Capítulo 11
Implementação do sistema de arquivos

Capítulo 11: Implementação do sistema de arquivos

- Estrutura do sistema de arquivos
- Implementação do sistema de arquivos
- Implementação de diretório
- Métodos de alocação
- Gerenciamento do espaço livre
- Eficiência e desempenho
- Recuperação
- Sistemas de arquivo estruturados em log
- NFS
- Exemplo: Sistema de arquivos WAFL

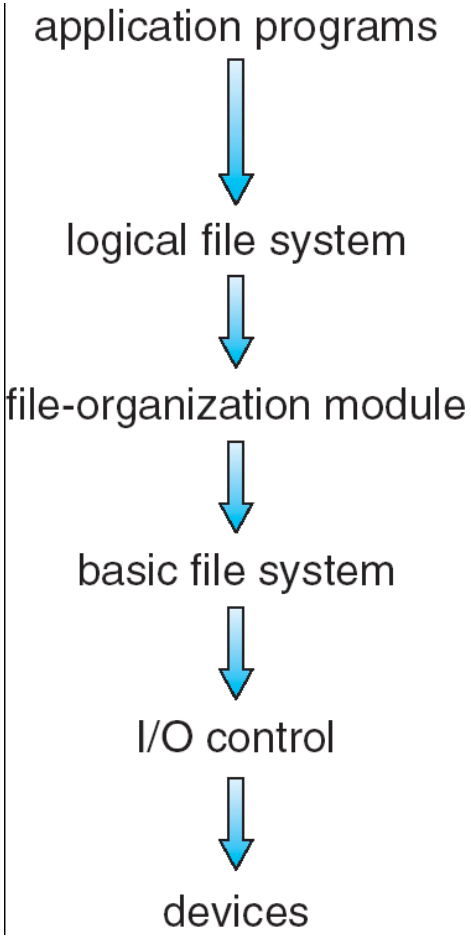
Objetivos

- Descrever os detalhes da implementação de sistemas de arquivos e estruturas de diretório locais
- Descrever a implementação de sistemas de arquivo remotos
- Discutir algoritmos e opções de alocação em bloco e bloco livre

Estrutura do sistema de arquivos

- Estrutura de arquivo
  - Unidade de armazenamento lógico
  - Coleta de informações relacionadas
- Sistema de arquivos reside no armazenamento secundário (discos)
- Sistema de arquivos organizado em camadas
- **Bloco de controle de arquivo –** estrutura de armazenamento consistindo em informações sobre um arquivo

Sistema de arquivos em camadas



Um bloco de controle de arquivo típico

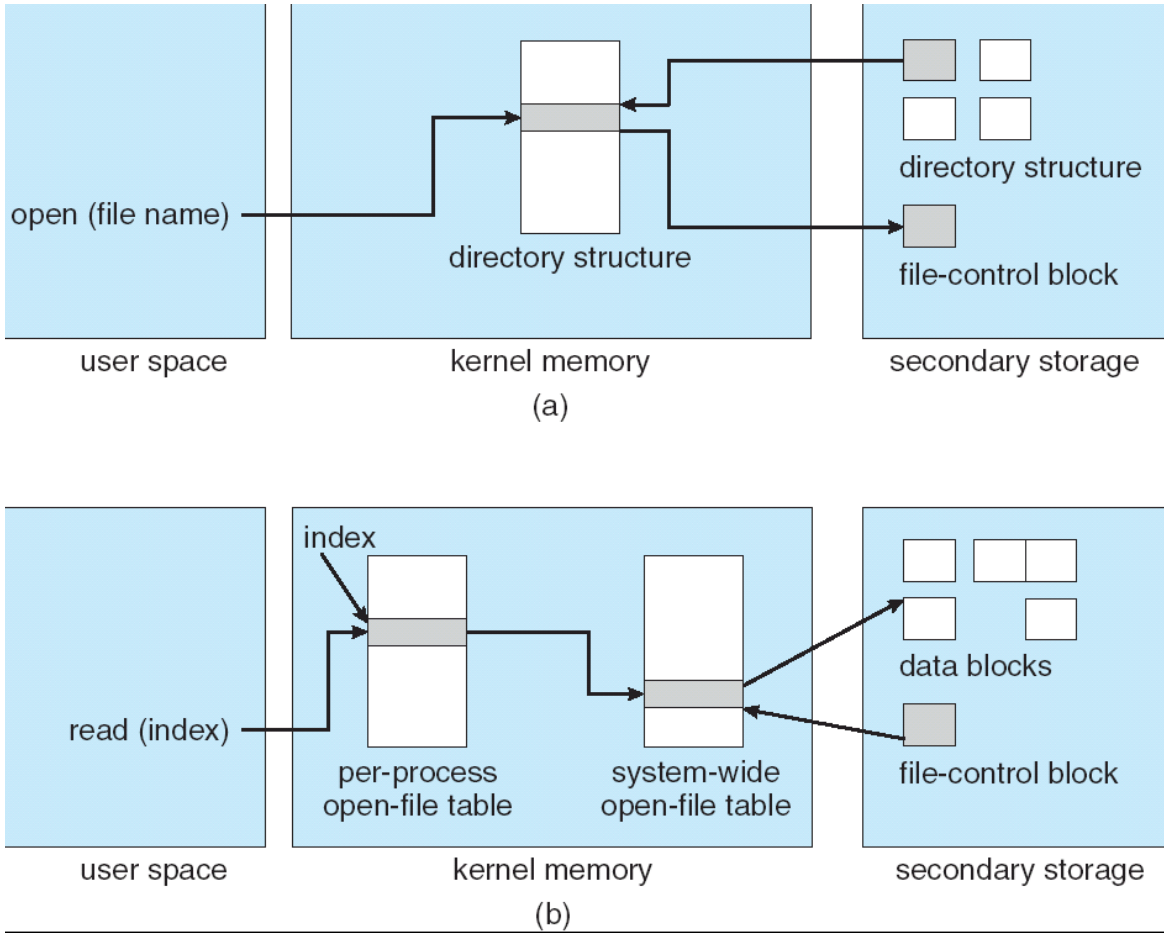
file permissions
file dates (create, access, write)
file owner, group, ACL
file size
file data blocks or pointers to file data blocks

*ACL: Access Control List

Estruturas de sistema de arquivos na memória

- A figura a seguir ilustra as estruturas necessárias do sistema de arquivos fornecidas pelos sistemas operacionais.
- Figura 12.3(a) refere-se à abertura de um arquivo.
- Figura 12.3(b) refere-se à leitura de um arquivo.

