



LIVRO

UNIDADE 4

Fundamentos de Sistemas Operacionais

GERENCIAMENTO DE DISPOSITIVOS

Juliana Schiavetto Dauricio

© 2015 por Editora e Distribuidora Educacional S.A

Todos os direitos reservados. Nenhuma parte desta publicação poderá ser reproduzida ou transmitida de qualquer modo ou por qualquer outro meio, eletrônico ou mecânico, incluindo fotocópia, gravação ou qualquer outro tipo de sistema de armazenamento e transmissão de informação, sem prévia autorização, por escrito, da Editora e Distribuidora Educacional S.A.

2015

Editora e Distribuidora Educacional S. A.
Avenida Paris, 675 – Parque Residencial João Piza
CEP: 86041 -100 – Londrina – PR
e-mail: editora.educacional@kroton.com.br
Homepage: <http://www.kroton.com.br/>

Sumário

Unidade 4 Gerenciamento de dispositivos	7
Seção 4.1 - Gerenciamento de memória: conceitos, tipos, características e virtualização	9
Seção 4.2 - <i>Swapping</i> : conceitos, tipos e suas características	23
Seção 4.3 - Memória virtual: conceitos, paginação, segmentação e virtualização	35
Seção 4.4 - Gerenciamento de dispositivos de entrada e saída: conceitos, rotinas, tipos e suas características	49

Palavras do autor

Olá, aluno. Bem-vindo a mais esta unidade de ensino. Vamos, a partir de agora, conhecer como o sistema operacional realiza a gerência de dispositivos.

Nesse sentido, vamos trabalhar os conceitos de:

- gerenciamento de memória: conceitos, tipos, características e virtualização;
- swapping: conceitos, tipos e suas características;
- memória virtual: conceitos, paginação, segmentação e virtualização;
- gerenciamento de dispositivos de entrada e saída: conceitos, rotinas, tipos e suas características.

Dessa forma, conseguimos evoluir para compreender todos os mecanismos computacionais que são de responsabilidade gerencial do sistema operacional, uma vez que já conhecemos os principais fundamentos de sistemas operacionais, como acontece o gerenciamento dos processos e threads, bem como a comunicação e escalonamento requeridos para o processamento das tarefas. Além desses, vimos também como se organizam os diretórios, quais são e as funções dos sistemas de arquivos.

Então, desde já ficam as recomendações de acesso ao material sempre que surgirem dúvidas de assuntos já trabalhados e desta unidade, ou mesmo de acesso aos conteúdos complementares recomendados ao longo de sua trajetória de estudos.

Além disso, é de extrema importância para a evolução do processo de ensino-aprendizagem que pretendemos construir que você realize as atividades propostas e tire suas dúvidas. No mais, vamos seguir em frente e bons estudos!

GERENCIAMENTO DE DISPOSITIVOS

Convite ao estudo

Olá, aluno, vamos continuar com os estudos e conhecer mais alguns aspectos fundamentais para o bom funcionamento dos sistemas operacionais, com uma abordagem voltada para a gerência de dispositivos. Com eles e os demais materiais estudados, você poderá desenvolver uma competência que é considerada fundamental para esta unidade curricular: o aluno deverá ser capaz de identificar quais são as principais funções de um sistema operacional, bem como ter conhecimento sobre como se dá o compartilhamento de recursos e a sua gerência.

Além disso, buscamos também como objetivos específicos desta unidade de ensino:

- conhecer os principais conceitos, as características da gerência de memória e o que é, além da virtualização de arquivos;
- conhecer o que é swapping, como o sistema operacional gerencia essa tarefa e aloca os devidos recursos, a fim de atender às demandas de processamento e alocação;
- conhecer os principais conceitos sobre memória virtual e os procedimentos necessários a essa implantação;
- conhecer como se dá a gerência de processos advindos das solicitações dos dispositivos de entrada e saída.

A fim de aproximar os conteúdos e teorias com a sua aplicação prática, para esta unidade vamos considerar o cenário de uma empresa, cujo foco está em

fazer pesquisa e desenvolvimento na área de tecnologias da informação. Com o intuito de diminuir o gargalo existente na indústria de telecomunicações, no que tange aos recursos de memória e processamento, vamos estudar os principais conceitos envolvidos nas arquiteturas utilizadas atualmente.

A partir disso, podemos propor alguns melhoramentos, voltando sempre à evolução para a aplicação em dispositivos móveis, uma vez que a tendência de ampla utilização, para as mais diversas áreas, é real e crescente.

Dedicação e curiosidade são seus ingredientes para esta jornada de estudos! Desde já, bons estudos e práticas.

Seção 4.1

Gerenciamento de memória: conceitos, tipos, características e virtualização

Diálogo aberto

De acordo com as tendências de mercado, você já deve ter observado que cada vez mais os dispositivos móveis ganham funcionalidade que vão além de simplesmente realizar chamadas telefônicas e enviar mensagens instantâneas.

A quantidade de memória suficiente nos dispositivos móveis se tornou um desafio para as empresas que precisam armazenar o sistema operacional, aplicativos, registros e, ainda, deixar espaço para que o usuário realize o download de outros serviços e aplicativos disponíveis para os seus respectivos dispositivos celulares. A memória é imprescindível e sua quantidade adequada é um desafio para as fabricantes.

Para esta aula, vamos trabalhar com um cenário em que uma empresa de pesquisa e desenvolvimento precisa realizar um levantamento de tipos de serviços que podem ser utilizados pelos usuários, que venham a minimizar os problemas com armazenamento em seus aparelhos móveis, e explicar a causa desse gargalo nesse mercado, uma vez que os princípios e técnicas são os mesmos. Através de pesquisas de boas práticas, vamos identificar quais ferramentas podem auxiliar na otimização dos espaços de memória em dispositivos móveis.

Já estudamos o que são os sistemas operacionais, suas responsabilidades e, principalmente, quais são os seus recursos de gerenciamento de processos, threads, sincronização e comunicação, bem como a realização do escalonamento que favorecerá a execução das tarefas. Compreendemos, também, como ocorre a gerência dos dispositivos.

Agora, precisamos entender como funciona um outro recurso administrado pelo sistema operacional e que é de extrema importância para a realização de suas atividades. Estamos falando de gerência de memória.

E, então, como funcionam os mecanismos de alocação de memória? Quais são os tipos de memória considerados nas arquiteturas de hardware atualmente?

As respostas dessas e de outras perguntas serão respondidas no decorrer dos

estudos desta unidade de ensino.

Para essa, ficamos com a distinção entre memória lógica e física, partições fixas e variáveis e o que é memória virtual.

Desde já, bons estudos e práticas a você!

Não pode faltar

Vamos iniciar os estudos sobre os recursos e mecanismos adotados pelo sistema operacional para realizar o gerenciamento de memória, no entanto é preciso esclarecer que aqui serão abordados conceitos gerais e não, exclusivamente, com o foco em uma arquitetura em especial. Por esse motivo, quando a pretensão for compreender o gerenciamento dos recursos de acordo com uma determinada arquitetura de computador, será preciso investigar quais são as suas especificidades. Isso agregará e facilitará na parametrização do sistema operacional que trabalhará.

Sendo assim, vamos iniciar, então, a descrição do que é memória, como ela está dividida e de que forma realiza a alocação de seus espaços para as mais variadas aplicações de software, informações e dados.

Quando vamos comprar um computador, precisamos nos atentar a alguns recursos, não é mesmo? Dentre eles, quais são os que você mais tende a observar? Claro que depende do que cada indivíduo necessita para realizar suas tarefas cotidianas, porém, quando vamos escolher um novo computador para comprar, queremos garantir que tenhamos basicamente: processamento eficiente e alta capacidade de armazenamento.

Com isso, acabamos por observar no ato da compra de uma máquina: qual é o tipo de processador, quanto tem de memória RAM? Qual a capacidade de armazenamento do hard disk – disco rígido (HD)? O sistema operacional que já vem licenciado e instalado na máquina? Vejamos um exemplo de quando iniciamos essa busca por algumas configurações comuns para a atualidade. Vamos escolher uma marca qualquer e especificar, de acordo com as respectivas famílias de processadores e capacidade de armazenamento que atendam às mais variadas tarefas que um usuário pode precisar. Sem contar os que utilizam para o trabalho mais recursos de máquina, no entanto configurações especiais requerem pesquisas mais rigorosas. Não se esqueça!



Refleta

"[...] o disco é um recurso compartilhado, sua utilização deverá ser

gerenciada unicamente pelo sistema operacional, evitando que a aplicação possa ter acesso a qualquer área do disco sem autorização, o que poderia comprometer a segurança e a integridade do sistema de arquivos” (MACHADO; MAIA, 2013).

Então, se você fosse comprar hoje um computador, qual configuração escolheria? Vou sugerir a seguinte: processador de última geração (família “i3, i5 ou i7”), memória RAM 4 gigabytes e disco rígido 500 GB são suficientes para você trabalhar com vários recursos e aplicações. Essa é apenas uma dica! Mas como essa escolha pode influenciar no meu desempenho profissional? Pois é, compreender o mecanismo de funcionamento desses recursos é importante para todos os tipos de usuários.

Nesse contexto, vamos estabelecer, de acordo com Stuart (2011), uma hierarquia utilizada pela maioria das arquiteturas de computadores quando se trata de gerenciamento de memória, pelo sistema operacional. Vejamos na Tabela 4.1, a seguir, como é que são considerados os tipos de armazenamento. Esses estão divididos em: permanente, secundário, principal, cache e registradores, hierarquizados de acordo com a sua eficiência, porte e valor de mercado. Observe:

Tabela 4.1 | Classificação de hierarquia de memória

Hierarquia de memória
Registradores
Cache
Memória principal
Memória secundária
Memória permanente

Fonte: Adaptado de Stuart (2011, p. 218).

Vamos iniciar a descrição da hierarquia especificada com a descrição dos registradores. Eles fazem parte da unidade central de processamento (CPU) e são rápidos, pois armazenam a informação ou instrução de processo que deverá ser executada imediatamente.

Já a memória cache é considerada mais lenta que os registradores, porém mais rápida que a memória principal. Ela existe, pois são necessários alguns mecanismos de armazenamento, para permitir que a CPU acesse e realize o processamento da instrução de processo mais rapidamente do que se tivesse que acessar determinado dado ou instrução diretamente em uma das outras memórias: principal, secundária ou permanente. Isso porque o modo como se realiza a busca da informação é diferenciado e cada uma possui uma função diante do contexto de gerenciamento de memória do sistema operacional, sendo que a memória cache ganha cada vez mais espaço para

armazenamento de acordo com as novas arquiteturas de computadores e em função do crescimento das aplicações e necessidades de software (STUART, 2011).

Seguindo a sequência apresentada na Tabela 1, temos a memória principal, conhecida também como de armazenamento primário, pois é nela que se concentra o gerenciamento de memória realizado pelo sistema operacional, ou seja, nela se concentram as aplicações que serão diretamente solicitadas pelo processador através do mecanismo de endereçamento. Não é considerada muito rápida, porém pode ser de capacidade de armazenamento relevante na ordem de alguns gigabytes. Como exemplo, podemos citar a memória RAM (Random Access Memory) e a ROM (Ready Only Memory).

A memória RAM permite apenas as operações de leitura e escrita. É volátil, ou seja, assim que o computador é desligado, ela perde todas as instruções ali carregadas e disponibilizadas para acesso rápido pelos registradores e processos de programas em execução. Na ROM, ficam gravadas as especificações do sistema e que, por padrão, não devem ser alteradas. Por esse motivo, a memória ROM fica disponível apenas para leitura.

Na sequência, temos a memória secundária, que é responsável pelo armazenamento permanente de dados e não permite endereçamento, como, por exemplo, os pen drives, os CDs e DVDs. É considerada mais lenta em função de não permitir endereçamento e ter de ser localizada por um sistema de arquivos, que é tratado de forma diferenciada e separado do sistema operacional.

Além dessas, para finalizar o quesito hierarquia, temos a memória permanente. Essa permite que se estabeleça em uma política de segurança da informação, que seja realizado o backup dos dados, por exemplo, como acontece em um datacenter ou centro de dados, em que há grande quantidade de dados e, se não tivermos uma política de segurança da informação bem definida, colocamos em risco as operações que dependem dessas informações.

A memória permanente garante o acesso à informação que foi trabalhada, sendo uma função externa às de responsabilidade do sistema operacional. Portanto, quando analisamos a hierarquia, temos a definição da mais rápida para as mais lentas, no entanto, com alta capacidade de armazenamento.

Agora que já identificamos como o sistema operacional reconhece a informação de cada tipo de memória segundo a sua hierarquia, vamos conhecer também os conceitos relacionados à memória lógica e física e em que essas se diferem.

Nesse sentido, temos a definição da memória lógica atrelada ao endereçamento realizado, ou seja, a busca da informação pelo processo se dá através do endereço lógico que uma instrução ocupa em memória. Estamos nos referindo às instruções de máquina em que há a manipulação de dados contidos nas variáveis indicadas nas aplicações (OLIVEIRA et. al, 2011).

Quando falamos em memória física, é a referência que se faz ao tipo de memória em que será armazenada a informação. No caso, podemos definir de acordo com a hierarquia especificada anteriormente qual delas será melhor utilizar e de acordo com o tipo de aplicação.

