

Introduction to Operating Systems

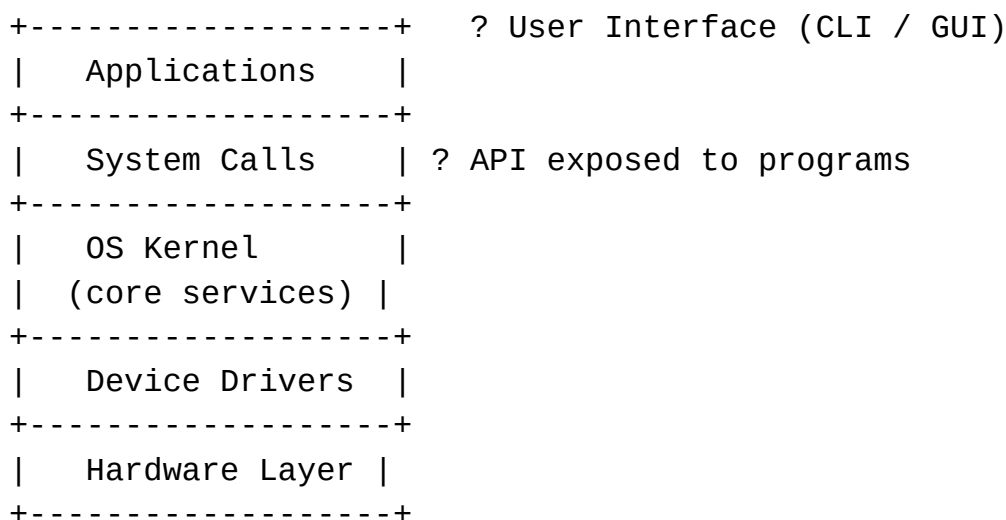
Introduction to Operating Systems

Key concepts, functions, and classifications – concise, easy-to-read notes

1. What Is an Operating System (OS)?

Aspect	Description
Definition	System software that manages hardware resources and provides services for application programs.
Goal	Hide hardware complexity, enable multiple programs to run concurrently, and give users a convenient interface.
Core Responsibilities	• Process management • Memory management • File system management • Device I/O control • Security & protection • User interface (CLI/GUI)

2. Basic OS Architecture



- **Kernel** – central part; runs in privileged mode, handles low-level tasks.
- **User mode** – where applications and most OS services run; cannot directly access hardware.

Introduction to Operating Systems

3. Major Functions

Function	What It Does	Typical Data Structures
Process Management	Creates, schedules, terminates processes; handles inter-process communication (IPC).	Process Control Block (PCB), ready/blocked queues
Memory Management	Allocates/reclaims RAM, implements virtual memory, paging, segmentation.	Page tables, frame allocation bitmap
File System	Organizes data on storage devices, provides hierarchical namespace, handles permissions.	Inodes, directory trees, FAT, NTFS, ext4 structures
I/O Management	Abstracts devices via drivers; buffers data, handles interrupts.	Device control blocks, request queues
Security & Protection	Enforces authentication, access control, isolation between processes.	Access control lists (ACLs), capability tables
Networking (optional in core)	Provides sockets, protocol stacks, routing.	Socket descriptors, routing tables

4. Process vs. Thread

	Process	Thread
Definition	Independent execution entity with its own address space.	Lightweight unit of execution sharing a process's address space.
Overhead	Higher (context switch includes memory map).	Lower (only register set & stack).
Communication	IPC mechanisms (pipes, message queues, shared memory).	Implicit via shared memory within the same process.
Use Cases	Isolated tasks, different applications.	Parallelism within a single application (e.g., web server workers).

5. Types of Operating Systems

Category	Typical Environments	Key Characteristics
----------	----------------------	---------------------

Introduction to Operating Systems

Batch OS	Early mainframes	Jobs queued, no interactive user.
Time-Sharing (Multitasking) OS	Mainframes, servers	CPU time sliced among users; interactive.
Real-Time OS (RTOS)	Embedded controllers, aerospace	Predictable response time; hard vs. soft deadlines.
Distributed OS	Clustered systems, cloud	Resources spread across networked nodes; appears as a single system.
Mobile OS	Smartphones, tablets	Power-aware, touch UI, sandboxed apps.
Desktop/Server OS	PCs	