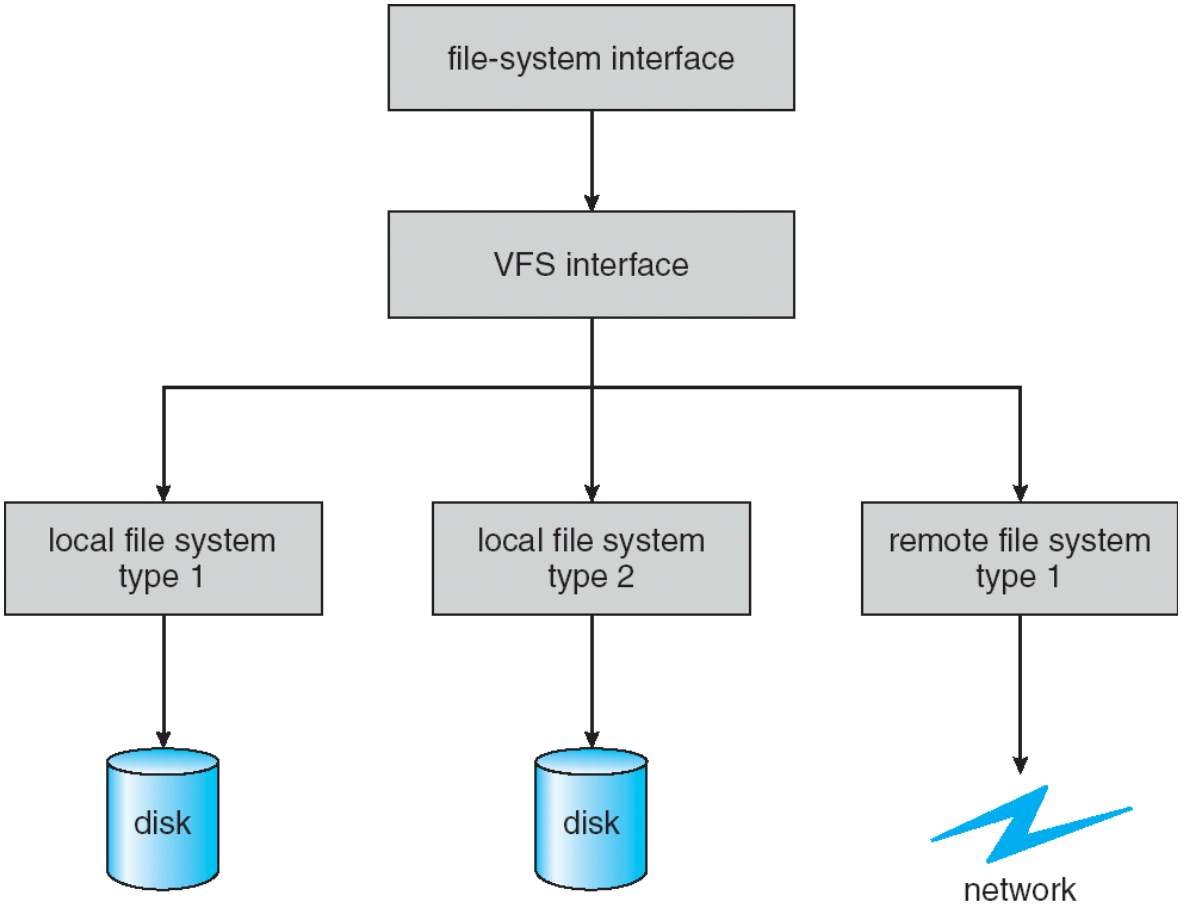
Estruturas de sistema de arquivos na memória



Sistemas de arquivo virtuais

- Virtual File Systems (VFS) oferecem um modo orientado a objeto para implementar sistemas de arquivos.
- VFS permite que a mesma interface de chamada do sistema (a API) seja usada para diferentes tipos de sistemas de arquivos.
- A API é para a interface VFS, e não qualquer tipo específico de sistema de arquivos.

Visão esquemática do Virtual File System



Implementação do diretório

- **Lista linear** dos nomes com ponteiro para os blocos de dados.
  - simples de programar
  - demorado para executar
- **Tabela de hash** –lista linear com estrutura de dados em hash.
  - diminui tempo de busca de diretório

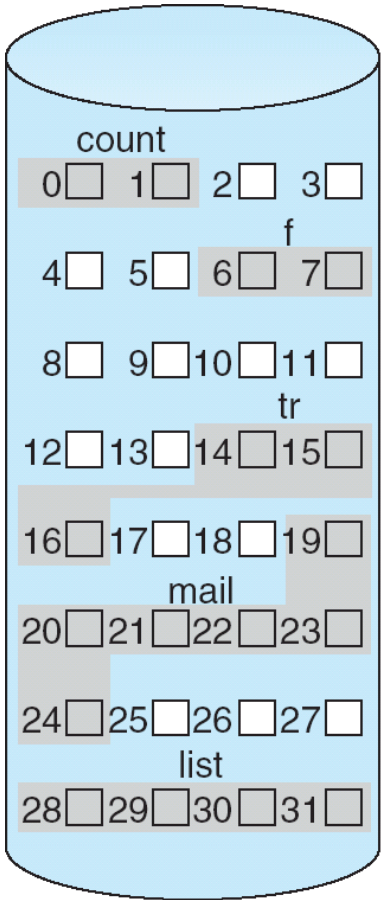
Métodos de alocação

- Um método de alocação refere-se a como os blocos de disco são alocados para arquivos:
- **Alocação contígua**
- **Alocação vinculada (interligada)**
- **Alocação indexada**

Alocação contígua

- Cada arquivo ocupa um conjunto de blocos contíguos no disco
- Simples – requer somente local inicial (# bloco) e tamanho (número de blocos)
- Acesso aleatório
- Desperdício de espaço (problema de alocação dinâmica de armazenamento)
- Arquivos não podem crescer

Alocação contígua de espaço em disco

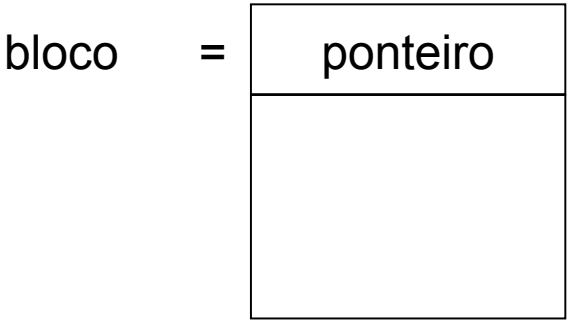


directory

file	start	length
count	0	2
tr	14	3
mail	19	6
list	28	4
f	6	2

Alocação vinculada

- Cada arquivo é uma lista vinculada (encadeada) de blocos de disco: blocos podem estar espalhados por todo o disco



