【Linux Programming】Day1

Class	Understanding Linux/Unix Programming
Date	@March 8, 2022

【Ch1】Unix Systems Programming: The Big Picture

1.2 What is systems programming?

1.2.1 The Simple Program Model

Many programs are based on the model in the figure below:

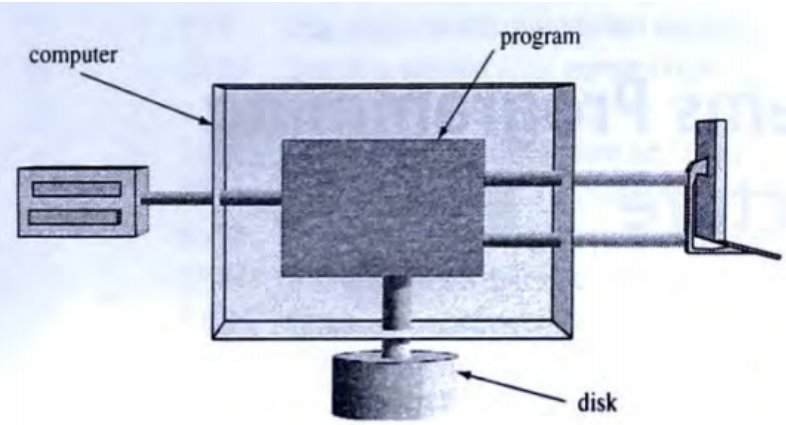

FIGURE 1.1
An application program in a computer.

A program is a piece of code that runs in a computer. Data go into the program, the program does something with the data, and data come out of the program.

In this model of a program, one writes clear, sensible code such as

```
// copy from stdin to stdout
int main() {
  int c;
  whil( (c = getchar()) != EOF)
    putchar(c);
}
```

This code corresponds to a visual model depicted in the figure below:

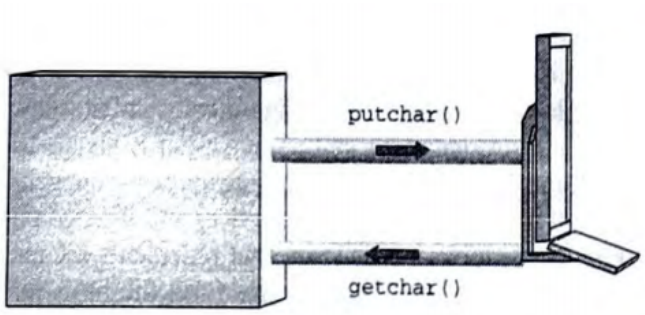

FIGURE 1.2
How application programs see user I/O.

1.2.2 Face Reality

What if we log into a multiuser system, like a typical UNIX machine?

In that case, the simple model of keyboard and monitor wired to the CPU is false. The real picture is depicted below.

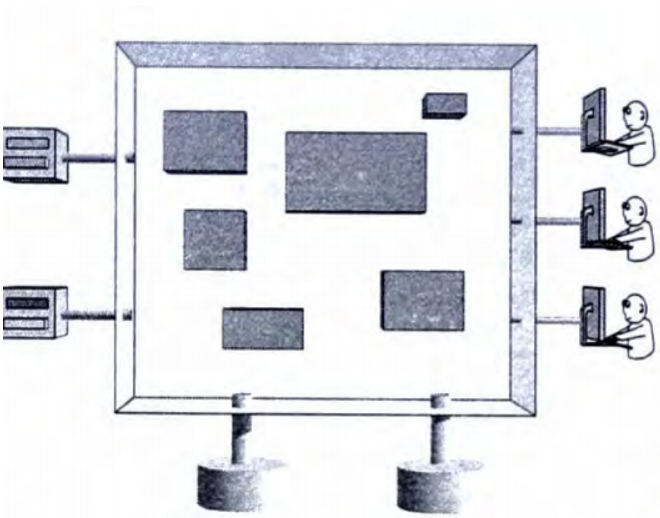

FIGURE 1.3
Reality: many users, programs, and devices.

