas neste trabalho, executou-se cada *benchmark* repetidas vezes e considerou-se o valor médio de seus resultados.

Ao comparar os sistemas operacionais foram utilizados os equipamentos e sistemas operacionais mostrados na tabela 3.

Com relação aos compiladores adotados, os aplicativos que rodaram no Linux foram gerados com o compilador *gcc*, adotando-se o padrão POSIX. Já para o Windows, foram gerados dois tipos de aplicativos: um gerado pelo compilador *DJGPP* e o outro pelo Microsoft *Visual C++*.

Tabela 3. Equipamentos e sistemas operacionais utilizados.

Configuração	Dual Pentium Pro 200 MHz		Celeron 500 MHz	
Cache	256 KB		128 KB	
RAM	128 MB		128 MB	
Sist. Operacional	Windows NT 4.0	Linux Kernel 2.0	Windows 2000	Linux Kernel 2.2

A versão gerada pelo *DJGPP* é compatível com a API do sistema MS-DOS, enquanto a versão gerada pelo *Visual C++* é compatível com a API Win32. A tabela 4 mostra os compiladores adotados para cada sistema.

Tabela 4. Compiladores e sistemas operacionais.

Sistema Operacional	Linux	Windows	
Compilador	<i>gcc</i>	<i>DJGPP</i>	<i>Visual C++</i>

Sinteticamente, os testes executados apresentam as seguintes condições comuns:

- os mesmos programas fontes;
- a coleta de dados foi realizada utilizando-se exatamente as mesmas máquinas;
- os procedimentos experimentais foram exatamente os mesmos.

As diferenças relevantes para a execução dos testes foram:

- os próprios sistemas operacionais;
- os programas executáveis.

## V. RESULTADOS OBTIDOS

Como mencionado foram realizados testes do sistema de memória e do sistema de arquivos. Os resultados obtidos nestes testes serão apresentados a seguir.

### V.1- Resultados dos Testes de Memória

As figuras 1 e 2 apresentam os resultados dos testes dos sistemas de memória do Windows e do Linux. Para cada gráfico, o eixo das abscissas representa a quantidade de memória usada no teste e o eixo das ordenadas contém a largura de banda medida.

De uma maneira geral, constata-se que as curvas apresentam um patamar superior paralelo ao eixo das abscissas, passando a sofrer inclinação decrescente e, finalmente, atingindo um patamar inferior, também paralelo ao eixo das abscissas.

O patamar superior de cada curva é a região possuindo os valores mais elevados, indicando a existência de memória livre suficiente à aplicação, nessa região não ocorreram operações de paginação.

A inclinação da curva apresenta valores intermediários entre o patamar superior e o inferior. Essa inclinação reflete

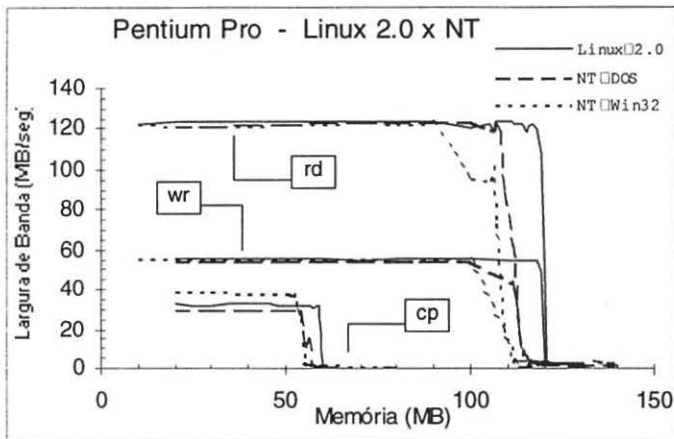


Fig. 1. Larguras de banda para o Windows NT 4.0 e o Linux kernel 2.0 no Pentium Pro.

o início de operações de paginação realizadas pelo sistema operacional, devido à escassez de memória livre disponível. A degradação aumenta à medida em que se aumenta a quantidade de memória usada.

