

Guilherme Pina Cardim

Pesquisa de Sistemas Operacionais I

Presidente Prudente - SP, Brasil

30 de junho de 2010

Guilherme Pina Cardim

Pesquisa de Sistemas Operacionais I

Pesquisa realizada para identificar diferenças
entre Windows e Linux.

Professor: Maurício Araújo Dias

DEMEC - DEPARTAMENTO DE MATEMÁTICA, ESTATÍSTICA E COMPUTAÇÃO
FCT - FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNESP - UNIVERSIDADE ESTADUAL PAULISTA

Presidente Prudente - SP, Brasil

30 de junho de 2010

Resumo

Pesquisa realizada para identificar diferenças entre Windows e Linux.

Sumário

Lista de Figuras

1	INTRODUÇÃO	p. 5
1.1	Windows	p. 5
1.2	Linux	p. 7
2	REQUISITOS DO COMPUTADOR	p. 9
2.1	Requisitos mínimos para Windows Vista Home Premium	p. 9
2.2	Requisitos mínimos para Windows 7	p. 9
2.3	Requisitos mínimos para Ubuntu	p. 9
2.4	Requisitos mínimos para Conectiva	p. 10
3	TRATAMENTO DE DEADLOCKS	p. 11
4	ALGORITMOS DE ESCALONAMENTO	p. 12
4.1	Escalonamento no Windows	p. 12
4.2	Escalonamento no Linux	p. 12
5	THREADS	p. 14
6	API - Application Programming Interface	p. 15
7	DLL - Dynamic-link library	p. 16
	Referências	p. 17

Lista de Figuras

1	Gráfico do uso dos SOs	p. 5
2	Microsoft Windows 1.0	p. 6
3	Microsoft Windows 7	p. 6
4	Ubuntu	p. 7
5	Conectiva	p. 8

1 INTRODUÇÃO

Um Sistema Operacional, ou simplesmente SO, é um programa computacional que administra os recursos do computador, sejam eles recursos de *hardware* ou de *software*. Esse programa tem por objetivo conseguir controlar todos os recursos do computador de forma segura.

Atualmente existem vários sistemas operacionais, alguns gratuitos enquanto outros não são. Contudo os mais utilizados em Desktops são o Windows e o Linux, ambos com variadas versões e distribuições. O gráfico a seguir, retirado do site <market-share.hitslink.com>, nos mostra a quantidade de usuários utilizando cada sistema operacional.

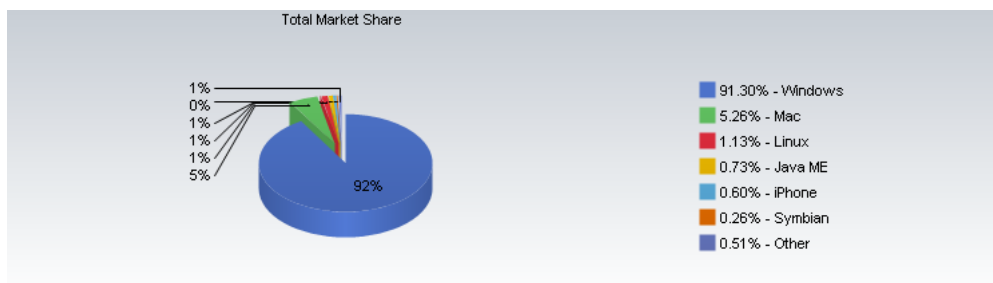


Figura 1: Gráfico do uso dos SOs

Apesar do gráfico mostrar o uso do MAC maior do que o uso do Linux, sabe-se que o uso do Mac é restrito aos computadores de mesmo nome produzido pela empresa Apple, dessa forma o presente relatório mostrará diferenças apenas dos Sistemas Operacionais Windows e Linux.

1.1 Windows

O Sistema Operacional Windows foi desenvolvido pela empresa norte americana Microsoft e possui ao longo de sua história o desenvolvimento de várias versões. As primeiras

versões foram chamadas de Windows 1.0, logo depois o Windows 2.0 e 3.x. Pode-se observar a imagem ilustrada pela Figura 2 que apresenta uma tela do Windows 1.0.

