



Архитектура компьютеров

Лекция 11 Файловые системы

Кафедра «Прикладная математика»
SpbSPU, 2015

Определение

- Файловая система – это часть операционной системы которая преобразует интерфейс блочного устройства в файлы, каталоги, журналы и т.д.
 - Обеспечивает
 - Управление местом на диске: блоки □ файлы
 - Именованное: Возможность поиска файлов по именам, а не адресам
 - Защита: ACL – access control list
 - Надежность: Данные не должны теряться из-за сбоев диска, сбоев ОС и т.д..
-

Управление диском

- Основные записи ФС:
 - **Файл:** Видимая пользователю упорядоченная группа блоков, имеющая имя
 - **Каталог:** Видимое пользователю отображение списка имен на файлы

 - Доступ диску, как к линейному списку секторов.
 - **Logical Block Addressing (LBA):** Каждый сектор имеет порядковый номер
 - Контроллер диска «разбирается» с плохими секторами (ранее OS/BIOS)
 - Контроллер преобразует номер \Rightarrow физическую позицию
 - ОС не знает про физическую структуру диска
-

Управление данными

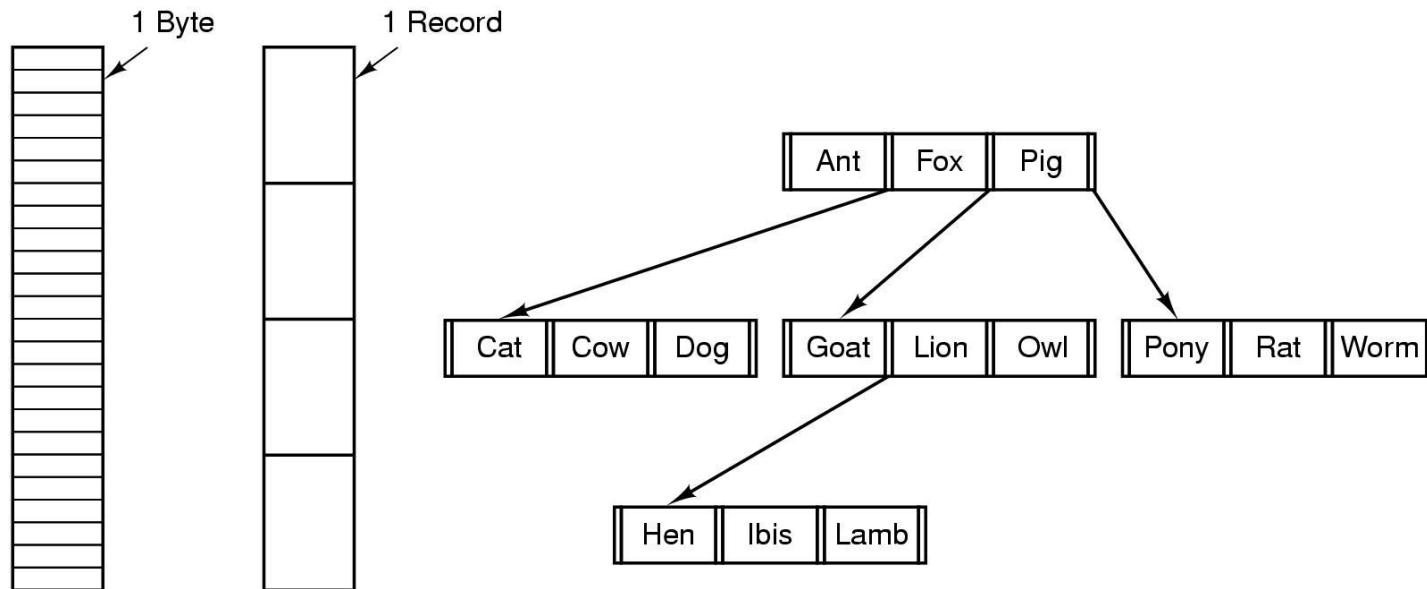
- Управление файлами
 - Зависит от ОС
 - FAT (File Allocation Table)
 - Использовалась от MS-DOS до Windows ME
 - NTFS (New Technology File System)
 - Используется Windows
 - Unix и Linux поддерживают множество ФС
 - XFS, JFS, ReiserFS, ext3, ext4 и т.д.
 - Mac OS X
 - HFS и HFS+
-

Цели ФС

- Длительное хранение информации:
 - Надо хранить много информации
 - Информация не должна «портиться» от использования
 - Должен быть возможен множественный одновременный доступ
 - Решение:
 - Записывать информацию в файлы
 - Файлы хранятся «вечно» пока владелец или администратор их не удалят
 - Файлы управляются ОС
-

Внутренняя структура файла

- Последовательность байтов: неструктурированные
- Последовательность записей: блокированные
- Сложные структуры
 - Например: блокировано – сегментированные, деревья и т.д.



Доступ к файлам

- Последовательный
 - Читаем от начала до конца
 - Нельзя прыгать, но можно перематывать
 - Лента
 - Произвольный
 - Читаем/пишем в любом порядке
-

Атрибуты файла

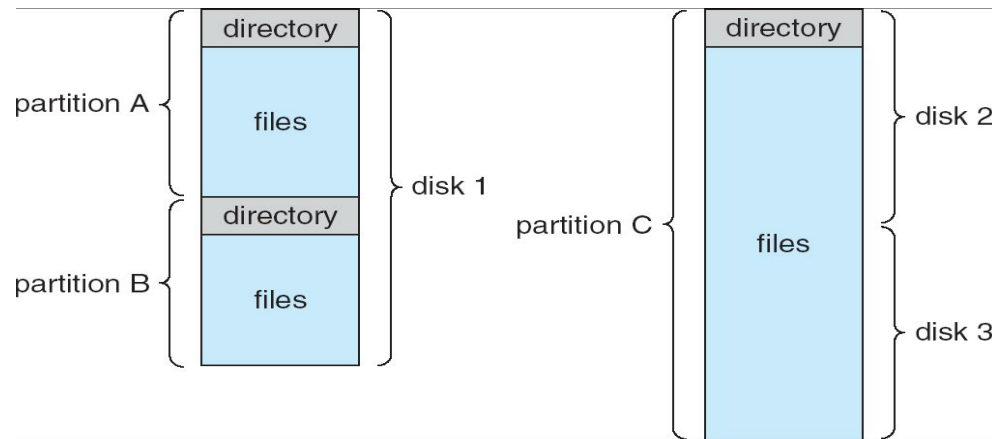
- Специфичная информация ОС
 - Размер файла, дата создания, дата модификации.
 - Сильно меняется в зависимости от ОС
 - Примеры:
 - Имя – Человекочитаемая информация
 - Идентификатор – Номер файла внутри ФС (NTFS)
 - Тип (расширение)– NTFS, FAT
 - Местоположение – Указатель на место где файл лежит на устройстве
 - Размер
 - ACL – определяет кто может писать, читать, выполнять файл
-