Finalmente, no patamar inferior, onde encontram-se os menores valores da curva, há o predomínio das operações de substituição de páginas.

Nas figuras 1 e 2, apresentam-se as curvas NT-DOS e Win2k que foram geradas pelos benchmarks compilados pelo DJGPP e as curvas NT-Win32, geradas pelos benchmarks compilados pelo Visual C++, produzindo programas objetos suportados pelo sistema operacional Windows. As curvas Linux 2.0 e Linux 2.2 foram obtidas pelos benchmarks compilados com o gcc, gerando programas executáveis compatíveis com o Linux. As curvas apresentam em suas legendas os conjuntos de letras rd, wr e cp que correspondem respectivamente aos benchmarks  $bw\_mem\_rd$ ,  $bw\_mem\_wr$  e  $bw\_mem\_cp$ .

As subseções seguintes contêm análises dos resultados obtidos.

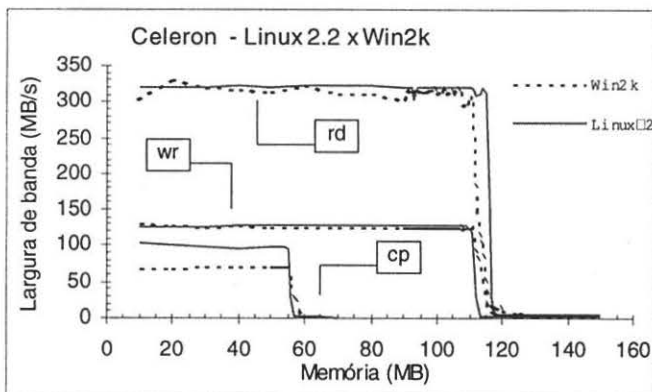


Fig. 2. Larguras de banda para o Windows 2000 e o Linux kernel 2.2 no Celeron.

### V.1.1 - Análise do Patamar Superior

A tabela 5 apresenta uma tabela contendo os valores médios referentes ao patamar superior dos gráficos apresentados nas figuras 1 e 2.

Tabela 5. Desempenho das plataformas (MB/s).

Equipamento	curvas	$bw\_mem\_rd$	$bw\_mem\_wr$	$bw\_mem\_cp$
Pentium Pro	NT-DOS	122,7	54,8	30,0
	NT-Win32	122,4	55,1	38,6
	Linux 2.0	123,4	55,1	32,3
Celeron	Win2k	311,7	126,0	70,3
	Linux 2.2	321,0	127,4	98,6

#### V.1.1.1 - Análise do patamar superior das curvas geradas pelo equipamento Pentium Pro

Para o Pentium Pro, verifica-se que, para os testes com os benchmarks  $bw\_mem\_rd$  e  $bw\_mem\_wr$ , as curvas mostradas na figura 1, são praticamente coincidentes, indicando um desempenho e uma estabilidade de desempenho similares entre si para operações de leitura e gravação em memória, independente do sistema operacional. Esse comportamento pode ser notado também pelos valores médios serem praticamente os mesmos, conforme tabela 5.

Os testes para o benchmark  $bw\_mem\_cp$  mostram que a curva NT-Win32 apresenta desempenho superior quanto operações de transferência de dados na memória executados no Windows com o benchmark compilado pelo Visual C++. Isto também é comprovado pelo valor médio dessa curva ser maior entre todos os testes para o Pentium Pro.

#### V.1.1.2 - Análise do patamar superior das curvas geradas pelo Celeron.

Observa-se da figura 2 que para o benchmark  $bw\_mem\_rd$ , as curvas estão próximas, porém apresentando alguma sinuosidade, indicando uma certa variação na estabilidade dos resultados. A curva do Windows 2000 apresentou maior oscilação, em relação à curva do Linux 2.2, indicando que o comportamento da aplicação no Linux foi um pouco mais estável para as operações de leitura em memória.