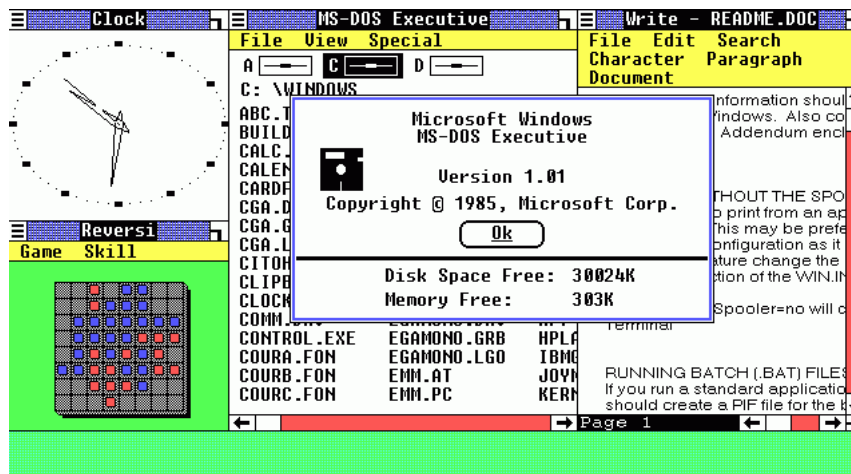


Figura 2: Microsoft Windows 1.0

Depois surgiram outras versões como, Windows 95, Windows 98, Windows 2000, Windows NT, Windows XP, Windows Vista e por fim e mais atual o Windows 7. Uma tela do Windows 7 pode ser visualizada na Figura 3.



Figura 3: Microsoft Windows 7

Pode-se observar o grande avanço gráfico entre a primeira e a última versão do Windows pelas duas imagens desses SOs apresentadas anteriormente.

Apesar de ser um produto comercial com alto preço, o Windows em suas diversas versões é o SO mais utilizado pelo fato de ter sido o primeiro SO a atingir os usuários

comuns, os quais normalmente possuem dificuldades para se adaptar aos novos SOs, uma vez que o Windows monopolizou o mercado durante décadas. Além disso, uma grande quantidade dos computadores com Sistemas Operacionais Windows são provenientes de cópias ilegais, o que ajudou a popularizar esse SO.

O Windows foi desenvolvido em C, C++ e Assembly e teve sua primeira aparição com o Windows 1.0 em 20 de novembro de 1985.

1.2 Linux

Desenvolvido por Linus Torvalds e milhares de contribuidores, o Linux é um Sistema Operacional desenvolvido em linguagem C, e está registrado sobre os termos da licença GNU, ou seja, o Linux pode ser usado, alterado e replicado como e quantas vezes for necessário sem pagar nada a empresa que o desenvolveu.

Por esse motivo, existem diversas distribuições Linux geradas a partir do código principal desenvolvido pelo Torvalds, entre elas pode-se destacar as distribuições Ubuntu, Debian, Fedora, Kurumin, Slackware, Mandriva, Conectiva, Red Hat, Gentoo, etc. Observe a tela do Ubuntu apresentada pela Figura 4 e a tela do Conectiva ilustrada na Figura 5.



Figura 4: Ubuntu

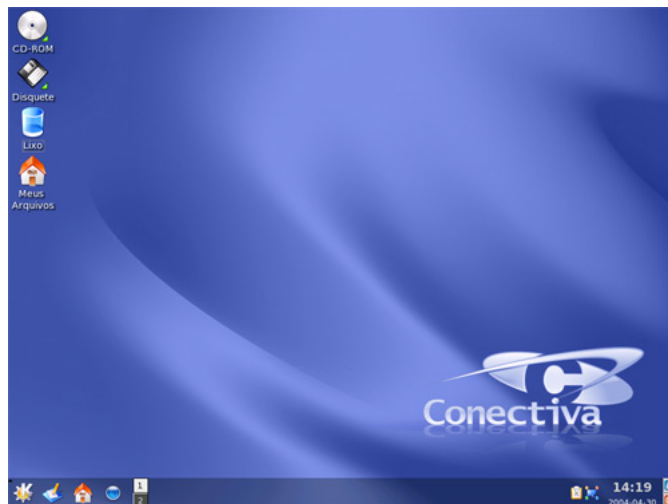


Figura 5: Conectiva

2 REQUISITOS DO COMPUTADOR

Cada Sistema Operacional necessita de uma configuração mínima de *hardware* do computador para poder funcionar normalmente.

