

Московский Физико-Технический Институт
Физтех-школа Аэрокосмических технологий
Институт Аэромеханики и Летательной Техники



Архитектура Компьютера и Операционные Системы.

Часть 2. Основы операционных систем

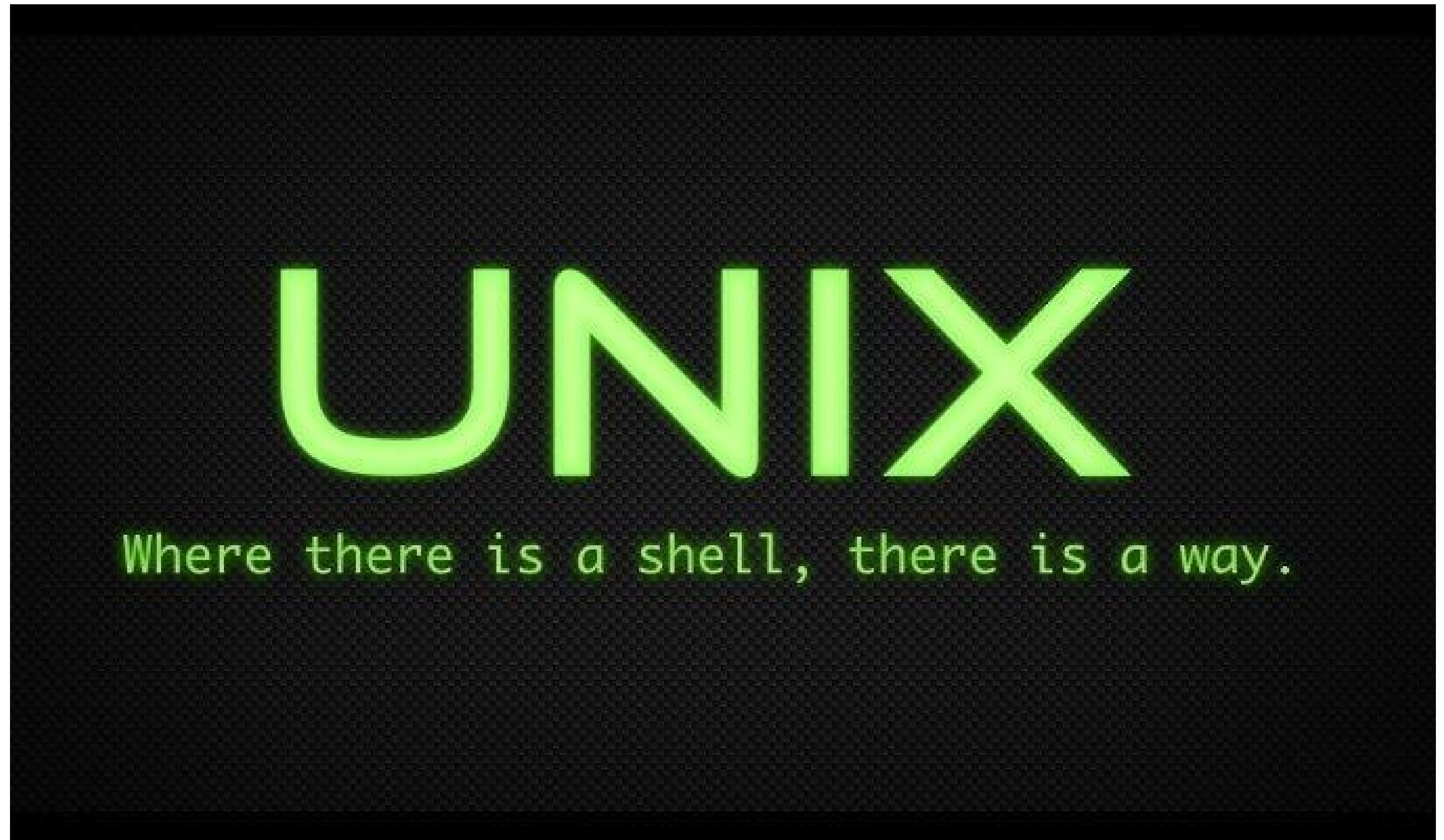
Лекция 2: Командная среда Unix

Новиков Андрей Валерьевич
д.ф.-м.н.

