Operating System Overview

CHAPTER 2

Operating System

- A program that
  - controls the execution of application programs
  - acts as an interface between applications and hardware
- OS
  - masks the details of the hardware from the programmer
  - provides the programmer with a convenient interface for using the system

User/Computer Interface

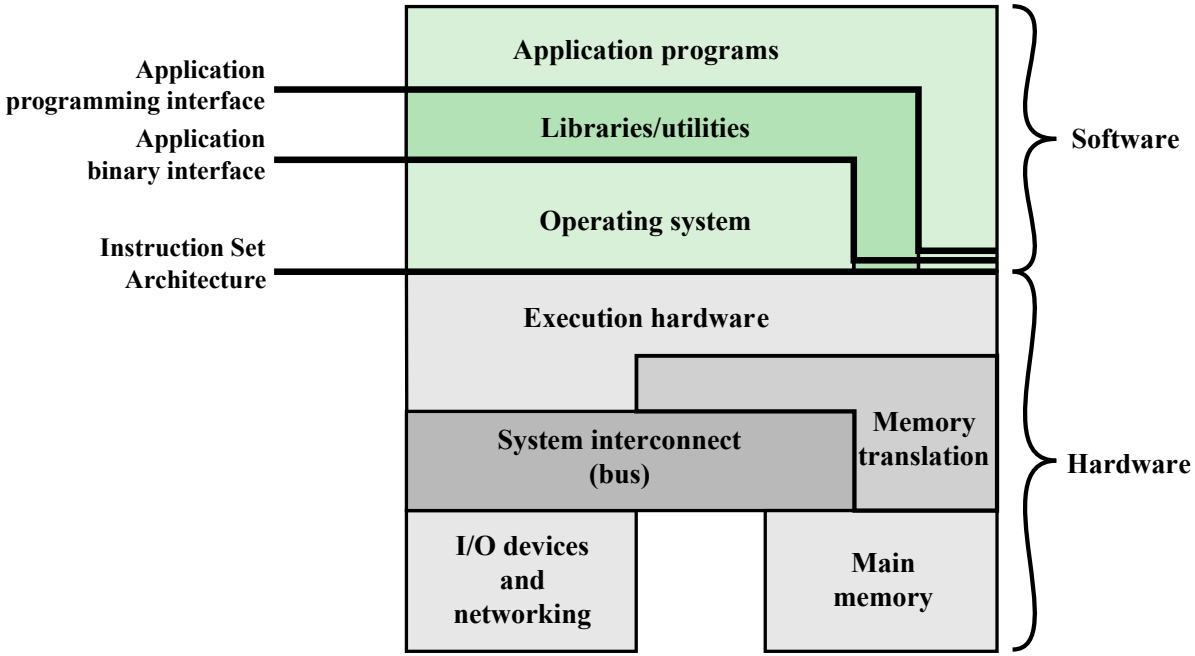


Figure 2.1 Computer Hardware and Software Structure

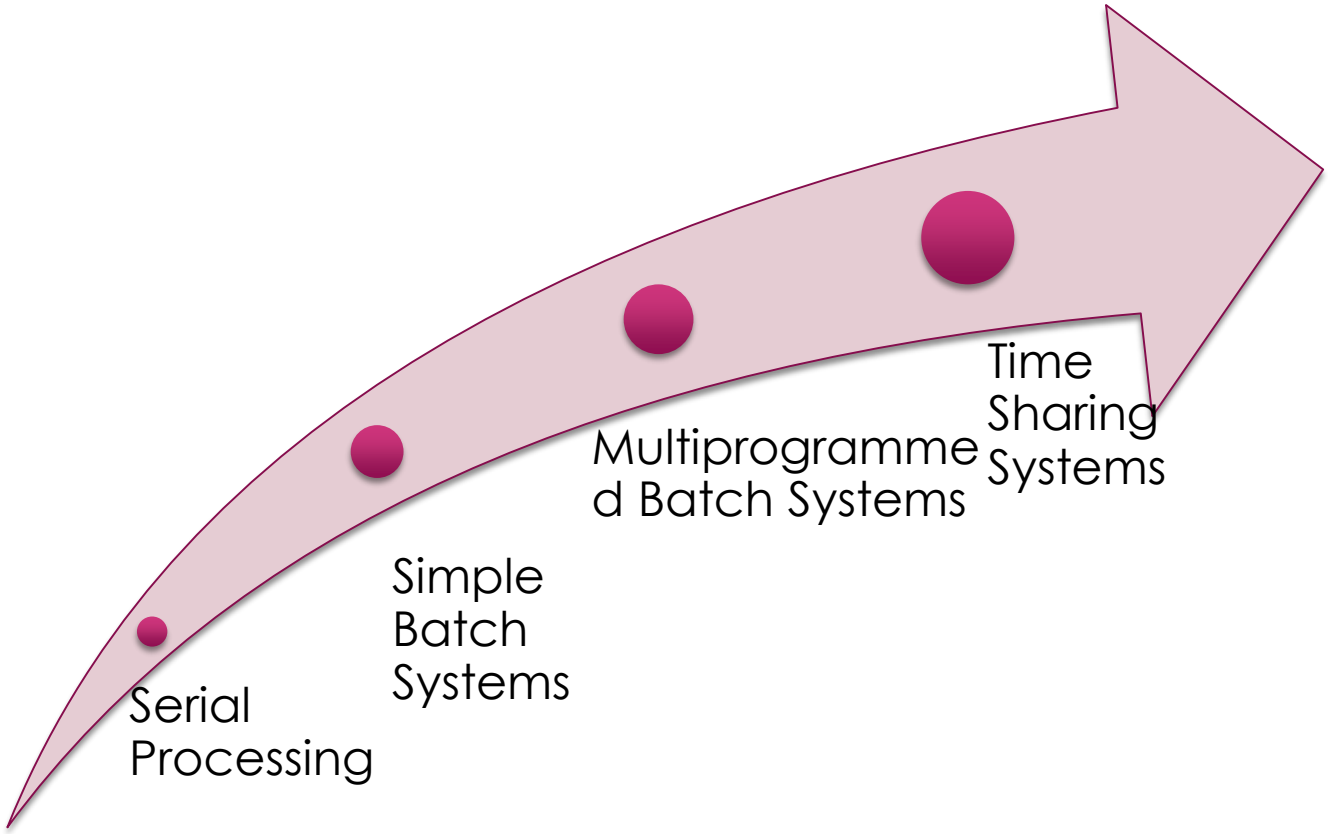
User/Computer Interface (2)

- Services Provided by the Operating System
  - Program development
  - Program execution
  - Access I/O devices
  - Controlled access to files