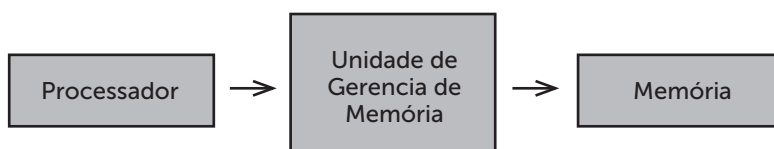
Temos, nesse contexto, a unidade de gerenciamento de memória, também conhecida como MMU (Memory Management Unit), cuja função é a de mapear os endereços lógicos em que estão as instruções nas memórias físicas.



Assimile

Observe na figura 4.1 qual é o fluxo que envolve o processador, a unidade de gerência de memória e o dado armazenado na memória:

Figura 4.1 | Fluxo de gerenciamento de recursos de memória e de processador

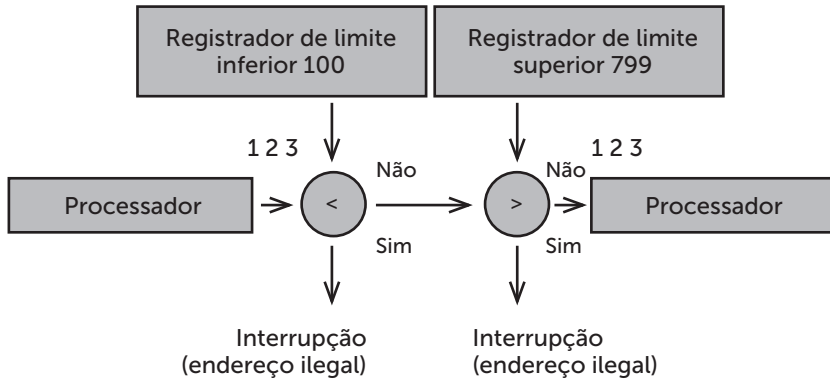


Fonte: Adaptado de Oliveira et. al. (2010, p. 155).

A seguir, será descrito o processo de acesso ao endereço lógico que é gerado pelo processo e com isso a MMU direcionará o endereço lógico para o mesmo correspondente na memória física. Uma observação interessante é que, quando o próprio processo gera o endereço lógico e físico, eles são idênticos tanto na memória lógica, responsável pelos endereçamentos, quanto na memória física que é correspondente à memória principal. Para realizar o gerenciamento de memória a MMU de um processo de usuário, que é limitado, são utilizados dois registradores de limite.

De acordo com o processo, o conteúdo de um registrador de limite passa a ser o parâmetro para que o processo defina o espaço que esse deverá ocupar na memória lógica. Para o exemplo dado por Oliveira et. al (2011), foi utilizado o limite que varia entre 100 a 799 para a alocação em memória. Na Figura 4.2, temos exemplificado o mecanismo de proteção da memória com o uso de registradores de limite. Tudo o que estiver fora desse intervalo será desconsiderado. Observe a seguir:

Figura 4.2 | Proteção da memória com dois registradores de limite



Fonte: Adaptado de Oliveira et. al. (2010, p. 155).



Pesquise mais

Vale a pena assistir a aula sobre tipos de memória e suas funções. Disponível em: <<https://www.youtube.com/watch?v=r-Ca80kDteA>>. Acesso em: 28 set. 2015.

É também de responsabilidade da gerência de memória manter em memória física ou principal a maior quantidade de processos residentes de forma que seja possível aproveitar ao máximo o compartilhamento de recursos, como, por exemplo, de processamento, ou seja, permitir que um número maior de processos leia as instruções e as execute (MACHADO; MAIA, 2013). As Figuras 1 e 2 visam demonstrar como ocorre a proteção das áreas de memória em que os processos trocam informações de endereçamento e alocação de forma protegida.

Além dos aspectos considerados, também precisamos aprender como acontece a divisão da memória em partes, processo esse também conhecido como particionamento, ou, ainda, alocação particionada.

Há três tipos de alocação de memória: alocação contígua simples, a técnica de overlay ou divisão da aplicação em módulos de forma a alocar de acordo com os espaços que cada um deles ocupará, e técnica de particionamento, como mencionado.

A alocação contígua simples é mais voltada à realidade dos primeiros sistemas operacionais que eram monoprogramáveis, sendo a memória principal dividida em duas grandes áreas: uma para alocar o sistema operacional e a outra para alocar as aplicações do usuário. Dessa forma, o desenvolvimento dessas aplicações de usuários tinha de respeitar as limitações da área de alocação das aplicações de usuário predeterminadas.

Já a técnica de overlay considera que, diante de uma aplicação, a divisão de módulos auxiliará na determinação do espaço de memória necessária a executar os módulos de forma independente. Isso quer dizer que é também no desenvolvimento da aplicação que deverá ser especificado o tamanho do módulo e como será a sua alocação de forma a permitir a execução de um módulo por vez.

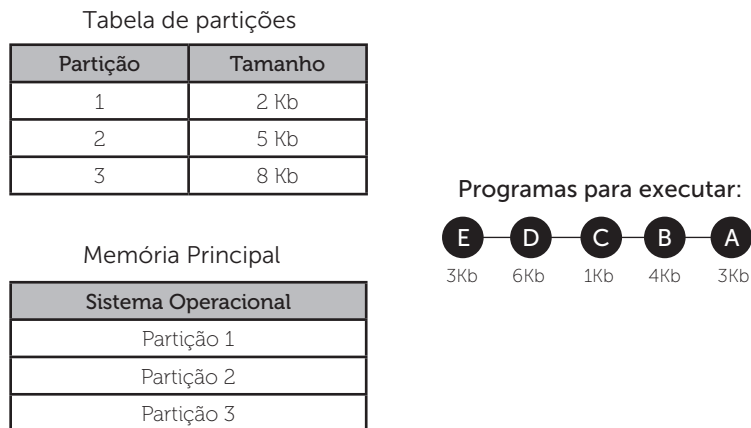
E a alocação particionada permite um maior aproveitamento dos espaços de memória principal. Nesse contexto, vamos conhecer os tipos de alocação de memória particionada existentes. São elas: estática ou dinâmica.



Exemplificando

Nos primeiros sistemas operacionais multiprogramáveis, as partições precisavam ter tamanho fixo, definido de acordo com as necessidades dos sistemas que seriam utilizados. Quando existia a necessidade de uma das partes ser desabilitada ou redimensionada, era preciso reconfigurar os sistemas e instalá-los de acordo com o tamanho das demais partições. A esse tipo de gerência de memória atribuiu-se o nome de alocação particionada estática ou fixa (MACHADO; MAIA, 2013). Observe na Figura 4.3 um exemplo desse mecanismo de gerência de memória:

Figura 4.3 | Alocação particionada estática



Fonte: Machado; Maia, (2013, p. 149).

Quando trabalhamos sob a perspectiva da alocação particionada dinâmica, eliminamos a obrigatoriedade de definição do tamanho das partições, de acordo com o tamanho dos programas que serão instalados. Então o que temos nesse tipo de particionamento? Como o sistema operacional disponibiliza os recursos?

Simple! Em alocação particionada dinâmica não temos mais a necessidade de definir o tamanho das partições, pois aos programas será disponibilizado apenas o espaço em memória que seja o suficiente para o seu armazenamento.

Com isso, há a otimização do recurso de alocação e, ainda, há a redução dos espaços vazios, ou seja, que porventura não estejam sendo utilizados por uma aplicação e necessite desfragmentar o disco para realocar e organizar a memória novamente. Esse é um problema que se resolve com o que chamamos de fragmentação externa.

Quando acontece de, nas alocações, sobrar em memória uma determinada quantidade que é considerada relativamente baixa para armazenar outra aplicação, será necessário criar uma área com aquele tamanho que sobrou, mesmo que pequeno, para armazenar outro tipo de informação ou aplicação que requeira menos espaço. Esse procedimento refere-se à fragmentação eterna.

Outra forma é a realocação dinâmica dos espaços disponíveis, ou seja, todos os pequenos espaços são organizados de forma contígua, o que libera mais espaços para armazenamento de outras aplicações.



Faça você mesmo

Verifique em seu computador quantas partições foram criadas e qual é o espaço disponível em disco. Realize o procedimento de desfragmentação, disponível no painel de controle e analise, em termos de eficiência e rendimento da máquina, se está mais rápida.

Atenção:

Devemos ter o cuidado em desfragmentar o SSD, por ser um disco de estrutura sólida e não utilizar leitura mecânica e sim processos elétricos, podendo comprometer a vida útil do equipamento. Então, toda atenção ao realizar os testes!

Sem medo de errar

Com o objetivo de auxiliar no desenvolvimento de uma solução de software que venha colaborar com a diminuição do gargalo existente hoje no que tange ao espaço de memória nos dispositivos móveis, vamos apresentar uma proposta que vem sendo utilizada pelo mercado e que nem todo o público-alvo desses equipamentos sabe que é uma ferramenta de apoio bastante útil.

A empresa de pesquisa de desenvolvimento de novos produtos mencionada está

trazendo alternativas já existentes para auxiliar tanto nas frentes de desempenho do produto, fabricação, diminuição de resíduos e descartes desnecessários, além de redução de custos para fabricantes e parceiros, que podem usufruir das soluções existentes no mercado, de forma a incentivar também as pequenas operações e empresas de produtos de software. Então, conheça o procedimento recomendado para identificar qual é o aplicativo que será capaz de auxiliar nessa tarefa de maximizar a gerência de memória de um dispositivo móvel.

1. Identifique a marca, o modelo e o sistema operacional do dispositivo móvel para descobrir qual tipo de aplicativo temos disponível e qual o usuário principal ou root do aparelho caso esse não esteja liberado.

2. Vamos limpar a memória cache do aparelho com o software chamado Cache Cleaner. Serão removidos todos os arquivos temporários do seu celular e esse não precisa necessariamente estar rooteado.

3. Outro software que pode ser elencado que auxilia na liberação de espaços de memória é o Link2SD. Esta é uma ferramenta que remove os aplicativos que já vêm instalados na memória interna do celular para o cartão de memória. No lugar do arquivo, fica um link, como um atalho que aponta para o endereço do aplicativo no cartão de memória externo. Quando instalado no dispositivo, o próprio aplicativo auxilia na identificação dos aplicativos padrão e indica o passo a passo para realizar a transferência.

4. Além dessas duas ferramentas, o backup de dados do aparelho móvel é sempre recomendado.

5. Quanto à desinstalação de arquivos possivelmente ocultos no dispositivo móvel, é possível utilizar para essa tarefa um aplicativo conhecido como Easy Uninstaller, que permite desinstalar vários arquivos a partir de uma seleção, em lote.

6. Outra opção é sempre salvar os seus arquivos como fotos e vídeos no cartão de memória externa, pois facilita o gerenciamento e a otimização da memória interna do dispositivo.



Atenção!

Leia o artigo que traz na íntegra o passo a passo e as melhores práticas para melhorar o desempenho de seu dispositivo móvel no que tange aos recursos de memória. Disponível em: <<http://imasters.com.br/mobile/android/espaco-precioso-economize-espaco-em-dispositivos-android/?t=1519021197&source=single>>. Acesso em: 25 set. 2015.



Lembre-se

A melhor estratégia a ser adotada por um sistema depende de uma série de fatores, sendo o mais importante o tamanho dos programas processados no ambiente. Independentemente do algoritmo utilizado, o sistema possui uma lista de áreas livres, com o endereço e tamanho de cada área (MACHADO; MAIA, 2013, p. 154).

Avançando na prática

Pratique mais	
Instrução Desafiamos você a praticar o que aprendeu, transferindo seus conhecimentos para novas situações que pode encontrar no ambiente de trabalho. Realize as atividades e depois compare-as com a de seus colegas.	
Gerenciamento de memória: conceitos, tipos, características e virtualização	
1. Competência de fundamento de área	O aluno deverá ser capaz de identificar quais são as principais funções de um sistema operacional, bem como ter conhecimento sobre como se dá o compartilhamento de recursos e a sua gerência.
2. Objetivos de aprendizagem	Conhecer os principais conceitos, características da gerência de memória e, o que é e a virtualização de arquivos.
3. Conteúdos relacionados	Gerenciamento de memória: conceitos, tipos, características e virtualização.
4. Descrição da SP	A empresa de pesquisa e desenvolvimento precisa entregar um relatório sobre ações do sistema operacional para realizar a alocação dinâmica em servidores virtualizados. Então, siga as instruções de um dos fornecedores de sistemas operacionais para iniciar as comparações:
5. Resolução da SP	<p>Quando se trata de virtualização de servidores, temos um gerenciador de tarefas de distribuição de recursos entre as máquinas virtuais criadas. Então, seguindo essa premissa, precisamos verificar de que forma acontece essa gerência no que tange aos recursos de memória e de processamento. Para tanto, veja as instruções do fornecedor quando o assunto é alocação dinâmica de recursos em servidores, a considerar o sistema operacional Windows Server:</p> <p>O passo a passo será o seguinte:</p> <ol style="list-style-type: none"> 1. Iniciar os recursos e adicioná-los ao pool de recursos disponíveis. 2. Notificar os dispositivos e aplicativos registrados sobre os recursos iniciados, de forma que estes dispositivos e aplicativos possam ajustar suas alocações de recursos. 3. No caso dos processadores: iniciar um balanceamento de recursos do sistema de device drivers participantes, para que possam desconectar e reconectar seus manipuladores de interrupção de hardware, e assim fazer uso dos novos recursos.