Операции над файлом

- Создать
 - Писать
 - Читать
 - Перейти на смещение
 - Удалить
 - Обрезать
-

ФС на диске

- Обычно – 1 диск = одна файловая система
 - Windows 7+: Компьютер имеет 2 ФС на двух разделах
 - Часть диска используется под SWAP
- Диск делится на разделы (partitions)
 - Раздел содержащий ФС - том
 - Структура каталогов содержит все файлы в томе

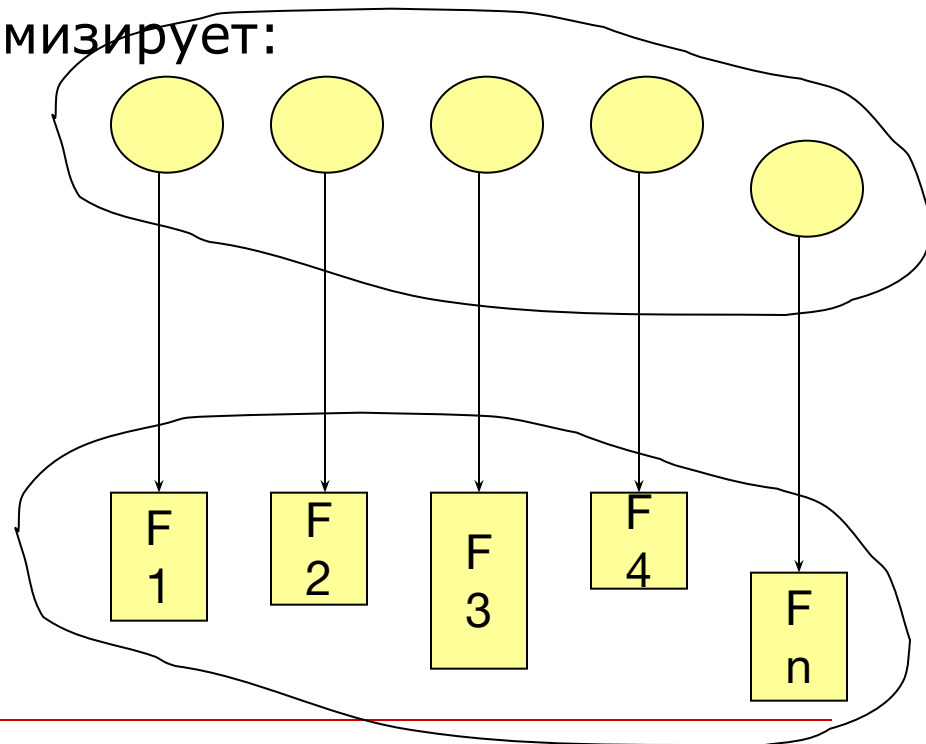


Каталоги

□ Каталог – это файл со спец. Атрибутом

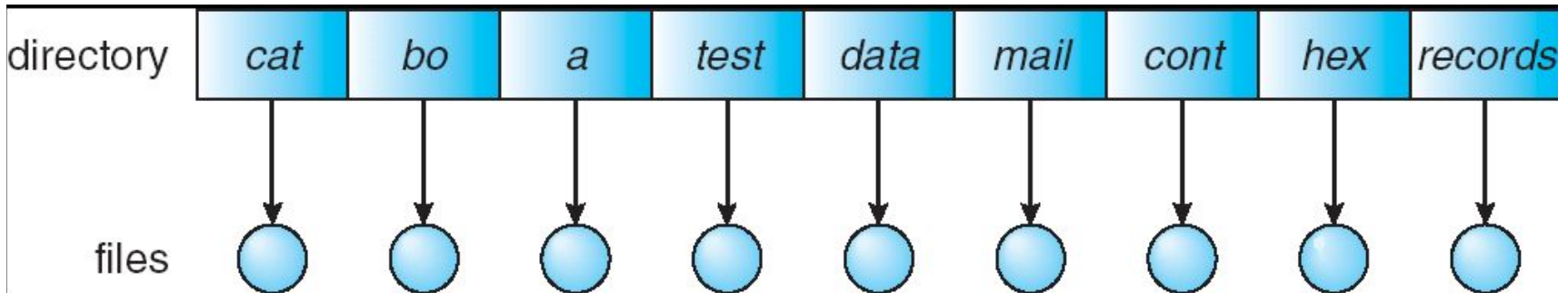
□ Структура каталогов оптимизирует:

- Поиск
- Создание
- Удаление
- Чтение списка файлов
- Переименование
- Переход по ФС