Alocação vinculada

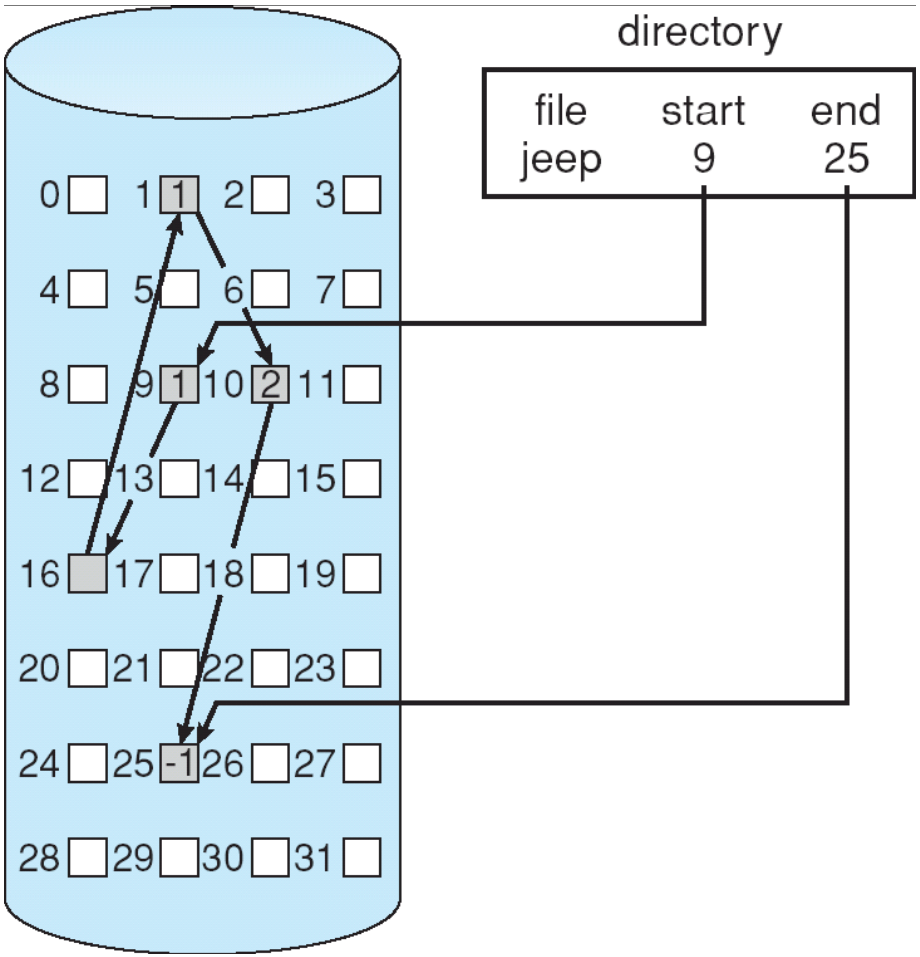
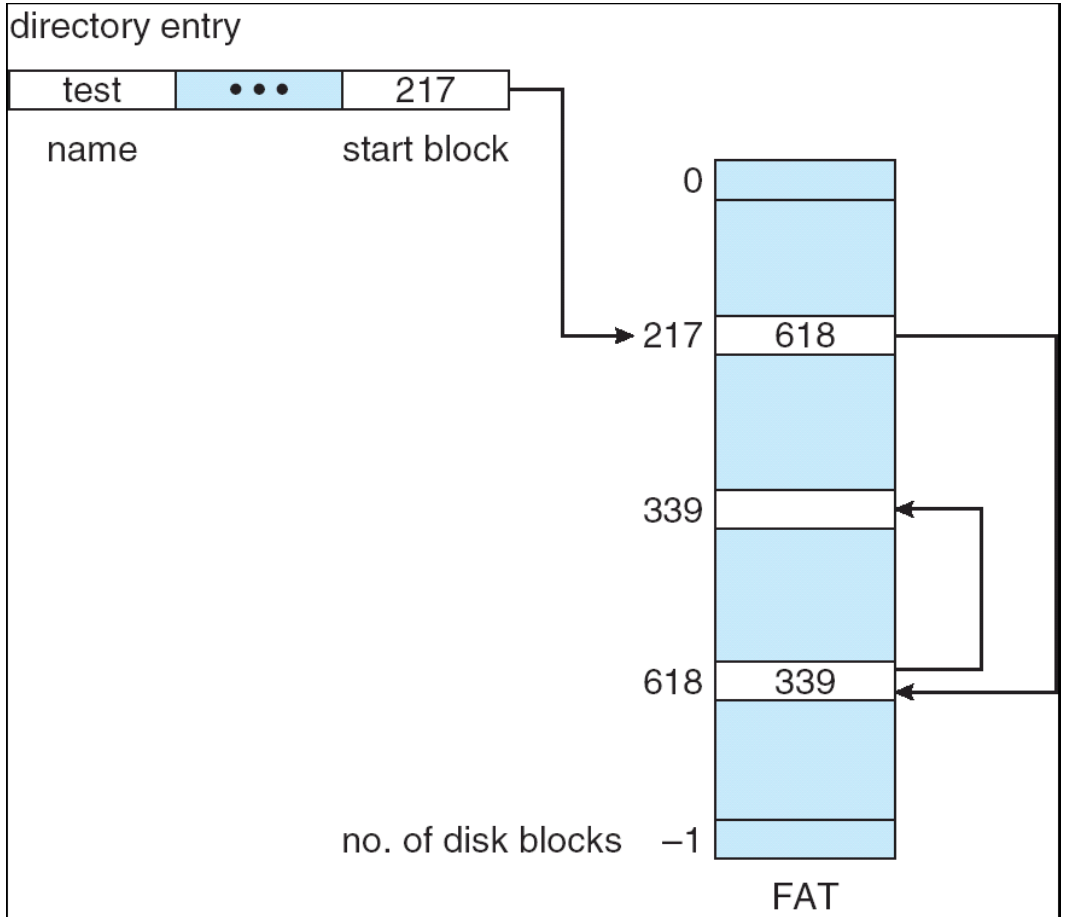


Tabela de alocação de arquivos (FAT)



Alocação indexada

- Reúne todos os ponteiros no *bloco de índice.*
- Visão lógica.

