Intro to Project 3: W_{isconsin}Sh_{ell}

CS537 Operating Systems Discussion Section 3

Yurun Yuan, Teaching Assistant

Content

- Project 3 Overview
- Makefile
- Multiprocessing
- Signaling
- IO redirection

Project 3 Overview: WisconsinShell

- Builtin commands
  - cd, exit
- Commands
  - You can assume that all commands being executed are located under /bin
  - Background job: `<cmd> <arg1> <arg2> … &`
- Job control
  - Jobs, fg, bg
- Signal
  - Ctrl+C, Ctrl+Z
- Pipes
  - cmd1 | cmd2

Makefile (will be graded in project 3)

```
BIN = signals multiprocess redirection

all: $(BIN)

signals: signals.c
    gcc -o signals signals.c

multiprocess: multiprocess.c
    gcc -o multiprocess multiprocess.c

redirection: redirection.c
    gcc -o redirection redirection.c

clean:
    rm -rf $(BIN) output.txt
```

Variables

Variables are used to store values that are reused throughout the Makefile.

Target Rules

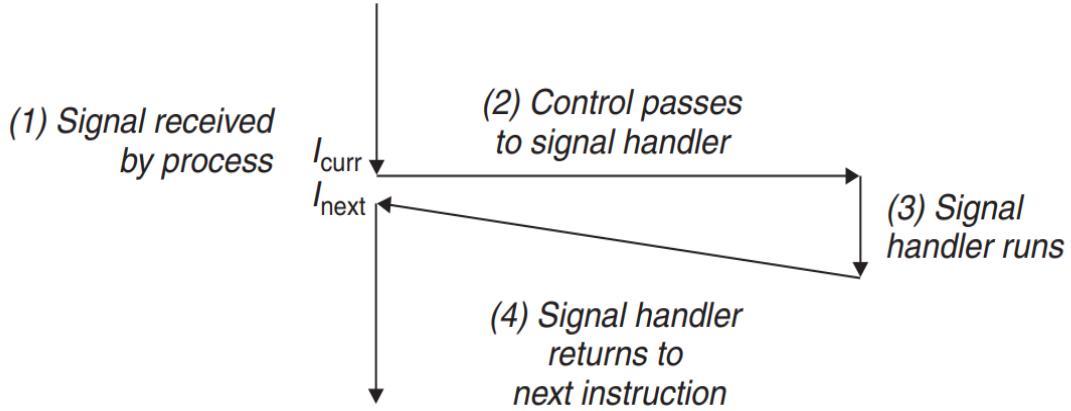
Target rules specify the target to be built and the dependencies required for that target.

Multiprocessing

- fork, exec, and waitpid

Signaling

```
1   #include "csapp.h"
2
3   void sigint_handler(int sig) /* SIGINT handler */
4   {
5       printf("Caught SIGINT!\n");
6       exit(0);
7   }
8
9   int main()
10  {
11      /* Install the SIGINT handler */
12      if (signal(SIGINT, sigint_handler) == SIG_ERR)
13          unix_error("signal error");
14
15      pause(); /* Wait for the receipt of a signal */
16
17      return 0;
18  }
```



Signaling (cont'd)

Sending Signals with the `kill` Function

Processes send signals to other processes (including themselves) by calling the `kill` function.

```
#include <sys/types.h>
#include <signal.h>

int kill(pid_t pid, int sig);
```

Returns: 0 if OK, −1 on error

If pid is positive, then signal sig is sent to the process with the ID specified by pid.

If pid equals 0, then sig is sent to every process in the process group of the calling process.

If pid equals -1, then sig is sent to every process for which the calling process has permission to send signals, except for process 1 (init), but see below.

If pid is less than -1, then sig is sent to every process in the process group whose ID is -pid.

IO Redirection

- How is IO redirection in shell implemented?
  - **`echo hello > output`**
  - Redirect stdout to file “output”
- How is pipe in shell implemented?
  - **`cmd1 | cmd2`**
  - The stdout of cmd1 is sent to the stdin of cmd2

IO Redirection (cont'd): review, what is a file descriptor?

