

# Взаимодействие процессов в UNIX

---

# IPC PIPE

Канал – однонаправленная передача данных между двумя процессами.

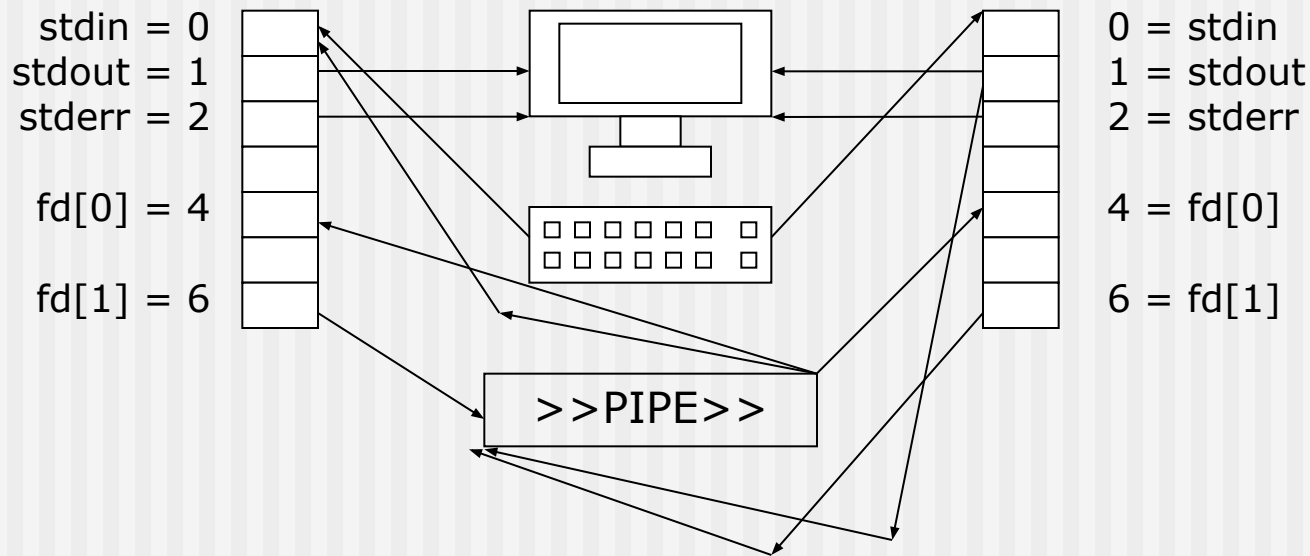
- Low level Piping

```
int pipe(int fd[2]);  
int filedes[2]={outdes, indes};
```

- Formatted Piping

```
FILE *popen(char *command, char *type);  
pclose(FILE *stream);
```

# Low-level piping



1. pipe(fd)
2. fork()
3. close(0)
4. dup2(fd[0],0)
5. close(fd[0])
6. close(fd[1])
7. exec\*("more")

1. close(1)
2. dup2(fd[1],1)
3. close(fd[0])
4. close(fd[1])
5. exec\*("ls -la")

# FIFO

```
int mknod(char *pathname, int  
    mode, int dev);
```

pathname – имя FIFO

mode – флаг S\_IFIFO, а также флаги  
владения, прав доступа и т.д.

dev – игнорируется

# Каналы

1. При чтении меньшего числа байт, чем есть в канале, остаток сохраняется
2. При чтении большего числа байт, чем есть в канале, возвращается доступное число байт
3. Если канал пуст и ни один процесс не открыл его на запись, будет возвращено 0 байт. Если какой-либо процесс открыл канал на запись, то `read` будет переведен в состояние ожидания (если канал не работает в режиме `NO_DELAY`).
4. Запись числа байт, меньше емкости канала гарантировано атомарно.
5. При записи числа байт, больше емкости канала, `write` блокируется, атомарность не гарантируется

# IPC FIFO

```
mknod("myfifo", S_IFIFO | 0666, 0);  
open("myfifo", O_RDONLY);  
open("myfifo", O_WRONLY);  
read(handle, buff, MAXBUFF);  
write(handle, "Hello, world!");  
close(handle);  
unlink("myfifo");
```

# Взаимодействие потоков в Win32

---

# Unnamed pipes

## Неименованные каналы

Однонаправленный локальный  
механизм взаимодействия

```
BOOL CreatePipe(  
    PHANDLE phRead,  
    PHANDLE phWrite,  
    LPSECURITY_ATTRIBUTES lpsa,  
    DWORD cbPipe );
```



# Named pipes

## Именованные каналы

Двухнаправленный механизм взаимодействия,  
МОЖЕТ ИСПОЛЬЗОВАТЬСЯ локально и удаленно

```
HANDLE CreateNamedPipe(  
    LPCTSTR lpszPipeName,  
    DWORD fdwOpenMode,  
    DWORD fdwPipMode,  
    DWORD nMaxInstances,  
    DWORD cbOutBuf,  
    DWORD cbInBuf,  
    DWORD dwTimeout,  
    LPSECURITY_ATTRIBUTES lpsa );
```

# Named pipes

## Именованные каналы

```
m_hPipe = ::CreateNamedPipe (  
    "\\.\pipe\ipcdemo", // Имя канала  
    PIPE_ACCESS_OUTBOUND, // Доступ только для записи  
    PIPE_TYPE_BYTE | PIPE_NOWAIT, // Запись байтов,  
    без ожидания  
    1, // По одному экземпляру за один раз  
    0, // Размер выходного буфера (байт)  
    0, // Размер входного буфера (байт)  
    0, // Время ожидания (мс)  
    NULL // Использовать дескриптор безопасности  
        // по умолчанию  
);
```

# Named pipes

## Именованные каналы

```
CString string = "\\\\" + strServerName +  
    "\\pipe\\ipcdemo";
```

```
hPipe = CreateFile(  
    string,  
    GENERIC_READ,  
    FILE_SHARE_READ,  
    NULL,  
    OPEN_EXISTING,  
    FILE_ATTRIBUTE_NORMAL,  
    NULL);
```