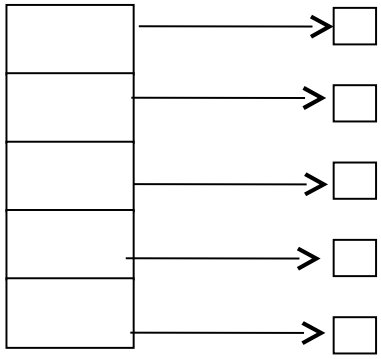
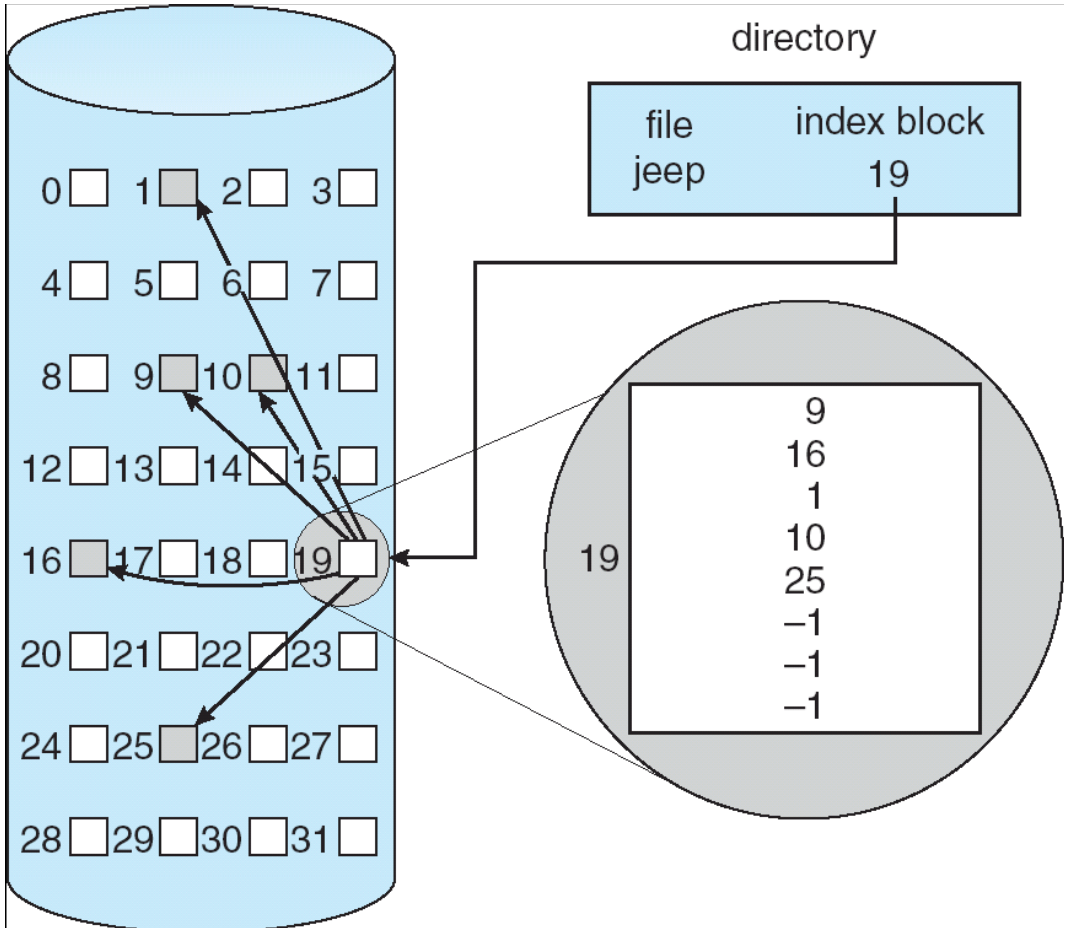
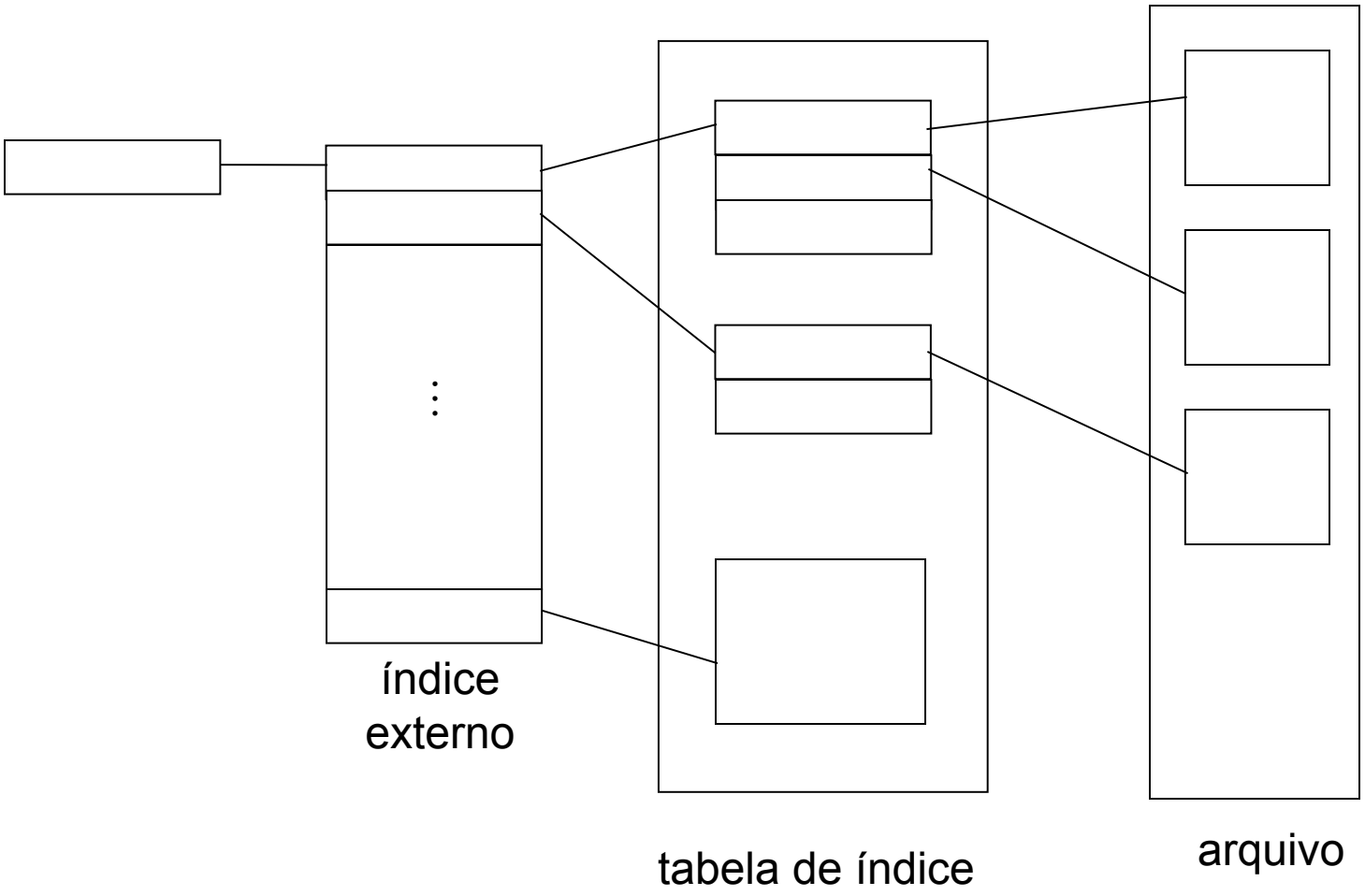