Nonetheless, programs that get input from the keyboard and send output to the display or to the disk work fine. Programs can assume the simply model and still get the correct result.

However, something more complicated is going on. Somehow all these various keyboards are connected to the various programs and there are lots of connections

inside that machine.

1.2.3 The Role of the Operating System

The role of an operating system is to manage and protect all the resources and to connect the various devices to the various programs.

The operating system is a program. The code for the operating system resides in the computer's memory. The memory also contains other programs. The operating system connects those programs to the outside world.

1.2.4 Providing Services to Programs

The part of the computer memory where the operating system is stored is called system space, and the part of the memory that contains user programs is called user space.

The operating system is called the kernel.

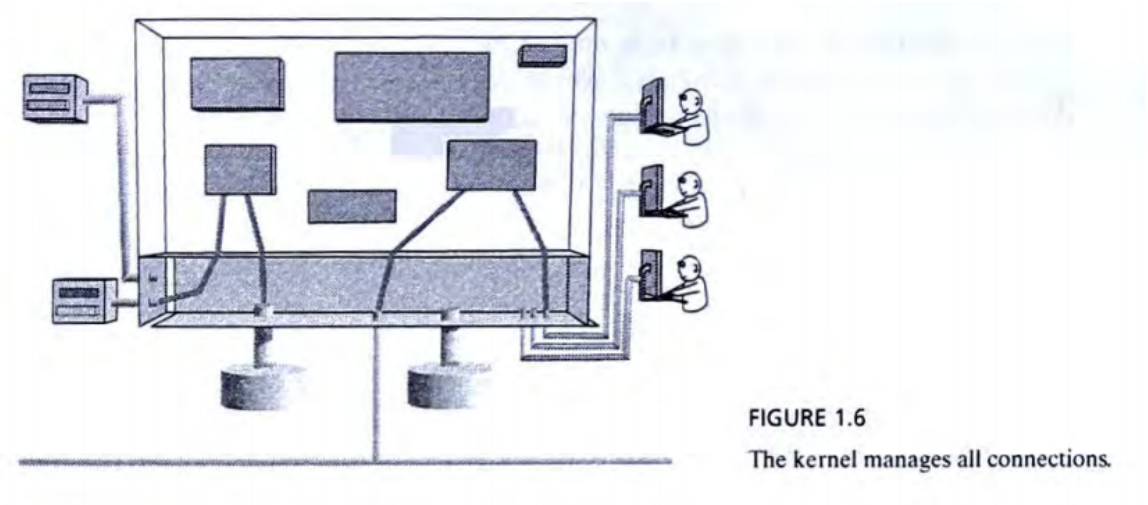

FIGURE 1.6
The kernel manages all connections.

Notice that the place where the wires connect is in system space; thus, the kernel is the only program with access to those devices.

User programs get data by asking the kernel. If a program wants to control any of those resources, it needs to ask the kernel.

Access to the external objects are services the kernel provides to user programs.

We shall study what services the kernel provides, the structure of those services, and how to write programs that work in this larger context.

1.3 Understanding Systems Programming

The kernel provides access to system resources. System programs use these services directly. *What are those services and how can we learn to use them?*

1.3.1 System Resources

Processors

A processor is the hardware that executes instructions.

Input/Output

All data that flow in and out of programs go through the kernel. This centralization ensures that data are transferred correctly-data go to the right place; efficiently-no extra time is wasted getting information from one place to another, and securely-no process is able to look at information it is not supposed to see.

Process Management

Unix uses the term process to refer to a program in action. A process consists of memory, open files, and other system resources a program needs to run. The kernel creates new processes, schedules processes, and arranges for them to work cooperatively.

Memory

The kernel keeps track of which processes are using which sections of memory and protects the memory of one process from being damaged by another process.

Timers

Some programs depend on time. They may perform an operation at set intervals; they may need to wait a while an then do something. The kernel makes timers available to processes.

Interprocess Communication

Processes need to communicate. The kernel provides various forms of interprocess communication.

Networking

A network connection computers is a broader form of interprocesss communication. A network allows processes on different computers to exchange data. Network access is a kernel service.