Жуковский

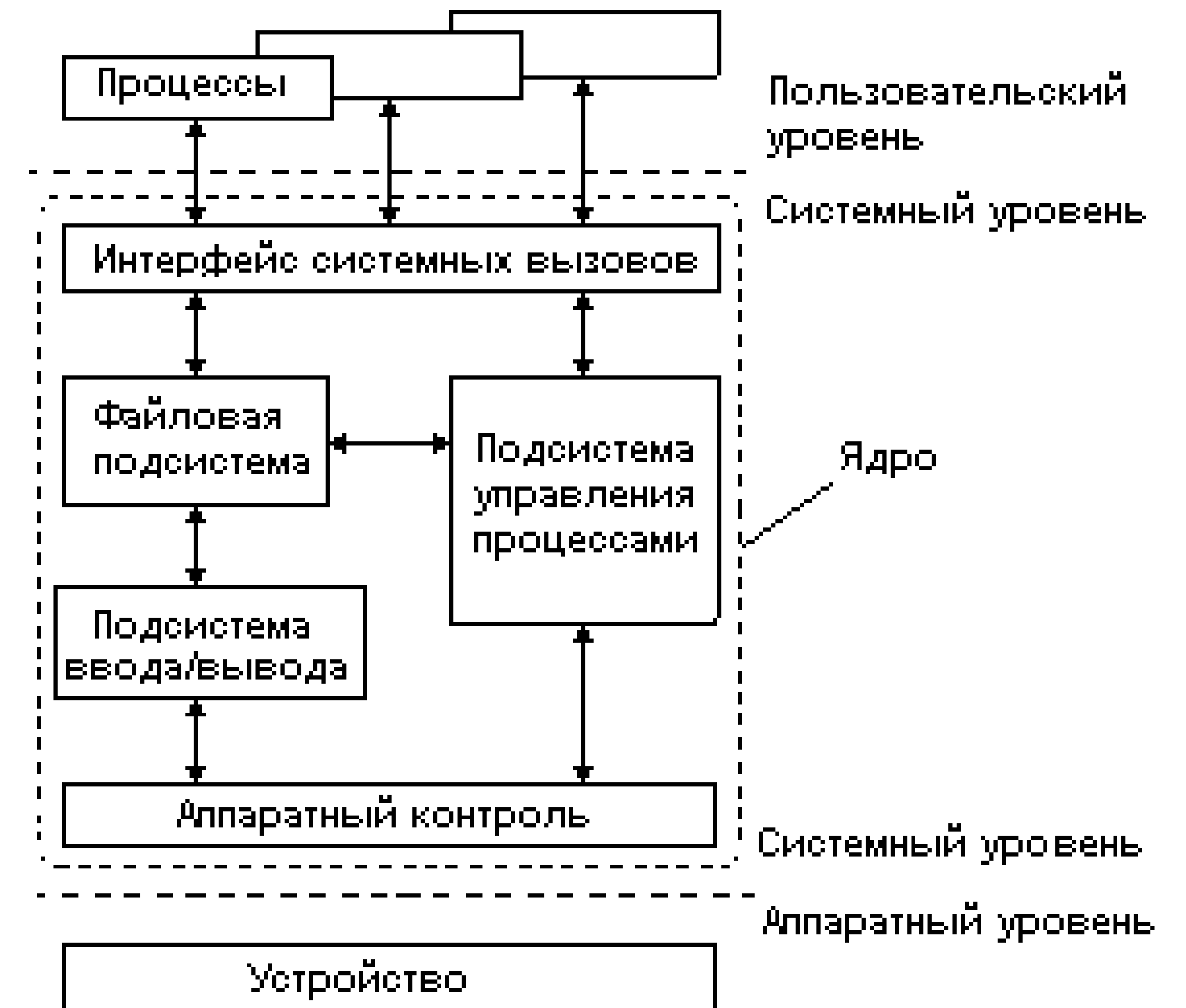
Оболочка - shell

- ❑ Командный интерпретатор
- ❑ Текстовый интерфейс командной строки
- ❑ Запуск программ, выполняющих полезные действия
- ❑ Автоматизация действий в виде программ
- ❑ Множество разновидностей:
 - sh (shell классическая)
 - bash (Bourne again shell)
 - ksh (Korn shell)
 - csh
 - zsh (Z shell)
 - ash (Almquist shell, минимальная)
 - ...



Системные вызовы

- ❑ Прямые точки входа пользовательских процессов в ядро
- ❑ Выполняются в привилегированном режиме для доступа к системным данным/устройствам
- ❑ Реализованы по-разному в различных ядрах
- ❑ Отображаются в вызовы функций языка C → интерфейс системных вызовов → стандартная UNIX-библиотека **libc**
- ❑ Стандарт **POSIX** = **P**ortable **O**perating **S**ystem Interface



Системные вызовы

```
#include <unistd.h> //<- int getuid()
#include <stdio.h>  //<- void perror(char*)

int main(int argc, char* argv[])
{
    int ret = getuid();
    perror("getuid result");
    return ret;
}
```

❑ Возвращаемое значение

- -1 – ошибка
- ≥ 0 – успех

❑ Точный код ошибки – в глобальной переменной **errno**

❑ При успешном завершении **errno** не меняется.

❑ Расшифровка последнего кода ошибки – функция **perror(char*)**

Синтаксис команд Unix

Команда и аргументы

- `$ command arg1 arg2 ... arg3`

☐ Аргументы, как опции:

