Universidade do Vale do Itajaí
Escola do Mar, Ciência e Tecnologia
Engenharia de Computação

Interface do sistema de arquivos

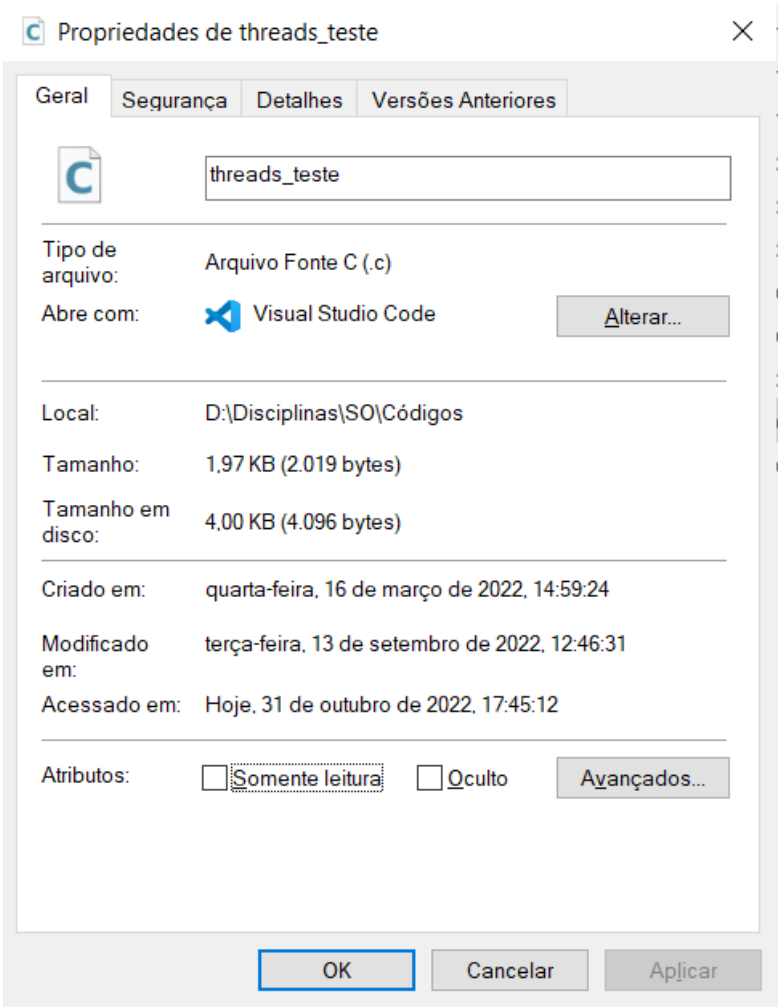
Conceito de arquivo

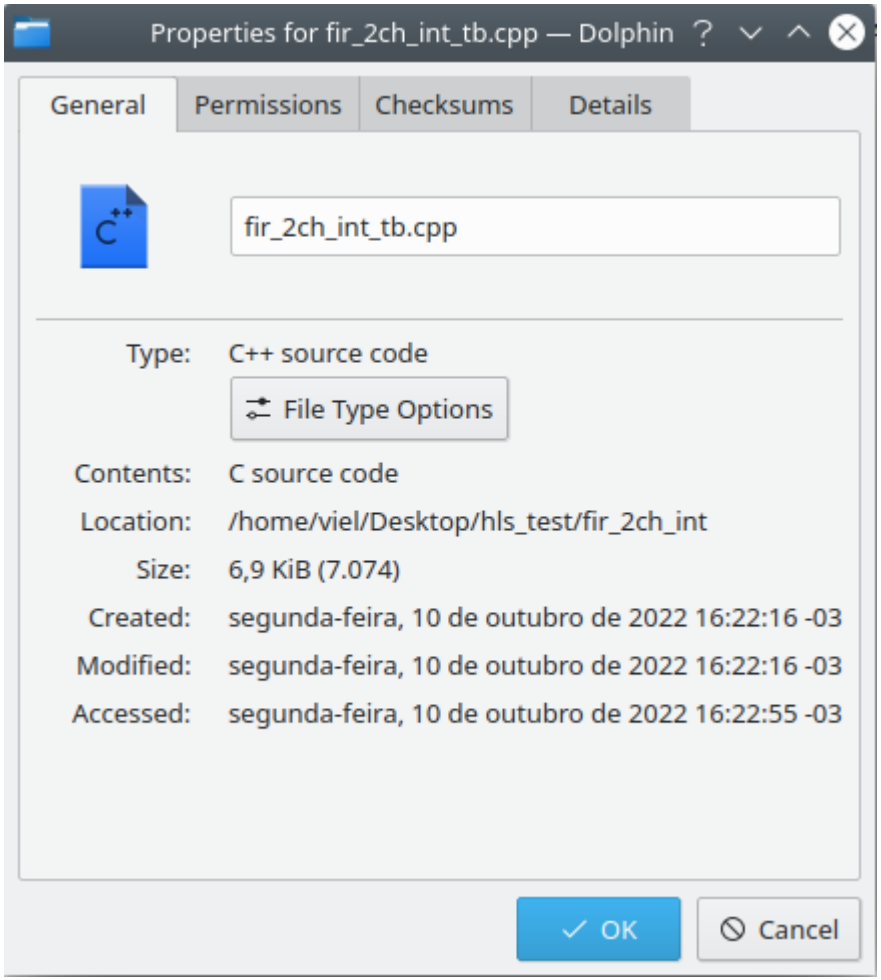
- Espaço de endereço lógico contíguo
- Tipos:
  - Dados
    - numérico
    - caracter
    - binário
  - Programa
- Conteúdo definido pelo criador do arquivo
  - Muitos tipos
    - Considere arquivo de texto, arquivo de código fonte, arquivo executável

Atributos de arquivo

- **Nome** – apenas informações mantidas em forma de leitura humana
- **Identificador** – tag (número) exclusivo identifica arquivo dentro do sistema de arquivos
- **Tipo** – necessário para sistemas que suportam diferentes tipos
- **Localização** – ponteiro para localização de arquivo no dispositivo
- **Tamanho** – tamanho do arquivo atual
- **Proteção** – controles que podem fazer leitura, escrita, execução
- **Hora, data e identificação do usuário** – dados para monitoramento de proteção, segurança e uso
- Informações sobre arquivos são mantidas na estrutura do diretório, que é mantida no disco
- Muitas variações, incluindo atributos de arquivo estendidos, como checkum de arquivo
- Informações mantidas na estrutura do diretório

Janela de informações do arquivo no Windows e Linux





Operações com arquivos

- Arquivo é um **tipo de dados abstratos**
- **Criar**
- **Escrever –** na localização do **ponteiro de escrita**
- **Ler –** pelo localicação do **ponteiro de leitura**
- **Reposicionamento dentro do arquivo - buscar**
- **Excluir**
- **Truncar**
- ***Abrir(F_i)*** – pesquisar a estrutura do diretório em disco para entrada ***F_i***, e mover o conteúdo da entrada para a memória
- ***Fechar (F_i)*** – mover o conteúdo de entrada ***F_i*** na memória para estrutura de diretório em disco

Arquivos Abertos

- Vários dados são necessários para gerenciar arquivos abertos:
  - **Tabela de arquivos abertos**: rastreia arquivos abertos
  - Ponteiro de arquivo: ponteiro para a última localização de leitura/gravação, por processo que tem o arquivo aberto
  - **Contagem de arquivos aberto**: contador de número de vezes que um arquivo é aberto – para permitir a remoção de dados da tabela de arquivos abertos quando os últimos processos os fecham
  - Localização do arquivo no disco: cache de informações de acesso a dados
  - Permissões de acesso: informações do modo de acesso por processo

Bloqueio de arquivos aberto

- Fornecido por alguns sistemas operacionais e sistemas de arquivos
  - Semelhante aos bloqueios de leitor-escritor: conceito de mutex ou semáforo com seção crítica
  - **Bloqueio compartilhado** semelhante ao bloqueio do leitor – vários processos podem adquirir simultaneamente
  - **Bloqueio exclusivo** semelhante a bloqueio do escritor
- Faz a mediação de acesso a um arquivo
- Obrigatório ou consultivo:
  - **Obrigatório** – o acesso é negado dependendo dos bloqueios mantidos e solicitados
  - **Consultivo** – processos podem encontrar status de fechaduras e decidir o que fazer