2.1 Requisitos mínimos para Windows Vista Home Premium

- Processador de 800 MHz;
- 512 MB de memória do sistema;
- Disco rígido de 20 GB com pelo menos 15 GB de espaço disponível;
- Suporte para gráficos Super VGA;
- Unidade de CD-ROM.

2.2 Requisitos mínimos para Windows 7

- Processador de 1 GHz ou superior de 32 *bits* (x86) ou 64 *bits* (x64);
- 1 GB de RAM (32 *bits*) ou 2 GB de RAM (64 *bits*);
- 16 GB de espaço em disco disponível (32 *bits*) ou 20 GB (64 *bits*)
- Dispositivo gráfico DirectX 9 com driver WDDM 1.0 ou superior.

2.3 Requisitos mínimos para Ubuntu

- Processador de 1 GHz ou superior;

- 1 GB de RAM;
- 15 GB de espaço em disco disponível;
- Dispositivo gráfico e monitor capaz de aceitar 1024 x 768;
- Unidade de CD-ROM ou entrada USB.

2.4 Requisitos mínimos para Conectiva

- Processador de 100 MHz ou superior;
- 64 MB de RAM;
- 2 GB de espaço em disco disponível;
- Adaptador de vídeo VGA.

3 TRATAMENTO DE DEADLOCKS

Tanto em Linux como em Windows o método de tratamento de deadlock escolhido foi o algoritmo da avestruz, ou seja, ambos os sistemas ignoram completamente a existência desse problema dentro da computação.

Essa técnica é utilizada por esses SOs, pelo fato deles necessitarem serem altamente interativos com usuários, não sabendo assim quais e quando os recursos serão utilizados. Além disso, os usuários precisam de respostas com extrema eficiência e rapidez, o que se torna praticamente impossível ao querer tratar um *deadlock* por qualquer estratégia que seja, pois estas reduzirão consideravelmente a performance do computador.

4 ALGORITMOS DE ESCALONAMENTO

O algoritmo de escalonamento é quem decide qual processo será executado em determinado momento. É ele que faz a troca de processos em execução dentro do processador.

4.1 Escalonamento no Windows

O Windows utiliza um algoritmo de escalonamento preemptivo por prioridades, onde são definidos 32 níveis de prioridades divididos em 6 classes:

- `REALTIME_PRIORITY_CLASS` (24);
- `HIGH_PRIORITY_CLASS` (13);
- `ABOVE_NORMAL_PRIORITY_CLASS` (10);
- `NORMAL_PRIORITY_CLASS` (8);
- `BELOW_NORMAL_PRIORITY_CLASS` (6);
- `IDLE_PRIORITY_CLASS` (4).

Os processos com prioridades de 1 a 15 podem alterar sua prioridade dinamicamente, enquanto que os processos com prioridades de 16 a 32 têm prioridade fixa.

O escalonador cria uma fila para cada um dos seis níveis de prioridade e escolhe o processo com prioridade mais alta para entrar em execução.

4.2 Escalonamento no Linux

No Linux, as tarefas do Kernel também são incluídas no escalonamento de processos. O Linux utiliza três métodos diferentes de escalonamento de processos, sendo que é a

classe de escalonamento de cada processo que determina qual algoritmo utilizar.

- SCHED_FIFO: *First-in-first-out*, o primeiro que entra na fila é o primeiro que será executado;
- SCHED_RR: *Round-Robin*;
- SCHED_OTHER: Escalonamento hierarquico com realimentação.

5 *THREADS*

Threads são partes, ou segmentos do processo que podem ser executadas em paralelo, sem dependência entre elas, ou seja, uma thread é cada uma das atividades que compõem um processo.