	<p>Os device drivers não participantes não estão incluídos, de forma que a realocação não interrompe seus serviços.</p> <p>4. Nos casos das pontes de hospedeiros de E/S: verificar os dispositivos conectados ao novo barramento de E/S e, se necessário, iniciar um novo balanceamento de recursos.</p> <p>5. Na conclusão dessas etapas, notificar o processador de serviços que a inclusão automática foi concluída.</p> <p>(Fonte: STANEK, Willian. Virtualização de servidores: particione e conquiste. Technet Microsoft. Disponível em: <https://technet.microsoft.com/pt-br/magazine/gg598495.aspx>. Acesso em: 30 set. 2015.</p> <p>Em seguida, teremos de saber também, como acontece com a adição de recursos. Temos também a possibilidade de realizar a substituição. Observe as orientações do mecanismo de funcionamento desse procedimento segundo o fornecedor do software:</p> <p>"A substituição automática está disponível apenas para memória e processadores (e apenas quando o recurso de substituição é idêntico ao recurso original). O processador de serviços controla uma operação de substituição por meio de:</p> <ol style="list-style-type: none"> 1. Seleção dos recursos sobressalentes disponíveis e necessários. 2. Ativação e inicialização dos recursos. Com memória, o estado dos módulos de memória antigos é copiado para os novos módulos de memória. 3. Notificação ao Windows Server 2008 sobre a iminente operação de substituição. O sistema operacional entra em um estado de suspensão pseudo-S4. Com processadores, o sistema operacional e o firmware do sistema copiam o estado dos processadores antigos para os novos. Com memória, todas as alterações de estado são copiadas para os novos módulos de memória. 4. Mapeamento dos recursos de substituição para a partição de hardware e substituição dos recursos antigos. 5. Notificação ao Windows Server 2008 sobre a conclusão da substituição. O sistema operacional sai do estado de suspensão e retorna as operações regulares. 6. Desativação dos recursos antigos e notificação ao gerenciador de serviços e ao aplicativo de gerenciamento do sistema que a substituição está concluída." <p>(Fonte: STANEK, Willian. Virtualização de servidores: particione e conquiste. Technet Microsoft. Disponível em: <https://technet.microsoft.com/pt-br/magazine/gg598495.aspx>. Acesso em: 30 set. 2015.</p> <p>Com isso, temos também de avaliar a possibilidade implantar essa solução de virtualização ou se mantém os melhoramentos previstos para os servidores locais.</p>
--	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------



Lembre-se

A primeira implementação de memória virtual foi realizada no início da década de 1960, no sistema Atlas, desenvolvido na Universidade de Manchester. Posteriormente, a IBM introduziria este conceito comercialmente na família System/370, em 1972. Atualmente, a maioria

dos sistemas implementa memória virtual, com exceção de alguns sistemas operacionais de supercomputadores (MACHADO; MAIA, 2013, p. 159).



Faça você mesmo

Leia e aprenda mais sobre virtualização: Disponível em: <<https://technet.microsoft.com/pt-br/magazine/gg598495.aspx>>. Acesso em: 29 set. 2015.

Faça valer a pena

1. Complete a frase de acordo com a alternativa que contém os respectivos conceitos apresentados:

"[...] a _____, também conhecida como _____, que tem por função _____ lógicos em que estão as instruções nas memórias físicas."

- a) unidade central de processamento/CPU (Central Processing Unit)/ controlar processos.
- b) unidade de gerenciamento de memória/MMU (Memory Management Unit)/mapear os endereços
- c) central de distribuição de memória/receptores de sinais/conectar os operadores.
- d) inserção de memória/recursos de alocação de memória virtual/ configurar os registradores.
- e) segurança dos dados/unidade central de processamento/registra os dados.

2. O trecho do texto abaixo indica qual tipo de memória? Assinale a alternativa correspondente:

"É considerada mais lenta que os registradores, porém, mais rápida que a memória principal. Ela existe, pois são necessários alguns mecanismos de armazenamento, para permitir que a CPU acesse e realize o processamento da instrução de processo mais rapidamente do que se tivesse que acessar determinado dado ou instrução diretamente em uma das outras memórias: principal, secundária ou permanente. Isso porque o modo como se realiza a busca da informação é diferenciado e cada uma possui uma função diante do contexto de gerenciamento de memória do sistema operacional, sendo que a memória cache ganha cada vez mais espaço para armazenamento de acordo com as novas arquiteturas de

computadores e em função do crescimento das aplicações e necessidades de software” (STUART, 2011).

- a) Memória secundária
- b) Memória principal
- c) Memória cache
- d) Memória externa
- e) Registradores

3. Assinale a alternativa que contém uma das principais responsabilidades da gerência de memória:

- a) Manter em memória física ou principal, a maior quantidade de processos residentes de forma que seja possível aproveitar ao máximo o compartilhamento de recursos.
- b) Permitir que um pequeno número de processos leia as instruções e as execute de acordo com a sua ordem na fila.
- c) Buscar a informação do processo nos registradores.
- d) Compartilhar a alocação dinâmica com os processadores.
- e) Executar instruções contidas nas memórias cache e registradores.

4. Assinale a alternativa que contém os tipos de particionamento que podem ser realizados:

- a) Estático e rotativo
- b) Dinâmico e rotativo
- c) Estático e predominante
- d) Estático e dinâmico
- e) Dinâmico e alternativo

5.5. Assinale a alternativa que contém a descrição da técnica de overlay:

- a) Determina o tamanho fixo de uma partição.
- b) Especifica o tamanho do módulo e como será a sua alocação de forma a permitir a execução de todos os módulos de uma vez.
- c) Delimita e compartilha os recursos de uma partição com outros sistemas operacionais.

- d) Estabelece um limite de memória e libera após a execução da instrução.
- e) Considera que, diante de uma aplicação, a divisão de módulos auxiliará na determinação do espaço de memória necessária a executar os módulos de forma independente.

6. Explique o conceito de alocação particionada estática.

7. Explique o conceito de alocação particionada dinâmica.

Seção 4.2

Swapping: conceitos, tipos e suas características

Diálogo aberto

Nesta aula, vamos estudar um conceito que está diretamente relacionado aos espaços reservados em memória principal para alocar processos residentes e em um determinado momento precisam ceder esse endereço ou posição de memória a outros processos não residentes, e como ocorre esse mecanismo de gerência de memória.

Nesse contexto, quando falamos em técnicas e mecanismos de gerência de memória, é importante conhecer o que é swapping.

Além disso, conhecer as formas de se ampliar a eficiência da multiprogramação em sistemas operacionais é como caminhar para identificar novas formas de compartilhamento de recursos e, ainda, visualizar outras possibilidades de desenvolvimento de soluções.

Mas você pode se perguntar: afinal, esse procedimento ainda ocorre nos sistemas operacionais atuais? Sim, e é cada vez mais comum inclusive para a gerência de memórias virtuais.

Com esta aula, vamos aprofundar os conhecimentos sobre esse procedimento que o sistema operacional realiza para gerenciar a memória, quando ocorre e porque motivos ele foi desenvolvido.

Além disso, precisamos estreitar nossos estudos junto ao mercado de trabalho e, dessa forma, diante do cenário de uma empresa de pesquisa e desenvolvimento, vamos propor um levantamento de técnicas para melhorar o nível de multiprogramação utilizando swapping em memórias virtuais.

Não se esqueça de visitar o seu material didático para esclarecer dúvidas de assuntos já abordados, tais como: gerência de processos, sincronização, diretórios e sistemas de arquivos.

Agora que você já sabe o que precisa pesquisar e o que vai aprender, mãos à obra!

Bons estudos e práticas a você!

Não pode faltar

Logo no início desta aula, vimos que estudaremos os conceitos relacionados à gerência de memória e que o foco está em compreender o que é swapping. Então, siga em frente e conheça mais uma técnica, utilizada em multiprogramação para otimizar recursos e, com isso, ampliar a eficiência do seu computador.

A técnica de swapping foi desenvolvida com o intuito de solucionar um problema comum em multiprogramação, que é a falta de espaço na memória principal. Então, ela propõe que, ao invés de um processo residente em memória principal, esse seja enviado por tempo determinado para a memória secundária, para dar espaço suficiente para que um processo não residente seja alocado e, com isso, após a sua execução, o espaço volta a ser liberado para que aquele processo residente retorne ao endereço de origem.



Assimile

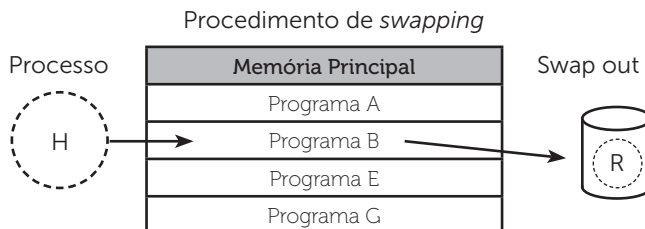
Nesta situação, o sistema escolhe um processo residente, que é transferido da memória principal para a memória secundária (swap out), geralmente o disco. Posteriormente, o processo é carregado de volta da memória secundária para a memória principal (swap in) e pode continuar sua execução como se nada tivesse ocorrido (MACHADO; MAIA, 2013, p. 156).

Observe no exemplo abaixo qual é a lógica dessa técnica e como o sistema operacional realiza essa gerência:



Exemplificando

Figura 4.4 | Técnica de *swapping*



Fonte: Adaptado de Machado; Maia (2013, p. 156).

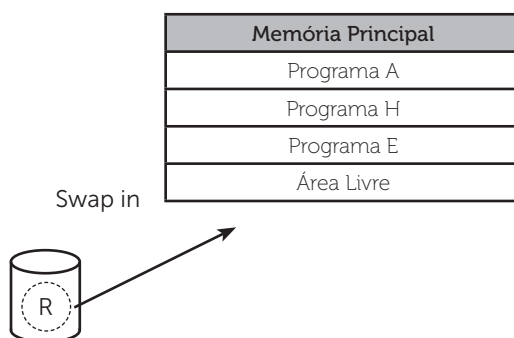
O que a Figura 4.4 apresenta é a identificação de um processo não residente em memória principal, que necessita de mais espaço para ser alocado e entrar em execução. Para que essa concessão aconteça, é

necessário que seja aplicado um algoritmo que verifique qual processo residente está em estado de espera e não será, portanto, alocado em seguida para execução. Com isso, é possível ceder o seu espaço ou endereço de memória, para que um outro processo utilize aquele recurso e entre efetivamente em execução.

Quando o processo escolhido para ser retirado da memória principal está em estado de espera ou de pronto, podendo ser considerado não residente, recebem o nome de *outswapped*.

Em seguida, quando o processo não residente alocado encerra a sua execução, ele é retirado da memória principal retorna ao local de origem.

Figura 4.5 | Processo de entrada de arquivo em *swap in*



Fonte: Machado; Maia (2013, p. 156).

A Figura 4.5 traz o procedimento de devolução do processo residente que estava alocado em disco, ou seja, memória secundária, para a área da memória principal que se encontra liberada.

A técnica de swapping somente poderá ser implementada a partir do uso de um registrador de alocação. Esse registrador de alocação é acionado no momento em que o programa ou aplicativo em uso é carregado e recebe o endereço inicial de alocação que foi disponibilizado para ele em memória.

A cada nova solicitação de memória, será somado ao conteúdo desse registrador e com isso, é gerado o seu endereço físico. Isso permite que um determinado programa possa ser carregado em qualquer posição de memória (MACHADO; MAIA, 2013).



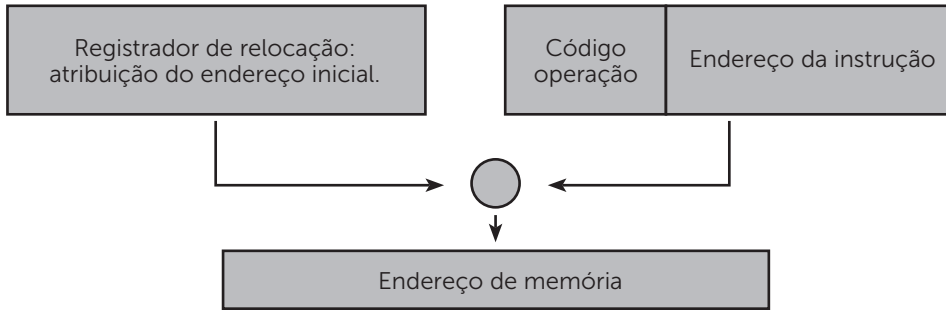
Pesquise mais

Aqui fica uma indicação de palestra sobre implantação de técnicas de gerência de memória virtual em softwares livres. Disponível em: <<https://>

www.youtube.com/watch?v=clU0oVnWvcU>. Acesso em: 2 out. 2015.

Na Figura 4.6, temos um exemplo do mecanismo de carregamento de um programa e a sua respectiva referência no registrador de relocação.

Figura 4.6 | Definição de endereço inicial do processo e endereço de memória



Fonte: Machado; Maia (2013, p. 157).

Mas, afinal, qual é a vantagem de se aplicar a técnica de swapping?

Vamos listar abaixo algumas delas, confira:

a. Ao se realizar swapping, é possível compartilhar mais recursos ou endereços na memória principal.

b. Com melhor aproveitamento da memória principal, conseguimos também, otimizar o processamento e maximizar a utilização dos recursos da máquina de modo geral.