Tabela de índice

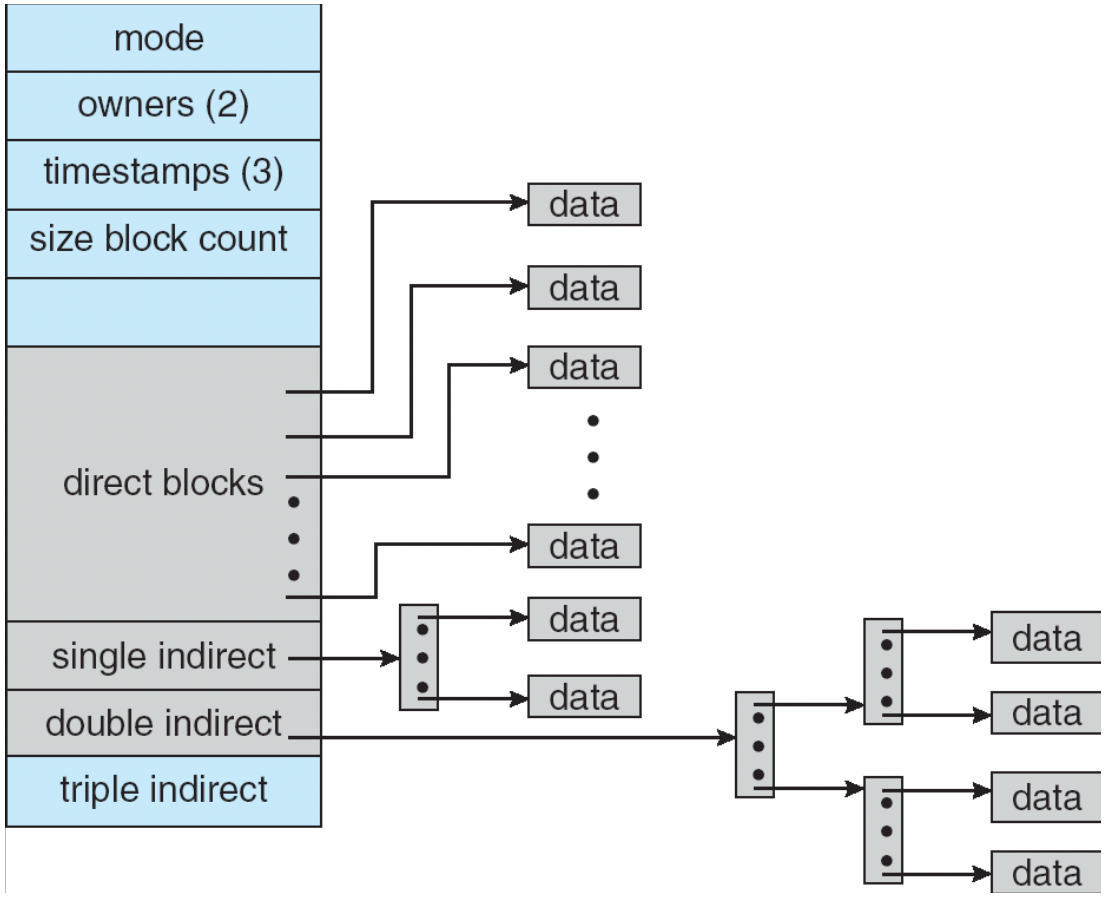
Exemplo de alocação indexada



Alocação indexada – mapeamento (cont.)

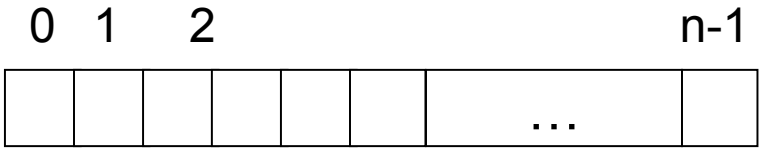


Esquema combinado: UNIX (4K bytes por bloco)



Gerenciamento de espaço livre

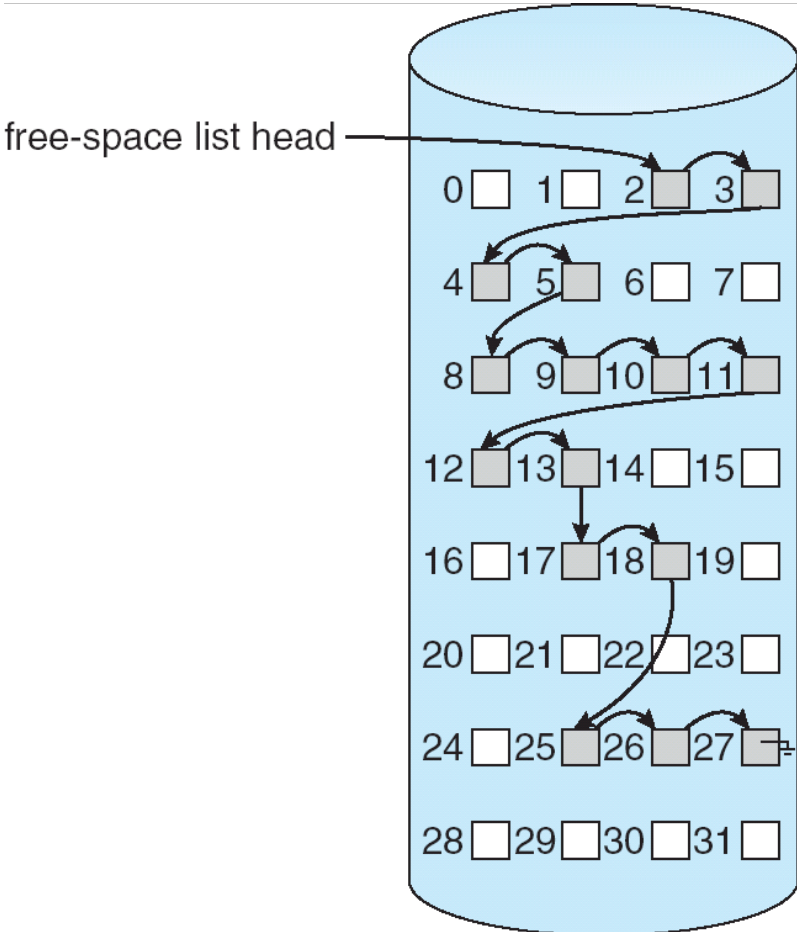
- Vetor de bits (*n* blocos)