  - System access
  - Error detection and response
  - Accounting

Resource Manager

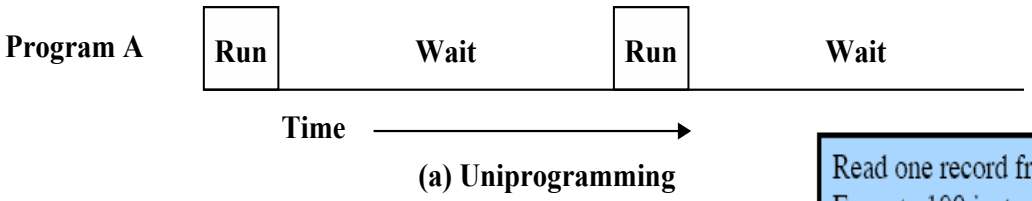
- OS is responsible for controlling the use of a computer's resources
  - I/O
  - Main memory
  - Secondary memory
  - Processor execution time
- Kernel
  - Portion of operating system that is in main memory
  - Contains most frequently used functions

Evolution of Operating Systems



Uniprogramming

- Processor must wait for I/O instruction to complete before preceding



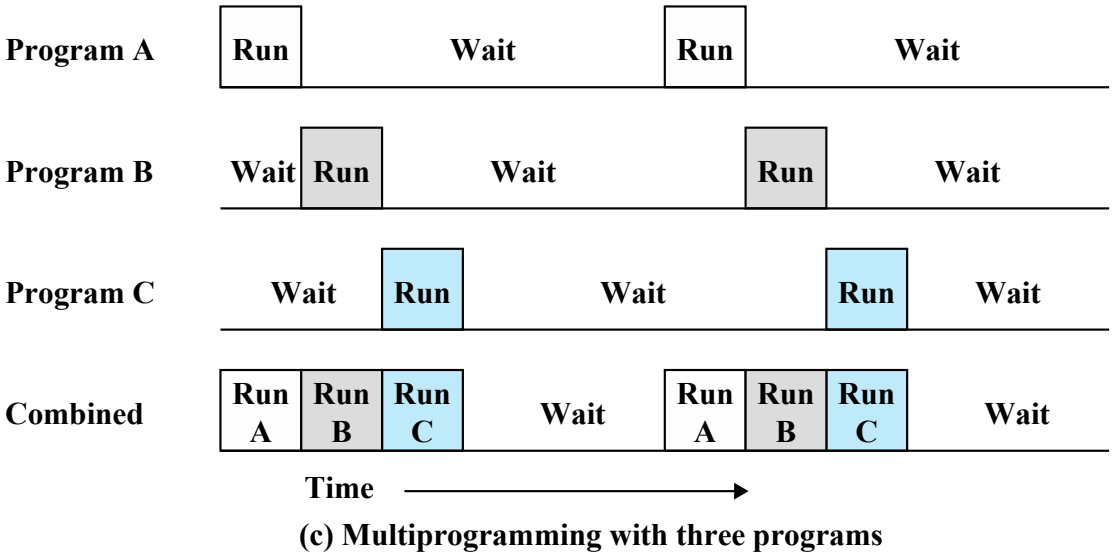
Read one record from file	15 μs
Execute 100 instructions	1 μs
Write one record to file	15 μs
TOTAL	31 μs