☐ короткий формат

- `-a`
- `-A`
- `-bc`

☐ длинный формат

- `--help`
- `--verbose`

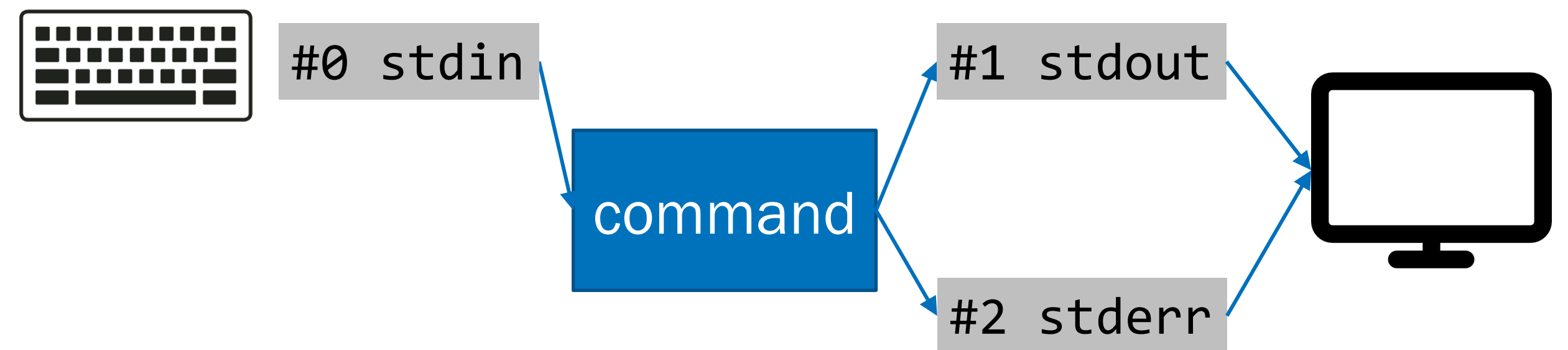
☐ Типичное описание в документации

- `command [arg1] [arg2] arg3`

☐ `[arg1]` – необязательный аргумент `arg1`