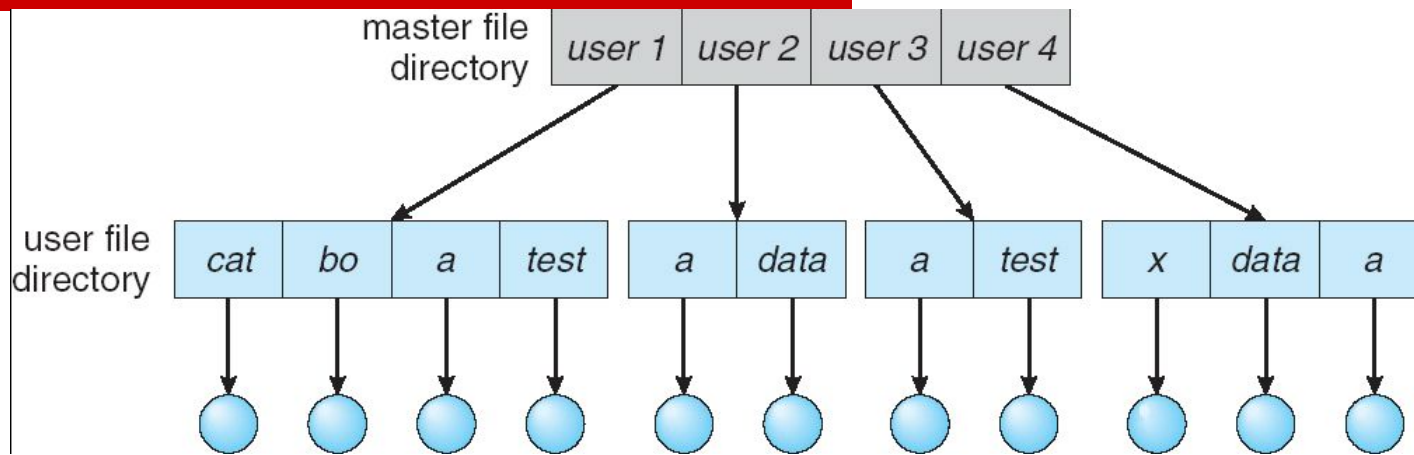
Одноуровневый каталог

- На диске один каталог
 - Он же корневой каталог



- Использовался на первом суперкомпьютере CDC 6600
 - +: ну очень простая логика
 - -: все остальное
-

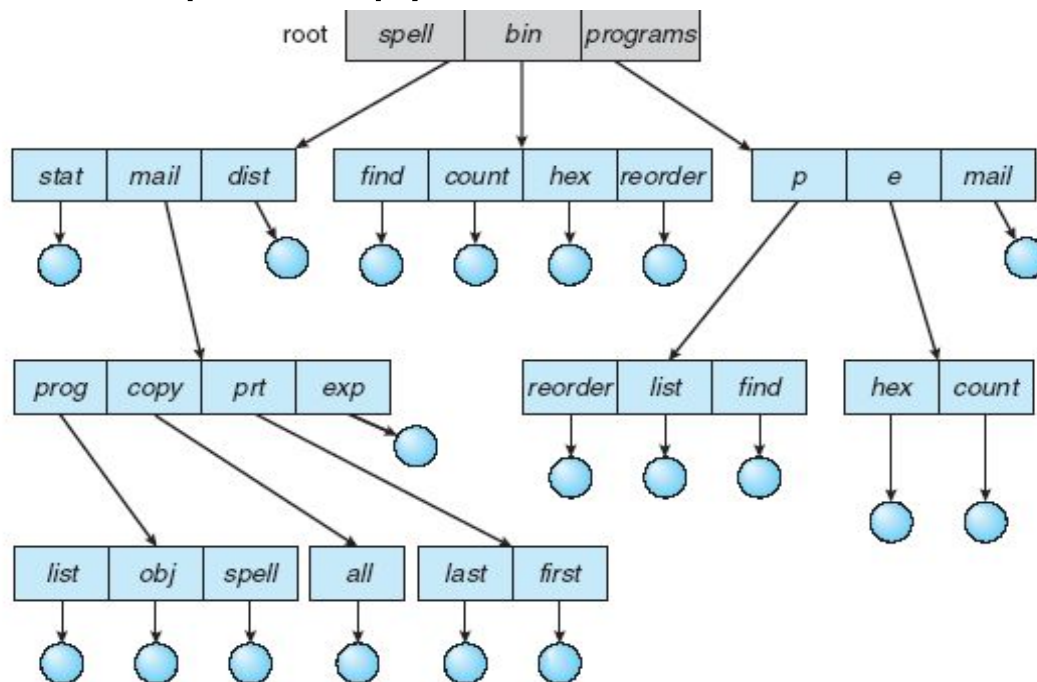
Двухуровневый каталог



- Каждый пользователь имеет свой каталог
 - Решает проблемы с коллизией имен
-

Древовидный каталог

- Дерево каталогов «произвольной» глубины
 - Атрибут отличает каталог от файла
 - Каталог содержит другие каталоги