Percent CPU Utilization $= \frac{1}{31} = 0.032 = 3.2\%$

Figure 2.4 System Utilization Example

Multiprogramming

- When one job needs to wait for I/O, the processor can switch to the other job (multitasking)



(c) Multiprogramming with three programs

Time Sharing

- Using multiprogramming to handle multiple interactive jobs
- Processor's time is shared among multiple users
- Multiple users simultaneously access the system through terminals

	Batch Multiprogramming	Time Sharing
Principal objective	Maximize processor use	Minimize response time
Source of directives to operating system	Job control language commands provided with the job	Commands entered at the terminal

Major Achievements

- Processes
- memory management
- information protection and security
- scheduling and resource management

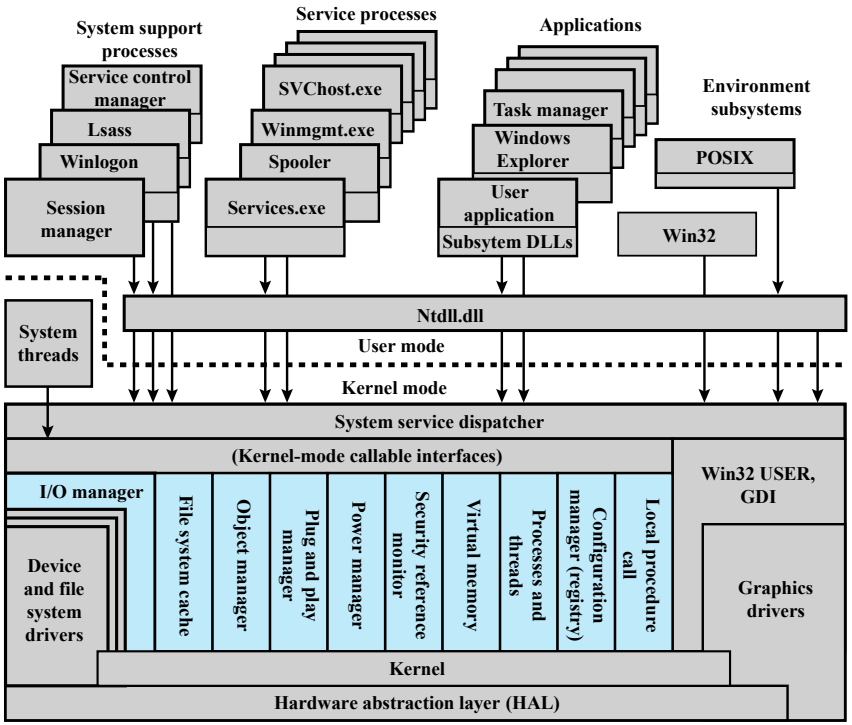
Modern Operating Systems

- Microkernel architecture
  - Assigns only a few essential function to the kernel and other OS services are provided by processes that run in user mode
- Multithreading
  - A technique in which a process, executing an application, is divided into threads that can run concurrently

Modern Operating Systems (2)

- Symmetric multiprocessing (SMP)
  - OS must provide tools and functions to exploit the parallelism of an SMP system
  - SMP → Multicore → Many-core
- Distributed operating systems
  - Provides the illusion of a single main memory space and a single secondary memory space, plus other unified access facilities, such as a distributed file system
- Object-oriented design

Windows Architecture



Lsass = local security authentication server
POSIX = portable operating system interface
GDI = graphics device interface
DLL = dynamic link libraries