☐ `arg3` – обязательный аргумент `arg3`

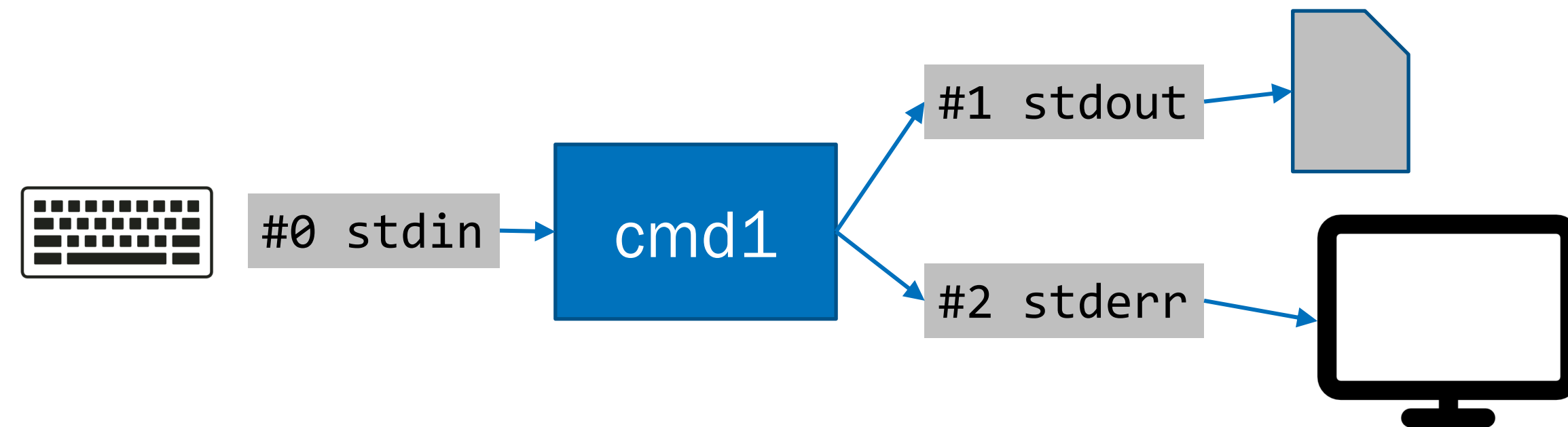
Потоки ввода-вывода



Перенаправление ввода-вывода

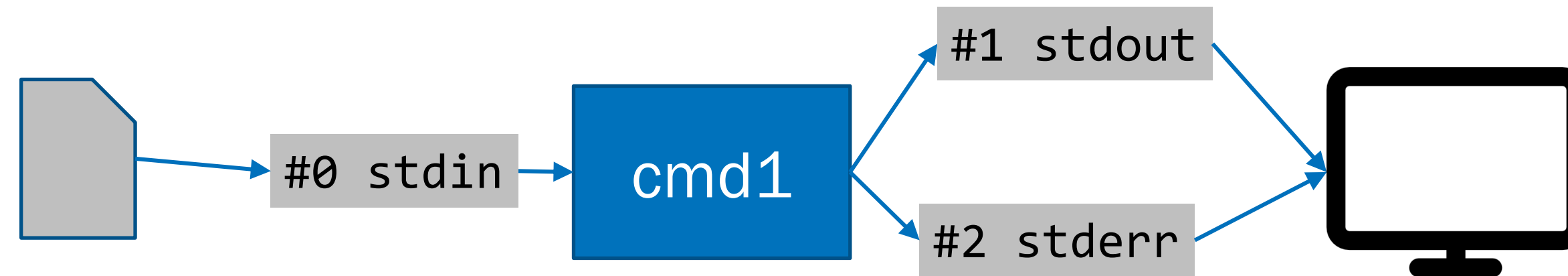
❑ Вывод в файл

```
$ cmd1 -a -b > out_file  
$ cmd1 -c -d >> out_file
```



