

# Файловая система в UNIX

Организация файловой системы в  
UNIX. Работа с файлами и  
директориями.

**Что такое файловая система?**

«

*Файловая система* - это часть операционной системы, назначение которой состоит в том, чтобы организовать эффективную работу с данными, хранящимися во внешней памяти и обеспечить пользователю удобный интерфейс при работе с этими данными.

»

# Физические носители информации

- магнитные диски
- оптические диски
- твердотельные накопители
- и т.д.

Структура разная, интерфейс одинаковый:

- драйвер устройства
- файловая система

# Логические диски

Физические носители принято делить на логические диски или разделы (partitions)

- несколько ОС;
- несколько видов ФС;
- различные категории файлов;
- внутренние требования ОС или ФС.

# Логическая структура ФС

В UNIX существуют файлы нескольких типов:

- регулярные файлы;
- директории;
- файлы типа FIFO;
- файлы устройств;
- сокеты (sockets);
- файлы связи (link).

# Полное имя файла

- Корневой узел – «/»
- Имя директории в пути, после него «/»
- Для каждой очередной директории в пути приписываем к уже получившейся строке справа «/» и имя соответствующей директории

# Файловая система s5fs

- System V File System (s5fs) – прародитель современной ФС в Linux
  - заголовок раздела: массив индексов узлов и суперблок;
  - логические блоки.

Файл:

- тип файла, права доступа;
- id владельца и его группы;
- размер файла;
- время последнего доступа;
- время последней модификации файла;
- время последней модификации узла.



# Понятие суперблока

- Тип ФС;
- Флаги состояния ФС;
- Размер логического блока в байтах;
- Размер ФС в ЛБ;
- Размер массива индексных узлов;
- Число свободных индексных узлов;
- Число свободных блоков для размещения данных;
- Часть списка свободных индексных узлов;
- Часть списка свободных блоков для размещения данных.

# Операции над файлами

Файлы:

- Последовательный доступ
  - чтение порции (read);
  - запись порции (write);
  - позиционирование на начале (rewind).
- Прямой доступ
  - чтение порции (read);
  - запись порции (write);
  - позиционирование на требуемой части данных (seek).
- Кроме того
  - создание нового объекта (new);
  - уничтожение существующего (free);
  - прочитать атрибуты (get attribute);
  - установить атрибуты (set attribute).

# Для директорий

- read = get record;
- write = add record;
- rewind (seek) = delete record.

# Работа с файлами

Команды:

- `chmod`, `cp`, `rm`, `mv`, `ls`

Системные вызовы:

- `open()`, `close()`, `read()`, `write()`

# Системный вызов creat

- Прототип системного вызова

*#include <fcntl.h>*

*int creat(char \*path, int mode);*

# Изменение атрибутов файлов

- `stat()`
- `fstat()`
- `lstat()`

# Изменение длины файла

- `ftruncate()`

# Изменение текущей позиции

- lseek()



# Создание связей

- Жесткая связь (hard link)
  - проблема удаления, счетчик ЖС
  - опасность зацикливания, запрет ЖС на директорию`ln, link()`
- Мягкая (символическая) связь (soft (symbolic) link)  
`symlink()`  
`unlink()` - удаление