bit[*i*] = 0: bloco[*i*] livre

1: bloco[*i*] ocupado

Lista vinculada de espaço livre em disco



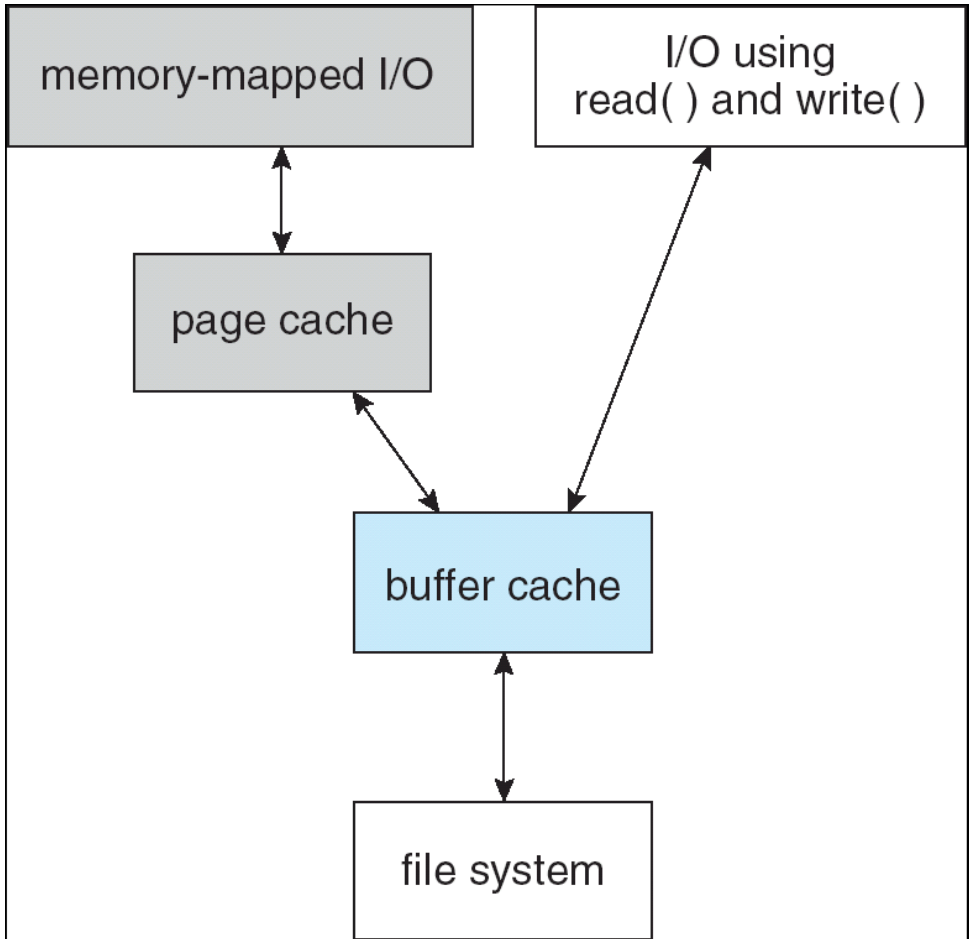
Eficiência e desempenho

- Eficiência dependente de:
  - Algoritmos para alocação de disco e diretórios
  - tipos de dados mantidos na entrada de diretório do arquivo
- Desempenho
  - cache de disco – seção separada da memória principal para blocos usados freqüentemente
  - free-behind e read-ahead – técnicas para otimizar acesso seqüencial

Cache de página

- Um **cache de página** guarda páginas ao invés de blocos de disco, usando técnicas de memória virtual
- E/S mapeada na memória usa um cache de página
- E/S de rotina pelo sistema de arquivos usa o cache de buffer

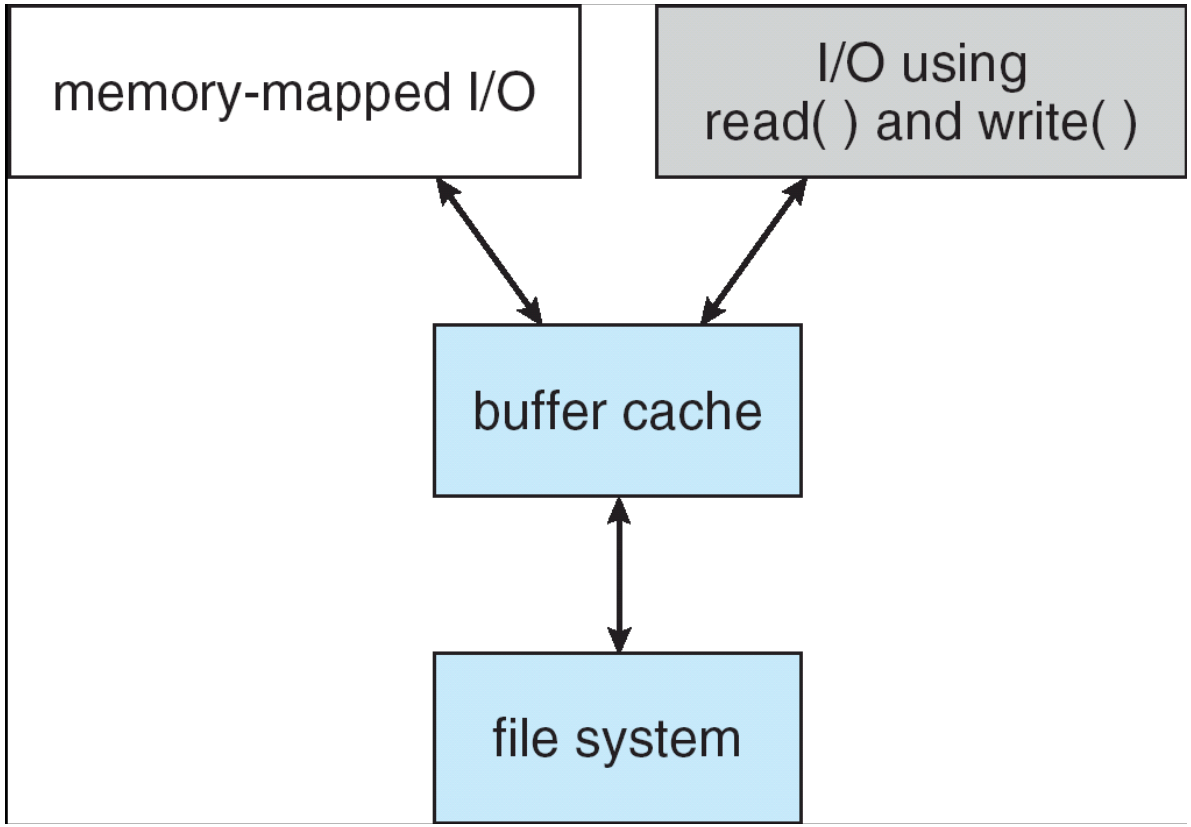
E/S sem um cache de buffer unificado



Cache de buffer unificado

- Um cache de buffer unificado usa o mesmo cache de página para guardar páginas mapeadas na memória e E/S normal do sistema de arquivos

E/S usando um cache de buffer unificado



Recuperação

- Verificação de consistência – compara dados na estrutura de diretórios com blocos de dados no disco, e tenta consertar inconsistências
- Usa programas do sistema para o **backup** de dados do disco para outro dispositivo de armazenamento (fita magnética, por exemplo)
- Recupera arquivo ou disco perdido, **restaurando** dados do backup

Sistemas de arquivos estruturados em log

- Sistemas de arquivos **estruturados em log** registram cada atualização no sistema de arquivos como uma **transação**
- Todas as transações são gravadas em um **log**
  - Uma transação é considerada **confirmada** depois de gravada no log
  - Porém, o sistema de arquivos pode ainda não estar atualizado
- As transações no log são gravadas assincronamente no sistema de arquivos
  - Quando o sistema de arquivos é modificado, a transação é removida do log
- Se o sistema de arquivos falhar, todas as transações restantes no log ainda precisam ser realizadas

O Network File System (NFS) da Sun

- Uma implementação e uma especificação de um sistema de software para acessar arquivos remotos pelas LANs (ou WANs)
- A implementação faz parte dos sistemas operacionais Solaris e SunOS rodando em estações de trabalho Sun usando um protocolo de datagrama não confiável (protocolo UDP/IP e Ethernet)