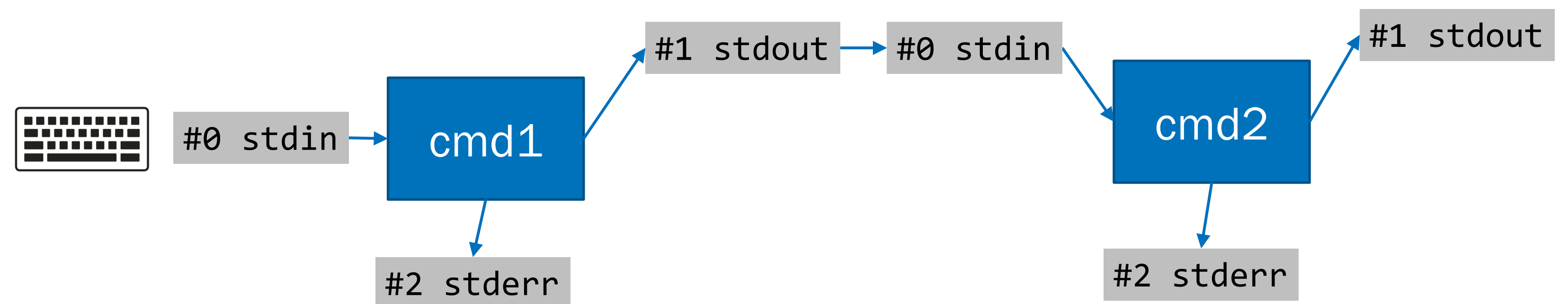
❑ Считывание из файла

```
$ cmd1 -e -f < in_file
```



❑ Конвейер

```
$ cmd1 -g -h | cmd2
```



Подстановки в оболочке (shell expansions)

В командной строке оболочка заменяет конструкции со спецсимволами *****, **?**, **\$**, **{}**, **[]**

❑ **a*z**

- ***** — любое количество любых символов
- заменяется на список имён файлов, имеющих в текущем каталоге, соответствующих шаблону
- подходят az, abcz, a_z, ...

❑ **a?z**

- **?** – один любой символ
- заменяется на список файлов имеющих в текущем каталоге, соответствующих шаблону
- подходят abz, acz, ...
- НЕ подходят az, abcz, ...

❑ **a[b-f0-9]z**

- **[]** – один любой символ из перечисленных
- список файлов из текущего каталога по шаблону