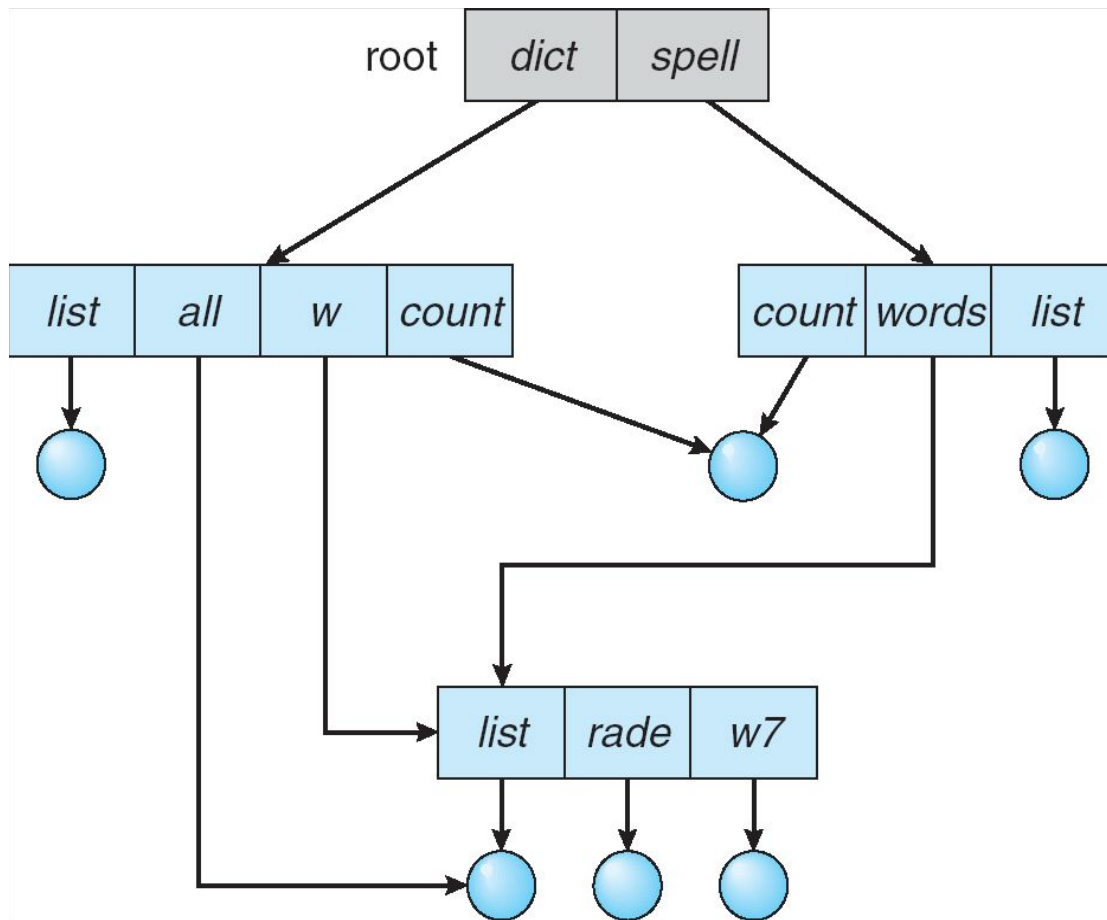


Пути к файлам

- Для доступа к файлу нужно:
 - Перейти в каталог с файлом
 - Указать **путь** до файла
 - Пути бывают абсолютными и относительными
 - Абсолютный: Путь от корневого каталога
 - Относительный: путь от текущего каталога
 - Большинство ОС имеют два спец. Имени файла:
 - "." текущий каталог
 - ".." родительский каталог
-

Ациклический граф

- Ссылки на файлы

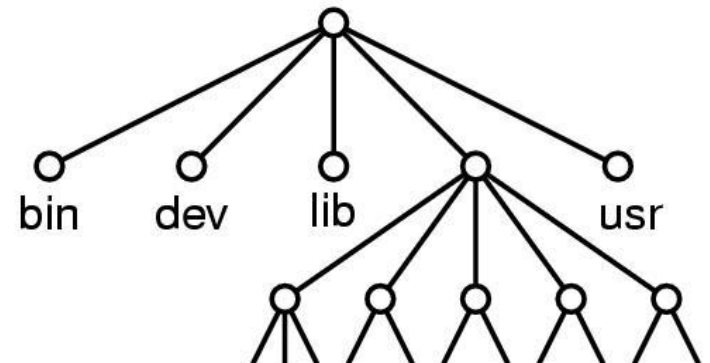
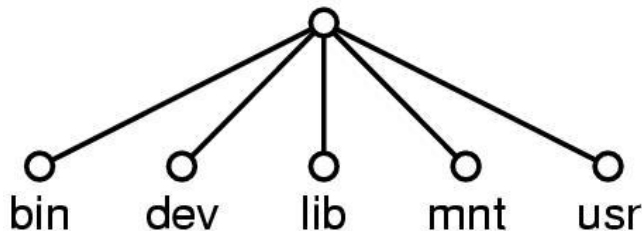


Ациклический граф

- Как реализовать разделение между пользователями файлов и каталогов:
 - Скопировать
 - Новый атрибут: ссылка
 - Ссылка ссылается на другой объект ФС
 - Ссылка игнорируется при обходе ФС
 - *In* в Linux, *fsutil* в Windows для жестких
 - *In -s* в UNIX, ярлыки в Windows для мягких
 - Проблемы
 - 1 файл – два имени
 - Если *dict* удаляет *count* ⇒ провисшая ссылка
 - Хранить обратные указатели
 - Удалять в случае первого доступа
 - Хранить счетчик ссылок у каждого файла
-

Монтирование ФС

- Монтирование позволяет подключить одну ФС к другой