No entanto, alguns pontos de atenção também podem ser listados como, por exemplo, o custo de processador com as operações de entrada e saída (swap out e swap in) junto à memória principal. Ou seja, é possível perder um pouco de eficiência quando se realizam essas operações, pois os recursos podem estar dedicados justamente ao swapping e, com isso, não ser alocados recursos suficientes para que os processos residentes na memória principal entrem em execução.

Quando temos esse tipo de situação acontecendo em gerência de memória, temos o que se chama de thrashing, que é um dos problemas mais graves que podem ocorrer e comprometer a eficiência da máquina.



Refleta

Mesmo com o aumento da eficiência da multiprogramação e, particularmente, da gerência de memória, muitas vezes um programa não podia ser executado por falta de uma partição livre disponível. A técnica de

swapping foi introduzida para contornar o problema da insuficiência de memória principal (MACHADO; MAIA, 2013, p. 156).

Outros fatores como não realizar swapping de código de processos devem ser levados em consideração, pois imagine se dois processos estejam dividindo o mesmo espaço de memória por swap, como deve ser tratado esse problema? A preferência fica em realizar swap apenas de dados e de elementos da pilha entre os processos que estão compartilhando do mesmo espaço em memória principal e deixar os segmentos em que estão alocados os códigos intactos até que seja finalizada a execução do processo.

Então, é importante reforçar a ideia de que não há vantagens em se aplicar a técnica de swapping quando o espaço liberado em memória não for o suficiente para alocar todos os dados necessários de um determinado processo.

Relembrando que o hardware responsável por realizar essa tarefa de seleção do processo que poderá realizar swapping é o de gerência de memória. Se a MMU conseguir decodificar uma quantidade relativamente grande de endereços, isso impacta em um ganho de flexibilidade tanto no que tange à escolha do processo que entrará em swapping quanto na velocidade. Pois se há a necessidade de alocar apenas parte do processo, como descrito no parágrafo anterior, temos menores tempos de leitura e gravação de dados (STUART, 2011).

E quais são os tipos de swapping utilizados? Abaixo, vamos listar os respectivos tipos de se organizar e estabelecer uma política de realização de swapping:

- Swapping por paginação ou paginação de swapping: a técnica de paginação é realizada pela unidade de gerência de memória. Com isso, torna-se mais viável utilizá-la, pois as páginas são de tamanhos relativamente pequenos e independente dessa característica, e os segmentos ou quadros de página têm o mesmo tamanho. Isso permite que sejam copiadas páginas de forma individual para a memória principal e memória secundária ou disco. É possível, através de swapping de paginação, fazer swap de processos inteiros (STUART, 2011). Para que a página seja devidamente identificada, é utilizado o bit de presença "P". Esse bit de presença verifica a existência da página em memória principal ou secundária.
- Strings de referência: lista de referência à sequência ordenada de acessos a página para leitura ou gravação.
- Políticas de substituição global (em que os processos podem solicitar mais páginas do que aquelas que a MMU alocou para swap) e local (em que o swap pode acontecer apenas entre as páginas que o processo possui).
- Mínimo de Belady: política de swap criada por Laszlo Belady, em que afirma que uma página que ainda não tenha sido utilizada será a mais viável escolha para se

realizar swap para memória secundária.

- FIFO: primeira página a entrar primeira a sair.
- Segunda chance: extensão de FIFO com uma verificação. Se o bit de acesso for "0", a página será enviada por swap para a memória secundária. Se o bit de acesso for "1", então o bit de acesso é deletado e a página reinserida ao final da fila.
- Algoritmo do relógio: infere em mais uma análise sob a política da segunda chance, pois organiza, a partir da ideia de ponteiros do relógio, as páginas ao seu redor e, então, se for preciso realizar swap em memória secundária, teremos aquela página que está sendo apontada para verificação do bit de acesso.
- Conjunto de trabalho: define-se como o conjunto de páginas que o processo utiliza em um determinado intervalo de tempo.
- Frequência de falha: a partir das falhas de paginação entre os processos, a tendência nessa política é verificar se há páginas sem uso ou com pouco uso. Se houver, será feito swap para memória secundária das páginas sem tanta utilidade, pertencentes ao processo que apresentou mais páginas residentes em memória principal do que em seu conjunto de trabalho.
- Segmentos paginados: representam os segmentos dos espaços de endereçamento e, se esses forem relativamente grandes, há a sua paginação, a fim de torna-lo menor.
- Arquivos mapeados na memória: uma vez um arquivo mapeado, este será armazenado. A alocação de páginas para um determinado processo ocorre toda vez em que há uma solicitação de mapeamento de um arquivo em memória.
- Cópia-na-escrita: por exemplo a fork (UNIX), em que cria um processo-filho que é a cópia do processo-pai, porém o que há em termos de swap é a cópia no mesmo segmento de memória, tanto do processo-filho quanto do processo-pai de seu mapeamento de endereço em memória virtual, sendo apenas liberados para leitura. Caso aconteça a tentativa de gravação, há uma interrupção e será registrada pela MMU.
- Desempenho: faz referência direta a eficiência e cuidados na escolha da política de substituição local e global. São considerados os fatores:
 - Tempos de acesso
 - Thrashing: como mencionado anteriormente



Faça você mesmo

Analise o exemplo:

Tempo médio de acesso:

Para colocar isso em perspectiva, considere o código executável em linha reta, ignorando quaisquer acessos a dados. Se cada página tiver 4096 bytes e cada instrução, 4 bytes, cruzaremos um limite de página a cada 1024 instruções. Isso implica que $f = 0,000976$ se nenhuma página estiver carregada. Se o tempo de acesso a memória for igual a 50nS (nanossegundos) e o tempo de acesso ao disco for igual a 10mS (milionésimos de segundos), então o tempo médio de acesso (t) será de 9760 nS. O efeito final é que a execução dessas instruções será 200 vezes mais lenta do que seria se todas as instruções estivessem residentes na memória (STUART, 2011, p. 252).

Nesta mesma referência, encontram-se outros exemplos no que tange à aplicação dos tipos de swapping listados acima e de que forma acontecem. Elabore um relatório com as diferenças e investigue quais delas os sistemas operacionais mais utilizam.

Sem medo de errar

Os procedimentos a seguir são sugeridos para adicionar uma memória virtual swap em um servidor Debian 7.0 Wheezy. Confira o tutorial sugerido no site Fator Binário, que traz informações sobre a configuração dessa memória. Vamos lá!

1. Verifique se a memória de troca está instalada:
 - `swapon -s`: mostrará uma tabela em branco
 - `free -m`: mostrará todos os valores zerados na linha "Swap:"
2. Verifique também o espaço livre no HD do servidor:
 - `df -h`
3. Vamos criar uma partição Swap em disco de 2G:
 - `fallocate -l 2G /swapfile`
4. Verifique se a partição Swap foi criada corretamente:
 - `ls -lh /swapfile`

Esse comando resultará a seguinte instrução:

- `-rw-r--r-- 1 root root 2.0G Apr 3 14:19 /swapfile`

5. Definir as permissões do arquivo e habilitações de sistema com a respectiva partição para a memória Swap:

- `chmod 600 /swapfile`
- `mkswap /swapfile`
- `swapon /swapfile`

6. Se quiser deixar a partição como permanente, será preciso editar o arquivo de nome:

• `/etc/fstab`: com a seguinte instrução para inserção de uma nova linha e salve em seguida:

- `/swapfile none swap sw 0 0`

7. Utilize o comando abaixo para evitar alguns erros de inicialização do servidor:

- `mount -a`

8. Você pode usar o seguinte comando para verificar a quantidade de memória em uso:

- `watch -n 5 free -m` : (o parâmetro `-n 5` define uma nova leitura a cada 5 segundos):



Atenção!

Leia o artigo na íntegra e aprenda mais! Disponível em: <<http://fatorbinario.com/linux-adicionando-memoria-virtual-swap-ao-vps/>>. Acesso em: 2 out. 2015.



Lembre-se

Para que a técnica de swapping seja implementada é essencial que o sistema loader que implementa a relocação dinâmica de programas. Um loader relocável que não ofereça esta facilidade permite que um programa seja colocado em qualquer posição de memória, porém, a relocação é realizada apenas no momento do carregamento (MACHADO; MAIA, 2013, p. 157).

Avançando na prática

Pratique mais	
Instrução Desafiamos você a praticar o que aprendeu, transferindo seus conhecimentos para novas situações que pode encontrar no ambiente de trabalho. Realize as atividades e depois compare-as com a de seus colegas.	
Swapping: conceitos, tipos e suas características	
1. Competência de fundamentos de área	O aluno deverá ser capaz de identificar quais são as principais funções de um sistema operacional, bem como ter conhecimento sobre como se dá o compartilhamento de recursos e a sua gerência.
2. Objetivos de aprendizagem	Conhecer o que é swapping, como o sistema operacional gerencia esta tarefa e aloca os devidos recursos, a fim de atender às demandas de processamento e alocação.
3. Conteúdos relacionados	Swapping: conceitos, tipos e suas características.
4. Descrição da SP	É uma situação comum que os usuários de computadores queiram destinar uma das partições de sua máquina a outro sistema operacional. Com isso, é preciso definir manualmente, muitas vezes, o espaço que será reservado às operações de swapping. Sua tarefa é investigar, de acordo com um fornecedor de sistemas operacionais, quais os procedimentos que podem ser aplicados. Siga em frente!
5. Resolução da SP	<p>Vamos instalar o Ubuntu em um notebook que já tem instalado o Windows e será preciso, portanto, fazer o que se chama de Dual Boot.</p> <p>Para isso, será preciso criar manualmente as partições necessárias para cada operação, como, por exemplo, a memória em que serão trocados arquivos (Swap) e a memória em que serão armazenados.</p> <p>Você pode seguir o passo a passo recomendado no artigo indicado no site "Mundo Ubuntu", que contém as informações detalhadas para a realização desse procedimento.</p> <p>Essa é uma prática interessante e te ajuda a fixar os conceitos estudados nesta aula.</p> <p>Siga as recomendações para:</p> <p>"Veja como criar um pendrive de boot com o Ubuntu 12.04. Leia esta matéria caso seu computador não esteja reconhecendo o boot pelo pendrive (boot pela USB)."</p> <p>E teste os procedimentos indicados!</p> <p>Fonte: Instalação do Ubuntu12.04 mostrando manualmente a criação de partições swap em dual boot com windows. Disponível em: <http://www.mundoubuntu.com.br/tutoriais/instalacao/77-instalacao-do-ubuntu-12-04-mostrando-manualmente-a-criacao-das-particoes-swap-e-home-em-dual-boot-com-o-windows-7>. Acesso em: 2 out. 2015.</p>



Lembre-se

Os primeiros sistemas operacionais que implementaram esta técnica

surgiram na década de 1960, como o CTSS do MIT e OS/360 da IBM. Com a evolução dos sistemas operacionais, novos esquemas de gerência de memória passaram a incorporar a técnica de swapping, como a gerência da memória virtual (MACHADO; MAIA, 2013, p. 157).



Faça você mesmo

Agora que você já conheceu a técnica, porque foi desenvolvida e como está sendo aplicada atualmente, elabore um relatório que contemple outros sistemas operacionais e o passo a passo para definir o espaço para troca de arquivos em memória, swapping.

Faça valer a pena

1. O que é swapping? Assinale a alternativa correspondente:

- a) Processo de gerência de memória secundária pelo sistema de arquivos.
- b) Técnica que estabelece políticas de trocas de espaços de memória principal para secundária para alocação de processos.
- c) Técnica que realiza a sobreposição de processos na memória principal.
- d) Técnica que auxiliar no processamento de dados de pilha apenas.
- e) Procedimento de recuperação de arquivos em RAM.

2. Analise as afirmações e assinale a alternativa correspondente:

I. Swapping faz a verificação de disponibilidade de troca de um processo da memória principal para a memória secundária.

II. Através de um algoritmo que é responsável por verificar qual processo residente está estado de espera, um processo pode ceder o seu espaço para que um outro processo utilize aquele recurso.

III. Quando o processo escolhido para ser retirado da memória principal está em estado de espera ou de pronto, podem ser considerados não residentes e, como foram eles os selecionados, recebem o nome de outswapped.

- a) V, V, V.
- b) F, F, F.
- c) V, F, F.

- d) F, V, V.
- e) V, V, F.

3. Swap out refere-se à:

- a) operação de saída de um processo da memória secundária para os registradores;
- b) operação de entrada de um processo da memória principal para os registradores;
- c) operação de saída de um processo da memória principal para ser armazenado em disco ou memória secundária;
- d) operação de entrada de um processo em memória principal;
- e) operação de entrada de um processo em memória cache e saída para RAM.

4. Swap in refere-se a:

- a) operação de saída de um processo da memória secundária para os registradores;
- b) operação de entrada de um processo da memória principal para os registradores;
- c) operação de saída de um processo da memória principal para ser armazenado em disco ou memória secundária;
- d) operação de entrada de um processo que estava em memória secundária e é enviado para memória principal;
- e) operação de entrada de um processo em memória cache e saída para RAM.

5. O que é thrashing no que concerne o gerenciamento de memória? Descreva.

6. Descreva o swapping por paginação.

7. Assinale a alternativa que contém a descrição da política de swapping de segmento de paginação.

- a) Alocação de páginas para um determinado processo ocorre toda vez

em que há uma solicitação de mapeamento de um arquivo em memória.

b) Cópia no mesmo segmento de memória, tanto do processo-filho quanto do processo-pai de seu mapeamento de endereço em memória virtual, sendo apenas liberados para leitura.

c) Representam os segmentos dos espaços de endereçamento e, se esses forem relativamente grandes, há a sua paginação, a fim de torná-lo menor.

d) É o conjunto de páginas que o processo utiliza em um determinado intervalo de tempo.

e) Lista de referência à sequência ordenada de acessos a página para leitura ou gravação.