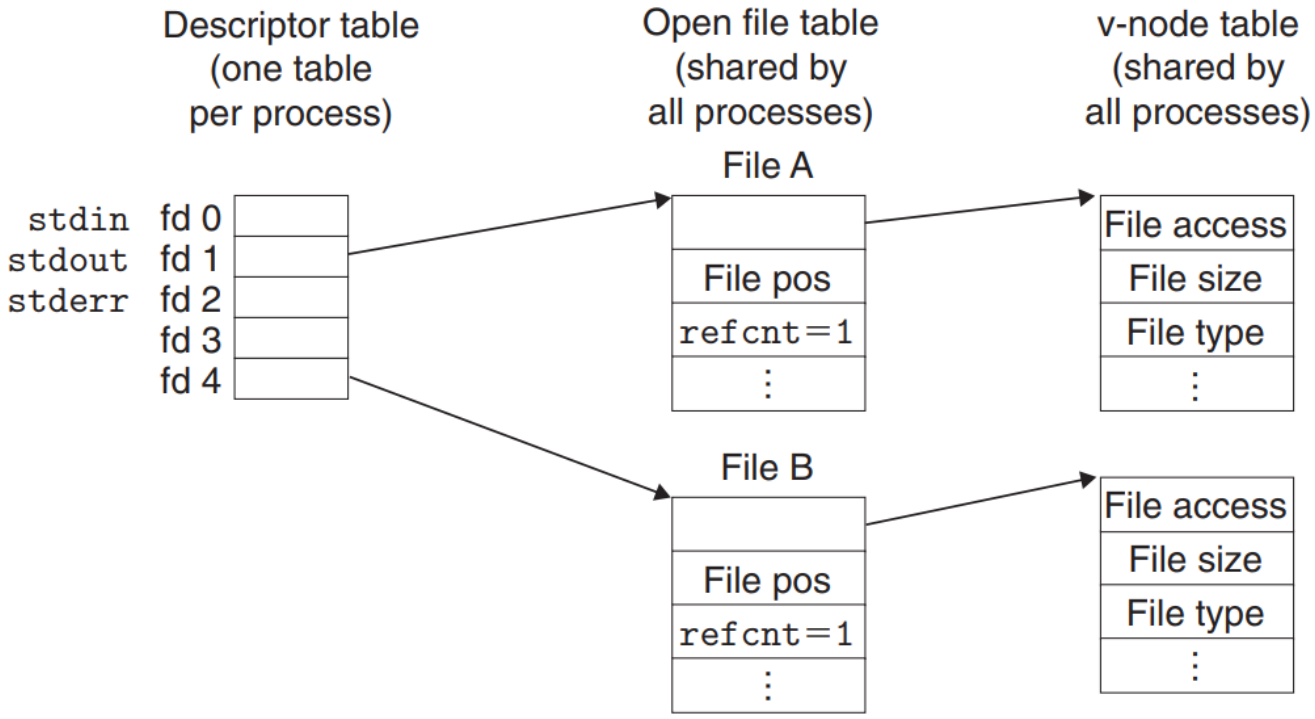
```
int fd = open("filename", MODE);
_ = read(fd, buf, count);
```

Special fd:

- stdin: 0
- stdout: 1
- stderr: 2

IO Redirection (cont'd): how does kernel represent open files

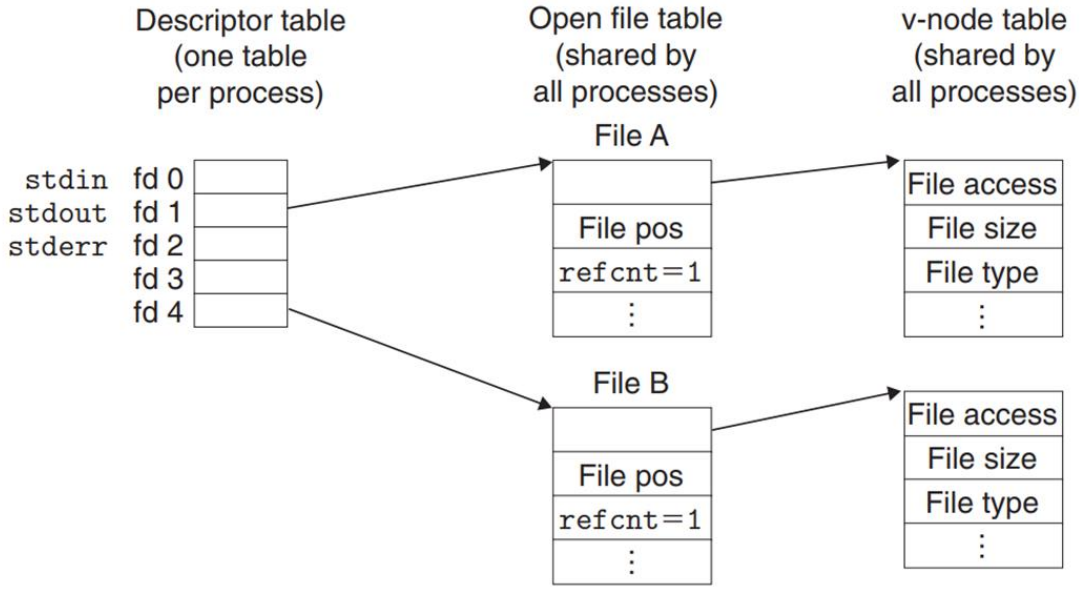
Figure 10.12
Typical kernel data structures for open files. In this example, two descriptors reference distinct files. There is no sharing.



