



西安邮电大学
XI'AN UNIVERSITY OF POSTS & TELECOMMUNICATIONS

Linux 编程技术



第4章 管道与重定向

——实现重定向(2)



主讲：黄茹

1. close then open

必须知道重定向文件的文件名才能使用open函数;

如果要重定向到管道或socket时, 无法使用open函数。

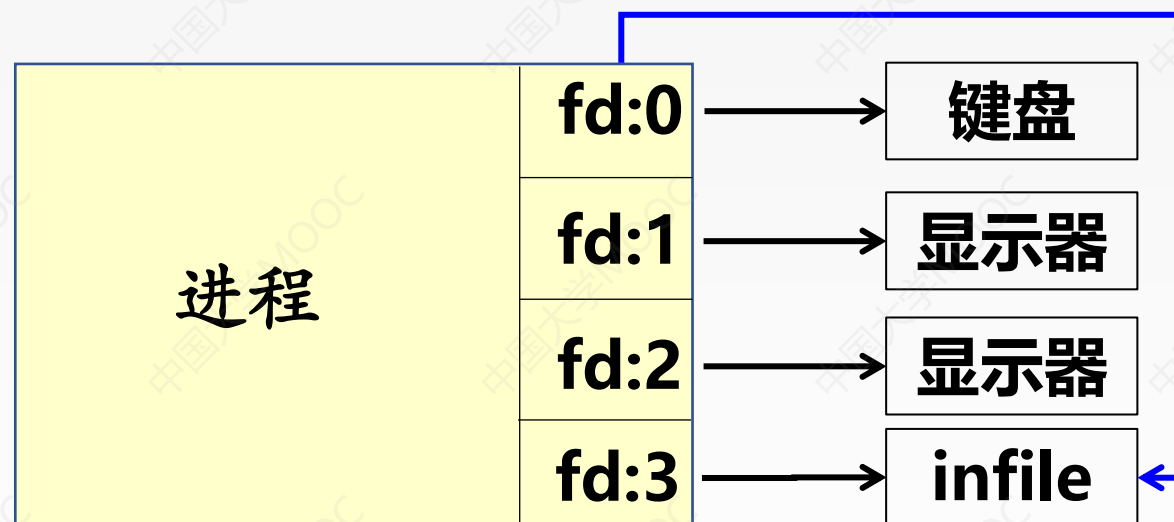
2. open...close...dup...close

在已知重定向文件的文件描述符时, 可省略open操作实现重定向。

dup	
功能	复制一个文件描述符
头文件	/usr/include/unistd.h
函数原型	int dup(int oldfd);
参数	oldfd 待复制的文件描述符
返回值	>-1 新的文件描述符
	-1 出错