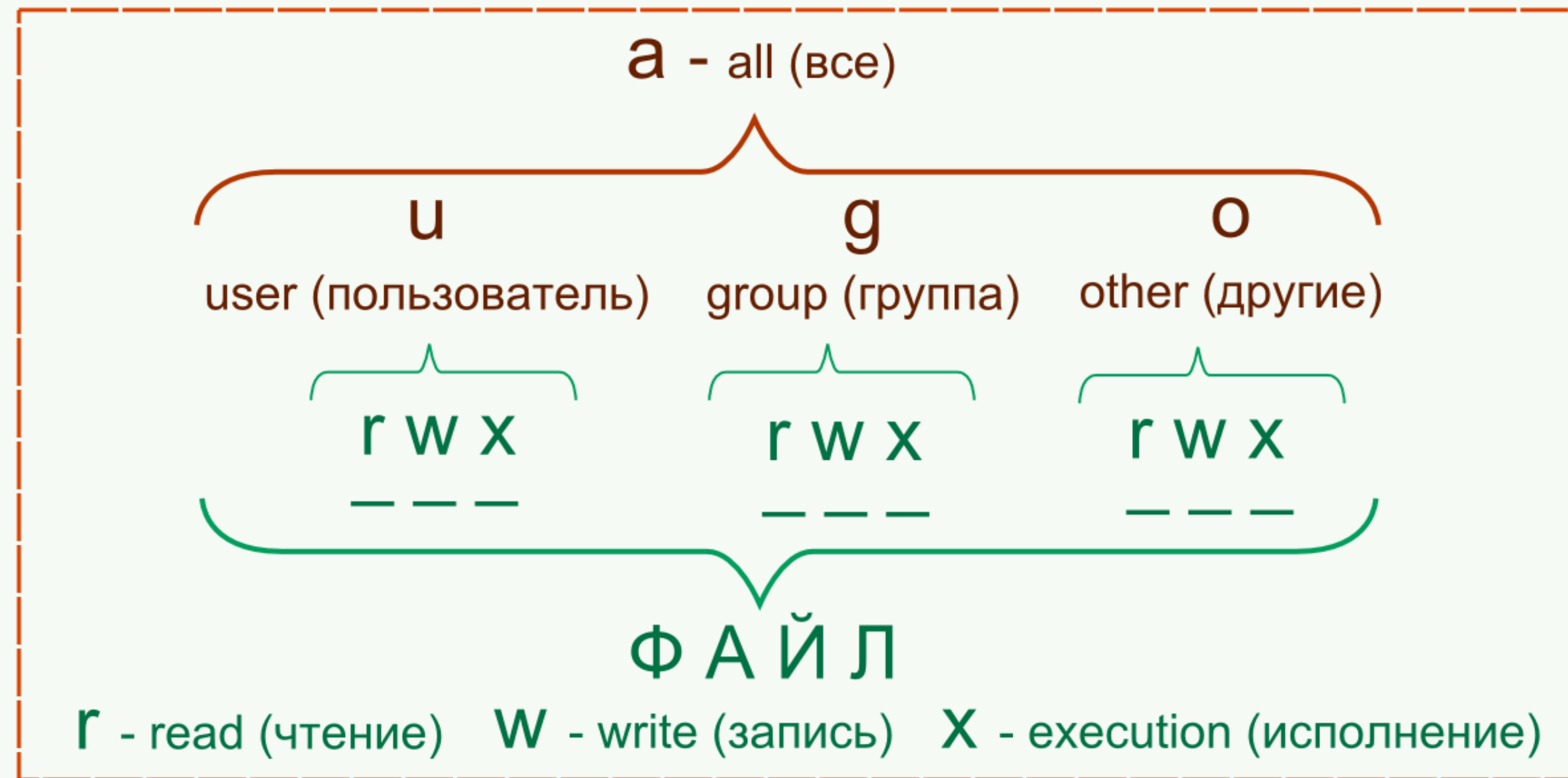
❑ **a{bc,de,gf}z**

- безотносительно к файлам или текущему каталогу
- генерирует список abcz adez agfz

❑ **\$var** или **\${var}**

- подставляет значение переменной по имени var
- задать значение как **var=value**

Права доступа



Примеры:

r w - r w - r w -	(все могут читать и изменять)
r w x - - - - -	(полный доступ имеет владелец файла)
r w - r - - r - -	(все могут читать, владелец также изменять)
r w x r - x r - x	(все могут читать и исполнять, владелец также изменять)

Права доступа. Числовая нотация

Примеры записи прав доступа в двоичной форме:

110 110 110	(все могут читать и изменять)
111 000 000	(полный доступ имеет владелец файла)
110 100 100	(все могут читать, владелец также изменять)
111 101 101	(все могут читать и исполнять, владелец также изменять)

Перевод представления прав доступа к восьмеричной форме:

гwx-представление	Двоичное число	Восьмеричное число	Значение
- - -	0 0 0	0	Все запрещено
- - x	0 0 1	1	
- w -	0 1 0	2	
- w x	0 1 1	3	
r - -	1 0 0	4	Только чтение
r - x	1 0 1	5	Чтение и исполнение
r w -	1 1 0	6	Чтение и запись
r w x	1 1 1	7	Все разрешено

Примеры записи прав доступа в восьмеричной форме:

6 6 6	(все могут читать и изменять)
7 0 0	(полный доступ имеет владелец файла)
6 4 4	(все могут читать, владелец также изменять)
7 5 5	(все могут читать и исполнять, владелец также изменять)