OPERATING SYSTEMS

Module1_Part3

Operating-System Services

Operating systems provide an environment for execution of programs and services to programs and users

- One set of operating-system services provides functions that are helpful to the user:

  User interface - Almost all operating systems have a user interface (**UI**).

  Eg: Command-Line (CLI), Graphics User Interface (GUI), touch-screen

  Program execution - The system must be able to load a program into memory and to run that program, end execution, either normally or abnormally (indicating error)

  I/O operations - A running program may require I/O, which may involve a file or an I/O device

Operating system services

File-system manipulation - The file system is of particular interest. Programs need to read and write files and directories, create and delete them, search them, list file Information, permission management.

Communications – Processes may exchange information, on the same computer or between computers over a network. Communications may be via shared memory or through message passing.

Error detection – OS needs to be constantly aware of possible errors

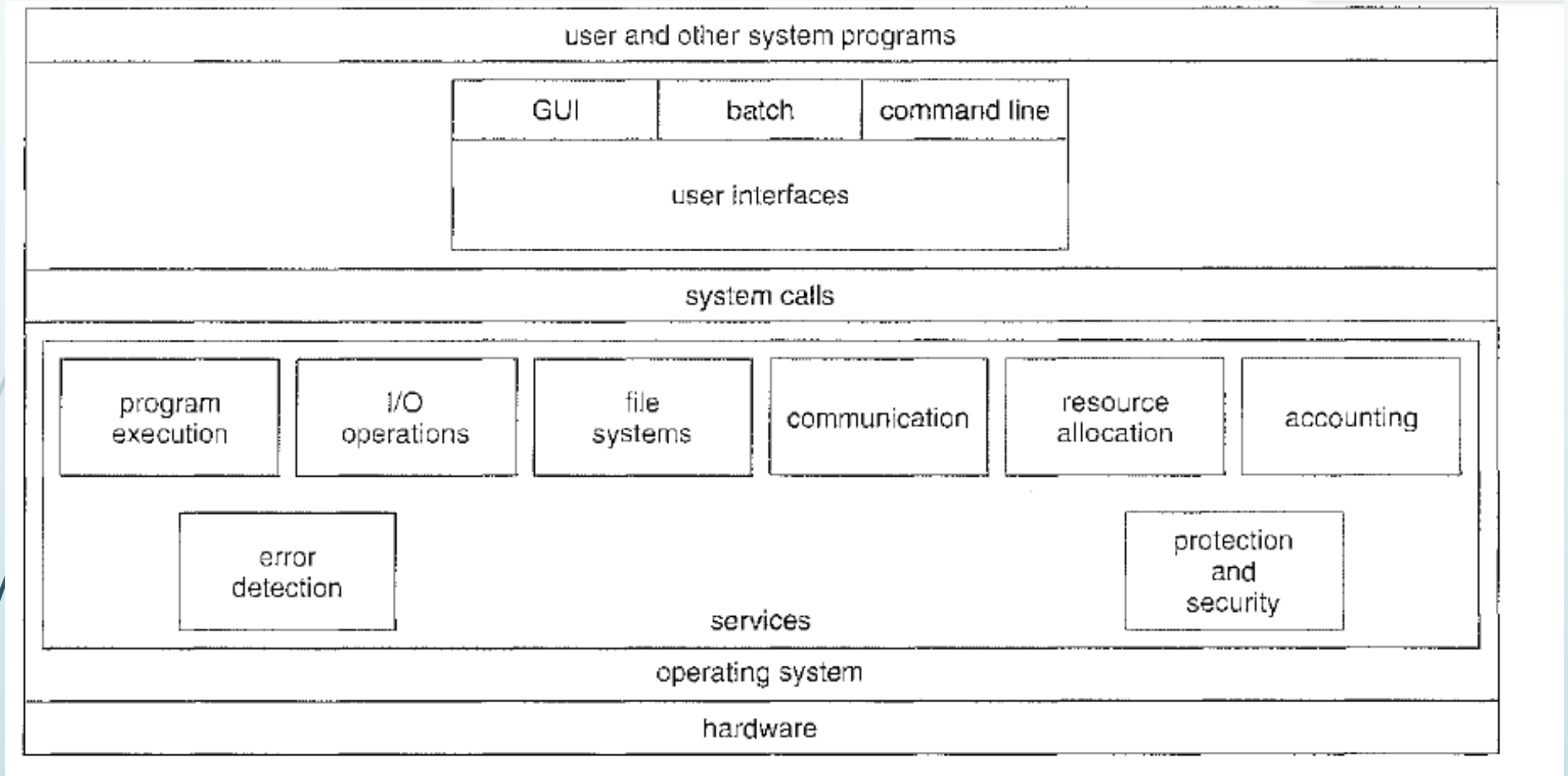
Resource allocation - When multiple users or multiple jobs running concurrently, resources must be allocated to each of them

