

Gestão de entradas e saídas

Capítulo 20 – Software de entrada/saída

(01) Questão 01. Por que os sistemas operacionais exigem de todos os drivers de dispositivo (device-drivers) a mesma interface padrão? Não seria mais apropriado deixar cada driver de dispositivo definir rotinas de interface que fazem sentido para aquele tipo específico de dispositivo?

Capítulo 21 – Discos Rígidos

(02) Questão 03. Você tem 4 discos rígidos de 8 TB cada, que pode organizar de diversas formas. Indique os arranjos RAID que escolheria para obter:

(a) O maior espaço útil de disco.

RAID 0 (linear): Esse modo está em criar discos virtuais, para distribuir os dados entre todos os discos e com isso você vai ganhar um considerável ganho de desempenho. Então se você tiver 2 discos de 80GB, você vai ter uma área útil de 160GB.

(b) A maior tolerância a falhas de disco.

RAID 6: é uma extensão do nível RAID 5 que utiliza blocos com códigos corretores de erros de Reed-Solomon, além dos blocos de paridade. Esta redundância extra demanda dois discos adicionais, mas permite tolerar falhas simultâneas de até dois discos.

(c) A maior velocidade média de leitura.

RAID 1: Esta abordagem oferece uma excelente confiabilidade, pois cada bloco lógico está escrito em dois ou mais discos distintos; caso um deles falhe, os demais continuam acessíveis. O desempenho em leituras também é beneficiado, pois a controladora pode distribuir as leituras entre as cópias dos dados. Contudo, não há ganho de desempenho em escrita, pois cada operação de escrita deve ser replicada em todos os discos.

(d) A maior velocidade média de escrita.

RAID 0 (striping): Suas características de suporte a grande volume de dados e alto desempenho em leitura/escrita tornam esta abordagem adequada para ambientes que precisam processar grandes volumes de dados temporários, como os sistemas de computação científica.

(e) Equilíbrio entre espaço útil, velocidades e tolerância a falhas.

RAID 5: esta abordagem armazena informações de paridade para tolerar falhas em blocos ou discos. Todavia, essas informações não ficam concentradas em um único disco físico, sendo distribuídas uniformemente entre eles. Essa estratégia elimina o gargalo de desempenho no acesso aos dados de paridade visto no RAID 4 (embora seja necessário, a cada escrita, ler blocos de todos os discos para poder calcular o novo bloco de paridade). Esta é uma abordagem de RAID popular, por oferecer um bom desempenho e redundância de dados, desperdiçando menos espaço que o espelhamento (RAID 1).

Justifique/explice suas respostas.

Gestão de arquivos

Capítulo 22 – O conceito de arquivo

(03) Questão 03. Apresente e comente as principais formas de atribuição de tipos aos arquivos. Quais são as vantagens e desvantagens de cada uma?

Os atributos dos arquivos variam de acordo com o sistema de arquivos utilizado. Os atributos mais usuais, presentes na maioria dos sistemas, são:

Nome: string que identifica o arquivo para o usuário, como "foto1.jpg", "documento. pdf", "hello.c", etc.;

Tipo: indicação do formato dos dados contidos no arquivo, como áudio, vídeo, imagem, texto, etc. Muitos sistemas operacionais usam parte do nome do arquivo para identificar o tipo de seu conteúdo, na forma de uma extensão: ".doc", ".jpg", ".mp3", etc.;

Tamanho: indicação do tamanho do conteúdo do arquivo, geralmente em bytes

Datas: para fins de gerência, é importante manter as datas mais importantes relacionadas ao arquivo, como suas datas de criação, de último acesso e de última modificação do conteúdo;

Proprietário: em sistemas multiusuários, cada arquivo tem um proprietário, que deve estar corretamente identificado;

Permissões de acesso: indicam que usuários têm acesso àquele arquivo e que formas de acesso são permitidas (leitura, escrita, remoção, etc.);

Localização: indicação do dispositivo físico onde o arquivo se encontra e da posição do arquivo dentro do mesmo;

(04) questão 04. Sobre as afirmações a seguir, relativas a formatos de arquivos, indique quais são incorretas, justificando sua resposta:

(a) Um *magic number* consiste de um atributo numérico separado que identifica o tipo de arquivo.

