Betriebssysteme

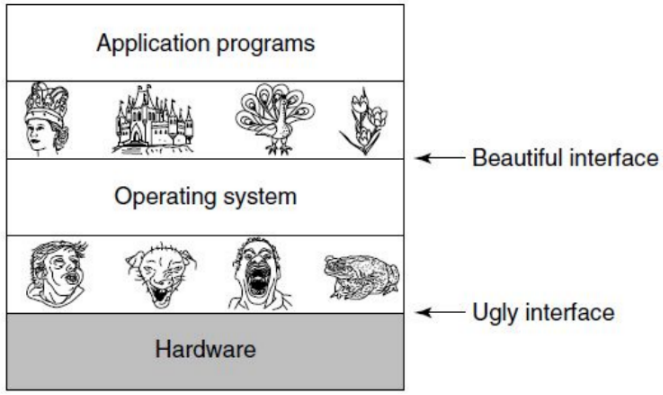
Jil Zerndt
FS 2025

Introduction to Operating Systems

Content:
- Basic Principles of Operating Systems
- Computer Hardware Review

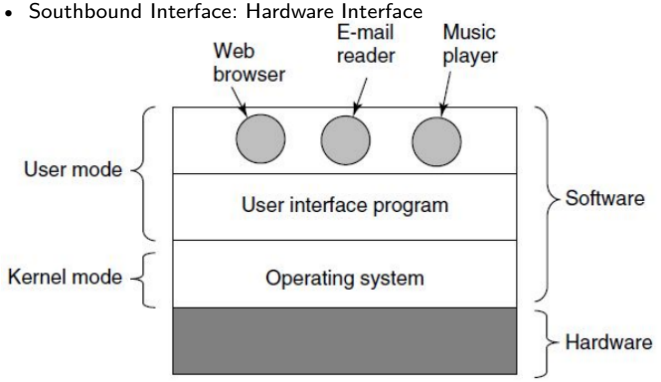
Basic Principles of Operating Systems

Operating System (OS) as an extended machine
- Software that manages computer hardware
- Provides services for programs
- Acts as intermediary between user and hardware

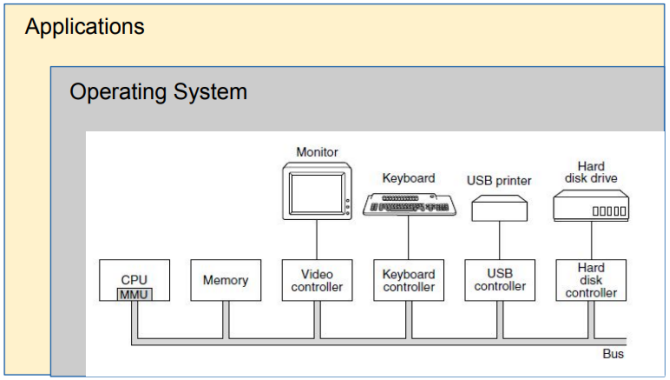


Hardware is very complicated: The job of the operating system is to create good abstractions and manage the abstract objects thus created.

Operating System (OS)
- User Mode vs Kernel Mode
- Northbound Interface: User Interface
- Southbound Interface: Hardware Interface



OS as a Resource Manager
- Process Management
- Memory Management
- File System Management
- Device Management
- Security



The OS controls resource usage, grants resource request, accounts for usage and mediates conflicting requests.