Tipos de arquivos – Nome, Extensão

file type	usual extension	function
executable	exe, com, bin or none	ready-to-run machine-language program
object	obj, o	compiled, machine language, not linked
source code	c, cc, java, pas, asm, a	source code in various languages
batch	bat, sh	commands to the command interpreter
text	txt, doc	textual data, documents
word processor	wp, tex, rtf, doc	various word-processor formats
library	lib, a, so, dll	libraries of routines for programmers
print or view	ps, pdf, jpg	ASCII or binary file in a format for printing or viewing
archive	arc, zip, tar	related files grouped into one file, sometimes compressed, for archiving or storage
multimedia	mpeg, mov, rm, mp3, avi	binary file containing audio or A/V information

Estrutura de arquivos

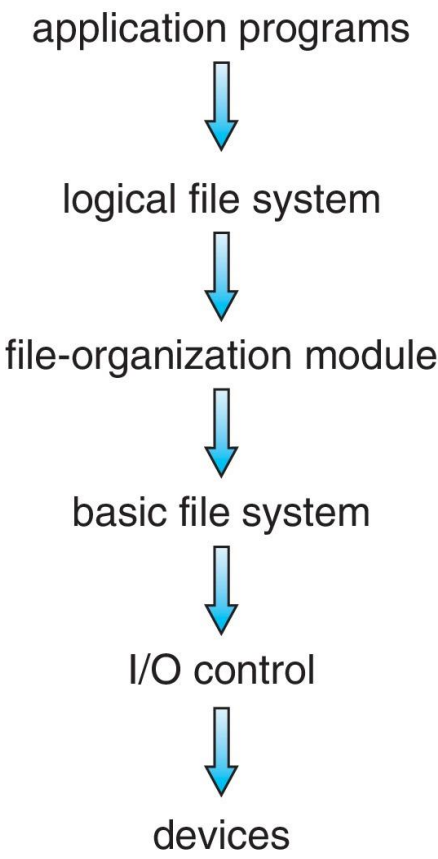
- Nenhum - sequência de palavras, bytes – também conhecido como puro (RAW)
- Estrutura de gravação simples
  - Linhas
  - Comprimento fixo
  - Comprimento variável
- Estruturas complexas
  - Documento formatado
  - Relocatable load file
- Pode simular os dois últimos com o primeiro método inserindo caracteres de controle apropriados
- Quem decide:
  - Sistema Operacional
  - Programa

Estrutura do sistema de arquivos

- Estrutura de arquivo
  - Unidade de armazenamento lógico
  - Coleta de informações relacionadas
- O sistema de arquivos reside no armazenamento secundário (discos)
  - Interface de usuário fornecida para armazenamento, mapeamento lógico para físico
  - Permitindo que os dados sejam armazenados, localizados e recuperados facilmente
- O disco fornece reescrita no local e acesso aleatório
  - Transferências de I/O realizadas em blocos de setores (geralmente 512 bytes)
- Bloco de controle de arquivo (FCB) – estrutura de armazenamento que consiste em informações sobre um arquivo
- Driver de dispositivo controla o dispositivo físico
- Sistema de arquivos organizado em camadas

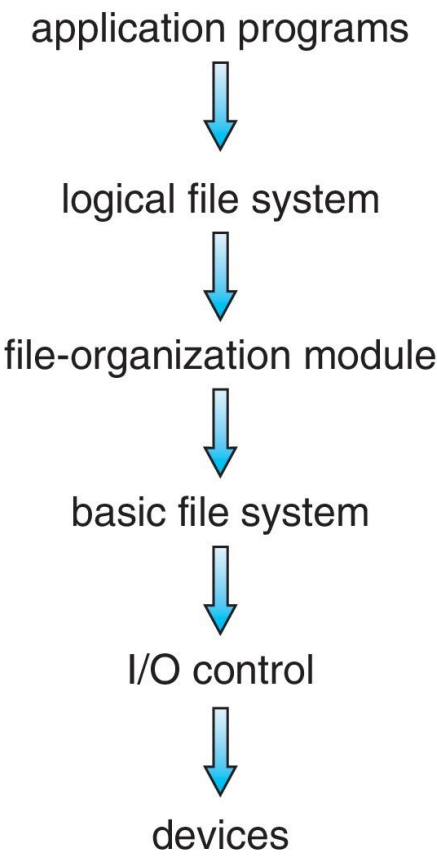
Camadas do sistema de arquivos

- Drivers de dispositivo gerenciam dispositivos de E/S na camada de controle de E/S
  - Comandos dados como “ler drive1, cilindro 72, trilha 2, setor 10, na localização de memória 1060” para o controlador de hardware
- Sistema de arquivos básico dado comando como “recuperar bloco 123” se traduz em driver de dispositivo
- Também gerencia buffers de memória e caches
  - Os buffers mantêm os dados em trânsito
  - Os caches armazenam dados usados com frequência
- O módulo de organização de arquivos compreende arquivos, endereços lógicos e blocos físicos
- Traduz o bloco lógico # para o bloco físico #
- Gerencia espaço livre e alocação de disco



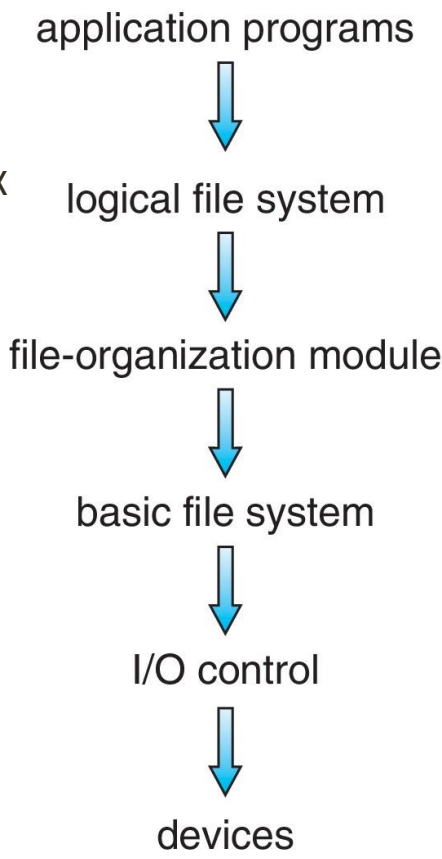
Camadas do sistema de arquivos

- O sistema de arquivos lógicos gerencia informações de metadados
  - Traduz o nome do arquivo em número de arquivo, identificador de arquivo, localização, mantendo blocos de controle de arquivo (inodes no UNIX)
  - Gerenciamento de diretório
  - Proteção
- A disposição em camadas é útil para reduzir a complexidade e a redundância, mas adiciona sobrecarga e pode diminuir o desempenho
- As camadas lógicas podem ser implementadas por qualquer método de codificação de acordo com o projetista do sistema operacional



Camadas do sistema de arquivos

- Muitos sistemas de arquivos, às vezes muitos dentro de um sistema operacional
  - Cada um com seu próprio formato (CD-ROM é ISO 9660; Unix tem UFS, FFS; Windows tem FAT, FAT32, NTFS, bem como disquete, CD, DVD Blu-ray, Linux tem mais de 130 tipos, com sistema de arquivos estendido ext3 e líder ext4; além de sistemas de arquivos distribuídos, etc.)
  - Novos: ZFS, GoogleFS, Oracle ASM, FUSE

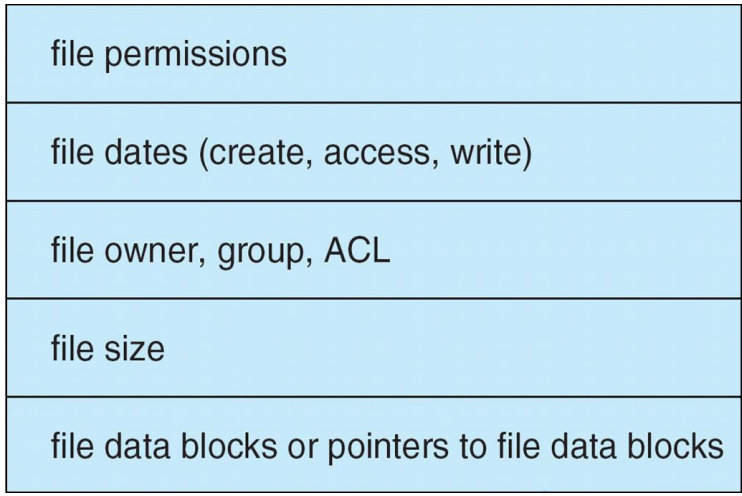


Operações do sistema de arquivos

- Temos chamadas de sistema no nível da API, mas como implementamos suas funções?
  - Estruturas em disco e em memória
- O bloco de controle de inicialização contém as informações necessárias ao sistema para inicializar o sistema operacional a partir desse volume
  - Necessário se o volume contiver sistema operacional, geralmente o primeiro bloco de volume
- Bloco de controle de volume (superbloco, tabela de arquivos mestre) contém detalhes do volume
  - Número total de blocos, número de blocos livres, tamanho do bloco, ponteiros ou matriz de bloco livre
- A estrutura de diretórios organiza os arquivos
  - Nomes e números de inode, tabela de arquivos mestre