 - Подключить оптический диск к ФС
 - Linux: `mount /dev/cdrom0 /mnt`
 - Windows: Diskpart assign [mount=<path>]



Монтирование по сети

- Та же идея, но ФС на другом компьютере
 - Реализуется через RPC
 - Очень часто используется в современных ФС
-

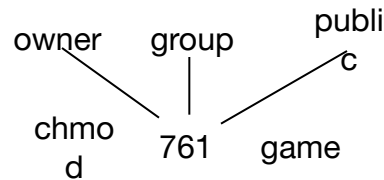
Защита файла

- Хозяин контролирует:
 - Что может быть сделано
 - Кем

 - Типы доступа
 - Чтение
 - Запись
 - Выполнение
 - Добавление
 - Удаление
 - Просмотр каталога
-

Права доступа Linux

- Режимы доступа: чтение, запись, выполнение
- Три класса пользователей RWX
 - Владелец** 7 \Rightarrow 1 1 1
RWX
 - Группа** 6 \Rightarrow 1 1 0
RWX
 - Все остальные** 1 \Rightarrow 0 0 1
- Для конкретного файла



Проблемы в Linux

- 1 владелец, одна группа
 - +: Компактно
 - -: не очень удобно
 - *ACL*: Пофайловый список который говорит, что можно сделать с файлом
 - +: Удобно
 - -: Много места, медленно
-

Блок управления файлом (ФСВ)

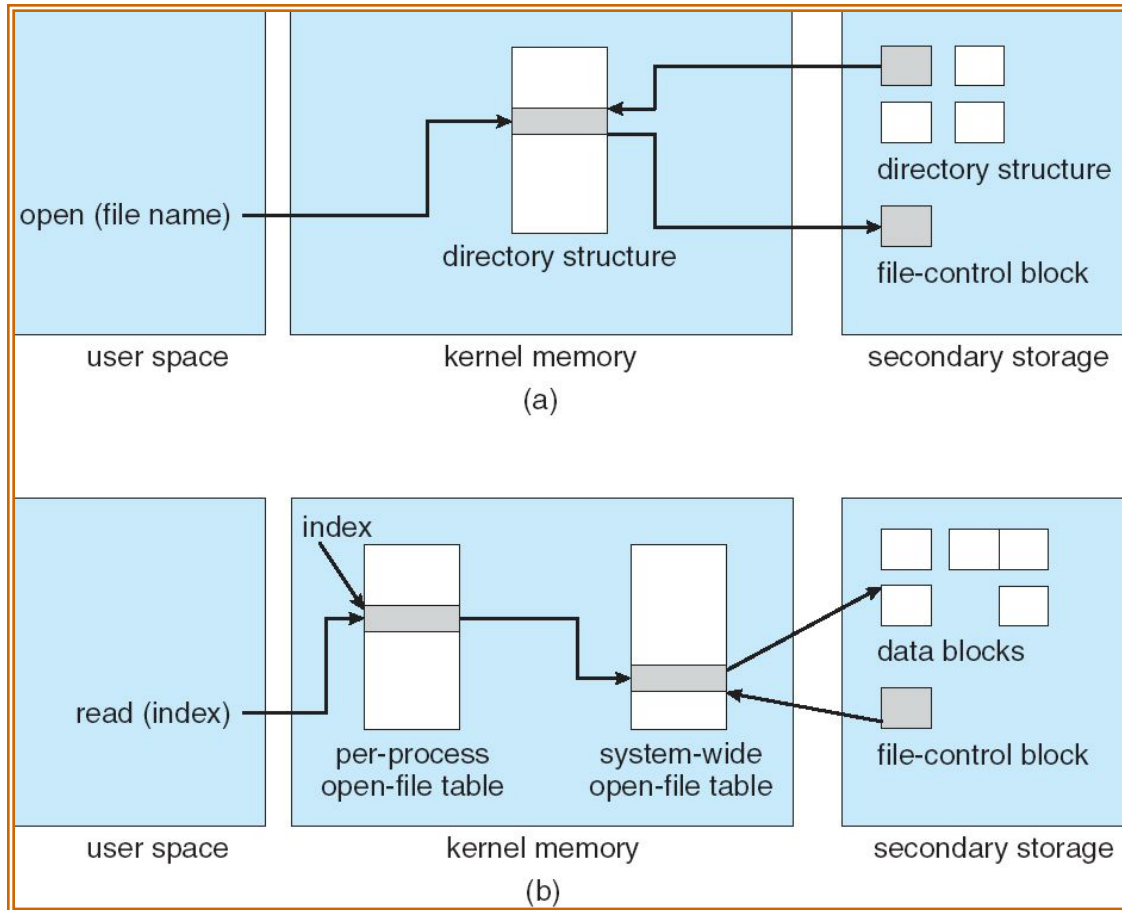
- ФСВ содержит всю информацию о файле
 - Linux называет эти структуры inode