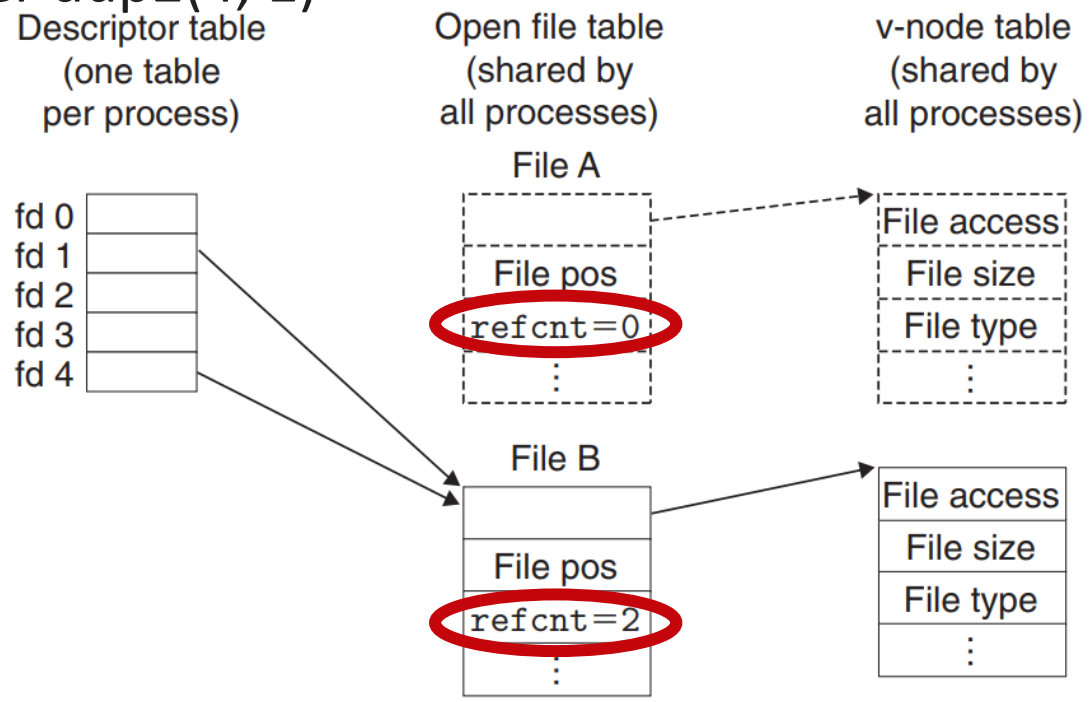
IO Redirection (cont'd): dup2

```
int dup2(int oldfd, int newfd);
```

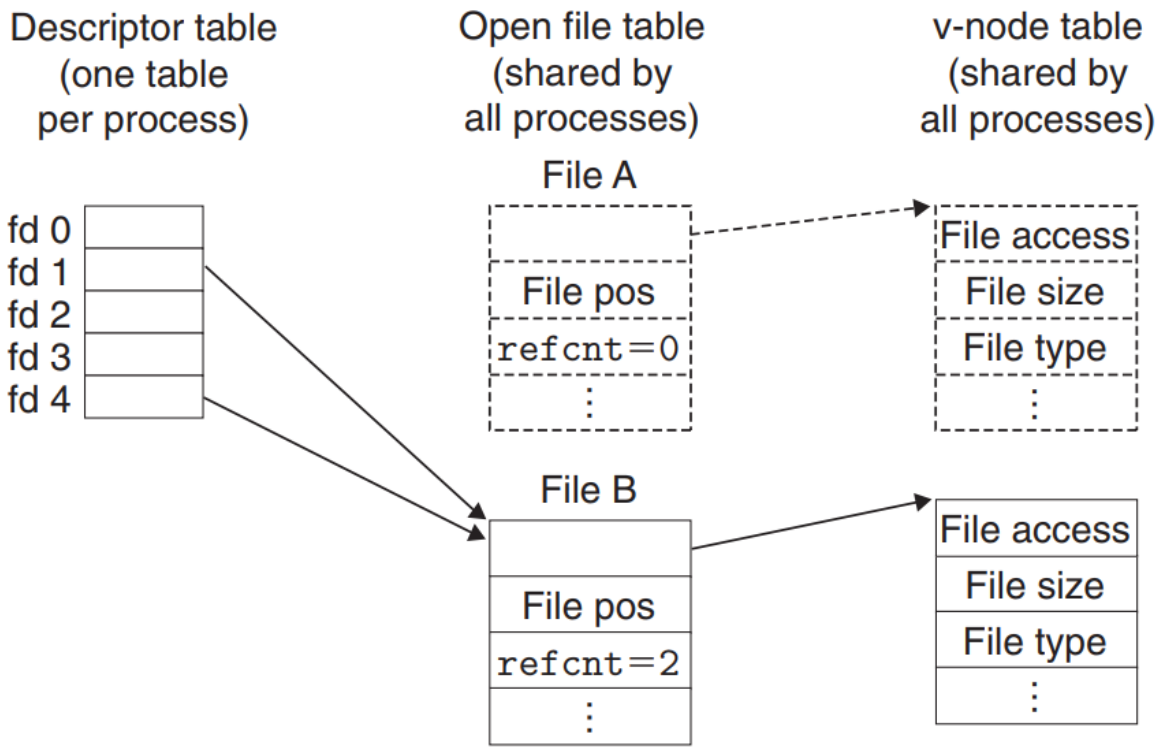
Before dup2(4, 1)



After dup2(4, 1)



IO Redirection (cont'd): how to implement IO redirection in shell




WISCONSIN
UNIVERSITY OF WISCONSIN–MADISON