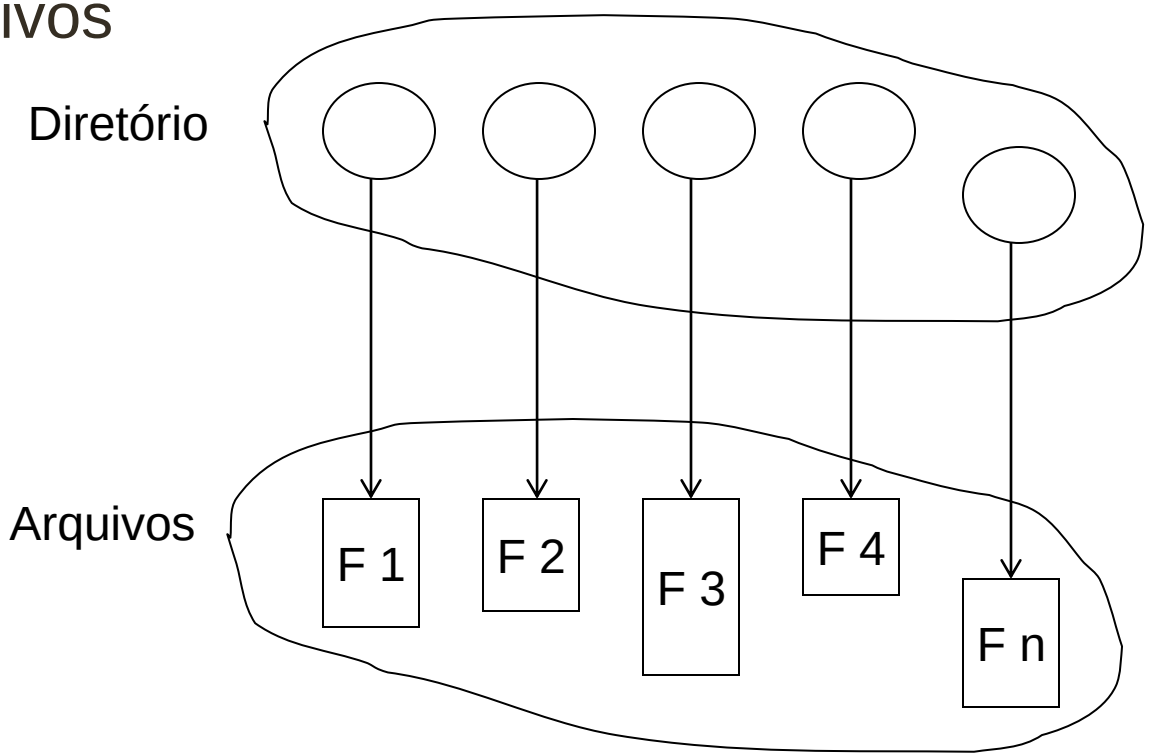
Operações do sistema de arquivos

- File Control Block (FCB) – de cada arquivo – contém muitos detalhes sobre o arquivo
- Normalmente número de inode, permissões, tamanho, datas
- O NFTS armazena na tabela de arquivos mestre usando estruturas de banco de dados relacionais



Estrutura de Diretório

- Uma coleção de nós contendo informações sobre todos os arquivos

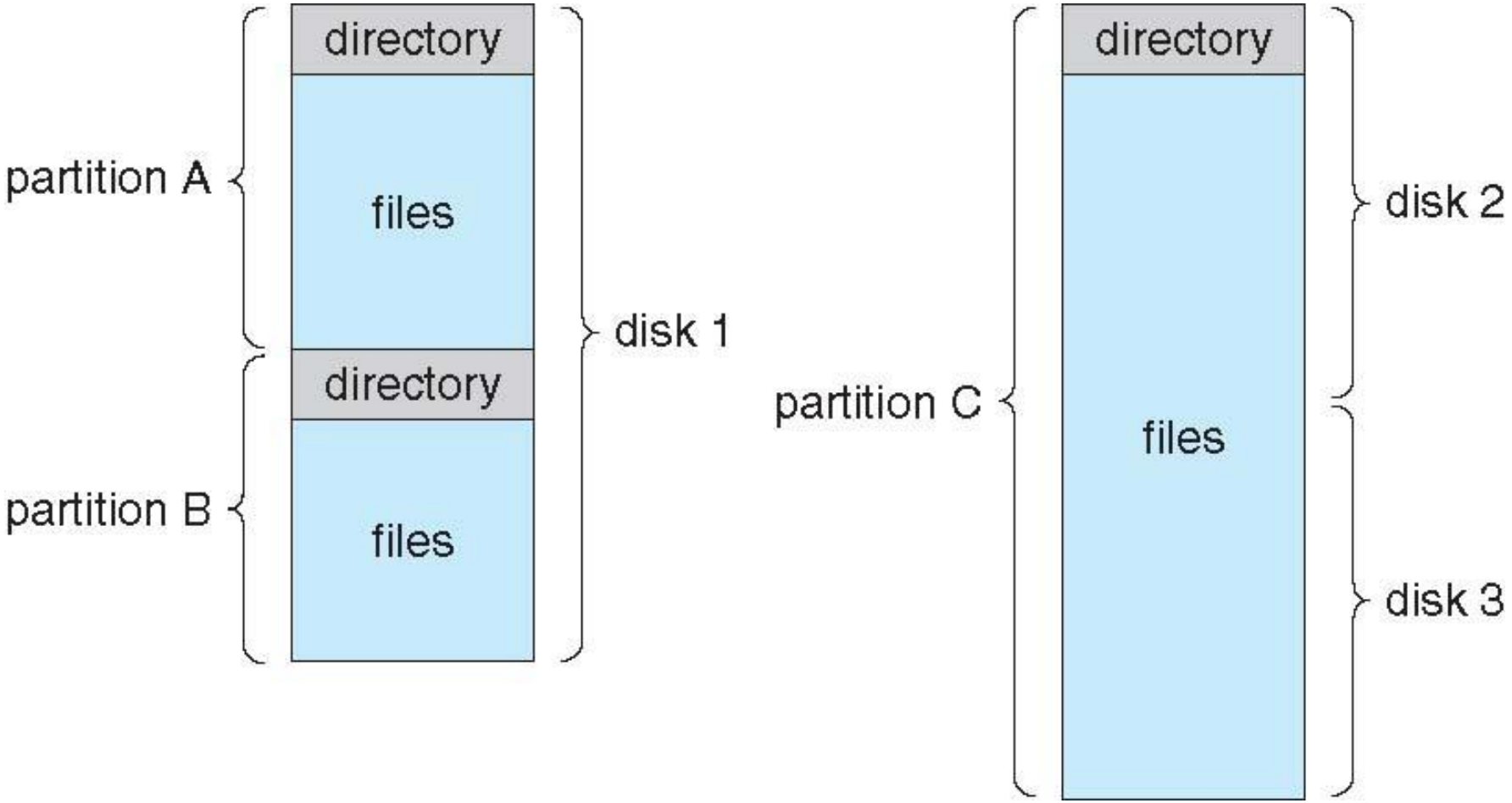


Tanto a estrutura do diretório quanto os arquivos residem em disco

Estrutura de disco

- O disco pode ser subdividido em **partições**
  - Discos ou partições podem ser **RAID** - protegido contra o falha
  - Disco ou partição podem ser usados **raw** – sem um sistema de arquivos, ou **formatado** com um sistema de arquivos
  - Partições também conhecidas como minidiscos, fatias
  - Entidade contendo sistema de arquivos conhecido como a **volume**
- Cada volume contendo sistema de arquivos também rastreia as informações desse sistema de arquivos em **diretório do dispositivo** ou **tabela de volume de conteúdo**
- Assim como **sistemas de arquivos de uso geral,** há muitos **sistemas de arquivos de propósito especial**
  - Frequentemente todos dentro do mesmo sistema operacional ou computador

Uma organização típica do sistema de arquivos



Tipos de sistemas de arquivos

- Falamos principalmente de sistemas de arquivos de uso geral
- Mas os sistemas frequentemente têm sistemas de arquivos, alguns de propósito geral e alguns especiais
- Considere que Solaris tem
  - tmpfs – sistema de arquivo volátil baseada em memória para I/O rápido e temporário
  - objfs – interface na memória do kernel para obter símbolos de kernel para depuração
  - ctfs – sistema de arquivos de contrato para o gerenciamento de daemons
  - lofs – sistema de arquivos loopback permite um sistema de arquivo para ser acessado no lugar de outro
  - procfs – interface do kernel para processar estruturas
  - ufs, zfs – sistemas de arquivos de propósito geral