Seção 4.3

Memória virtual: conceitos, paginação, segmentação e virtualização

Diálogo aberto

Olá, aluno. Vamos à penúltima aula desta unidade curricular e com ela aprenderemos o que é memória virtual. Apesar de já termos tido contato com esse conceito em algumas de nossas atividades, vamos conhecer todos os relacionados.

Até aqui, estudamos nesta unidade o que é a gerência de memória, quais são os seus tipos, como é realizada a alocação de recursos e os tipos de particionamento. No caso, nós fizemos uma breve descrição dos particionamentos do tipo fixo e dinâmico.

Além disso, estudamos também a técnica de swapping. Essa técnica requer que se estabeleça uma política para a realização da troca de processos da memória principal para a memória secundária. Dessa forma, elencamos algumas, como, por exemplo, swapping por paginação, que é um conceito que será abordado também nesta seção, quando descrevermos o que significa para o sistema operacional separar em páginas a sua disponibilidade em memória para alocação de processos.

Nesse contexto, vamos estudar o que é memória virtual e para que serve. Diante dessa técnica, é possível estabelecer critérios de seleção e organização de processos para execução e aumento da velocidade de processamento.

É muito comum encontrarmos, por exemplo, um computador que esteja mais lento, com tempos de resposta mais longos. Nesse contexto, precisamos avaliar o que acontece com o hardware e o software para uma possível modificação.

No caso, supondo que a empresa de pesquisa e desenvolvimento está comprometida em ensinar mecanismos de otimização de memória virtual, vamos trabalhar para aumentar o espaço disponível a ela. A sua tarefa agora é auxiliar na investigação do processo necessário para inserir esse acréscimo de memória.

Ficam as seguintes recomendações: leia atentamente seu material didático, realize as atividades de aprendizagem propostas e acesse os links em destaque para ampliar o seu conhecimento acerca desse assunto.

Desde já, desejamos bons estudos e práticas a você!

Não pode faltar

Você sabe qual é a diferença entre a memória RAM, a técnica de swapping e a memória virtual? Então, este será nosso ponto de partida nessa aula. Siga em frente!

Vamos recapitular as respectivas definições conceituais. A primeira delas que devemos observar é a memória virtual. O conceito de memória virtual foi criado para solucionar alguns problemas como o da fragmentação do disco. A memória virtual também é útil quando se utiliza a técnica de overlay, em que apenas parte de um processo é dimensionada e alocada a uma área de memória principal, sendo este processo parcial dividido em módulos. Porém podem faltar recursos para alocar os processos, ainda que divididos em módulos, em um espaço que seja suficiente. Então, a gerência desse serviço é mais complexa.

A memória virtual se mostra como uma solução em função de trabalhar com uma quantidade maior de processos que compartilham a memória principal, uma vez que, destes processos já estão alocadas apenas as partes residentes. Isso faz com que menos recursos de processamento sejam utilizados.



Refleta

Memória virtual é uma técnica sofisticada e poderosa de gerência de memória, em que as memórias principal e secundária são combinadas dando ao usuário a ilusão de existir uma memória muito maior que a capacidade real da memória principal. O conceito de memória virtual fundamenta-se em não vincular o endereçamento feito pelo programa dos endereços físicos da memória principal. Desta forma, programas e suas estruturas de dados deixam de estar limitados ao tamanho da memória física disponível, pois podem possuir endereços associados à memória secundária (MACHADO; MAIA, 2013, p. 159).

Existem três técnicas para implementar memória virtual: paginação, segmentação e segmentação com paginação. É comum associar o conceito de memória virtual com vetores, em função de, apesar de os dados serem alocados em posições distintas, não haver a necessidade de saber a exata posição de um determinado dado. Basta que instruções de código sejam desenvolvidas para que no contexto de software possam trazer a instrução de acordo com a sua respectiva identificação. A diferença é que um aplicativo ou software não fará a referência direta aos endereços de memória física do processo, e sim ao seu respectivo endereço de memória virtual. Em termos

de processamento, por esse necessitar apenas do endereço físico, ou seja, temos a conversão do endereço virtual ao seu em memória principal. A esse procedimento de conversão de endereços atribui-se o nome de mapeamento, a considerar que um processo é constituído pelo seu espaço de endereçamento e pelos contextos de hardware e de software.

Já a memória RAM (Random Access Memory) é um hardware que armazena, de acordo com uma limitação que pode ser da ordem de gigabytes, uma determinada quantidade de informações, porém, temporariamente. O acesso aos dados é realizado de forma não sequencial e, por esse motivo, ela tem em seu nome o termo “randômica” (aleatória), em que o acesso é feito a um processo sempre que solicitado, de maneira aleatória, não obedecendo a uma ordem de acesso.

A velocidade de um computador para processar informações está diretamente relacionada ao tamanho de sua memória RAM, em função de ser essa a responsável por armazenar o processo de forma que não tenha de ser mapeado o seu endereço em memória secundária, e seja acessado diretamente e em sua respectiva posição e enviado para ser executado. É considerada volátil, por precisar de uma fonte de energia que a alimente.

Então, os processos saem da fila de execução ou porque foram concluídos e, com isso, já liberaram o espaço para que outro processo seja alocado, ou porque o computador foi desligado e não há mais a necessidade de armazenar dados de processo para execução imediata.



Assimile

E quando o problema parece ser o tamanho da memória virtual? Nesse caso, você precisa analisar bem o problema e diferenciar se há a necessidade de alteração de MV ou se é mais viável adicionar memória RAM em seu computador. Veja abaixo um artigo da Microsoft acerca dessa situação que requer uma análise mais criteriosa de viabilidade:

“Se faltar ao seu computador a quantidade de memória RAM necessária para executar um programa ou uma operação, o Windows usa a memória virtual para compensar. Para descobrir a quantidade de RAM existente no computador, consulte descobrir a quantidade de RAM existente no computador. A memória virtual combina a RAM do computador com espaço temporário no disco rígido. Quando a RAM fica insuficiente, a memória virtual move os dados da RAM para um espaço chamado arquivo de paginação. Isso libera a RAM para que o computador possa concluir seu trabalho. Quanto mais RAM um computador tem, mais rápido ele irá executar os programas. Se a falta de RAM estiver diminuindo o desempenho do computador, é possível que você fique tentado

a aumentar a memória virtual para compensar. Entretanto, como o computador pode ler dados da RAM com muito mais rapidez do que de um disco rígido, a melhor solução é adicionar RAM.”

Fonte: Microsoft. Disponível em: <<http://windows.microsoft.com/pt-br/windows/what-is-virtual-memory#1TC=windows-7>>. Acesso em: 5 out. 2015.

Mas, afinal, como a técnica de swapping interage com a memória principal?

Esse identifica a oportunidade de realizar uma troca da memória principal para a memória secundária (swap out) de um processo que tenha o mesmo tamanho em termos de segmentos alocados e, assim que o processo alocado encerra a execução, libera este espaço para o processo original residente retornar da memória secundária para a memória principal (swap in).

Foi na universidade de Manchester que houve a primeira implementação de memória virtual. O sistema operacional era o Atlas, desenvolvido em 1962 nesta universidade.



Pesquise mais

No site de um desenvolvedor conhecido de sistemas operacionais e outros softwares, a Microsoft disponibiliza artigos que podem auxiliar na compreensão e aplicação dos conceitos sobre memória virtual aqui estudados. Disponível em: <<http://windows.microsoft.com/pt-br/windows/what-is-virtual-memory#1TC=windows-7>>. Acesso em: 5 out. 2015.

A principal vantagem é que um software pode fazer referência a um processo que esteja fora da memória principal, pois deixaram de ser limitados ao tamanho da memória física para que possam ser selecionados e entrar em execução.

Sendo assim, a memória secundária é utilizada pelo sistema operacional como uma forma de expandir a memória de trabalho. Apenas uma parte do processo que é residente permanece na memória principal.

E, quanto ao mapeamento dos endereços que mencionamos, como é esse mecanismo?

Com um algoritmo de mapeamento inserido na decodificação do endereço virtual para o endereço físico do processo, temos condições de não alocar contigualmente os processos em memória principal, considerando-se que em sistemas operacionais modernos a unidade de gerência de memória fica responsável por essa conversão ou tradução de endereços (MACHADO; MAIA, 2013).

Nesse contexto de mapeamento de endereços, podemos elencar uma estrutura de dados em tabela que contém as informações de cada processo. Assim, a cada vez que ele for solicitado, será acessada a sua respectiva tabela com as informações de seu endereço virtual. Observe o exemplo de tabela de endereços virtuais utilizada no mapeamento de blocos de dados:



Exemplificando

Tabela 4.2 | Espaço virtual x tamanho do bloco

End. Virtual	Tamanho do bloco	Número de blocos	Nº de entradas na tabela de mapeamento
2^{32} endereços	512 endereços	2^{23}	2^{23}
2^{32} endereços	4 K endereços	2^{30}	2^{30}
2^{64} endereços	4 K endereços	2^{52}	2^{52}
2^{64} endereços	64 K endereços	2^{48}	2^{48}

Fonte: Machado; Maia (2013, p. 163).

A tabela representa uma relação inversamente proporcional em que, quanto maior o bloco, teremos menos entradas na tabela de endereços virtuais.

Esses blocos a que nos referimos na tabela podem ser de tamanho fixo (paginação) ou de tamanho variável (segmentação).

Vamos agora falar dos conceitos de memória virtual por paginação. Já falamos anteriormente que, em função do tamanho dos segmentos de endereços alocados para os processos, muitas vezes o sistema operacional tratará essas informações em blocos menores, chamados de páginas, e criará uma de mesmo tamanho para fazer referência a esta na memória secundária.

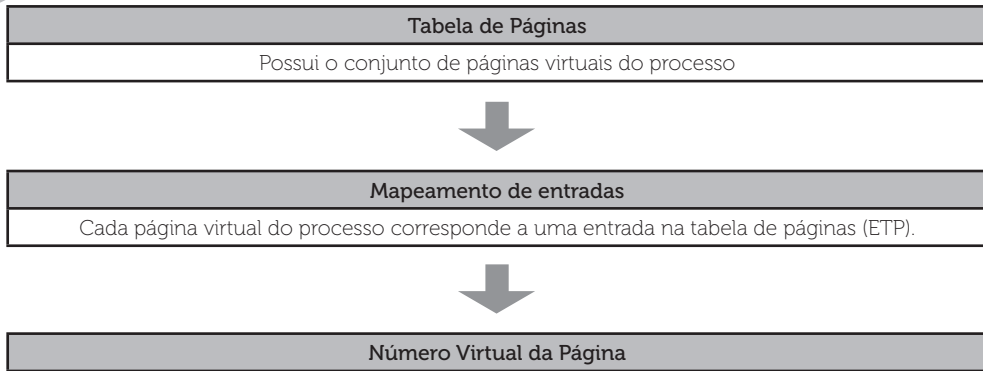
Para as páginas da memória principal, damos o nome de páginas reais, também chamadas de frames. E páginas virtuais quando se trata da respectiva paginação em memória virtual.

O fluxo de mapeamento de memória virtual pode ser definido da seguinte forma:

Tabela 4.7 | Fluxo de mapeamento de memória virtual

Processo
Possui uma tabela de páginas própria.





Fonte: Adaptado de Machado; Maia (2013, p. 163).

Estamos dizendo, na Figura 4.7, que todo processo possui a sua respectiva tabela de páginas e que cada uma das páginas representa uma entrada na tabela de páginas. Além disso, que a sua localização pode ser realizada a partir o número virtual da página, também chamado de (NVP), que é uma espécie de índice para localizar a informação do processo na memória virtual.

A ETP, ou entrada na tabela de páginas, é composta por informações como o bit de validade do processo, que indica através da representação "0" que está na página principal e "1", que o processo se encontra na memória virtual.

Além disso, se o processo estiver na memória virtual e a unidade de gerência de memória identificar a partir do bit de validade que a página não está também na memória principal, acontece um procedimento chamado de page fault, pois identificou que está faltando a correspondência na memória principal. Com isso, uma cópia do bloco de dados da memória virtual é enviado para a memória principal e é corrigido o problema, porém, em função de aumentar as operações de entrada e saída, podemos ter perdas de eficiência da máquina. Por esse motivo, controla-se esse tipo de comportamento do sistema por meio de um indicador, conhecido como taxa de paginação do processo. Ao fato de a página virtual ser enviada para a sua respectiva tabela de processo na memória principal, atribui-se o nome de page in, ou entrada de página.

Quando esse procedimento de page in se torna frequente, pode acarretar em perda de eficiência da máquina.

Da mesma forma, quando uma página está identificada como page fault, o seu processo passa do estado de execução para o estado de espera enquanto acontece a transferência da memória secundária para a memória principal. Com isso, há a troca de contexto e a respectiva atualização da informação na tabela de mapeamento e a restauração das informações do processo para que possa novamente entrar em procedimento de escalonamento, retornando à fila de processos que se encontram em estado de pronto (MACHADO; MAIA, 2013).