dup遵循最低文件描述符原则。

过程分析



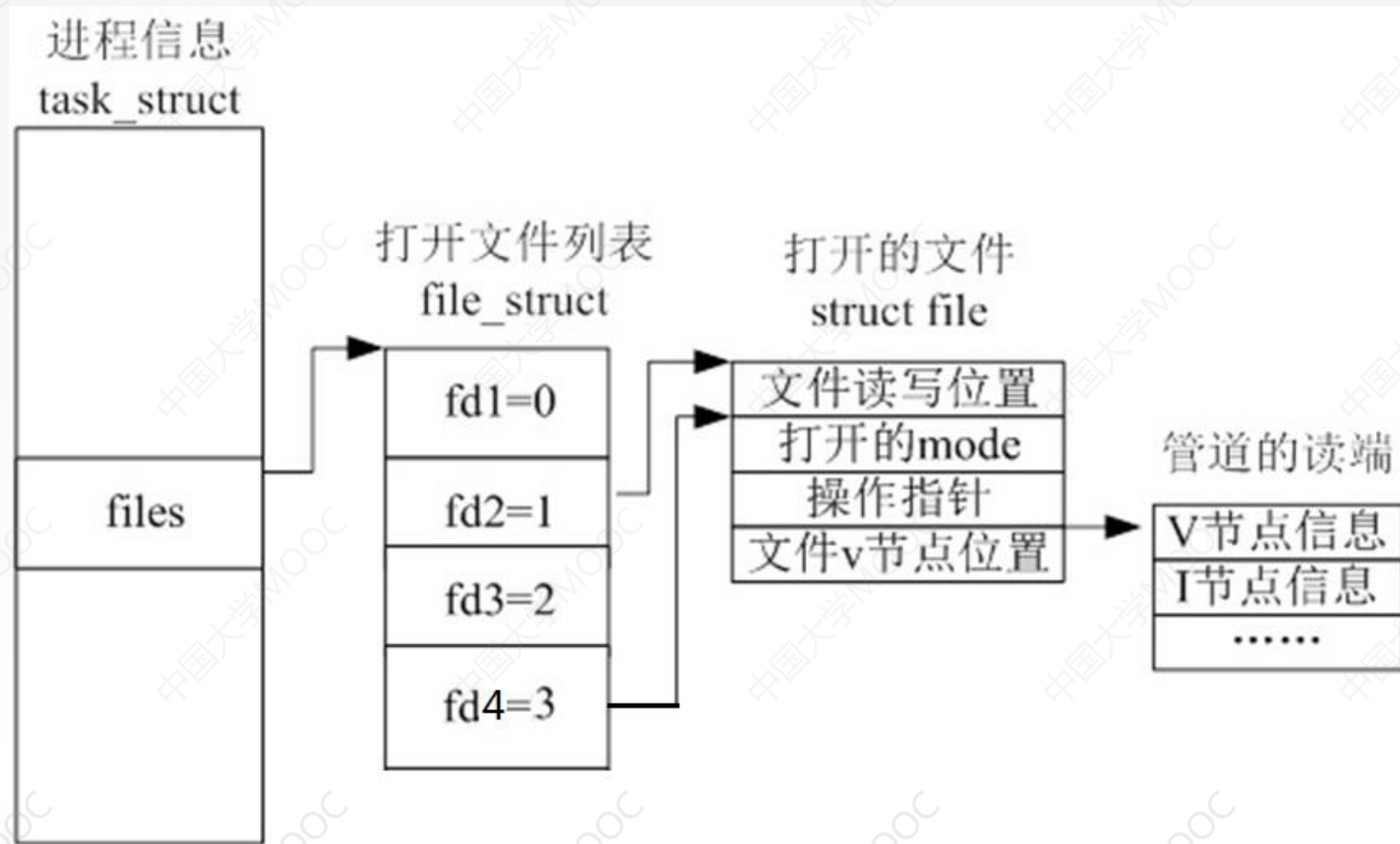
- `fd=open(infile,O_RDONLY)`
- `close(0)`
- `dup(fd)`
- `close(fd)`

dup2	
功能	复制一个文件描述符
头文件	/usr/include/unistd.h
函数原型	int dup2(oldfd, newfd);
参数	oldfd 待复制的文件描述符
	newfd 复制oldfd后得到的新描述符
返回值	>-1 新的文件描述符
	-1 出错

dup2在复制文件描述符时，如果newfd对应有打开的文件，那么系统会先关闭newfd，然后再复制。

使用dup或dup2所得的新文件描述符和oldfd共用一个文件表项

```
fd4=dup (1) ;
```



```
#include <stdio.h>
#include <stdlib.h>
#include <fcntl.h>

main() {
    int fd,newfd;    char buf[80];
    read(0,buf,80);  write(1,buf,80);
    fd=open("/etc/passwd",O_RDONLY);
    close(0);
    newfd=dup(fd);
    if(newfd!=0)    {perror("dup");    exit(1);  }
    close(fd);
    read(0,buf,80);  write(1,buf,80);
}
```



```
#include <stdio.h>
#include <stdlib.h>
#include <fcntl.h>

main() {
    int fd,newfd;    char buf[80];
    read(0,buf,80);  write(1,buf,80);
    fd=open("/etc/passwd",O_RDONLY);
    newfd=dup2(fd,0);

    if(newfd!=0)    {perror("dup");    exit(1); }
    close(fd);
    read(0,buf,80);  write(1,buf,80);
}
```




西安邮电大学
XI'AN UNIVERSITY OF POSTS & TELECOMMUNICATIONS

Linux 编程技术

谢谢大家!