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

Множественный доступ к файлу

- В OS два типа таблиц:
 - Таблица процесса
 - Информация о использовании (текущий указатель файла)
 - Системная таблица
 - Создаётся при первом доступе к файлу
 - Место на диске
 - Время доступа
 - Размер файла
 - Считает как много процессов имеют доступ (чтоб не удалить)
-

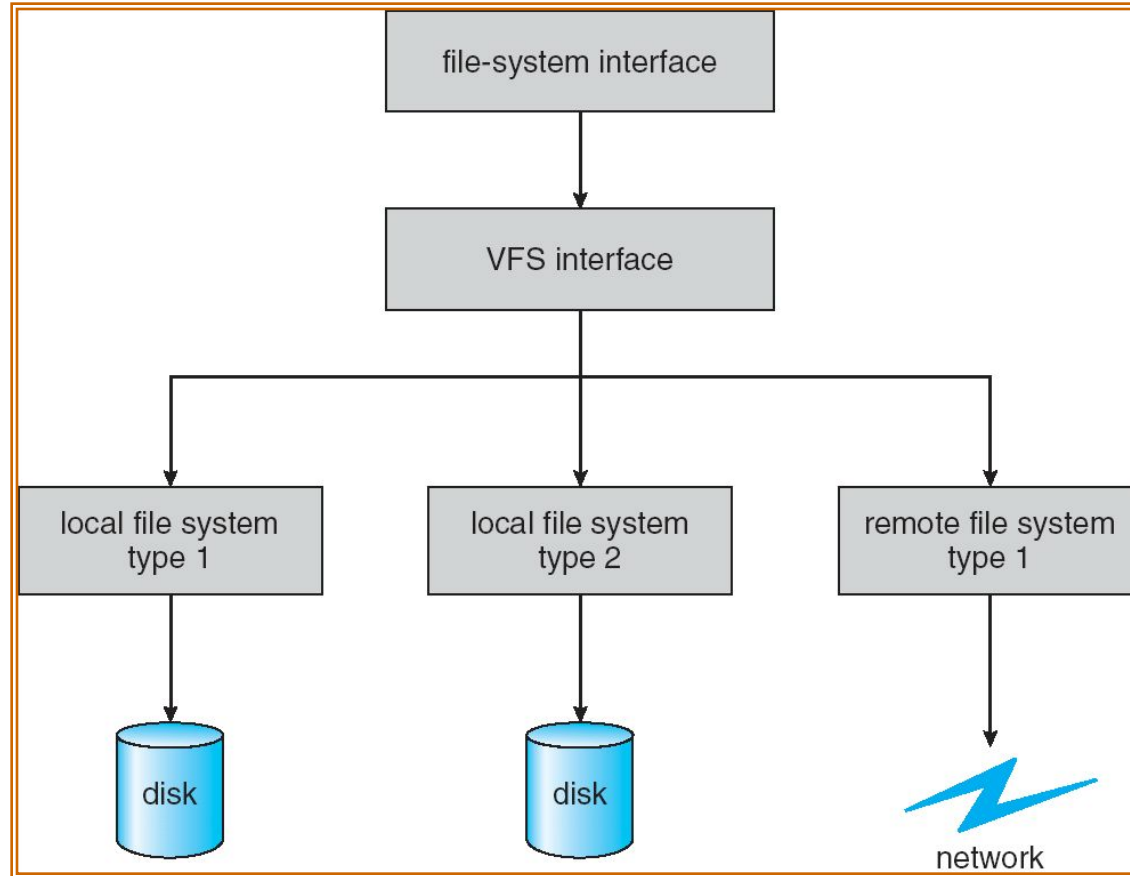
Открытие файла и чтение



Виртуальные файловые системы VFS

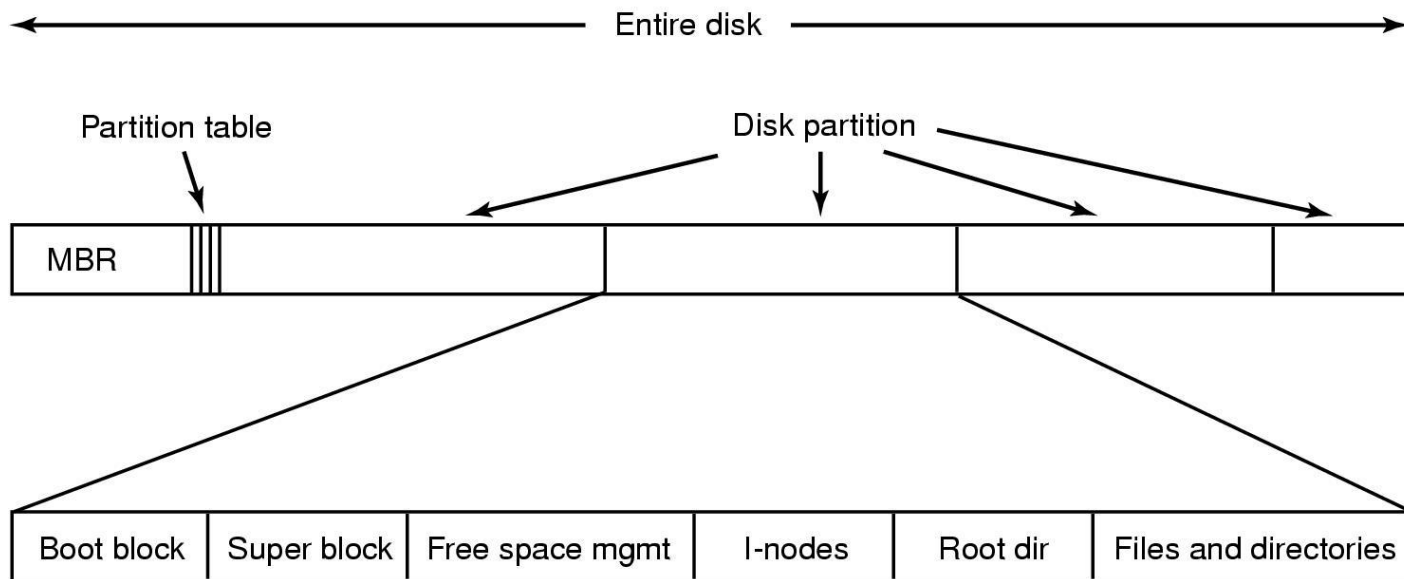
- SAMBA – для подключения сетевых дисков
 - FUSE – file system in user space
 - VFS
 - Дает один API для разных ФС
 - Позволяет пересекать границы ОС
-

VFS



Структура ФС

- Первый блок каждого тома – загрузочный блок
 - Загружается MBR и исполняется

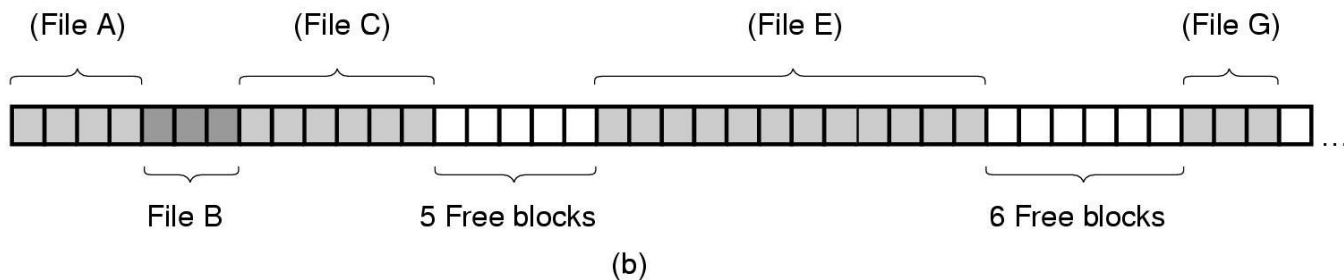
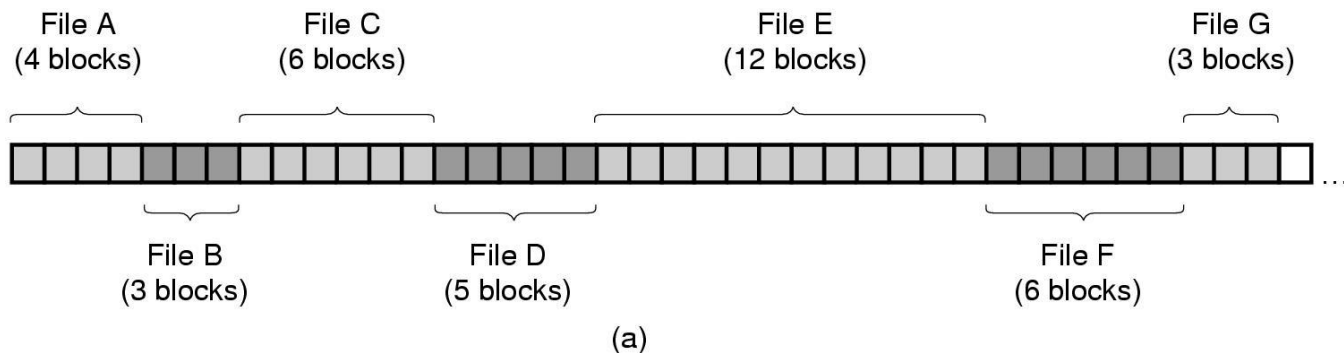


Хранение файлов

- Последовательное
 - Индексно-последовательное
 - Списанием
 - Библиотечное
 - Индексное
-