Você agora deve estar se perguntando: de que forma o sistema operacional gerencia a demanda de processos no contexto da alocação de recursos em memória virtual? Para contornar essa situação e permitir o controle de demanda, foi criada a política de busca de páginas. Essa pode ser classificada em: paginação por demanda, ou seja, há a transferência da memória secundária para a principal apenas quando há a referência, e, ainda, há a paginação antecipada, voltada também para o controle de demanda, no entanto com o acréscimo de páginas, caso o processo venha precisar já tem algumas alocadas e a disposição.

Além da política de buscas, temos também as seguintes políticas de:

1. Alocação de páginas.
2. Substituição de páginas.
3. Working set.
4. Algoritmos de substituição de páginas:
 - a. Ótimo.
 - b. Aleatória.
 - c. FIFO (First in First out).
 - d. LFU (Least- Frequently- Used).
 - e. LRU (Least- Recently- Used).
 - f. NRU (Not- Recently- Used).
 - g. FIFO com buffer de páginas.
 - h. FIFO circular.
5. Política para estabelecer o tamanho da página.
6. Paginação com múltiplos níveis.
7. Tradução de endereços virtuais em endereços reais.
8. Proteção de memória.
9. Compartilhamento de memória.

Além dos conceitos aqui apresentados, não podemos esquecer que no que diz respeito ao conceito de memória virtual também temos a sua implementação realizada por segmentação. Nesse sentido, essa pode ser realizada de acordo com uma segmentação por paginação, ou, ainda, realizando swapping de memória virtual. Além

disso, um ponto de atenção que precisamos reforçar aqui é a questão do trashing, que pode acontecer nas operações diretamente relacionadas à memória virtual.




Faça você mesmo

De acordo com as políticas mencionadas, elabore um relatório que contenha as respectivas definições e um exemplo de aplicação de cada uma delas! Bons estudos.

Sem medo de errar

Vamos, então, aprender a configurar o tamanho da memória virtual de sua máquina a partir das recomendações da Microsoft, fornecedora do sistema operacional Windows. Confira abaixo o procedimento adotado:

“Se você receber avisos de que a memória virtual é insuficiente, aumente o tamanho mínimo do arquivo de paginação. O Windows define o tamanho mínimo inicial do arquivo de paginação como a quantidade de memória RAM instalada no computador, e o tamanho máximo, como três vezes a quantidade de RAM instalada no computador. Se você vir avisos nesses níveis recomendados, aumente os tamanhos mínimo e máximo.

1. Para abrir Sistema, clique no botão Iniciar , clique com o botão direito em Computador e clique em Propriedades.
2. No painel esquerdo, clique em Configurações avançadas do sistema. Se você for solicitado a informar uma senha de administrador ou sua confirmação, digite a senha ou forneça a confirmação.
3. Na guia Avançado, em Desempenho, clique em Configurações.
4. Clique na guia Avançado e, em Memória virtual, clique em Alterar.
5. Desmarque a caixa de seleção Gerenciar automaticamente o tamanho do arquivo de paginação de todas as unidades.
6. Em Unidade [Rótulo do Volume], clique na unidade que contém o arquivo de paginação que você deseja alterar.
7. Clique em Personalizar Tamanho e digite um novo tamanho em megabytes na caixa Tamanho inicial (MB) ou Tamanho máximo (MB). Em seguida, clique em Definir e em OK.”

Fonte: Microsoft. Disponível em: <<http://windows.microsoft.com/pt-br/windows/change-virtual-memory-size#1TC=windows-7>>. Acesso em: 05 out. 2015.



Atenção!

Veja a referência completa do artigo e mais informações sobre o procedimento no site: <<http://windows.microsoft.com/pt-br/windows/change-virtual-memory-size#1TC=windows-7>>. Acesso em: 5 out. 2015.



Lembre-se

Existe um forte relacionamento entre a gerência de memória virtual e a arquitetura de hardware do sistema computacional. Por motivos de desempenho, é comum que algumas funções da gerência da memória virtual sejam implementadas diretamente no hardware. Além disso, o código do sistema operacional deve levar em consideração várias características específicas da arquitetura, especialmente o esquema de endereçamento do processador (MACHADO; MAIA, 2013, p. 159).

Avançando na prática

Pratique mais	
Instrução Desafiamos você a praticar o que aprendeu, transferindo seus conhecimentos para novas situações que pode encontrar no ambiente de trabalho. Realize as atividades e depois compare-as com a de seus colegas.	
Memória virtual: conceitos, paginação, segmentação e virtualização	
1. Competência de fundamentos de área	O aluno deverá ser capaz de identificar quais são as principais funções de um sistema operacional, bem como ter conhecimento sobre como se dá o compartilhamento de recursos e a sua gerência.
2. Objetivos de aprendizagem	Conhecer os principais conceitos sobre memória virtual e os procedimentos necessários a essa implantação.
3. Conteúdos relacionados	Memória virtual: conceitos, paginação, segmentação e virtualização.
4. Descrição da SP	Uma vez que o foco dessa pesquisa repousa sobre memória virtual e suas formas de implementar, realize um levantamento em uma base de referência, que compare o procedimento de implementação de memória virtual por paginação com o de segmentação.
5. Resolução da SP	Quando falamos em implementação de memória virtual por meio de segmentação, podemos elencar as seguintes características que também a diferem do processo de paginação, a começar pelos atributos de sua tabela de entrada de segmento (ETS). Observe a seguir:

Tabela 4.3: Campos da ETS

Campo	Descrição
Tamanho	Especifica o tamanho do segmento.
Bit de validade	Indica se o segmento está na memória principal.
Bit de modificação	Indica se o segmento foi alterado.
Bit de referência	Indica se o segmento foi recentemente referenciado, sendo utilizado pelo algoritmo de substituição.
Proteção	Indica a proteção do segmento.

Fonte: Machado; Maia (2013, p. 183).
Além dessas características, observe os apontamentos de Machado; Maia (2013, p. 184), que comparam os procedimentos de paginação com segmentação:

Tabela 4.4: Paginação x segmentação

Característica	Paginação	Segmentação
Tamanho dos blocos de memória	Iguais	Diferentes
Proteção	Complexa	Mais simples
Compartilhamento	Complexo	Mais simples
Estruturas de dados dinâmicas	Complexo	Mais simples
Fragmentação interna	Pode existir	Não existe
Fragmentação externa	Não existe	Pode existir
Programação modular	Dispensável	Indispensável
Alteração do programa	Mais trabalhosa	Mais simples

Fonte: Machado; Maia (2013, p. 184).

Vantagens da segmentação com relação à paginação:

- a. É compatível com estruturas de dados dinâmicas, como pilhas e filas. Na paginação, a alteração do vetor requer a criação de novas páginas e a alteração da ETP.
- b. As transferências de memória secundária para a memória principal acontecem apenas com a referência dos segmentos. Os softwares precisam ser desenvolvidos em módulos para aumentar o grau de multiprogramação.
- c. Quanto à fragmentação em paginação, temos esse problema internamente, enquanto que, na segmentação, a fragmentação externa pode ocorrer.
- d. O mapeamento em segmentação é mais simples, pois mapeia estruturas lógicas e não as páginas (MACHADO; MAIA, 2013).



Lembre-se

Outra técnica é a segmentação com paginação, em que o espaço de endereços de memória é dividido em segmentos e cada segmento divide-se em páginas.



Faça você mesmo

Em sua concepção, tratando-se de gerência de memória seja por paginação, segmentação ou segmentação por paginação, qual é a importância de se seguir uma política que determine como serão realizados os compartilhamentos, buscas, substituições e acessos? Você, diante da arquitetura de computadores utilizada atualmente, já se encontra em condições de avaliar outras possibilidades de se fazer gerência de memória visando otimizar recursos e aumentar eficiência de máquina?

Faça valer a pena

1. Analise as afirmações e assinale a alternativa correspondente:

- I. Foi criada para solucionar alguns problemas, como o da fragmentação do disco, ou mesmo quando se utiliza a técnica de overlay.
- II. Trabalha com uma quantidade maior de processos que compartilham a memória principal.
- III. O acesso aos dados é realizado de forma não sequencial e, por esse motivo, ela tem em seu nome o termo “randômica” (aleatória), em que o acesso é feito a um processo sempre que solicitado, de maneira aleatória, não obedece a uma ordem de acesso.

Estamos respectivamente falando de:

- a) RAM, Swap e memória virtual;
- b) swap, RAM e memória virtual;
- c) memória virtual, swap e RAM;
- d) memória virtual, memória virtual e RAM;
- e) RAM, RAM, swapping.

2. Complete as lacunas da frase com as opções de uma das alternativas respectivamente:

"O conceito de _____ fundamenta-se em não vincular o endereçamento feito pelo programa dos endereços físicos da _____. Desta forma, programas e suas estruturas de dados deixam de estar limitados ao tamanho da memória física disponível, pois podem possuir endereços associados à _____" (MACHADO; MAIA, 2013, p. 159).

- a. memória secundária/memória virtual/memória principal
- b. memória virtual/memória principal/memória secundária
- c. memória virtual/memória secundária/memória híbrida
- d. memória virtual/memória virtual/memória secundária
- e. memória virtual/memória secundária/memória principal

3. São técnicas de implementação de memória virtual:

- a. Paginação, segmentação e segmentação com paginação.
- b. Swapping, trashing, FIFO.
- c. Randômico, integral, distorção.
- d. Cliente Servidor, roteirização.
- e. Escalonamento, sincronização e threads.

4. É verdadeiro o que se afirma em:

I. A principal vantagem em memórias virtuais é que um software pode fazer referência a processos que estejam fora da memória principal, pois deixaram de ser limitados ao tamanho da memória física para que possam ser selecionados e entrar em execução.

II. A principal vantagem é que em memórias virtuais o acesso ao dado é idêntico ao acesso randômico e não utiliza identificadores de processos ou bit de validade.

III. O bit de validade indica o tempo de execução dos processos que estão na memória virtual.

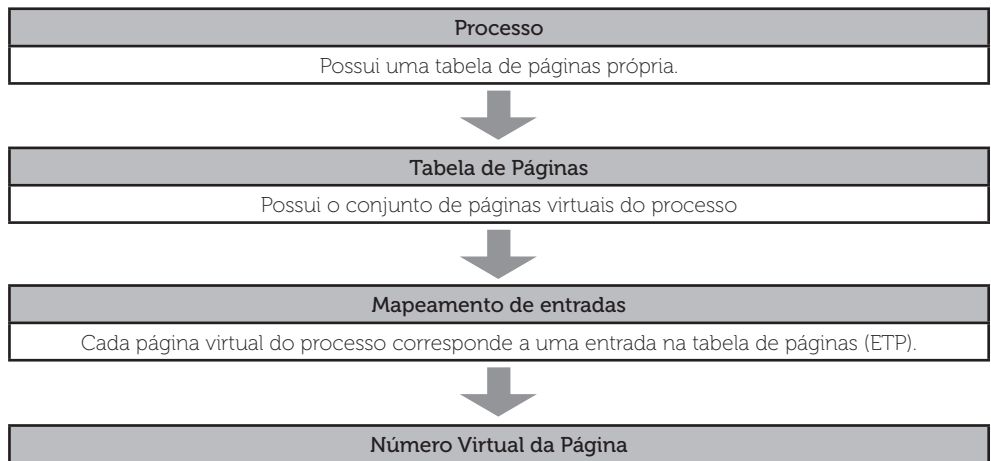
- a. I e II.
- b. Apenas I.
- c. Apenas II.
- d. II e III.
- e. I, II e III.

5. Explique o mecanismo de mapeamento de processos em memória virtual:

6. Qual é a relação com a memória virtual dos conceitos de page fault e page in?

7. Analise a figura e assinale a alternativa correspondente:

Figura 4.8 | Fluxo de mapeamento de memória virtual



Fonte: Adaptado de Machado; Maia (2013, p. 163).

A figura representa:

a) que todo processo possui a sua respectiva tabela de páginas e que cada uma das páginas representa uma entrada na tabela de páginas. Além disso, que a sua localização pode ser realizada a partir o número virtual da página;

b) a ETP, ou entrada na tabela de páginas, que é composta por informações como o bit de validade do processo que indica através da representação "0" que está na página principal e "1", que o processo se encontra na memória virtual;

c) o processo na memória virtual e a unidade de gerência de memória identificar a partir do bit de validade que a página não está também na memória principal, acontece um procedimento chamado de page fault;

d) a estrutura de página virtual enviada para a tabela de processo na memória principal e se chama relatório de paginação;

e) o fluxo de operações do sistema operacional.

Seção 4.4

Gerenciamento de dispositivos de entrada e saída: conceitos, rotinas, tipos e suas características

Diálogo aberto

Estamos encerrando os estudos dos fundamentos de sistemas operacionais. Chegou o momento de conhecer como acontece a gerência de dispositivos de E/S e a sua importância para o sistema operacional. Com isso, vamos aproximar a teoria e a prática a partir da implementação de técnicas e, também, da compreensão das funções dos drivers para as operações de leitura e escrita.

Você precisará identificar uma forma de a empresa de P&D em questão desenvolver as melhorias necessárias, competentes ao driver responsável pela recepção, codificação e transmissão dos dados que são provenientes das ações do usuário no sistema, com as operações de E/S.

Para resolver essa situação, você precisa apresentar um tipo de driver, os seus modos de operação, bem como descrever brevemente de que modo acontecem as operações de interpretação das operações de leitura e escrita.