(b) A forma mais comum de identificação de tipo de arquivo é o uso de extensões ao seu nome.

(c) Arquivos de texto em sistemas DOS e UNIX diferem nos caracteres de controle usados para identificar o fim de arquivo.

(d) Para a maioria dos núcleos de sistema operacional, arquivos são quase sempre vistos como meras sequências de bytes.

(e) ELF e PE são dois formatos típicos de arquivos de configuração.

(f) O padrão MIME é usado no Linux para identificação de tipos de arquivos pelo sistema operacional.

Capítulo 23 – Uso de arquivos

(05) Questão 03. Comente as principais formas de acesso a arquivos. Qual o uso mais apropriado para cada uma delas?

Acesso sequencial: os dados são sempre lidos e/ou escritos em sequência, do início ao final do arquivo. Para cada arquivo aberto por uma aplicação é definido um ponteiro de acesso, que inicialmente aponta para a primeira posição do arquivo. A cada leitura ou escrita, esse ponteiro é incrementado e passa a indicar a posição da próxima leitura ou escrita. Quando esse ponteiro atinge o final do arquivo, as leituras não são mais permitidas, mas as escritas podem sê-lo, permitindo acrescentar dados ao final do mesmo. A chegada do ponteiro ao final do arquivo é normalmente sinalizada ao processo através de um flag de fim de arquivo (*EoF – End-of-File*).

Acesso aleatório (ou direto): indica a posição no arquivo onde cada leitura ou escrita deve ocorrer, sem a necessidade de um ponteiro de posição corrente. Assim, caso se conheça previamente a posição de um determinado dado no arquivo, não há necessidade de percorrê-lo sequencialmente até encontrar o dado desejado. Essa forma de acesso é muito importante em gerenciadores de bancos de dados e aplicações congêneres, que precisam acessar rapidamente as posições do arquivo correspondentes aos registros desejados em uma operação.

(06) Questão 04. Apresente e explique os quatro principais tipos de travas sobre arquivos compartilhados disponíveis no sistema operacional.

Travas obrigatórias (*mandatory locks*): são impostas pelo núcleo de forma incontornável: se um processo obtiver a trava de um arquivo, outros processos que solicitarem acesso ao mesmo arquivo serão suspensos até que aquela trava seja liberada. É o tipo de trava mais usual em sistemas Windows.

Travas recomendadas (*advisory locks*): não são impostas pelo núcleo do sistema operacional, mas gerenciadas pelo suporte de execução (bibliotecas). Os processos envolvidos no acesso aos mesmos arquivos devem travá-los explicitamente quando forem acessá-los. Contudo, um processo pode ignorar essa regra e acessar um arquivo ignorando a trava, caso necessário. Travas recomendadas são úteis para gerenciar concorrência entre processos de uma mesma aplicação. Neste caso, cabe ao programador implementar os controles necessários nos processos para impedir acessos conflitantes aos arquivos compartilhados. Em relação ao compartilhamento das travas de arquivos, elas podem ser:

Travas exclusivas (ou travas de escrita): garantem acesso exclusivo ao arquivo: enquanto uma trava exclusiva estiver ativa, nenhum outro processo poderá obter outra trava sobre o mesmo arquivo.

Travas compartilhadas (ou travas de leitura): impedem outros processos de criar travas exclusivas sobre aquele arquivo, mas permitem a existência de outras travas compartilhadas.

(07) Questão 05. Apresente e explique as quatro principais semânticas de acesso a arquivos compartilhados em um sistema operacional.

Semântica imutável: de acordo com esta semântica, se um arquivo pode ser compartilhado por vários processos, ele é marcado como imutável, ou seja, seu conteúdo somente pode ser lido e não pode ser modificado. É a forma mais simples de garantir a consistência do conteúdo do arquivo entre os processos que compartilham seu acesso, sendo por isso usada em alguns sistemas de arquivos distribuídos.