Индексно-последовательное

- Файлы друг за дружкой и каталог их индекс

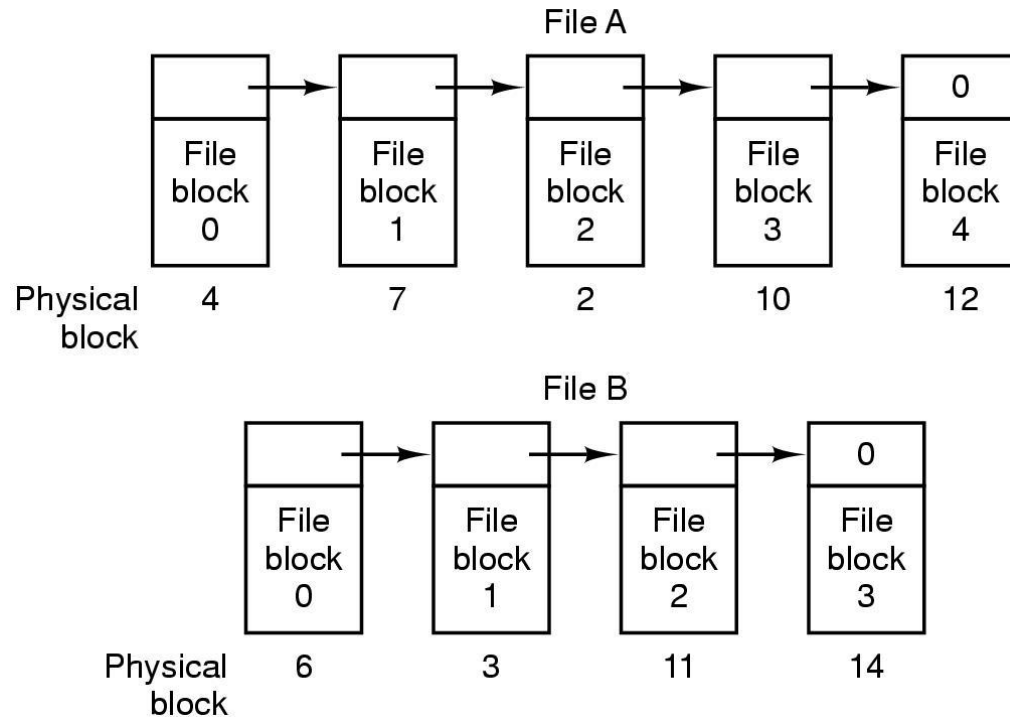


Индексно-последовательное

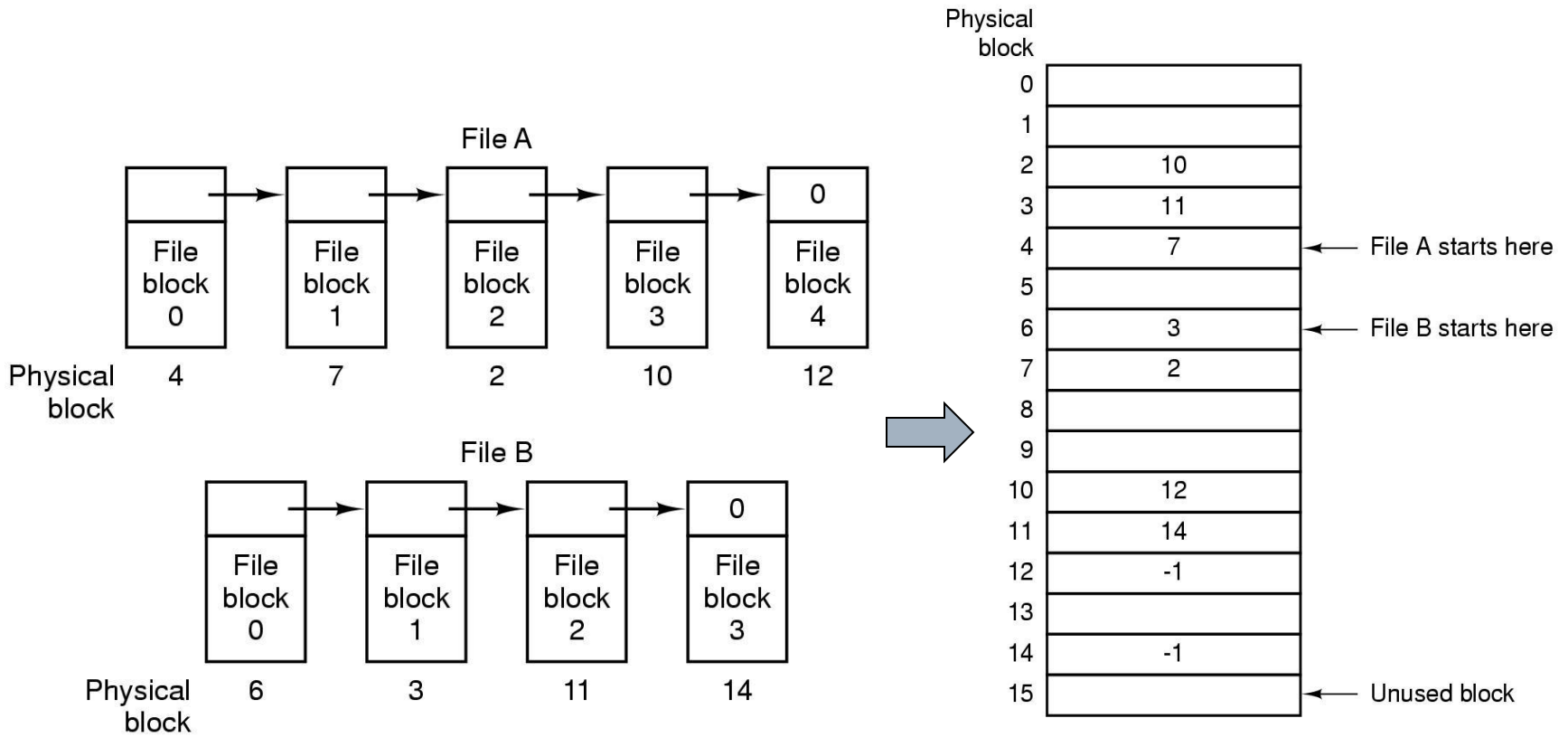
- +:
 - Просто
 - Быстрый доступ. Файл читает зараз
 - -:
 - Фрагментация - проблема
 - Надо знать размер файла заранее
 - Используется CDRoms, DVDs, Blur-ray (CDFS)
-