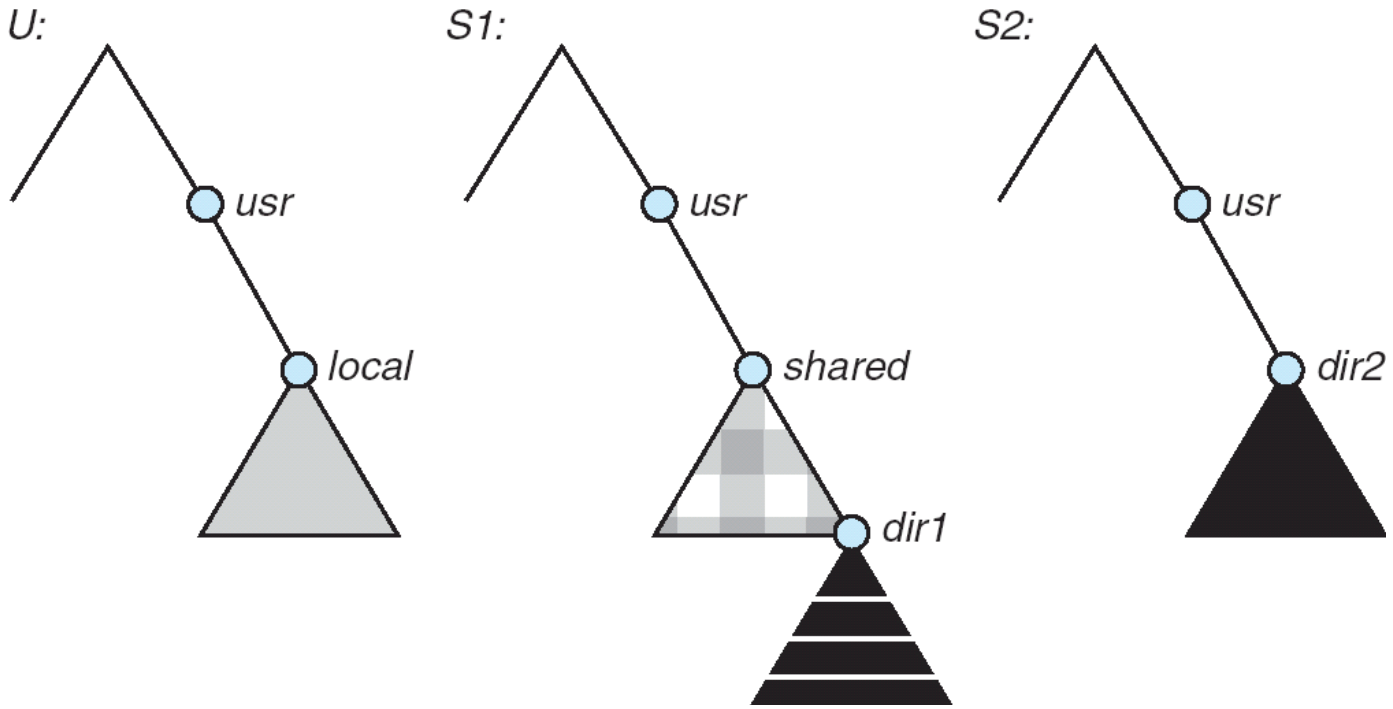
NFS (cont.)

- Estações de trabalho interconectadas vistas como um conjunto de máquinas independentes com sistemas de arquivos independentes, permitindo o compartilhando entre esses sistemas de arquivos de modo transparente
  - Um diretório remoto é montado sobre um diretório do sistema de arquivos local
    - O diretório montado se parece com uma sub-árvore integral do sistema de arquivos local, substituindo a sub-árvore descendo do diretório local
  - Especificação do diretório remoto para a operação mount não é transparente; o nome de host do diretório remoto precisa ser fornecido
    - Arquivos no diretório remoto podem então ser acessados de modo transparente
  - Sujeito a verificação de direitos de acesso, potencialmente qualquer sistema de arquivos (ou diretório dentro de um sistema de arquivos) pode ser montado remotamente em cima de qualquer diretório local

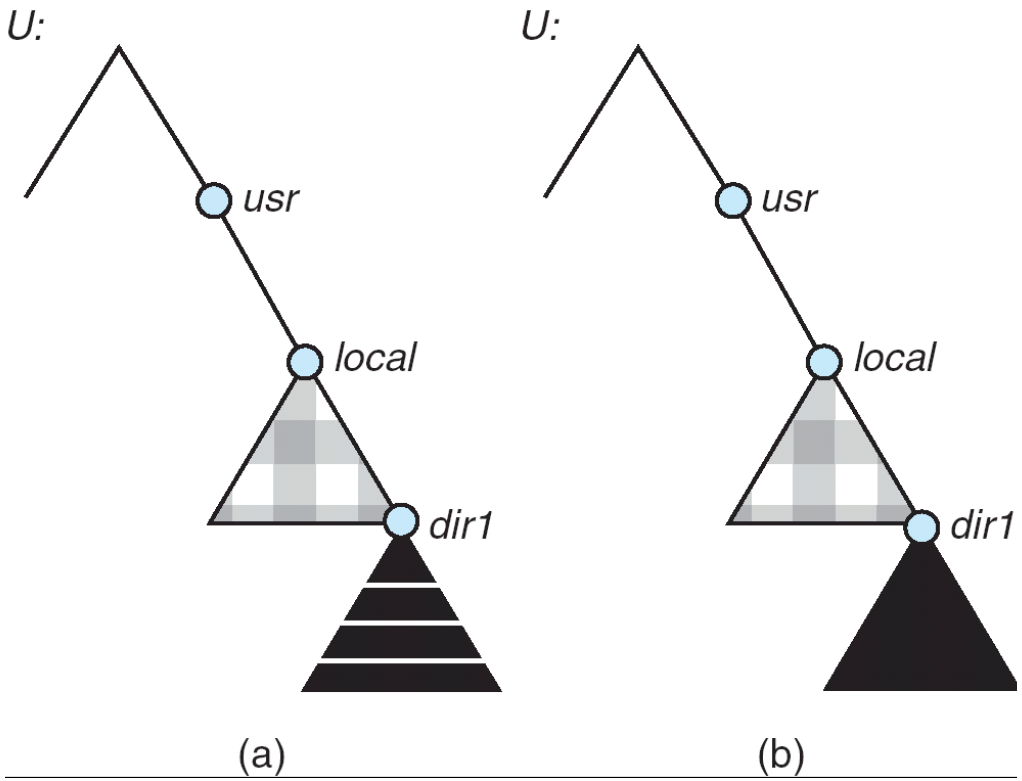
NFS (cont.)