A utilização de threads melhora o desempenho do computador, uma vez que elas poderão ser executadas separadamente, dentro do processador, possibilitando que o escalonador de processos escalone diretamente threads ao invés de processos, pois esses necessitam de mais memória e possuem um contexto maior para recarregá-lo no processador, dessa forma executando threads estaremos melhorando a performance do computador.

Atualmente tanto o Windows, como o Linux, permitem a implementação de algoritmos com threads.

A programação e processamento em multithread foi uma das grandes novidades apresentadas a partir da apresentação da família Windows 32 *bits*.

6 API - Application Programming Interface

A API do Windows é uma interface de interação entre a aplicação e serviços do sistema operacional, ou seja, a API do Windows é uma interface voltada para funções de controle do sistema. Como exemplos dessas funções pode-se utilizar as funções de criação de janelas, criação de menus, envio de comunicação entre janelas ativas, manipulação de threads/processos, manipulação de arquivos, suporte gráfico para desenho, gerência de memória, entre outras várias utilidades, uma vez que o Windows chama essas funções para controle interno.

Existe sim API no Linux, contudo ela trabalha de modo diferente da API do Windows.

7 DLL - Dynamic-link library

É uma implementação feita pelo Windows para o conceito de bibliotecas compartilhadas nos sistemas operacionais Windows, independente da versão. Uma dll pode conter códigos, dados e até mesmo recursos para uso compartilhado.

Em Linux existe sim dll's, contudo elas são chamadas simplesmente de bibliotecas.

Referências

Campos, Augusto. O que é uma distribuição Linux. BR-Linux. Florianópolis, março de 2006. Disponível em <<http://br-linux.org/faq-distribuicao>> acessado em 30 de junho de 2010.

Ferreira, Luis L.; Viamonte, Maria J. Sistemas Operacionais I - Escalonamento. Fevereiro de 2006. Disponível em <<http://www.dei.isep.ipp.pt/llf/docs/Escalonamento-VI.pdf>> acessado em 30 de junho de 2010.

Paula, Fábio B. INTRODUÇÃO - O Surgimento do Lunux. Disponível em <<http://www.vivaolinux.com.br/artigo/O-Surgimento-do-Linux/>> acessado em 30 de junho de 2010.

Muniz, Denis B. Api Do Windows, Referência! Disponível em <<http://scriptbrasil.com.br/forum/index.php?showtopic=91710>> acessado em 30 de junho de 2010.

Silva, Adenilson C. da. Programação multithread no Windows 32bits: como fazer 2 tarefas ao mesmo tempo. Julho de 2002. Disponível em <<http://www.linhadecodigo.com.br/Artigo.aspx?id=39>> acessado em 30 de junho de 2010.

Tanenbaum, A. S. Sistemas Operacionais Modernos. 2ª ed, Prentice Hall.

<<http://x'/operating-system-market-share.aspx?qprid=8>> acessado em 30 de junho de 2010.

<http://pt.wikipedia.org/wiki/Microsoft_Windows> acessado em 30 de junho de 2010.

<http://en.wikipedia.org/wiki/Microsoft_Windows> acessado em 30 de junho de 2010.

<<http://pt.wikipedia.org/wiki/Linux>> acessado em 30 de junho de 2010.

<<http://en.wikipedia.org/wiki/Linux>> acessado em 30 de junho de 2010.

<<http://www.microsoft.com/brasil/windows/products/windowsvista/editions/homepremium/default.aspx>> acessado em 30 de junho de 2010.

<<http://windows.microsoft.com/pt-BR/windows7/products/system-requirements>>

acessado em 30 de junho de 2010.

<<https://help.ubuntu.com/community/Installation/SystemRequirements>> acessado em 30 de junho de 2010.

<http://www.conectiva.com/doc/livros/online/10.0/instalacao/pt_BR/ch01s02.html> acessado em 30 de junho de 2010.

<<http://kernelbook.sourceforge.net/kernel-api.html/>> acessado em 30 de junho de 2010.

<<http://pt.wikipedia.org/wiki/DLL>> acessado em 30 de junho de 2010.

<<http://www.linuxforums.org/forum/coffee-lounge/37811-does-linux-use-dlls.html>> acessado em 30 de junho de 2010.