Operações realizadas em diretório

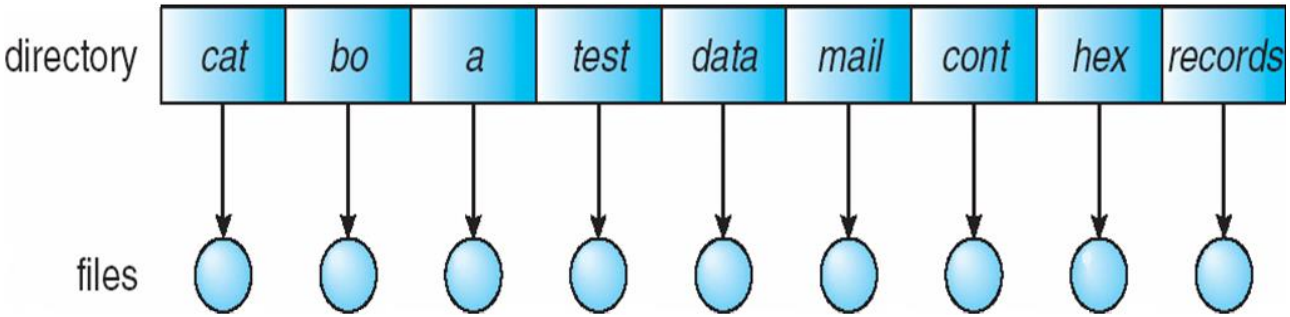
- Procure por um arquivo
- Crie um arquivo
- Exclua um arquivo
- Liste um diretório
- Renomeie um arquivo
- Atravesse o sistema de arquivos

Organização de Diretórios

- O diretório é organizado logicamente para obter
  - Eficiência – localização de um arquivo rapidamente
  - Nomeação – conveniente para os usuários
    - Dois usuários podem ter o mesmo nome para arquivos diferentes
    - O mesmo arquivo pode ter vários nomes diferentes
  - Agrupamento – agrupamento lógico de arquivos por propriedades, (por exemplo, todos os programas Java, todos os jogos, ...)

Diretório de nível único

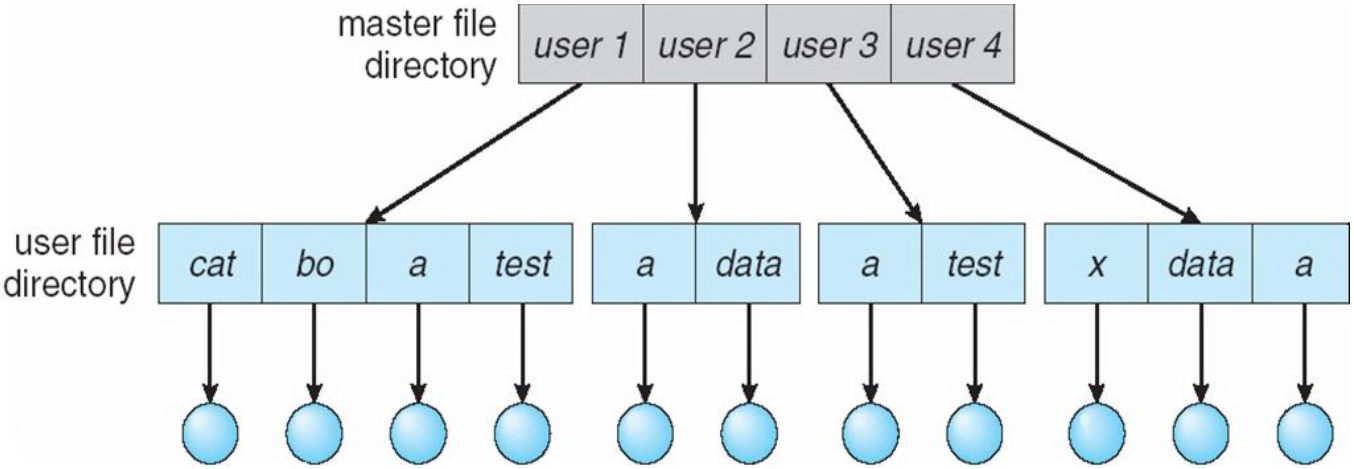
- Um único diretório para todos os usuários



- Problema de nomeação
- Problema de agrupamento

Diretório de dois níveis

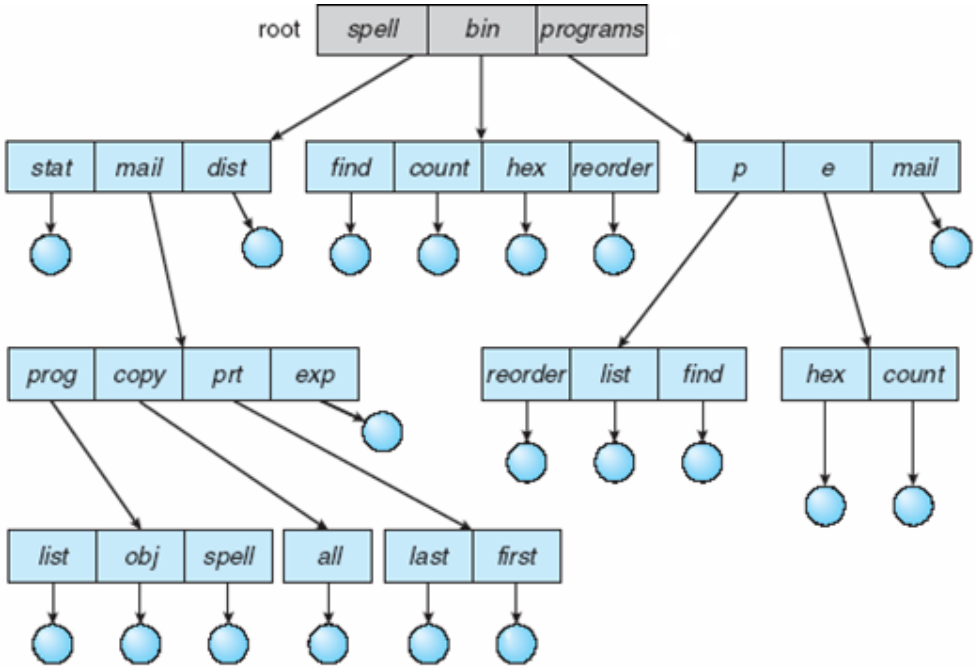
- Diretório separado para cada usuário



  - Nome do caminho
  - Pode ter o mesmo nome do arquivo para diferentes usuários
  - Busca eficiente
  - Sem capacidade de agrupamento

Diretórios estruturados por árvores

- Busca eficiente
- Capacidade de agrupamento
- Diretório atual (diretório de trabalho)
  - `cd /spell/mail/prog`
  - `type list`



Diretórios estruturados por árvores (Cont.)

- Nome de caminho **absoluto** ou **relativo**
- A criação de um novo arquivo é feita no diretório atual
- Exclua um arquivo

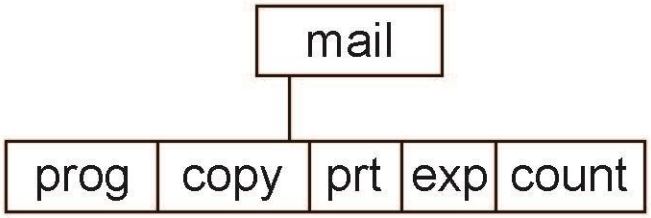
```
rm <file-name>
```

- A criação de um novo subdiretório é feita no diretório atual

```
mkdir <dir-name>
```