Accounting - To keep track of which users use how much and what kinds of computer resources

Protection involves ensuring that all access to system resources is controlled

Security of the system from outsiders requires user authentication, extends to defending external I/O devices from invalid access attempts

Operating system services



CLI -- command line interpreter

Sometimes implemented in kernel, sometimes by systems program

Sometimes multiple flavors implemented – **shells**

Primarily fetches a command from user and executes it

GUI—Graphical user interface

users employ a mouse-based window and-menu system

Depending on the mouse pointer's location, clicking a button on the mouse can invoke a

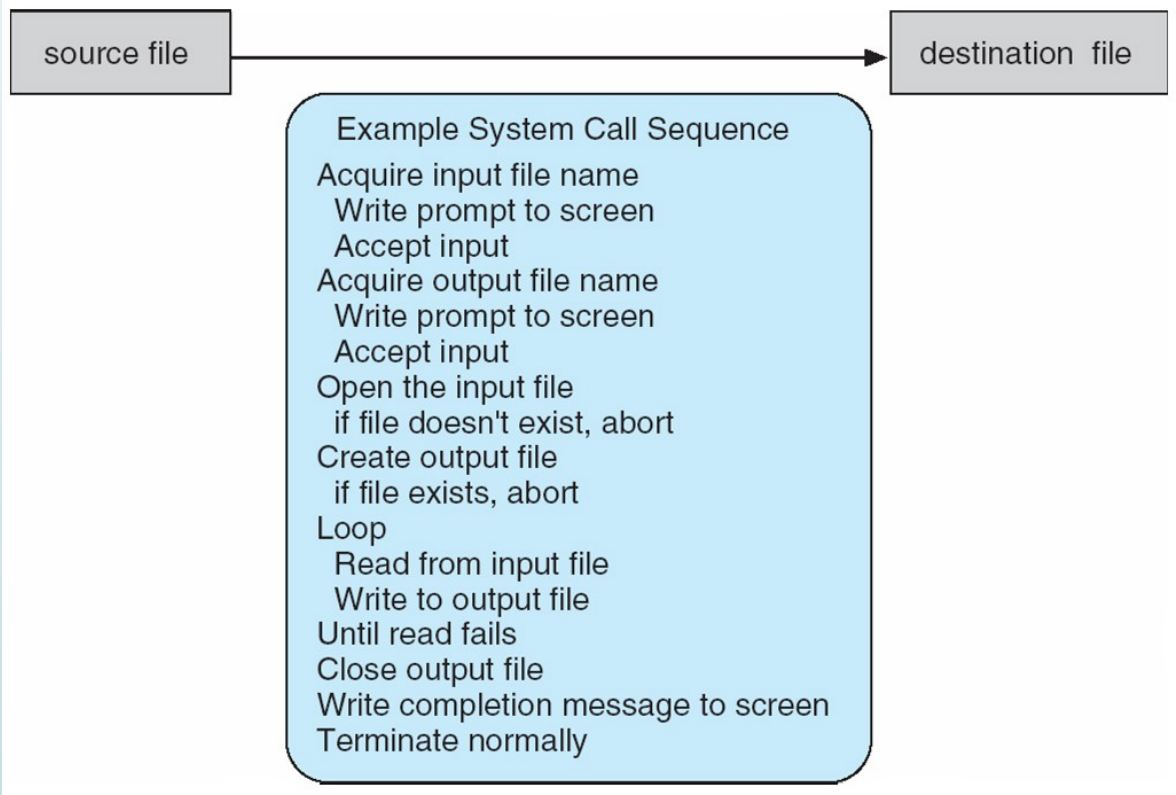
program, select a file or directory.

System calls

- A system call is a mechanism that provides the interface between a process and the operating system. A system call is a programmatic way a program requests a service from the kernel.
- To understand how an operating system works, you first need to understand how system calls work.
- System calls are very similar to function calls, which means they accept and work on arguments and return values. The only difference is that system calls enter a kernel, while function calls do not.
- System call offers the services of the operating system to the user programs via API (Application Programming Interface).

System calls

System call sequence to copy the contents of one file to another file



System call implementation

- Typically, a number is associated with each system call
  - **System-call interface** maintains a table indexed according to these numbers
- The system call interface invokes the intended system call in OS kernel and returns status of the system call and any return values
- The caller need not know anything about how the system call is implemented
  - Just needs to obey API and understand what OS will do as a result call
  - Most details of OS interface hidden from programmer by API

API: System Call to Open a File