Colored area indicates Executive

Figure 2.14 Windows Architecture

UNIX Architecture

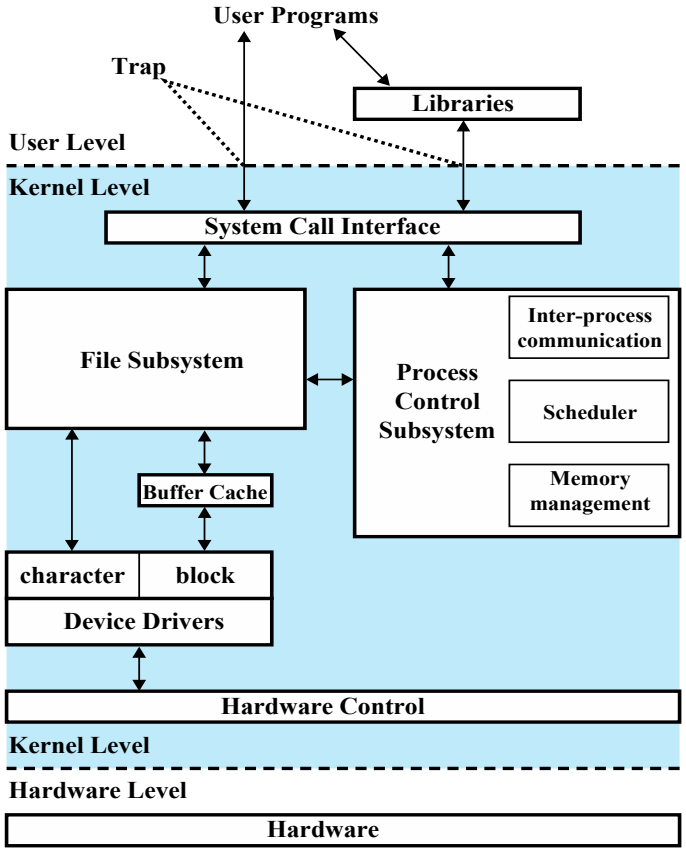


Figure 2.16 Traditional UNIX Kernel

Modern UNIX Architecture

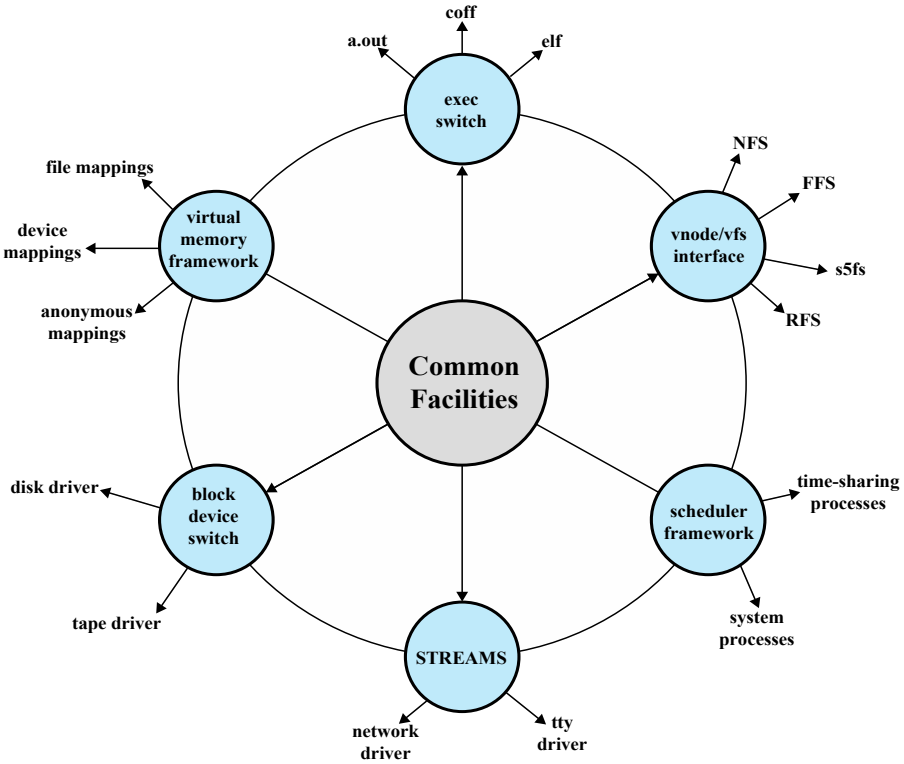


Figure 2.17 Modern UNIX Kernel [VAHA96]