Além disso, aqui conseguimos atingir o objetivo de conhecer como acontece a gerência de dispositivos de E/S pelo sistema operacional, o que nos permite identificar, nos computadores que trabalhamos, a necessidade de se atualizar algum driver, por exemplo, ou mesmo que tipo de problema e a sua respectiva solução, o que certamente pode nos ajudar a desempenhar as tarefas auxiliadas pelo uso do computador, em nosso cotidiano.

Não se esqueça de recorrer ao seu material, mesmo às unidades de ensino anteriores, para relembrar conceitos já estudados, como gerência de processos, de memória, sistemas de arquivos e os assuntos a esses interligados.

Além disso, resolver os exercícios e tirar suas dúvidas é muito importante. Então, leia com atenção o seu livro didático e acompanhe regularmente as atualizações em conteúdo pertinentes à disciplina.

Não pare por aqui, continue avançando e conheça ainda mais sobre o universo da

tecnologia da informação, pois essa fará parte de toda a sua trajetória profissional.

Ficamos por aqui. Obrigado pela sua atenção e dedicação aos estudos!

Não pode faltar

Considerada como uma das mais complexas tarefas que o sistema operacional realiza, a gerência dos dispositivos de entrada e saída é responsável por facilitar a comunicação entre as solicitações dos usuários e as aplicações. Para tanto, é preciso que a arquitetura de comunicação entre os dispositivos de E/S e os respectivos hardwares e softwares estejam interligados e possam atender a essas demandas.

Vamos conhecer melhor a arquitetura em camadas e seus elementos, ilustrados na Figura 1, e também seremos apresentados aos principais conceitos relacionados a:

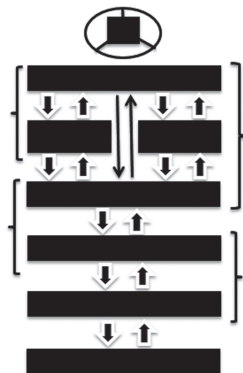
- subsistema de E/S
- device Driver
- controlador
- dispositivos de E/S.
- discos magnéticos ou SGBD

Além disso, conheceremos também como se aplicam questões de segurança da informação na gerência de dispositivos de entrada e saída.



Assimile

Figura 4.9 | Arquitetura de camadas da gerência de dispositivos



Fonte: Machado; Maia (2013, p. 209).

A arquitetura de gerência de dispositivos é dividida em camadas que se distribuem da seguinte forma:

1. Dispositivos ou periféricos de entrada e saída (E/S): mecanismos que permitem a interação entre usuário e máquina de forma amigável e segura junto às aplicações.

2. Controlador: esse hardware faz a interface entre a solicitação do usuário e o driver. É composto por memória e registradores programados para enviar as instruções ao respectivo driver. Aqui é que são armazenadas as sequências de bits que o dispositivo de E/S envia. Quando não houver erros, o controlador enviará a informação do bloco para a memória principal.

3. Device driver: nessa camada, há instruções que realizarão a comunicação das solicitações enviadas pelo controlador, ao subsistema de E/S. Sua função principal é interpretar ou traduzir as instruções recebidas, para comandos que são compreensíveis tanto para os controladores quanto para o subsistema de E/S.

4. Subsistema de E/S: tem a função de distinguir, de acordo com cada dispositivo, as solicitações e executar as rotinas de comunicação que realiza entre as aplicações, dos sistemas de arquivos e dos sistemas de gerenciamento de bancos de dados, além das solicitações advindas do device driver.

5. Sistema de arquivos, sistemas de gerenciamento de bancos de dados e as aplicações se relacionam de forma a gerar as demandas de processos.

Com isso, observamos o isolamento no tratamento de informações de dispositivos para com as informações do processo.



Pesquise mais

O artigo “Como funcionam os dispositivos de entrada e saída”, disponível em: <http://www.devmedia.com.br/como-funcionam-os-dispositivos-de-entrada-e-saida/28275>, traz informações sobre como o sistema operacional consegue interpretar as solicitações que vêm dos dispositivos de entrada e saída. Acesso em: 5 out. 2015.

Em função da quantidade de tipos de dispositivos de E/S ser bastante diversificada, temos de trabalhar com a camada de subsistema de E/S. Essa tem por função, como mencionado, isolar as atividades e solicitações dos periféricos, aplicações, sistemas de arquivos e SGBDs, de forma a permitir que os periféricos interajam com qualquer um deles e suas demandas sejam tratadas de forma diferente.

Com isso, é preciso ter um conjunto de rotinas, também chamado de rotinas de entrada e saída, para fazer essa tarefa de comunicação otimizada. Devemos fixar a forma

de como se estabelece essa comunicação, sendo que as rotinas de E/S é que geram as demandas de execução das operações de E/S.

As operações de entrada e saída podem estar organizadas logicamente, ou, de forma estruturada, quando armazenadas em dispositivos como pen drives ou disco rígido, por exemplo, ou, ainda, de forma não estruturada, quando o dispositivo não permite esse tipo de configuração.

Temos, com isso, algumas formas de se estabelecer comunicação. São elas:

- Chamada explícita: caracteriza-se pela chamada realizada diretamente pelo sistema operacional, a partir de comandos de alto nível.
- Chamada implícita: caracteriza-se pela realização das operações de E/S para leitura e gravação, através de programação de alto nível (linguagem C, por exemplo). Além desta, ainda pode realizar as chamadas de rotinas contidas nas bibliotecas de funções dessas linguagens de programação.

A camada de subsistema de E/S também considera a comunicação a partir do sincronismo, permitindo que aconteça, portanto, de forma síncrona ou assíncrona. Leia e analise a reflexão a seguir, sobre essa classificação:



Refleta

As operações de E/S podem ser classificadas conforme o seu sincronismo. Uma operação é dita síncrona quando o processo que realizou a operação fica aguardando no estado de espera pelo seu término. A maioria dos comandos das linguagens de alto nível funciona desta forma. Uma operação é dita assíncrona quando o processo que realizou a operação não aguarda pelo seu término e continua pronto para ser executado. Neste caso, o sistema deve oferecer algum mecanismo de sinalização que avise ao processo que a operação foi terminada (MACHADO; MAIA, 2013, p. 210).

Esse subsistema de E/S também trabalha com uma técnica chamada buffering, que deriva do espaço de memória intermediária conhecida como buffer, em que são carregados os blocos de dados de forma a facilitar o acesso ao dado e se mais algum processo solicitar informações, e essas estiverem no intervalo do bloco de dados, não haverá a necessidade de se realizar mais uma operação de E/S (MACHADO; MAIA, 2013).

O subsistema de E/S verifica deve receber a programação correspondente a cada dispositivo, ou seja, o driver para cada um e, assim, sempre que um periférico for reconhecido pela máquina o driver deverá ser imediatamente chamado pelo subsistema e instalado.

Quando falamos de Driver, estamos nos referindo diretamente ao processo de comunicação com o subsistema de E/S e também, com os controladores. Em uma visão mais ampla, temos o reconhecimento do dispositivo pelo subsistema de E/S. Já em uma visão mais próxima, micro, temos o drive, que contém as especificações de cada um dos dispositivos de E/S. Os drivers recebem as informações e as codifica para que o controlador ou o subsistema possam reconhecer aquelas instruções e chamar os processos inerentes a cada uma das tarefas ou operações de E/S.

Um driver pode conter as instruções e comandos de um dispositivo, bem como de um grupo de dispositivos, de forma a facilitar a programação do driver e o espaço ocupado para armazenar instruções de mesmo tipo. Algumas das características dos drivers devem ser consideradas, como, por exemplo:

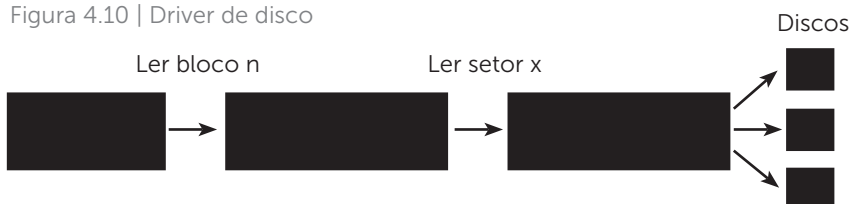
- número de registradores do controlador;
- comandos do dispositivo; e ainda,
- o fluxo e processos de funcionamento do dispositivo a que se refere.



Exemplificando

Quando o sistema recebe uma solicitação de E/S para a leitura de dados, podemos citar como exemplo o funcionamento do modo síncrono de se realizar esta operação. A informação de localização deve chegar ao cilindro, trilha e setor. Observe na Figura 4.10 como ocorre esse procedimento de comunicação síncrona:

Figura 4.10 | Driver de disco



Fonte: Machado; Maia (2013, p. 212).

Assim, quando o driver recebe essa demanda de leitura de um bloco, deverá informar ao disco todos os dados de localização daquele determinado bloco de dados.

O controlador de entrada e saída tem a função de intermediar a comunicação com os dispositivos de E/S. É um hardware e assemelha-se a uma placa ou vem implementado no próprio chip do processador.

Ele é composto por:

- memória; e
- registradores específicos que realizam a comunicação com o driver.

No controlador, forma-se o bloco de dados das operações de leitura e o mesmo é transferido, se não for identificado nenhum tipo de erro, para a memória principal. Então, podemos dizer que há, respectivamente, uma transferência de blocos de dados que estão armazenados internamente no buffer e são transferidos para o buffer de E/S. Essa tarefa é realizada pelo processador ou por um controlador que utilize a técnica chamada de DMA (Direct Memory Access ou Acesso Direto à Memória Principal).



Pesquise mais

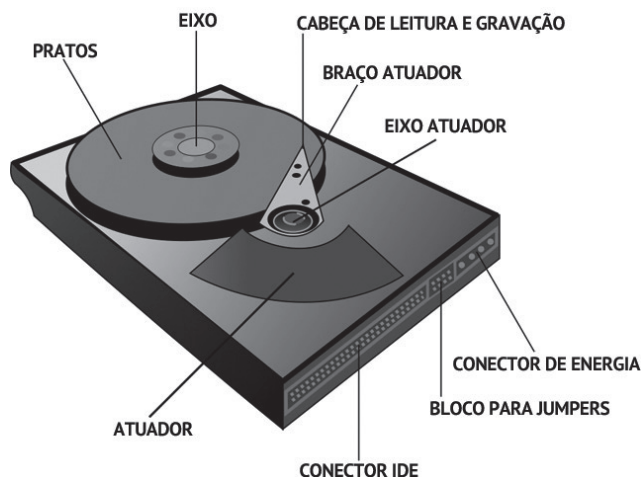
Leia sobre DMA. Disponível em: <<http://www.hardware.com.br/termos/dma>>. Acesso em: 8 out. 2015.

Mas você sabe ou se recorda quais são os dispositivos de entrada e saída? Vamos rever esta informação. Acompanhe a seguir quais são os dispositivos que fazem a comunicação entre o sistema operacional e o ambiente externo:

Podemos classificar como dispositivos de entrada de dados: teclado, mouse, microfone, pen drive, CDs, DVDs. Já como os de saída de dados, naturalmente, podemos citar as impressoras, o monitor, bem como aqueles que podem servir tanto para a entrada como para a saída de dados: modems, pen drives, CDs, DVDs, HDs externos, entre outros. Atualmente, temos até os aparelhos celulares como uma forma de se estabelecer essa comunicação também e solicitações de entrada e saída. Então, quando conectamos o nosso celular via cabo ou bluetooth, também estamos realizando operações de E/S. Observe a necessidade de quando se conecta um deles a uma outra máquina ou dispositivo, de um mecanismo que reconheça e permita que seja dada sequência nas operações tanto de leitura quanto de gravação e interface com as aplicações e sistemas de arquivos requeridos.

Além desses, ainda temos os discos magnéticos, que merecem atenção especial em função da quantidade de dados que podemos armazenar e, justamente por isso, fatores como segurança da informação e proteção devem ser levados em consideração. Esses discos magnéticos são compostos por discos sobrepostos que são unidos por um eixo vertical, que permite a sua rotação e em uma velocidade constante. Os discos são compostos por trilhas, compostas por setores. As trilhas dos discos sobrepostos formam um cilindro, sendo que para cada disco há um procedimento que permite a leitura e gravação de dados.

Figura 4.11 | Hard disk ou disco rígido



Fonte: <<https://www.oficinadanet.com.br/post/8632-como-funciona-um-disco-rigido-hd>>. Acesso em: 8 out. 2015.



Faça você mesmo

Elabore um relatório que contenha os tempos de leitura e gravação em disco: seek, latência rotacional e transferência e também traga quais são as técnicas de RAID e as diferenças. Investigue formas de implementação advindas dos fornecedores de software.

Sem medo de errar

Agora, vamos conhecer como é que se estabelece a comunicação entre a solicitação de ação do sistema pelo usuário através dos dispositivos de entrada e saída. O intuito é fazer com que seja compreendido esse procedimento, que servirá como fundamento para os melhoramentos que a empresa de pesquisa e desenvolvimento precisa investigar. Siga em frente e aproveite para testar algumas das soluções!