Exemplo: se no diretório atual **/mail**

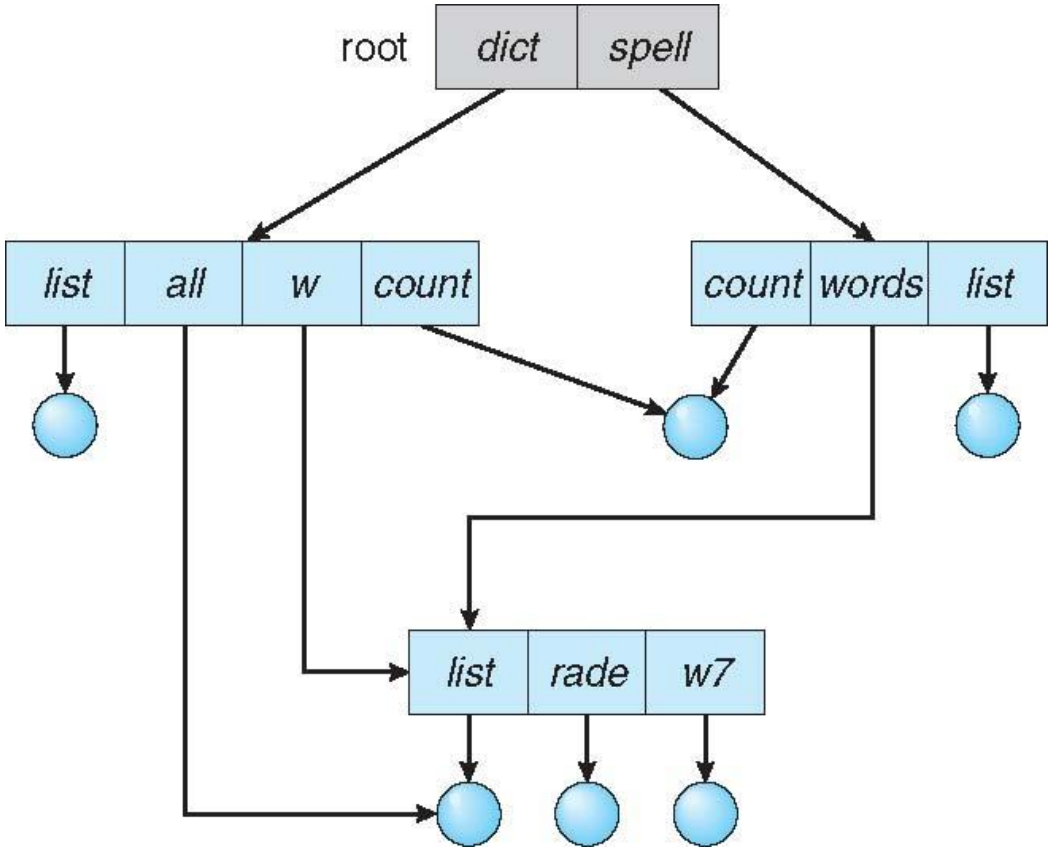
```
mkdir count
```



Deletar “mail” ⇒ excluindo todo a subárvore enraizado por “mail”

Diretório em grafo acíclico

- Tenha subdiretórios e arquivos compartilhados

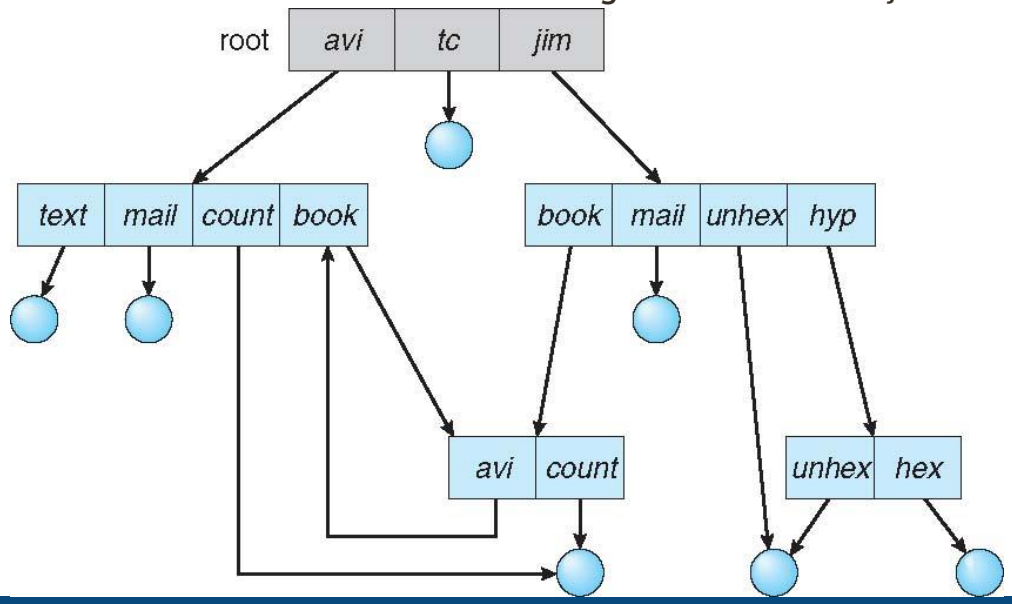


Diretório em grafo acíclico

- Dois nomes diferentes
- Se ***dict*** Exclui ***list*** $\Rightarrow$ ponteiro pendurado
- Soluções:
  - Backpointers, para que possamos excluir todos os ponteiros
    Tamanho variável registra um problema
  - Backpointers usando uma organização de lista encadeada
  - Solução de contagem de entrada
- Novo tipo de entrada do diretório
  - **Link** – outro nome (ponteiro) para um arquivo existente
  - **Resolver o link** – siga ponteiro para localizar o arquivo

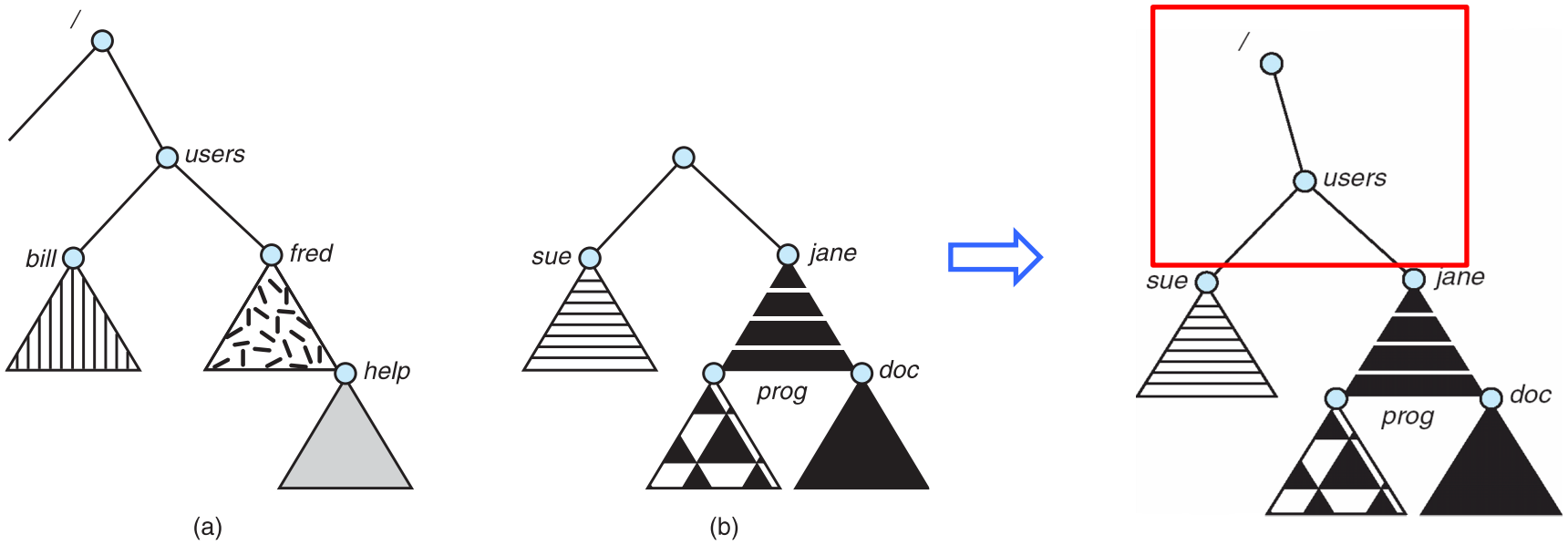
Diretório de grafo geral

- Como garantimos que não há ciclos?
  - Permitir apenas links para arquivar, não para subdiretórios
  - **Garbage collection** – solução parecida com a implementada em linguagens como Java e Rust
  - Toda vez que um novo link é adicionado use um algoritmo de detecção de ciclo para determinar se ele está OK



Montagem do sistema de arquivos

- Um sistema de arquivos deve ser **montado** antes que possa ser acessado
- Um sistema de arquivos desmontado (b) é montado em um **ponto de montagem**

