Paul Kummer

Dr. Hanku Lee

Operating Systems CSIS 430-01

09/04/2021

Summarization of Chapter 1.5: “Operating System Concepts”

1.5.1 Processes

With an operating system, programs are executed, and their execution requires system resources. From the operating system’s perspective, each program is a process that gets a specified allotment of CPU time and memory. The space that a process gets is referred to as the address space. Since many programs can be running at the same time with multiprogramming, whenever the execution of one program is paused to start the execution of another, the computer must save the state of the currently running program. This means that the pointers to the stack and program counter must be preserved so that the program can resume when it is allowed to run again. All the information about currently running programs is stored in a process table, allowing many programs to pick up where they left off when they are suspended. The process will create a core image, its current state or address space, when it receives an alarm signal, indicating that it is about to lose privileges to use resources.

Often, processes can spawn additional processes to aid in completing a program's task. These processes from another process are called child processes. They often communicate with other processes, referred to as interprocess communication, to work towards a common goal in the parents' process.

All users of a system are assigned a user identification, UID, that is unique to that user. This UID is attached to all the processes that a user starts. Users can also be members of groups and be associated with a group id, GID. Certain groups may have certain privileges that apply to all members of that specific group. There are some users that have special elevated privileges and can be referred to as super user or administrator, depending on what operating system is used.

1.5.2 Address Spaces

Modern computers allow multiple processes to run and be stored in memory. Each process has its own allocated space in memory that it can use. The operating system has built in protections to prevent one process from exceeding its memory allocation. However, some processes may need and request more memory than is available. The operating system will that create virtual memory to allow for a larger address space. This address space will cause part of a larger permanent drive to act like memory, but it will be much slower.

1.5.3 Files

Operating systems provide its users with an abstract model of files to have a logical structure organizing data that is stored on permanent storage, but the structure is independent of the storage device. The operating system also provides ways of creating, removing, accessing, and closing files.

Files in most modern operating systems are part of a directory. A directory is a file that allows organization of files and can contain other directories. Files and directories have path names that indicate where they are located from the root directory. The root directory is the directory that every file is located within.

Every process in an operating system will have a current working directory. The working directory is the location where the process resides. For instance, if a program is launched from the directory C:\Users\pakum\Desktop\operatingSystems, the working directory is within "operatingSystems". The process can use system calls to access data in different directories, but by default, the process will look within the working directory for any files that are attempting to be accessed.

Some devices attached to a computer may have their own internal file structure. With an operating system like UNIX, devices can be mounted. After a device is mounted, a path from the root directory will be appended to the directory of the device's files. This will in effect create a file structure with one tree.

1.5.4 Input/Output

Computers almost always have input and output devices. The most common devices are keyboards, mice, monitors, and printers. These devices are controlled by the operating system with the use of device drivers. Some drivers are generic and apply to many devices, but there are also hardware specific drivers.

1.5.5 Protection

Some of the information contained in computers must be secured and is not intended for everyone to access. Modern computers have ways of protecting this information by having files permission levels for certain users. In UNIX, every file has a 3-bit protection code that specifies if users have read (r), write (w), or execute (x) privileges. The three bits represent read, write, execute in that order. If all three bits are set to one, then a user will have read, write, and execute privileges. If a user is a super user, or has elevated privileges, they can change the 3-bit protection code with he chmod command in the shell.

1.5.6 The Shell

The shell is not part of the operating system but has the ability to make systems calls similar to the operating system. A shell is started whenever a user logs into a machine. There are different variations to the shell like sh, bash, csh, and ksh. By using a shell, the user is able to interface with the operating system.

When using the shell, it will wait for user input, which is indicated with a prompt. The user can type in a command and the shell will create a child process that executes what the user wants and return to a prompt after the child process terminates. Output from one command can be redirected as input to another command with the use of piping (|). For example, if a person wanted to print the current date on a printer, they could type date | /dev/lp at the prompt. This would then execute a child process to get todays date. That process would terminate, and output would then be piped to the printer to be printed.

Today most users use a graphic user interface (GUI) to interface with an operating system. There can be different ways to interface with the same operating system, giving them different

looks and usability. This is evident through the different flavors of Linux like KDE, XFE, Mint, and Gnome.

1.5.7 Ontogeny Recapitulates Phylogeny

Computers go through and have been through many changes to become what they are today. They have also changed what is believed to be possible and what is currently possible. Technology is often a driving force that begins many changes. For example, the rapid storage and exchange of patient information from nurse to nurse and hospital to hospital was never possible and even thought of until computers and software advanced to a point where it is now almost unheard of for nurses to chart on physical paper. Technology will continue to change and so will out views of what is possible and how we live.

Whenever computers advance their technological capabilities, what may have once been the bottleneck restricting performance may no longer be the case. Instead, a part of the computer that may have been the fastest may now be a bottleneck, which could give rise to using technology that was abandoned because the cost of its use didn't improve performance at the time. Technology evolves much like biological beings in the sense that they adapt to their conditions and morph to fit the current environments.

Large Memories

The original computers had very limited amounts of memory and were therefore programmed using assembly language. When memory started getting larger, higher level programming languages were used like FORTRAN and COBOL. Then computers became smaller and more affordable, leading to programming resorting to assembly language due to smaller memories. There has been a back and forth in programming languages, which is mostly due to the size of the memories available.

Protection Hardware

Early computers had no hardware protection, so only one program could run at a time and if that program had bugs, the operating system could fail and cause the computer to crash. Then, computers had hardware protection introduced, allowing more than on program to be stored in memory and run. This gave rise to multiprogramming. However, like the fluctuation of memory on computers as they became more portable, hardware protection was lost when computer became smaller and monoprogramming came back into practice. Later multiprogramming made its way into these smaller minicomputer systems when hardware protection was developed. There are common cycles with technology that happen as hardware becomes more compact and new technologies are invented.

Disks

Data storage on the first computers was done on magnetic tapes. This was then replaced by a hard disk developed by IBM. The hard disk could store very little and was expensive to rent. Then the hard disk improved allowing more storage and all its files stored in what could be considered a single directory. Files, if they shared a same name, would be overwritten because the operating systems had no way to differentiate the files. Eventually hierarchical file systems were developed allowing files of the same names to be stored in different directories. Hard disks were not immune to the cycles of technological development, and parts of hard disks had to regress in technological advances as the hardware became smaller.

Virtual Memory

In modern computers, if a computer attempts to a run a program that is larger than its physical memory, the computer will allocate some of its hard drive space to be used like memory to make up for the lack of physical memory. The space on the hard drive acting like memory is called virtual memory. It is much slower than regular memory because the read/write speeds are significantly slower. Like all things relating to computers, virtual memory has disappeared from existence just to reemerge in different technology.

Works Cited

Tanenbaum, A., & Boschung, H. T. (2015). Modern operating systems (4th ed.). Pearson.