Para o programa  $bw\_mem\_wr$ , verifica-se do gráfico que as curvas são retilíneas e praticamente sobrepostas, isso significa que houve estabilidade de resultados e que o desempenho das aplicações foram similares para os sistemas operacionais quanto a operações de gravação em memória. A tabela 5 apresenta os valores médios das curvas, e mostra que os valores médios são próximos entre si, com discreta vantagem para o Linux 2.2.

Quanto ao programa  $bw\_mem\_cp$ , observa-se que a curva gerada pelo Linux 2.2 apresentou-se mais retilínea

e esteve acima da curva do Windows 2000. Isso indica que a aplicação no Linux obteve melhor desempenho e estabilidade dos resultados para operações de transferência de dados em memória.

Destaca-se o fato de que o desempenho das operações de transferência em memória apresentaram-se melhores para o Windows NT no Pentium Pro, para o executável gerado pelo Visual C++ em contraste com a curva gerada no Celeron, onde o melhor desempenho foi obtido no Linux, com o executável gerado pelo compilador gcc. Esse comportamento pode ter ocorrido devido a otimização gerada pelos compiladores.

#### V.1.2 - Análise da Capacidade de Processar sem sofrer Degradação

A tabela 6 apresenta dados obtidos a partir das curvas, onde tem-se o valor da quantidade de memória usada pela aplicação onde ocorre o início da declividade da mesma. Esse valor indica a capacidade máxima de memória antes do aplicativo sofrer degradação.

Tabela 6. Valor máximo antes de degradar.

Máquina	curvas	bw_mem_rd	bw_mem_wr	bw_mem_cp
Pentium Pro	NT-DOS	107 MB	100 MB	53 MB
	NT-Win32	90 MB	100 MB	51 MB
	Linux 2.0	117 MB	118 MB	59 MB
Celeron	Win2k	107 MB	105 MB	55 MB
	Linux 2.2	111 MB	109 MB	55 MB

A tabela 6 mostra a capacidade de processar sem sofrer degradação foi melhor para o Linux. Com exceção da curva gerada pelo *bw\_mem\_cp* no Celeron, onde houve a mesma banda de memória sem degradar, nas demais curvas, o Linux apresentou a maior banda de memória sem degradar.

#### V.1.3 - Análise do patamar inferior das curvas

Para o patamar mais baixo, as curvas são praticamente coincidentes, indicando que o comportamento dos sistemas quanto ao desempenho são similares para essa região.

#### V.2- Testes dos Sistemas de Arquivos

Os resultados dos testes de criação de arquivos do *benchmark lat\_fs* estão apresentados nas figuras 3 e 4. A figura 3 refere-se aos testes executados com o equipamento Pentium Pro comparando os resultados produzidos pelas aplicações NT-DOS e NT-Win32 executadas sobre o Windows NT 4.0 com sistema de arquivos NTFS e a aplicação Linux executada sobre o sistema operacional Linux kernel 2.0, com sistema de arquivos ext2.

A figura 4 refere-se ao gráfico gerado pelo equipamento Celeron, comparando os resultados referentes a criação de arquivos entre o sistema Windows 2000 e Linux 2.2.

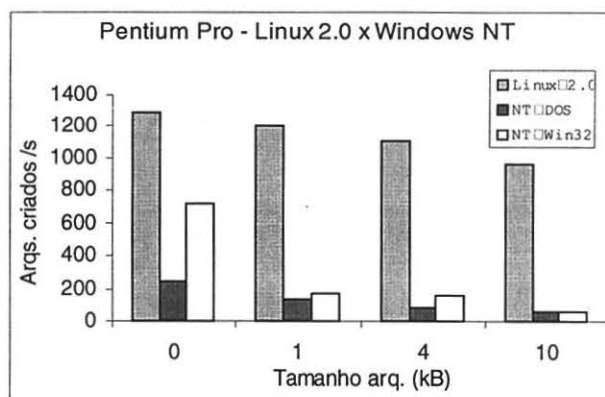


Fig. 3. Criação de arquivos Pentium Pro.