Linux Kernel Components

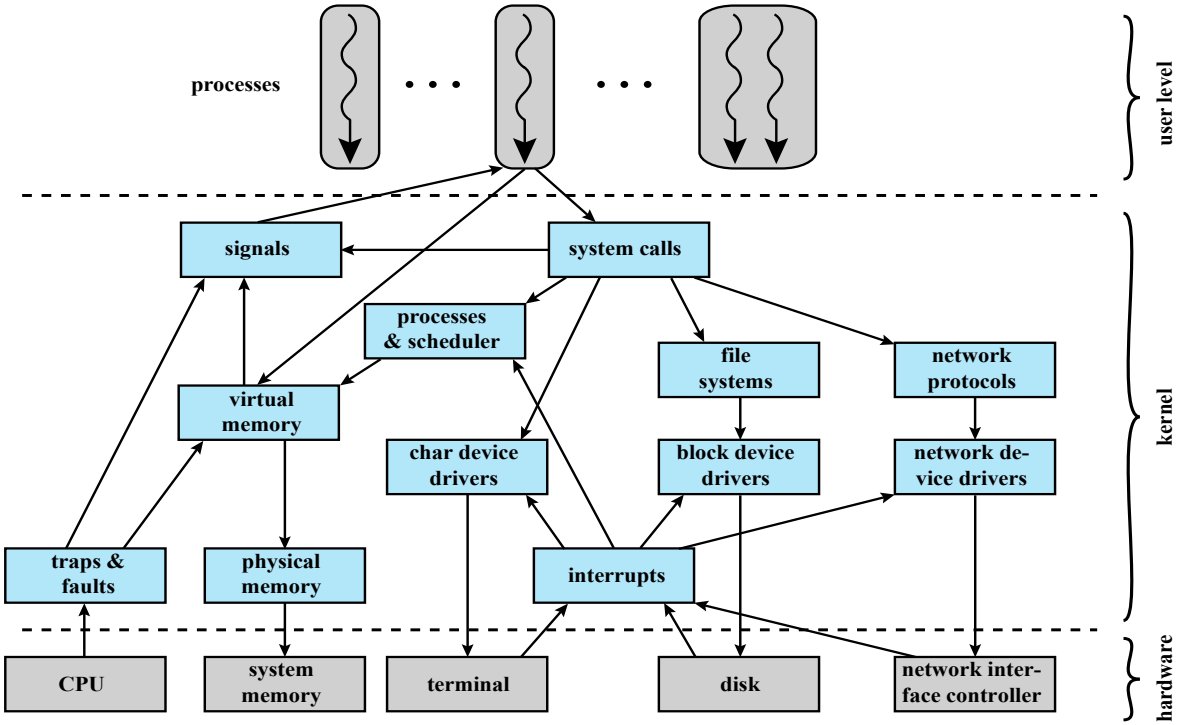


Figure 2.19 Linux Kernel Components

Android System Architecture

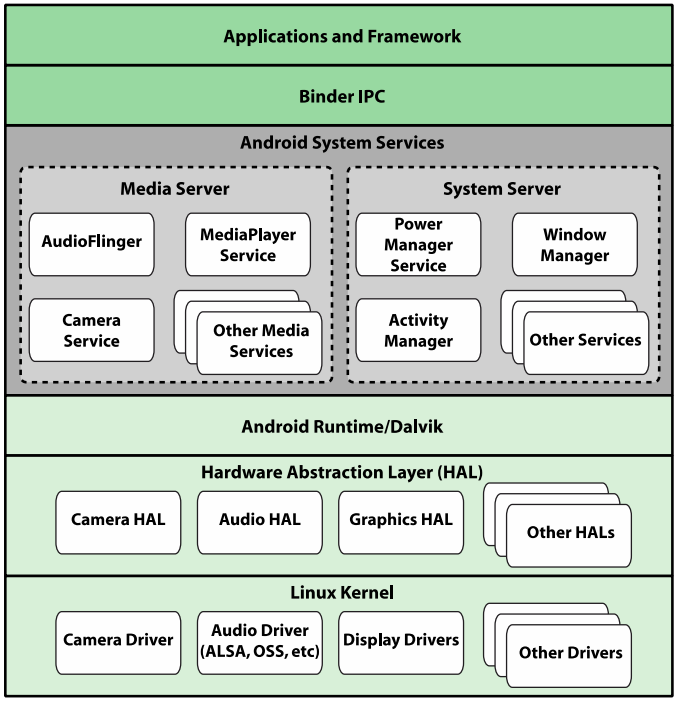


Figure 2.21 Android System Architecture