Como a empresa de P&D pretende encontrar as melhores práticas e disseminá-las, vamos apresentar, aqui, a interpretação do processo realizada por Vila, que é um pesquisador nessa área. Confira a seguir os seus apontamentos e explicações de conceitos que podem auxiliar nas ações de melhoramentos desse mecanismo:

- **Line Discipline:** pode ser definida como uma interface entre os processos de usuários e os drivers de terminal, ou seja, responsáveis por realizar a codificação e decodificação das solicitações de usuários em informações ou instruções de máquina. Esse driver pode trabalhar de dois modos de operação:

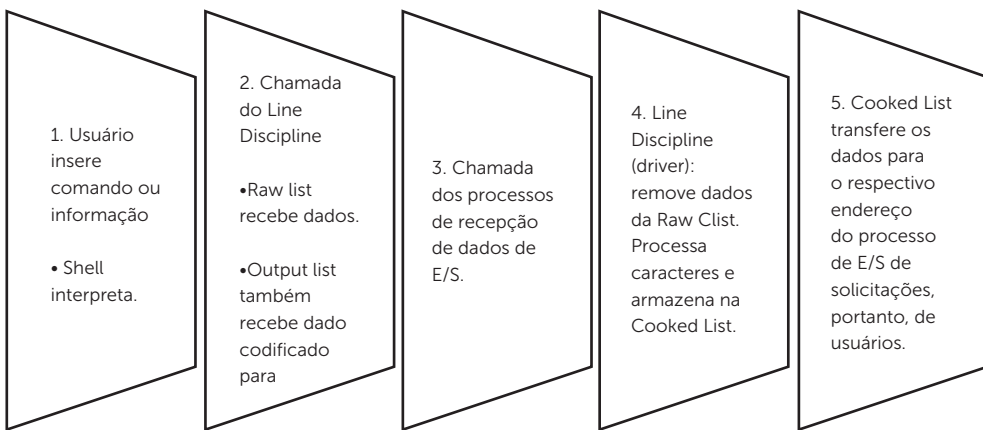
- Raw e Cooked: responsáveis pela transmissão e codificação, respectivamente, dos caracteres de entrada para os processos oriundos de solicitações de usuários. Essa informação deve passar pelo interpretador de comandos, também conhecido como shell.

Além desses conceitos, para entender melhor sobre como é o funcionamento das ações de leitura e escrita pelo sistema operacional, também podemos citar a "Clist" ou também conhecida por lista de caracteres. A Clist trabalha com uma lista encadeada de blocos de caracteres, chamada de Cblocks. Essa armazena um vetor do tipo caractere com as informações enviadas pelos dispositivos de E/S e aponta para o próximo bloco de endereços, considerando o tamanho do vetor, ou seja, considera os endereços de início e fim, como se fosse um tipo de buffer e faz, efetivamente, três operações de leitura e escrita:

- Raw Clist: responsável por armazenar os dados de entrada.
- Cooked Clist: responsável por armazenar os dados de entrada processados pelos modos de operação mencionados no line discipline.
- Output Clist: essa função do Cblock armazena os dados de saída que deverão ser exibidos para o usuário.

Agora, podemos entender como funciona uma operação de leitura:

Figura 4.12 | Processo de leitura e escrita de dados de E/S



Fonte: Adaptado de VILA, Fabrício. Universidade Federal de Campina Grande. Disponível em: <http://fubica.lsd.ufcg.edu.br/hp/cursos/so/LabSO/ent_saida.html#e1-d>. Acesso em: 7 out. 2015.



Atenção!

Um sistema operacional possui diversos drivers. Cada um será responsável por uma tarefa diferente e é constituído, portanto, de funções distintas, por exemplo, existem drivers de disco, drivers de rede, drivers de vídeo.



Lembre-se

O device driver, ou somente driver, tem como função implementar a comunicação do subsistema de E/S com os dispositivos, por intermédio dos controladores de E/S. Enquanto o subsistema de E/S trata de funções ligadas a todos os dispositivos, os drivers tratam apenas dos seus aspectos particulares (MACHADO; MAIA, 2013, p. 211).

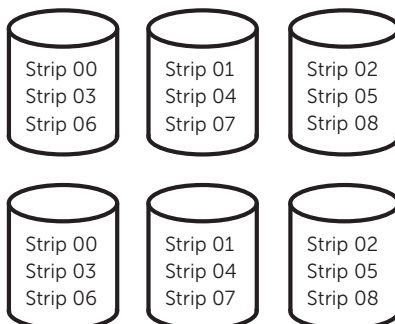
Avançando na prática

Pratique mais	
Instrução Desafiamos você a praticar o que aprendeu, transferindo seus conhecimentos para novas situações que pode encontrar no ambiente de trabalho. Realize as atividades e depois compare-as com a de seus colegas.	
Gerenciamento de dispositivos de entrada e saída: conceitos, rotinas, tipos e suas características	
1. Competência de fundamentos de área	O aluno deverá ser capaz de identificar quais são as principais funções de um sistema operacional, bem como ter conhecimento sobre como se dá o compartilhamento de recursos e a sua gerência.
2. Objetivos de aprendizagem	Conhecer como se dá a gerência de processos advindos das solicitações dos dispositivos de entrada e saída.
3. Conteúdos relacionados	Gerenciamento de dispositivos de entrada e saída: conceitos, rotinas, tipos e suas características.
4. Descrição da SP	Todos sabemos que as instituições financeiras movimentam muitos dados e precisam garantir a segurança da informação em seu aspecto mais amplo e, também, criar mecanismos que permitam que os serviços oferecidos on-line e mesmo entre agências, estejam sempre à disposição de seus clientes. Então, uma das técnicas que utilizam é a conhecida como "Espelhamento" ou RAID1. O representante do órgão fiscalizador dos bancos entrou em contato com a empresa de P&D para solicitar melhorias e possibilidades de se realizar com segurança esse procedimento de replicação dos dados em outros discos secundários, que permitam a garantia da oferta dos serviços bancários aos seus clientes. Então, a fim de atender a essa solicitação, a empresa deve elaborar um relatório que evidencie esse mecanismo de espelhamento e explique como funciona e a melhor forma de se implementar. Sua tarefa é explicar esse mecanismo. Siga em frente!
5. Resolução da SP	A técnica conhecida como espelhamento permite a replicação de dados do disco principal para um disco secundário. A essa replicação, podemos atribuir um conceito chamado de redundância, que é basicamente o fato de repetir ou permitir o acesso à informação, quantas vezes for necessário.

Dessa forma, quando o disco principal vir a ter algum tipo de falha, teremos a mesma informação disponível em um disco secundário. Essa gerência de discos está diretamente interligada às responsabilidades do sistema de arquivos.

Então, para todas as operações de E/S que forem solicitadas, há a sincronização entre o disco principal e seus espelhos, para que aconteça a replicação dos dados. Observe a Figura 4.13, que ilustra esta atividade do sistema:

Figura 4.13: Espelhamento



Fonte: Adaptado de Machado e Maia (2013, p. 218).



Lembre-se

O overhead adicional exigido nesta operação é pequeno, e o benefício da proteção justifica a implementação. Apesar da vantagem proporcionada pela redundância oferecida por esta técnica, a capacidade útil do subsistema de discos com a implementação do RAID1 é de apenas 50% (MACHADO; MAIA, 2013, p. 218).



Faça você mesmo

Siga as orientações do fornecedor para fazer a implantação. Abaixo um tutorial disponibilizado pela Microsoft para configurar RAID1 no Windows Server 2008 R2 Server Core:

“Para criar um disco espelhado (Mirror), também conhecido como RAID 1, são necessários no mínimo dois discos. Em um volume espelhado todos os dados são gravados em dois discos diferentes. Se um disco falhar, os dados serão preservados no outro disco. Neste exemplo, um volume espelhado de 1000MB será criado usando 1000MB em cada disco. Antes de criar um volume espelhado você deve:

Carregar o comando DISKPART:

DISKPART

Listar os discos existentes:

LIST DISK

Dentro do DISKPART, selecione os discos que irá usar no seu volume RAID 1 (lembrando que são somente 2 discos):

Exemplo:

SELECT DISK 1

Converter o disco para dinâmico:

CONVERT DYNAMIC

Se o disco estiver como Read Only, você não poderá convertê-lo para dinâmico. Então desative o atributo:

ATTRIBUTES DISK CLEAR READONLY

Agora com os discos convertidos para dinâmico siga os próximos passos para criar o volume espelhado:

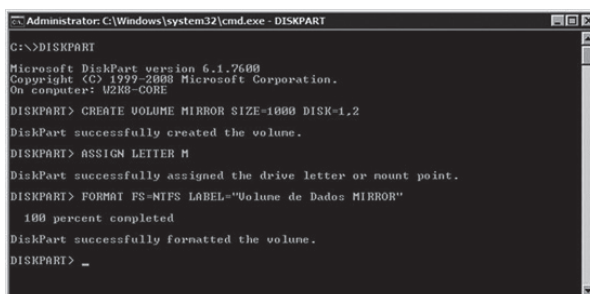
CREATE VOLUME MIRROR SIZE=1000 DISK=1,2

Defina a letra da unidade para o volume criado:

ASSIGN LETTER M

Formate a partição recém-criada:

FORMAT FS=NTFS LABEL="Volume de Dados MIRROR"



```
Administrator: C:\Windows\system32\cmd.exe - DISKPART
C:\>DISKPART
Microsoft DiskPart version 6.1.7600
Copyright (C) 1999-2008 Microsoft Corporation.
On computer: UZRB-CORE

DISKPART> CREATE VOLUME MIRROR SIZE=1000 DISK=1,2
DiskPart successfully created the volume.
DISKPART> ASSIGN LETTER M
DiskPart successfully assigned the drive letter or mount point.
DISKPART> FORMAT FS=NTFS LABEL="Volume de Dados MIRROR"
100 percent completed
DiskPart successfully formatted the volume.
DISKPART> _
```

Fonte: DONDA, Daniel. <<http://social.technet.microsoft.com/wiki/contents/articles/3993.aspx>>. Acesso em: 07 out. 2015.

Faça valer a pena

1. Associe os conceitos na tabela e assinale a alternativa correspondente:

I. Dispositivos de E/S	() Esse hardware faz a interface entre a solicitação do usuário e o driver.
II. Controlador	() Nessa camada, há instruções que realizarão a comunicação das solicitações enviadas pelo controlador ao subsistema de E/S.
III. Device driver	() Tem a função de distinguir de acordo com cada dispositivo, as solicitações e executar as rotinas de comunicação que realiza entre as aplicações, dos sistemas de arquivos e dos sistemas de gerenciamento de bancos de dados, além das solicitações advindas do device driver.
IV. Subsistema de E/S	Mecanismos que permitem a interação entre usuário e máquina de forma amigável e segura junto às aplicações.

- a) I, II, III e IV.
- b) IV, III, II e I.
- c) II, III, IV e I.
- d) III, II, I e IV.
- e) IV, I, II, e III.

2. É verdadeiro o que se afirma em:

I. Em função da quantidade de tipos de dispositivos de E/S ser bastante diversificada, temos de trabalhar com a camada de Subsistema de E/S, que faz a interface com o processo diretamente.

II. A camada de subsistema de E/S tem por função isolar as atividades e solicitações dos periféricos, aplicações, sistemas de arquivos e SGBDs, de forma a permitir que os periféricos interajam com qualquer um deles e suas demandas sejam tratadas de forma diferente.

III. Temos de fixar a forma de como se estabelece essa comunicação, sendo que as rotinas de E/S é que geram as demandas de execução das operações de E/S.

- a) Apenas III.
- b) I e III.
- c) Apenas II.
- d) II e III.
- e) Apenas I.

3. Explique brevemente o modo de funcionamento dos device drivers ou simplesmente driver:

4. Assinale a alternativa que contém os componentes de um driver:

- a) Número de registradores do controlador; comandos do dispositivo e o fluxo de processos de funcionamento do dispositivo a que se refere.
- b) Quantidade de bits de um bloco de dados e o seu endereço de memória.
- c) Endereço de memória e registradores utilizados.
- d) Mapa de bits da memória e alocação da RAM.
- e) Quantidade de registradores e informações da RAM.

5. As afirmações abaixo referem-se respectivamente a que tipo de interação ou classificação das operações de E/S? Assinale a alternativa correta:

I. Caracteriza-se pela chamada realizada diretamente pelo sistema operacional, a partir de comandos de alto nível.

II. Caracteriza-se pela realização das operações de E/S para leitura e gravação, através de programação de alto nível (linguagem C, por exemplo). Além dessa, ainda podem realizar as chamadas de rotinas contidas nas bibliotecas de funções dessas linguagens de programação.

- a) Roteamento/ mapa de bits.
- b) Chamada explícita/ chamada implícita.
- c) Chamada de processo/ device driver.
- d) Rotina de E/S / chamada implícita.
- e) Chamada implícita/ chamada explícita.

6. O controlador de operações de E/S é composto por:

- a) registradores e mouse
- b) teclado e memória
- c) registro e banco de dados
- d) processos e aplicações
- e) memória e registradores

7. Como o controlador realiza a transferência de blocos de dados que estão armazenados internamente no buffer, transferindo-os para o buffer de E/S da memória principal?

Referências

Entrada e Saída. VILAR, Francisco. Universidade Federal de Campina Grande. Disponível em: <http://fubica.lsd.ufcg.edu.br/hp/cursos/so/LabSO/ent_saida.html#e1->. Acesso em: 6 out. 2015.

MACHADO, Francis B.; MAIA, Luiz P. **Arquitetura de Sistemas Operacionais**. 5. ed. Rio de Janeiro: LTC, 2013.

STUART, Brian L. **Princípios de sistemas operacionais**: projetos e aplicações. São Paulo: Cengage Learning, 2010.