Das figuras 3 e 4, fica claro que os testes executados no Linux obtiveram um desempenho muito melhor na criação de arquivos em disco rígido.

As diferenças nos resultados obtidos são expressivos e mostram claramente um melhor desempenho do Linux nestes testes.

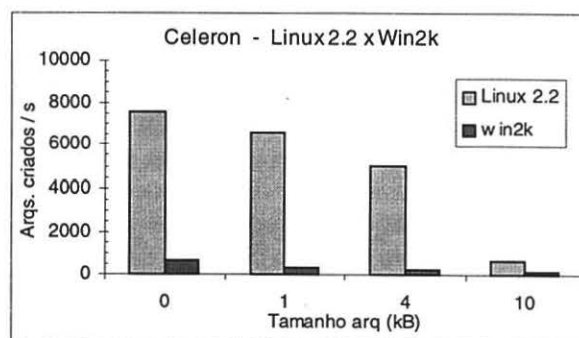


Fig. 4. Criação de arquivos Celeron.

A figura 5 apresenta os resultados dos testes referentes a remoção de arquivos, executados pelo programa *lat\_fs*, com o equipamento Pentium Pro com as versões NT-DOS e NT-Win32 executadas sobre o Windows NT 4.0, com a versão Linux 2.0, executado sobre o sistema operacional Linux kernel 2.0.

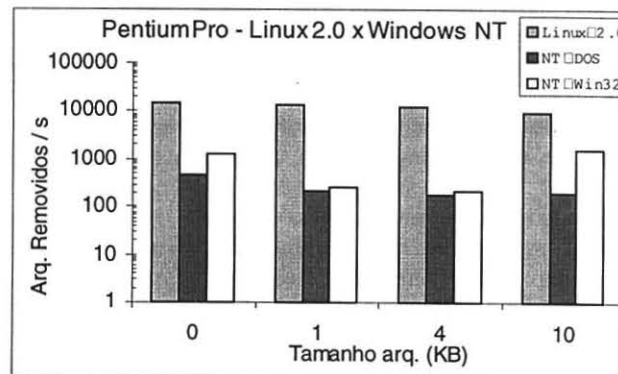


Fig. 5. Remoção de arquivos - Pentium Pro.

A figura 6 apresenta os testes realizados no equipamento Celeron, e também referem-se a remoção de arquivos, nos sistemas operacionais Windows 2000 com sistema de arquivos NTFS e Linux kernel 2.2 com sistema de arquivos ext2.

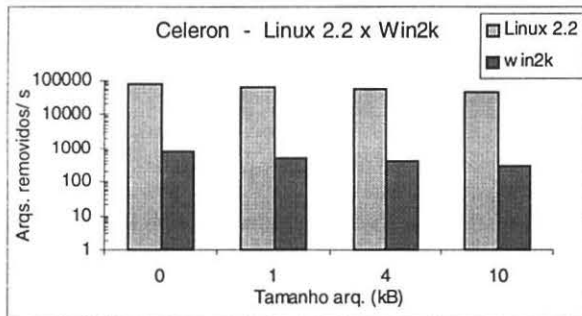


Fig. 6. Remoção de arquivos - Celeron.

Das figuras 5 e 6, constata-se a nítida vantagem de desempenho do Linux quanto à remoção de arquivos. O desempenho do Linux foi de cerca de 2 a 3 ordens de grandeza melhor.

## VI. DISCUSSÃO E CONCLUSÃO

Este trabalho aborda questões relativas à comparação de desempenho de dois sistemas operacionais muito diferentes, o Windows e o Linux. Os programas usados neste trabalho foram divididos em dois grupos: os que testaram operações em memória (*bw\_mem\_rd*, *bw\_mem\_wr*, *bw\_mem\_cp*) e o que realizou testes com o sistema de arquivos (*lat\_fs*).

