Хранение данных

Лекция №9 - АКОС 2019-2020

Хранилища

- На физическом уровне:
  - носители
  - интерфейсы подключения
  - высокоуровневые схемы (RAID)
- На логическом уровне:
  - организация файловых систем

Способы хранения информации

- HDD - магнитный носитель высокой плотности
- Характеристики:
  - интерфейс подключения
  - ёмкость (1Кбайт = 1000 байт!!!)
  - время доступа - обусловлено перемещением головки
  - скорость вращения (5400, 7200, 10К, 15К)
  - скорость передачи данных (от 70 до 200Мб/с)
  - объем буфера (от 8 до 128Мб)

Способы хранения информации

- SSD - флеш-память
- SLC - 1 бит на ячейку, MLC - 2 бита на ячейку, TLC - 3 бита на ячейку
- Характеристики:
  - интерфейс подключения (SATA или M2)
  - емкость
  - время доступа/скорость чтения
  - **количество циклов записи**

Ограниченный срок службы SSD

- SLC - 100 000 циклов
- MLC - от 3000 до 40 000 циклов
- TLC - менее 2000 циклов

- Проблема решается балансировкой нагрузки ячеек
- Ячейки, ресурс которых близок к ненадежному, можно помечать как «израсходованные», - операция **trim**

Интерфейсы подключения (немного истории)

- Параллельный интерфейс IDE - 40-pin, два устройства на линию. Master и Slave.
- Параллельный интерфейс SCSI - 50, 68 или 80 контактов на линии. Устройства подключаются последовательно, должны быть специальные «терминаторы» у последнего устройства.

Интерфейс ATA

- Логический уровень параллельного интерфейса IDE и последовательного SATA

- ATA PIO Mode: передача данных через порты ввода-вывода процессора; UDMA - без использования процессора
- Данные передаются в виде команд
- Расширенный интерфейс ATAPI (в том числе для SATA) - добавляет команды для эмуляции SCSI

Интерфейс SCSI

- Универсальный интерфейс для дисков, CD, и разной перефирии
- По аналогии с ATA - передача данных в виде пакетов команд, но набор команд более универсальный
- Современная реализация - Serial Attached SCSI (SAS)

Advanced Host Controller Interface (SATA AHCI)

- Нативный интерфейс для SATA без Legacy времен IDE/ATA
- Поддерживает горячую замену и управление питанием
- BIOS (обычно) позволяет выбирать режим работы между ATAPI/AHCI
- Но Windows работает только в том режиме, который был на этапе установки. Be careful!

RAID

(Reduntant Array of Inexpensive Disks)

- Повышение надежности: дублирование данных
- Увеличение скорости работы: чередование данных
- Требует нескольких дисков
- Может быть реализован как программно, так и аппаратно

RAID

(Reduntant Array of Inexpensive Disks)

- RAID-0:
  - от 2-х дисков
  - только чередование данных
  - скорость чтения/записи - (почти) пропорциональна количеству дисков
  - смерть одного из дисков приводит к потере всех данных

RAID

(Reduntant Array of Inexpensive Disks)

- RAID-1:
  - от 2-х дисков
  - только дублирование данных
  - скорость чтения (в среднем) пропорциональна количеству дисков
  - скорость записи лимитируется самым медленным диском

RAID

(Reduntant Array of Inexpensive Disks)

- Исторически сложившиеся уровни:
  - RAID-2: чередование данных + код Хэмминга
  - RAID-3: побайтное чередование данных + отдельный диск для битов четности
  - RAID-4: блочное чередование данных + отдельный диск для битов четности
- RAID-5 и RAID-6: чередование данных + биты четности на каждом диске

Реализации RAID

- Аппаратная
  - отдельный чип, как правило в серверных системах
  - требуется поддержка на уровне ОС (драйвер)
  - в Linux - управление с помощью dmraid
  - Be careful!
  - в системе обычно /dev/dmX

Реализации RAID

- Программная
  - дополнительная нагрузка на процессор
  - составляется отдельное «устройство» из существующих дисков с помощью mdadm
  - нет зависимости от конкретной реализации
  - /dev/mdX

Другие программные реализации

- Logical Volume Manager (LVM)
- На уровне некоторых файловых систем:
  - ZFS
  - BtrFS

Разбиение дисков на разделы

- DOS-style: MBR, 4 первичных раздела
- GPT: неограниченное количество разделов
- Logical Volume Manager: несколько дисков или разделов можно объединять в одно «устройство» (группа томов)

Файловые системы

- Дисковое пространство (раздел или группа томов) - это просто хранилище без структуры
- Задача файловой системы - организовать порядок в этом хранилище

Доступ к файлу

- С точки зрения пользователя или API - каждый файл имеет абсолютный путь (текст)
- С точки зрения процесса - открытый файловый дескриптор
- С точки зрения операционной системы: пара значений {st_dev, st_ino}
- С точки зрения файловой системы - inode

Основные концепции ФС

на примере ext2

- Пространство делится на блоки фиксированного размера
- Блок - минимально адресуемый объем данных в ФС
- Непрерывные посделовательности блоков объединяются в группы
- Некоторые блоки в группе имеют специальное назначение

Фрагментация

- Размеры файлов могут вариьроваться
- Нельзя непрерывно разместить все файлы

- Это может приводить (не всегда!) к потере производительности на механических HDD
- Проблема не актуальна для многозадачных систем, особенно серверных
- В старых Windows была утилита «дефрагментация диска»

Основные концепции ФС

на примере ext2

- Специальные блоки:
  - битовая маска занятых/свободных блоков
  - таблицы связей inode и отдельных блоков

- Суперблок
  - специальный блок с фиксированным размещением на диске
  - содержит сводную информацию о файловой системе

tune2fs -l имя_раздела

Поиск данных по номеру inode

- Нумерация inode начинается с 1

  block_group = (inode-1)/INODES_PER_GROUP
- Далее находится группа по глобальной таблице Block Group Descriptor
- Номер элемента в таблице группы - остаток по модулю

  index = (inode-1) % INODES_PER_GROUP
- Номер блока в группе

  containing_block = (index * INODE_SIZE) / BLOCK_SIZE

Содержимое inode в ext2/3/4

- 12 прямых указателей на блоки данных
- 3 косвеных указателя для дополнительных таблиц 1, 2 и 3 уровней
- Метаданные файла - доступны через системный вызов stat

Man page: stat

Содержимое inode в ext2/3/4

- Сумарный размер структуры inode = 128 байт
- INODE_SIZE = 256

Виды файлов

- Регулярный файл
- **Каталог**
- Файлы-устройства (блочные и символьные)
- Символические (но не жёсткие) ссылки
- Именованные каналы (FIFO)
- Сокеты

Каталог

- Список записей типа struct dirent
- Как **минимум**, в POSIX структура содержит:
  - номер inode
  - размер записи
  - имя файла длиной не более 256 байт
- Конкретная реализация структуры не регламентирована стандартом POSIX и различается в разных UNIX-системах

Варианты файловых систем для Linux

- ext2/3/4 - самые стабильные
- XFS - ориентирована на большие файлы
- BtrFS - много фич, но до сих пор не считается стабильной
- ReiserFS - когда-то была самой эффективной для большого количества маленьких файлов