Semântica UNIX: toda modificação em um arquivo é imediatamente visível a todos os processos que mantêm aquele arquivo aberto; Essa semântica é a mais comum em sistemas de arquivos locais, ou seja, para acesso a arquivos nos dispositivos locais;

Semântica de sessão: considera que cada processo usa um arquivo em uma sessão, que inicia com a abertura do arquivo e que termina com seu fechamento. Modificações em um arquivo feitas em uma sessão somente são visíveis na mesma sessão e pelas sessões que iniciarem depois do encerramento da mesma, ou seja, depois que o processo fechar o arquivo; assim, sessões concorrentes de acesso a um arquivo compartilhado podem ver conteúdos distintos para o mesmo arquivo. Esta semântica é normalmente aplicada a sistemas de arquivos de rede, usados para acesso a arquivos em outros computadores;

Semântica de transação: uma transação é uma sequência de operações de leitura e escrita em um ou mais arquivos, emitidas por um processo e delimitadas por comandos de início e fim de transação (begin ... end), como em um sistema de bancos de dados. Todas as modificações parciais do arquivo durante a execução de uma transação não são visíveis às demais transações, somente após sua conclusão. Pode-se afirmar que a semântica de transação é similar à semântica de sessão, mas aplicada a cada transação (sequência de operações) e não ao período completo de uso do arquivo (da abertura ao fechamento).

Capítulo 24 – Sistemas de arquivos

(08) Questão 01. Apresente a arquitetura de gerência de arquivos presente em um sistema operacional típico, explicando seus principais elementos constituintes.

Os principais elementos que realizam a implementação de arquivos no sistema operacional estão organizados e detalhados a seguir:

Dispositivos: como discos rígidos ou bancos de memória flash, são os responsáveis pelo armazenamento de dados.

Controladores: são os circuitos eletrônicos dedicados ao controle dos dispositivos físicos. Eles são acessados através de portas de entrada/saída, interrupções e canais de acesso direto à memória (DMA).

Drivers: interagem com os controladores de dispositivos para configurá-los e realizar as transferências de dados entre o sistema operacional e os dispositivos. Como cada controlador define sua própria interface, também possui um driver específico. Os drivers ocultam as diferenças entre controladores e fornecem às camadas superiores do núcleo uma interface padronizada para acesso aos dispositivos de armazenamento.

Gerência de blocos: gerencia o fluxo de blocos de dados entre as camadas superiores e os dispositivos de armazenamento. Como os discos são dispositivos orientados a blocos, as operações de leitura e escrita de dados são sempre feitas com blocos de dados, e nunca com bytes individuais. As funções mais importantes desta camada são efetuar o mapeamento de blocos lógicos nos blocos físicos do dispositivo e o *caching/buffering* de blocos.

Sistemas Operacionais:

Alocação de arquivos: realiza a alocação dos arquivos sobre os blocos lógicos oferecidos pela camada de gerência de blocos. Cada arquivo é visto como uma sequência de blocos lógicos que deve ser armazenada nos blocos dos dispositivos de forma eficiente, robusta e flexível.

Sistema de arquivos virtual: o VFS (Virtual File System) constrói as abstrações de diretórios e atalhos, além de gerenciar as permissões associadas aos arquivos e as travas de acesso compartilhado. Outra responsabilidade importante desta camada é manter o registro de cada arquivo aberto pelos processos, como a posição da última operação no arquivo, o modo de abertura usado e o número de processos que estão usando o arquivo.

Interface do sistema de arquivos: conjunto de chamadas de sistema oferecidas aos processos do espaço de usuários para a criação e manipulação de arquivos.

Bibliotecas de entrada/saída: usam as chamadas de sistema da interface do núcleo para construir funções padronizadas de acesso a arquivos para cada linguagem de programação.

(09) Questão 18. Explique como é efetuada a gerência de espaço livre através de bitmaps.

Além de manter informações sobre que blocos são usados por cada arquivo no disco, a camada de alocação de arquivos deve manter um registro atualizado de quais blocos estão livres, ou seja, não estão ocupados por nenhum arquivo ou metadado. Isto é importante para obter rapidamente blocos no momento de criar um novo arquivo ou aumentar um arquivo existente.

Capítulo 25 – Diretórios e atalhos

(10) Questão 02. Do ponto de vista lógico, quais as principais diferenças entre a estrutura de diretórios Unix e Windows?