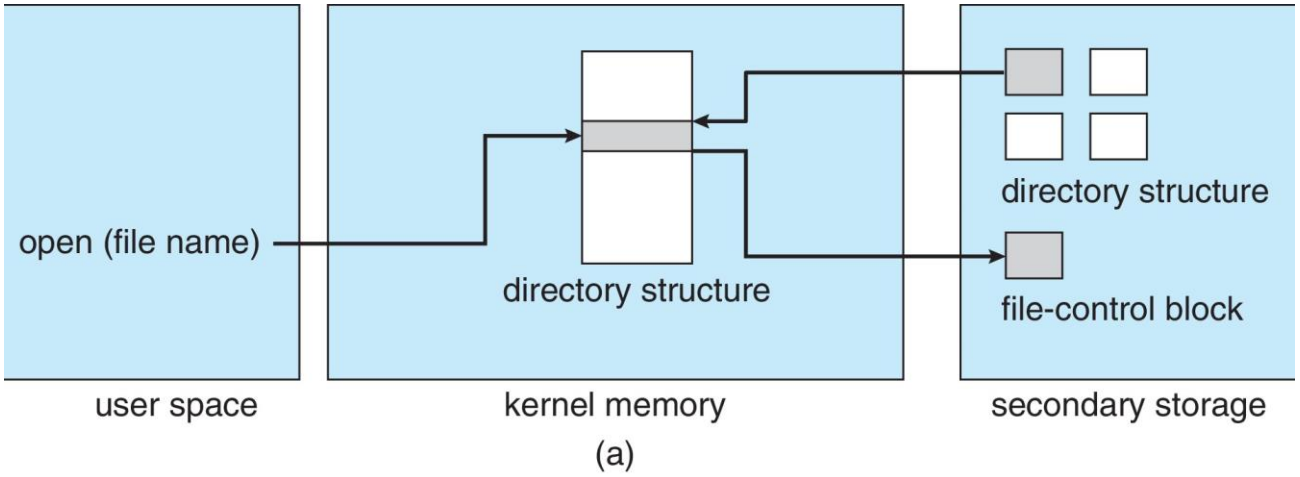


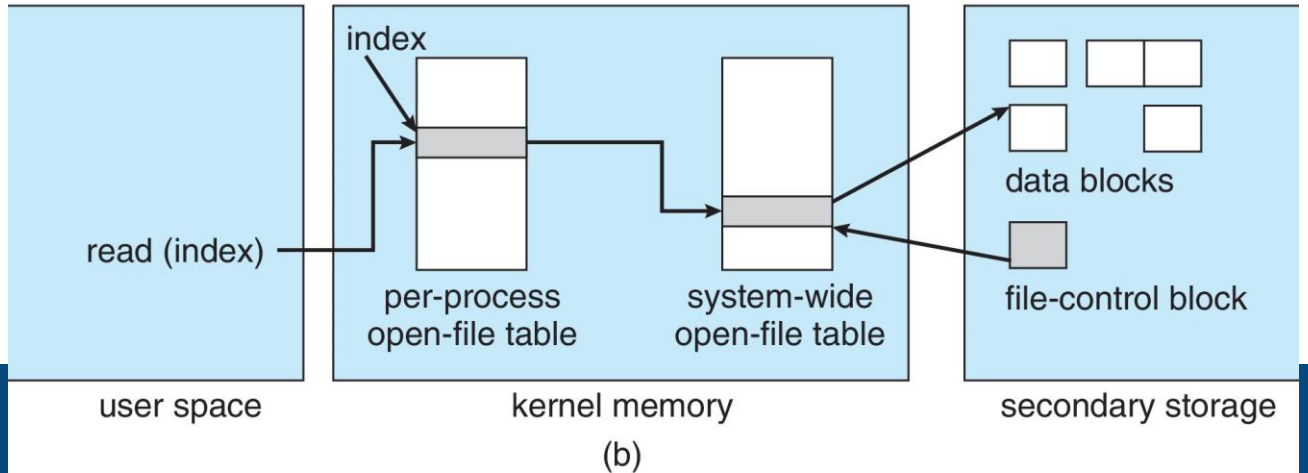
Estruturas do sistema de arquivos na memória

- Tabela de montagem que armazena montagens do sistema de arquivos, pontos de montagem, tipos de sistema de arquivos
- Tabela de arquivo aberto em todo o sistema contém uma cópia do FCB de cada arquivo e outras informações
- A tabela de arquivos abertos por processo contém ponteiros para as entradas apropriadas na tabela de arquivos abertos de todo o sistema, bem como outras informações

Estruturas do sistema de arquivos na memória

- a) refere-se à abertura de um arquivo e b) refere-se à leitura de um arquivo





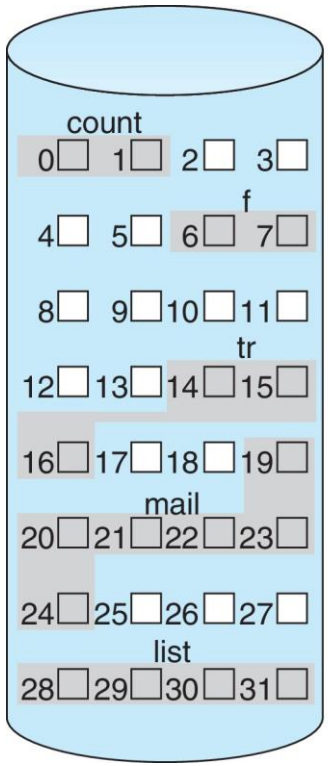
Implementação de diretório e indexação de arquivos

- Diretório são implementados como listas lineares (também usando o conceito de árvore) ou tabelas hash
  - Lista Linear: lista linear de nomes de arquivos com ponteiro para os blocos de dados, sendo simples de programar, mas demora para executar
  - Tabela hash: lista linear com estrutura de dados hash ajuda a diminuir o tempo de pesquisa de diretório, mas provoca colisões (ex: dois nomes de arquivos para o mesmo local)
    - Bom apenas se as entradas forem de tamanho fixo ou usar o método chained-overflow

- Quanto a forma de alocação (e indexação) dos arquivos, temos
  - Alocação contínua, alocação lincada e alocação indexada