Список

- Каждый файл последовательность связанных блоков

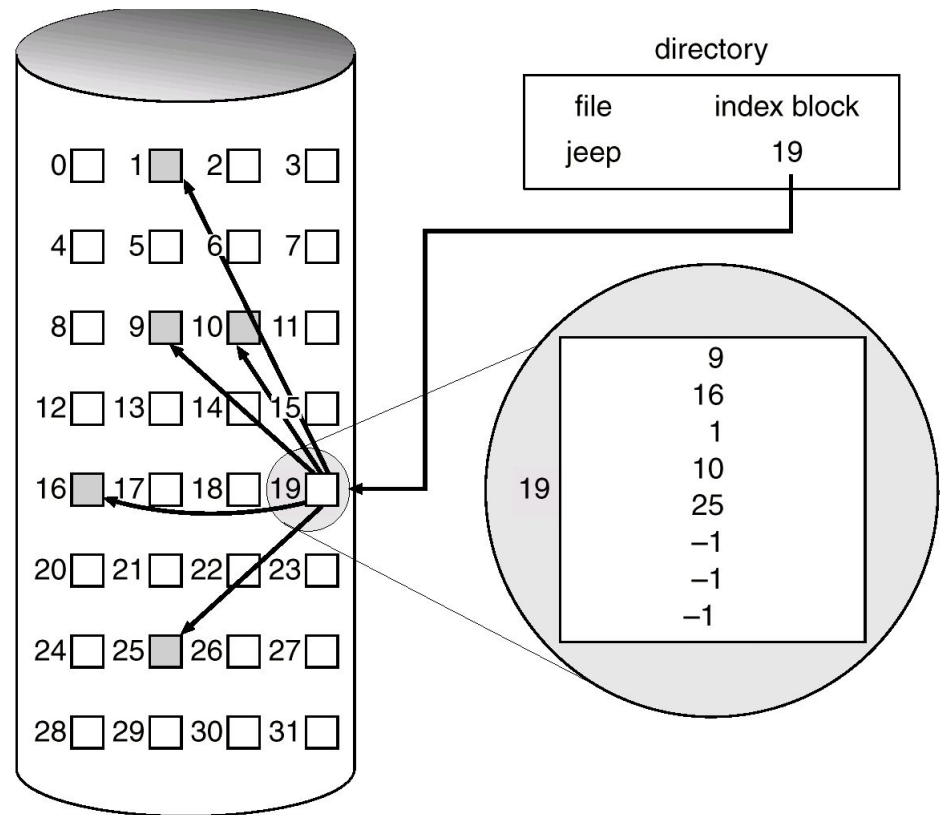


FAT

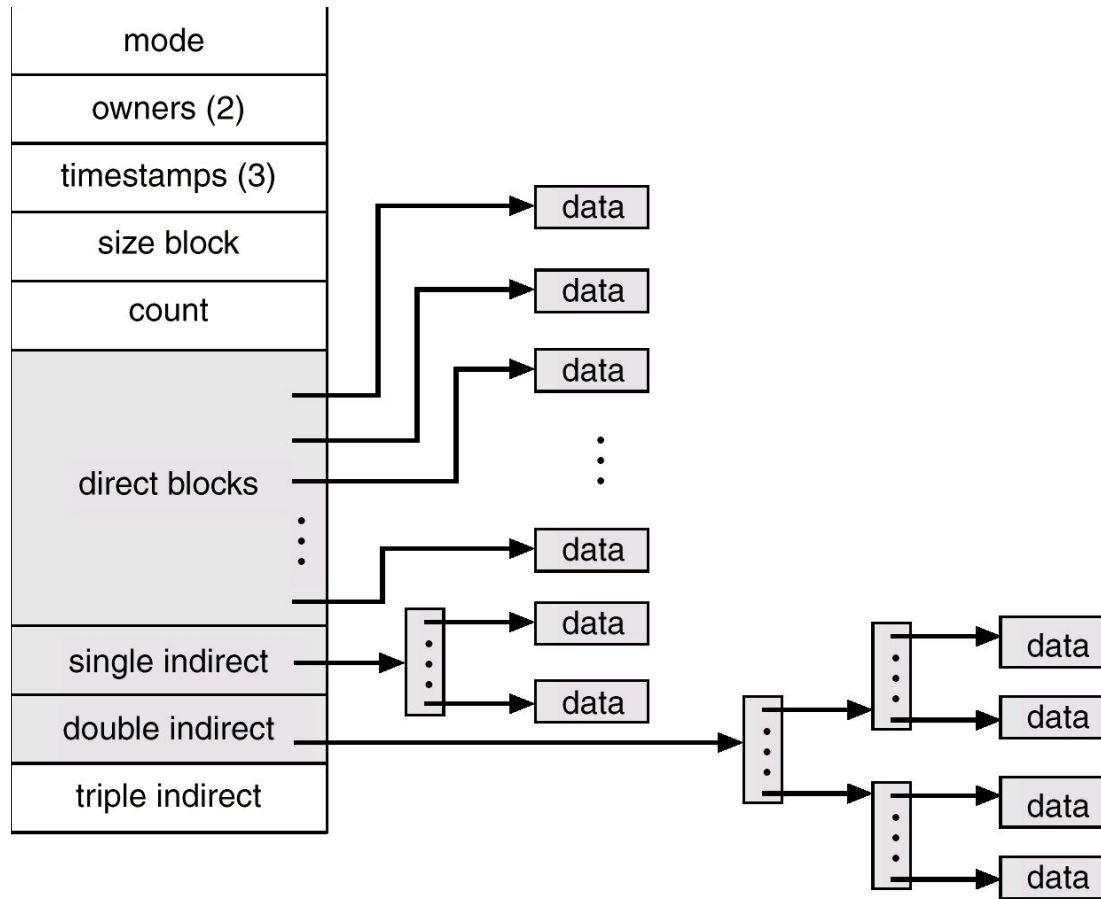


Индексное хранение

- Индексный блок содержит указатели на каждый блок



UFS – Unix File System

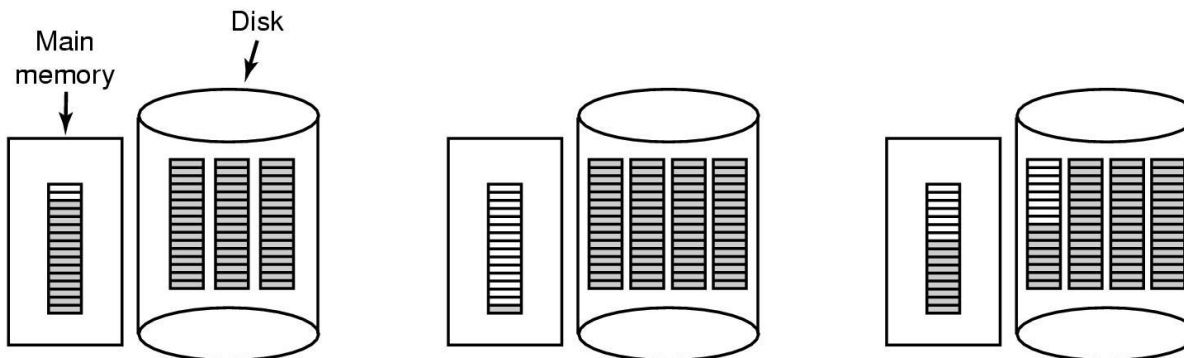


Журналирование

- Метод обеспечения надежности
 - Все операции в ФС реализуются как транзакции
 - Либо случились, либо нет
 - Транзакция считается совершенной после того как инф. Об этом записана в журнал
 - Запись в журнал всегда последнее действие
-

Управление свободным местом

- Свободные блоки пишутся связным списком
 - +: Только один блок в памяти
 - -: Фрагментация



- Хранение отображений
 - Меньше места
 - Выделяемые блоки ближе друг к другу
-

Следующая лекция

Процесс загрузки и BIOS