Os *benchmarks* realizaram testes de operações básicas e simples para estudar de forma isolada quais são suas influências no desempenho de todo o sistema. Como descrito, o Linux apresenta arquitetura diferente do Windows, porém, existem operações comuns a ambos que puderam ser comparadas. Para realizar tal comparação mantiveram-se condições ambientais similares. As medidas foram avaliadas e comparadas entre sistemas diferentes, mas para um mesmo hardware, comparando-se o quão rápidas foram as operações para a mesma memória física, mesmo disco rígido, enfim, para a mesma máquina.

O trabalho não analisou o comportamento em um ambiente multiusuário, onde tem-se várias aplicações em execução simultânea.

Nos testes de memória, verificou-se que a largura de banda de cada sistema variava de acordo com o uso da memória. As curvas mostravam um comportamento característico com um patamar superior e um patamar inferior. A degradação de desempenho iniciava-se quando começava a faltar memória; neste instante iniciam-se as operações de paginação de memória e o aumento da degradação ocorre na medida em que a aplicação requer mais memória até atingir o instante em que ela não existe mais.

Quanto ao *desempenho no patamar superior* para o programa *bw\_mem\_cp* convém ressaltar que notou-se um resultado não conclusivo em que nenhum sistema apresentou melhor resultado para ambas as máquinas adotadas. Isto pode ser explicado devido ao *benchmark* utilizar a função *memcpy* da biblioteca da linguagem C, e assim, o desempenho fica fortemente dependente do compilador usado.

Verificou-se que o sistema operacional Linux teve capacidade de processamento com maior aproveitamento da memória, antes de degradar. Isso ocorreu devido ao mencionado sistema operacional ocupar menos memória, fazendo com que exista mais memória livre para as aplicações e, conseqüentemente, menos operações de paginação.

Os testes foram conclusivos quanto ao sistema Linux apresentar melhor desempenho em criar e remover arquivos em relação aos demais sistemas. Essa diferença ocorre devido ao fato do sistema de arquivos ext2 do Linux ter uma operação mais simples que o NTFS usado no Windows.

Como trabalho futuro, podemos citar testes com outros programas do pacote *Lmbench*, bem como com outros pacotes (*HBenchOS* e *ParkBench*) que estão em andamento. Outras plataformas de hardware estão sendo utilizadas nestes trabalhos (Athlon, Pentium Xeon e Alpha).

## VII. REFERÊNCIAS

- [BALS97] BALS, A. D. *Linux Benchmarking HOWTO*, Linux Documentation Project, August 1997. <http://metalab.unc.edu/LDP>
- [BECK98] BECK, M. et al. *Linux kernel internals*. 2<sup>nd</sup> edition. Addison-Wesley, 1998.
- [CARD98] CARD, R. et al. *The Linux kernel book*. Wiley, 1998.
- [GALL00] GALLI, D. L. *Distributed operating systems: concepts & practice*. Prentice-Hall, 2000.
- [IDG00] <http://www.idc.com/itforecaster/itf20000808.stm>, August 8, 2000.
- [LMBE96] *Lmbench - Tools for Performance Analysis*. <http://www.bitmover.com/lmbench>.
- [MICR98] MICROSOFT. *Suporte às Tecnologias Principais do Microsoft Windows NT 4.0*. Manual do Aluno. Microsoft Corporation, 1998.
- [MCVO96] McVOY, L. & STAELIN, C. *Lmbench: portable tools for performance analysis*. In: USENIX Conference, January 1996. Proceedings. p.279-284.
- [PATT96] PATTERSON, D. et al. *Computer Architecture: A Quantitative Approach*. 2<sup>nd</sup> Edition. Morgan Kaufmann, 1996.
- [SOLO98] SOLOMON, D. A. *Desvendando o Windows NT*. Rio de Janeiro, Editora Campus, 1998.
- [TANE92] TANENBAUM, A. S. *Modern Operating Systems*. Prentice Hall, 1992.
- [TOKH85] TOKHEIM, L. R. *Introdução aos Microprocessadores*. McGraw-Hill, 1985.