- NFS foi criado para operar em ambiente heterogêneo de diferentes máquinas, sistemas operacionais e arquiteturas de rede; especificações NFS independentes desses meios
- Essa independência é alcançada por meio dos primitivos RPC em cima de um protocolo External Data Representation (XDR) usado entre duas interfaces independentes de implementação

Três sistemas de arquivo independentes



Montagem no NFS



Montagens

Montagens em cascata

Protocolo de montagem do NFS

- Estabelece conexão lógica inicial entre servidor e cliente
- Operação mount inclui nome do diretório remoto a ser montado e nome da máquina servidora que o armazena
  - Requisição de mount é mapeada na RPC correspondente e encaminhada para servidor de mount executando na máquina servidora
  - Lista de exportação – especifica sistemas de arquivos locais que o servidor exporta para montagem, junto com os nomes das máquinas que têm permissão para montá-los
- Seguindo uma requisição de mount de acordo com sua lista de exportação, o servidor retorna um descritor de arquivo – uma chave para outros acessos
- Descritor de arquivo – um identificador do sistema de arquivos e um número de inode para identificar o diretório montado dentro do sistema de arquivo exportado
- A operação mount muda apenas a visão do usuário e não afeta o lado do servidor

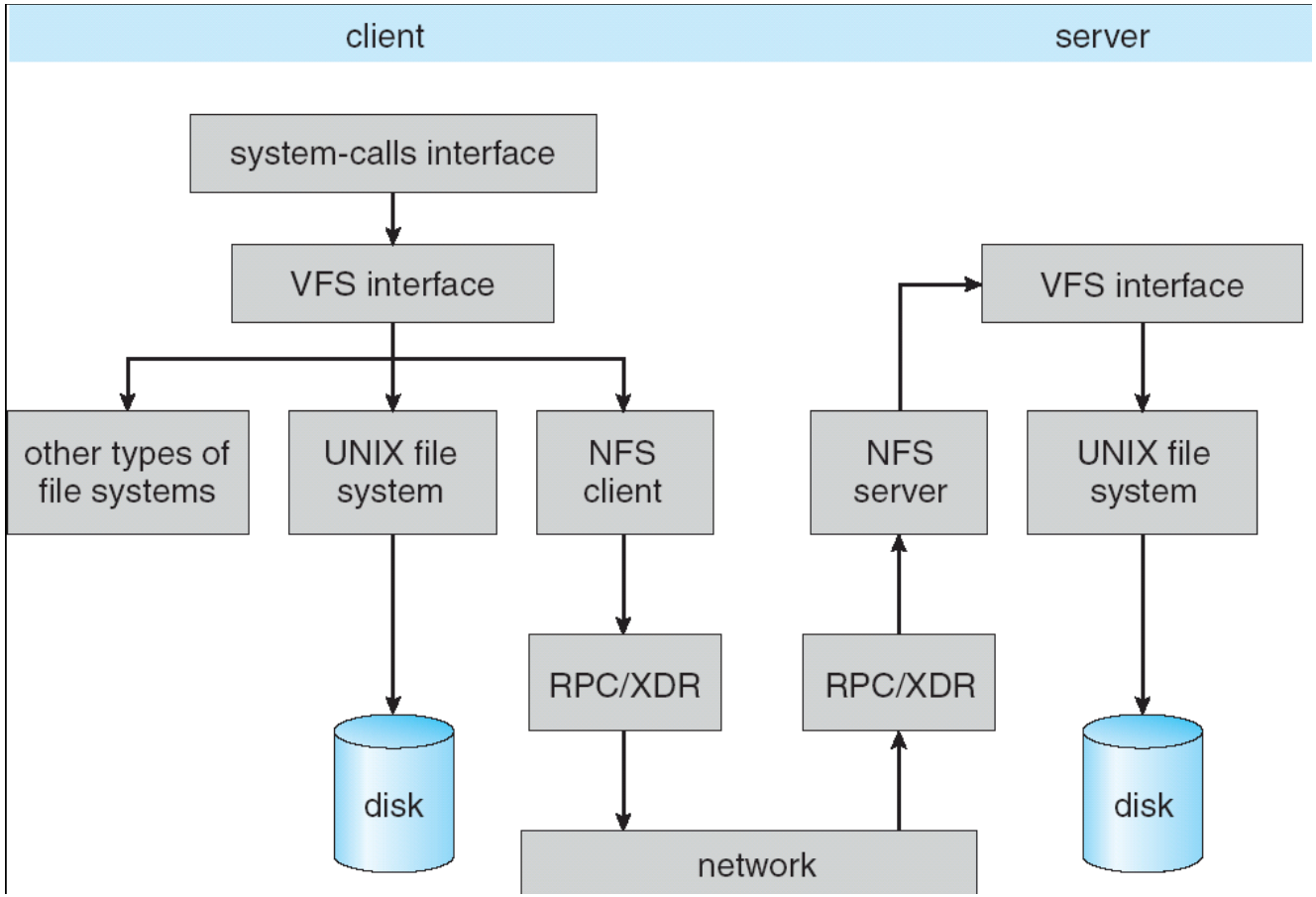
Protocolo NFS

- Oferece um conjunto de chamadas de procedimento remoto para operações de arquivo remoto. Os procedimentos admitem as seguintes operações:
  - procurar um arquivo dentro de um diretório
  - ler um conjunto de entradas de diretório
  - manipular links e diretórios
  - acessar atributos do arquivo
  - ler e gravar arquivos
- Dados modificados precisam ser confirmados no disco do servidor antes que resultados sejam retornados ao cliente
- O protocolo NFS não oferece mecanismos de controle de concorrência

Três principais camadas da arquitetura NFS

- Interface de sistema de arquivos do UNIX (baseada nas chamadas **open, read, write** e **close,** e **descritores de arquivo**)
- Camada do *Virtual File System* (VFS) – distingue arquivos locais dos remotos, e arquivos locais são diferenciados também de acordo com seus tipos do sistema de arquivos
  - O VFS ativa operações específicas ao sistema de arquivos para lidar com requisições locais, de acordo com seus tipos
  - Chama as procedures do protocolo NFS para requisições remotas
- Camada de serviço do NFS – camada inferior da arquitetura
  - Implementa o protocolo NFS

Visão esquemática da arquitetura do NFS



Tradução de nome de caminho do NFS

- Realizada dividindo-se o caminho em nomes componentes e realizando-se uma chamada de pesquisa do NFS separada para cada par de nome de componente e vnode de diretório
- Para tornar a pesquisa mais rápida, um cache de pesquisa do nome de diretório no cliente mantém os vnodes para nomes de diretório remotos

Operações remotas do NFS

- Correspondência quase um-para-um entre chamadas de sistema regulares do UNIX e as RPCs do protocolo NFS (exceto abrir e fechar arquivos)
- NFS adere ao paradigma de serviço remoto, mas emprega técnicas de buffering e caching por questão de desempenho
- Cache de blocos de arquivo – quando um arquivo é aberto, o kernel verifica com o servidor remoto se deve apanhar ou revalidar os atributos em cache
  - Blocos de arquivo em cache são usados apenas se os atributos em cache correspondentes estiverem atualizados
- Cache de atributo de arquivo – o cache de atributo é atualizado sempre que novos atributos chegam do servidor
- Clientes não liberam blocos de escrita adiada até que o servidor confirme que os dados foram gravados em disco

Final do Capítulo 11