Implementação de diretório e indexação de arquivos

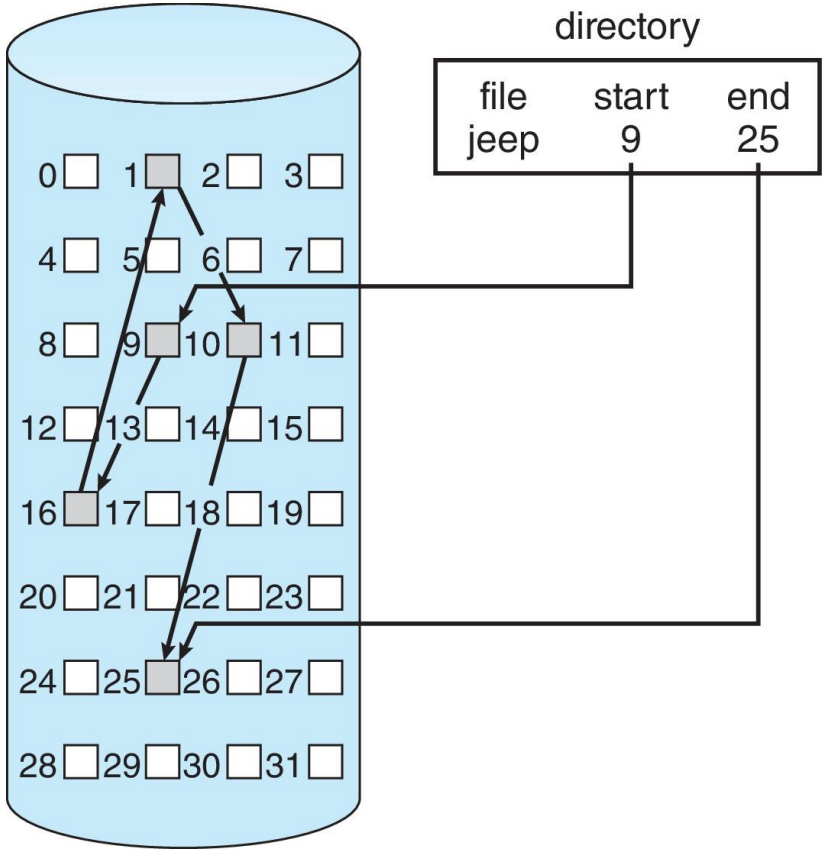
Alocação contínua



directory

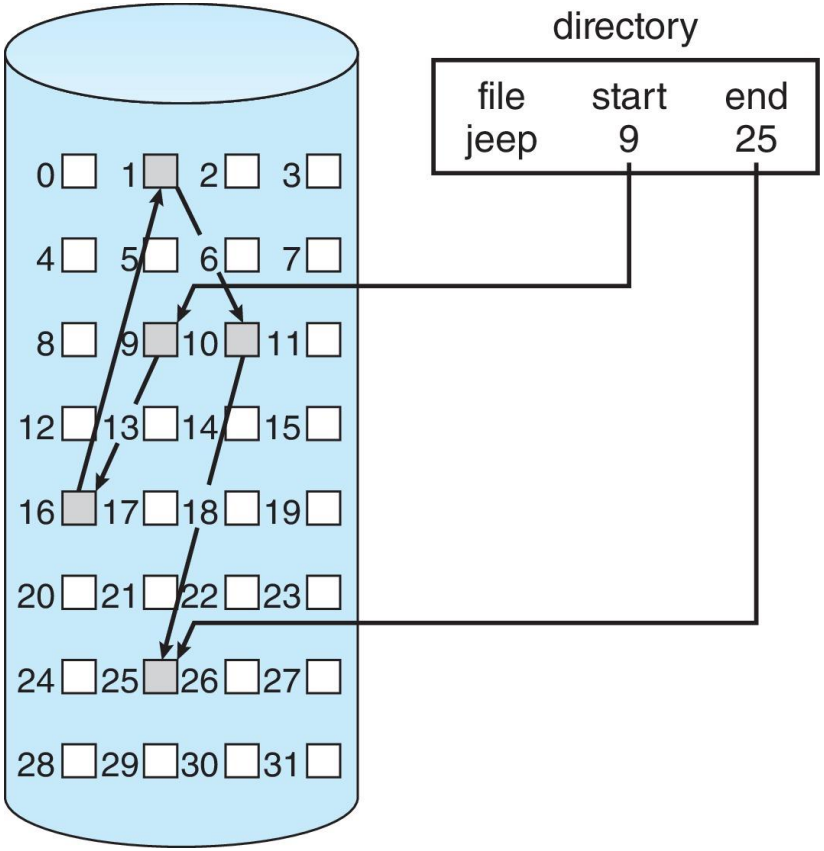
file	start	length
count	0	2
tr	14	3
mail	19	6
list	28	4
f	6	2

Alocação lincada

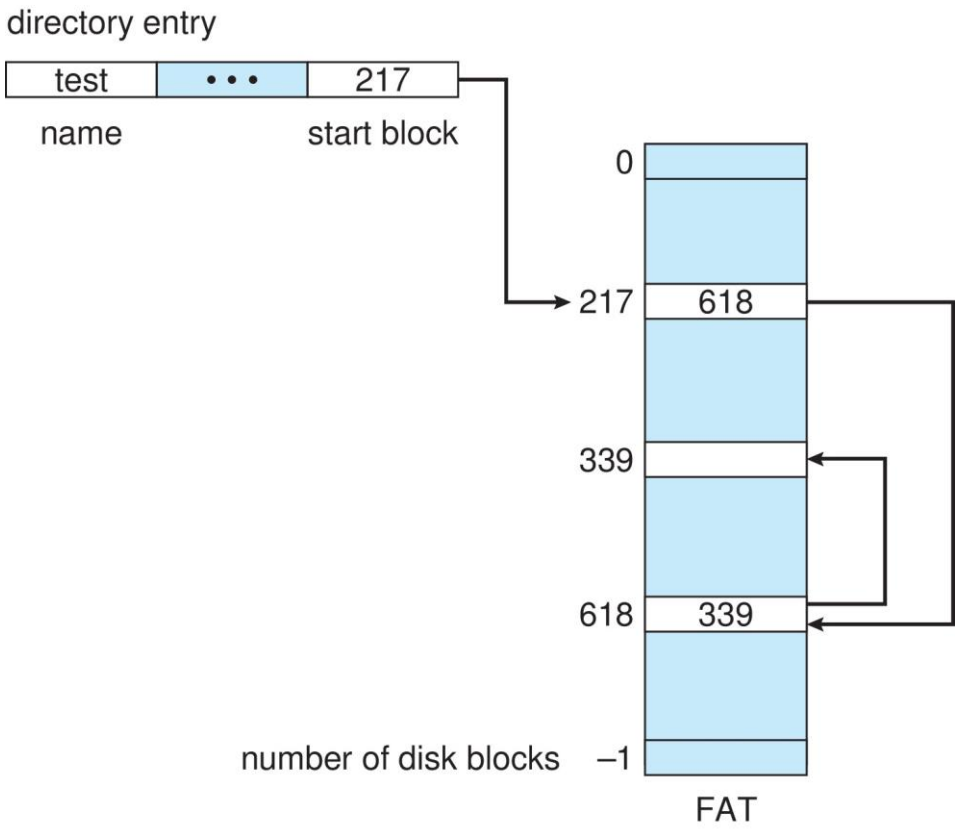


Implementação de diretório e indexação de arquivos

Alocação lincada

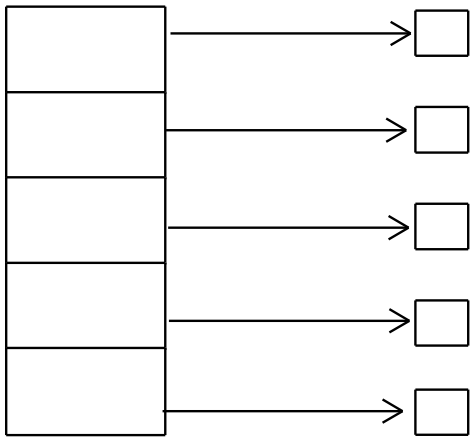


Esquema FAT

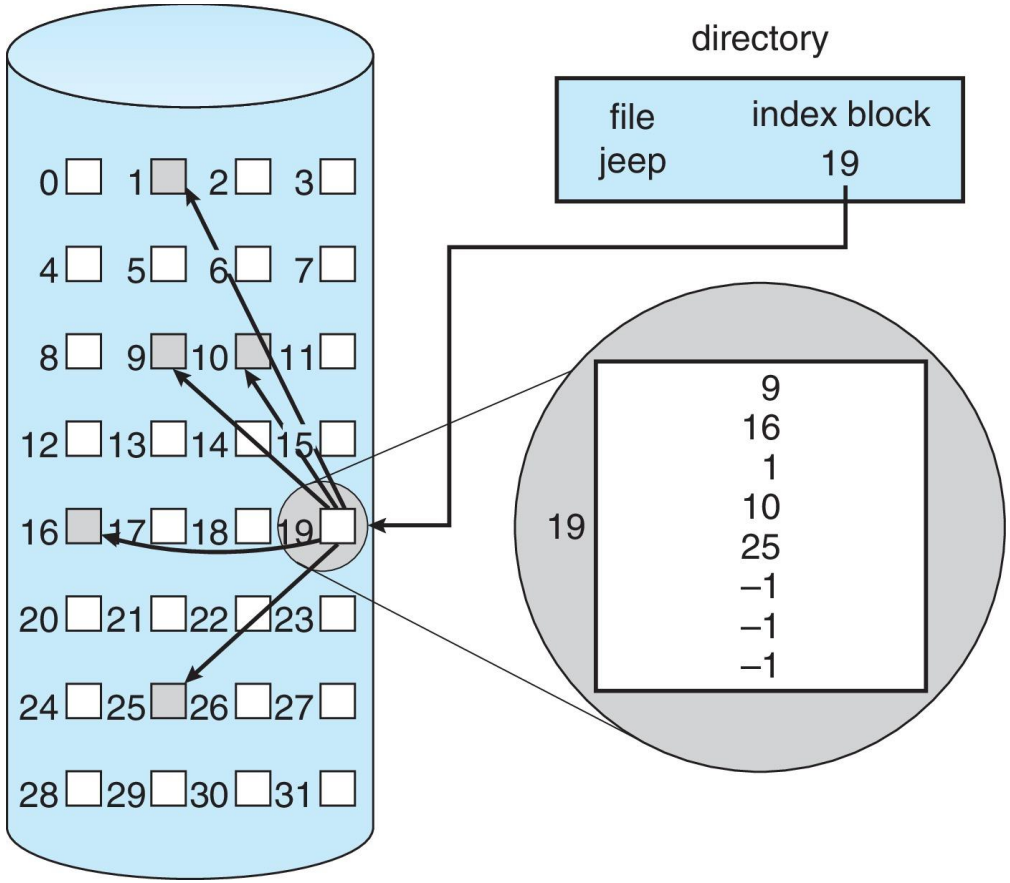


Implementação de diretório e indexação de arquivos

Alocação indexada

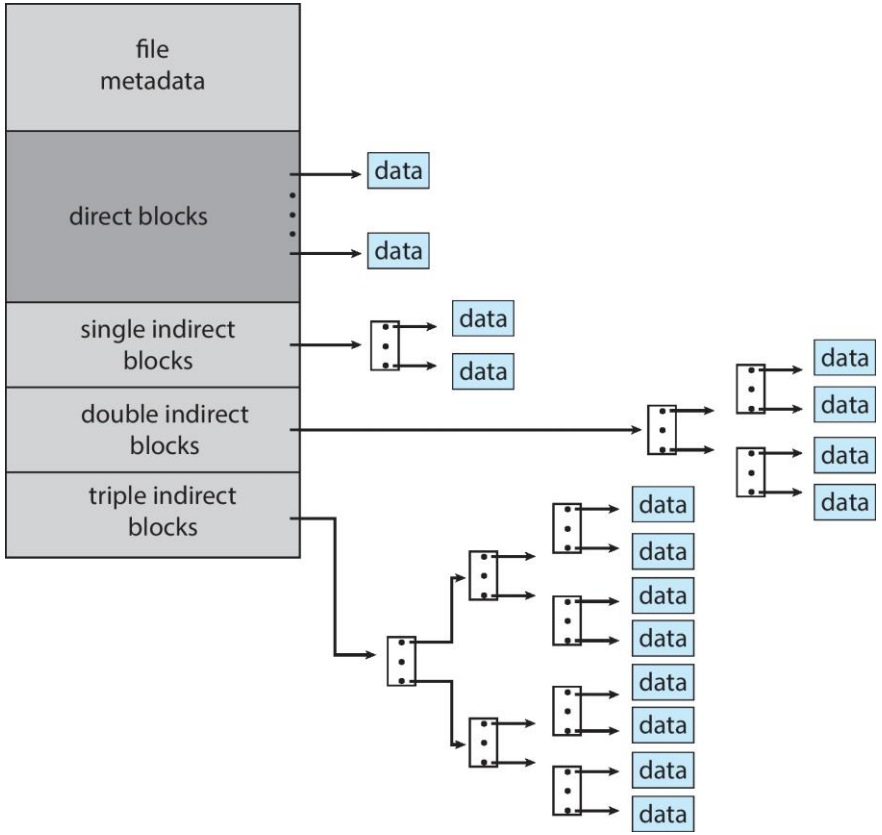


index table



Implementação de diretório e indexação de arquivos – UNIX UFS

4K bytes por bloco, endereços de 32 bits



Mais blocos de índice do que podem ser endereçados com ponteiro de arquivo de 32 bits

Proteção

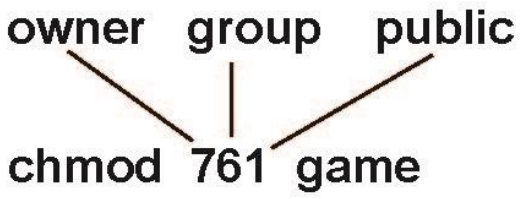
- O proprietário/criador de arquivos deve ser capaz de controlar:
  - o que pode ser feito
  - por quem
- Tipos de acesso
  - **Ler**
  - **Escrever**
  - **Executar**
  - **Acrescentar**
  - **Excluir**
  - **Listar**

Listas e Grupos de Acesso

- Modo de acesso: ler, escrever, executar
- Três classes de usuários no Unix/Linux

			RWX
a) **owner access**	7	⇒	1 1 1
b) **group access**	6	⇒	RWX 1 1 0
c) **public access**	1	⇒	RWX 0 0 1

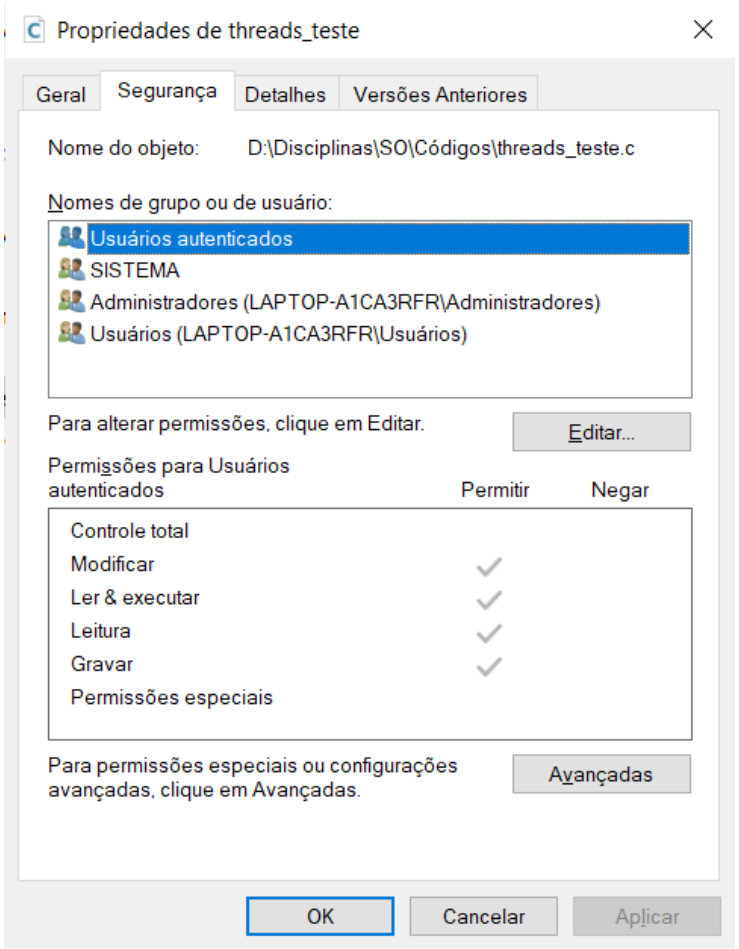
- Peça ao gerente para criar um grupo (nome único), digamos G, e adicione alguns usuários ao grupo
- Para um determinado arquivo (digamos jogo) ou subdiretório, defina um acesso apropriado

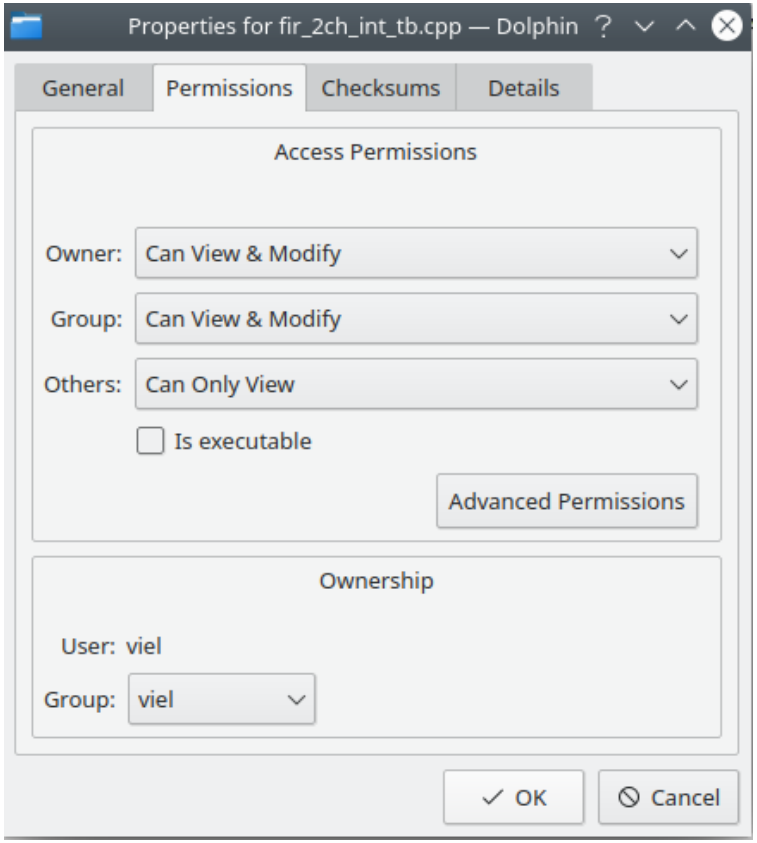


- Anexar um grupo a um arquivo

chgrp G game

Gerenciamento da lista de controle de acesso do Windows 10 e Linux Kubuntu





Um exemplo de listagem de diretório Linux

- ls -all

```
drwxrwxr-x 3 viel viel   4096 out 10 16:23 .
drwxrwxr-x 4 viel viel   4096 out 10  2021 ..
-rw-rw-r-- 1 viel viel   8427 out 10 16:22 fir_2ch_int.cpp
-rw-rw-r-- 1 viel viel  42517 out 10 16:22 fir_2ch_int_din_i.txt
-rw-rw-r-- 1 viel viel  42517 out 10 16:22 fir_2ch_int_din_q.txt
-rw-rw-r-- 1 viel viel  25261 out 10 16:22 fir_2ch_int_dout_i_cmodel.txt
-rw-rw-r-- 1 viel viel  25162 out 10 16:22 fir_2ch_int_dout_q_cmodel.txt
-rw-rw-r-- 1 viel viel   5579 out 10 16:22 fir_2ch_int.h
-rw-rw-r-- 1 viel viel   7074 out 10 16:22 fir_2ch_int_tb.cpp
drwxrwxr-x 5 viel viel   4096 out 10  2021 proj
-rw-rw-r-- 1 viel viel   5670 out 10 16:22 README
-rw-rw-r-- 1 viel viel   6401 out 10 16:22 run_hls.tcl
-rw-rw-r-- 1 viel viel 180045 out 10  2021 vivado_hls.log
```