Computer Hardware Review

CPU Central Processing Unit

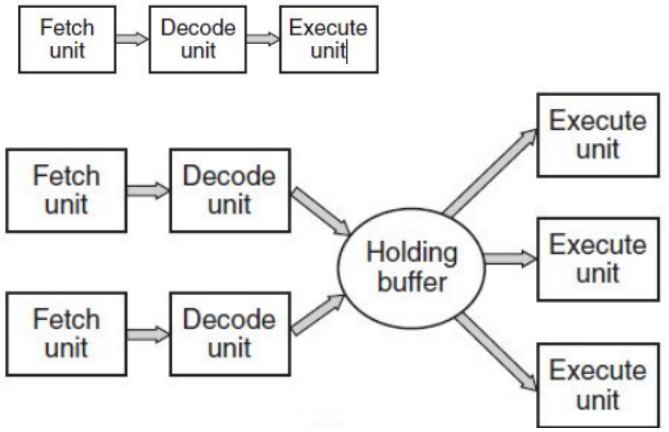
- Basic cycle: Fetch, Decode, Execute
- CPUs feature some registers to hold key variables and temporary results
- Special registers for internal use: Program Counter (PC), Stack Pointer (SP), Program Status Word (PSW), etc.

CPUs and their Instruction Sets are architecture-specific:
- ARM, RISC, X86, etc.
- Instructions are classified along Execution Privileges, enforced by CPU:
  - Intel: User Mode (limited set of instructions) vs Kernel Mode (full set of instructions)
  - ARM: UnPrivileged Mode vs Privileged Mode

CPU cycles

Basic cycle of every CPU:
- Fetches the first instructions from memory into registers
- Decodes instructions (determining type and operands)
- Executes instructions
- fetch, decode, execute, ...

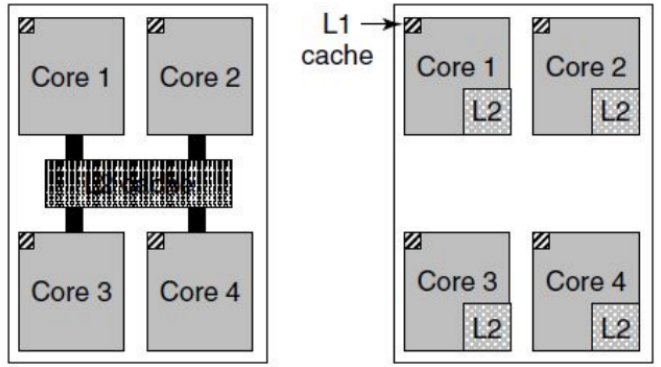
CPUs have multiple cores, each having multiple execution units and parallel pipelines:



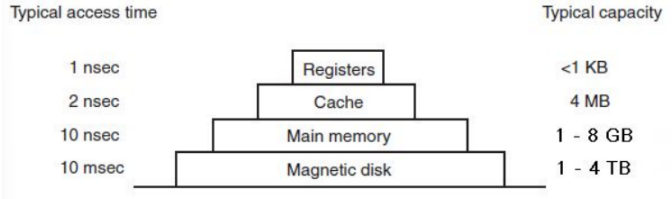
CPU Caches

CPUS may have multiple levels of caches:
- L1: Small, fast, close to CPU
- L2: Larger, slower, further away
- L3: Even larger, even slower, even further away
- Caches are used to store frequently accessed data and instructions

Example: Quad-core chip with a shared L2 cache and a quad-core chip with separate L2 caches for each core.



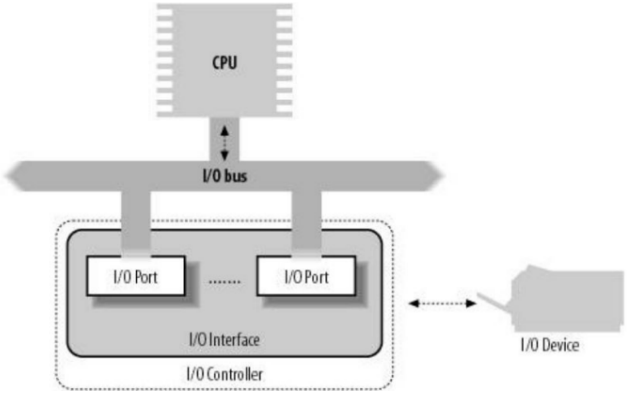
Memory The memory system is constructed as a hierarchy of layers, according to access time:



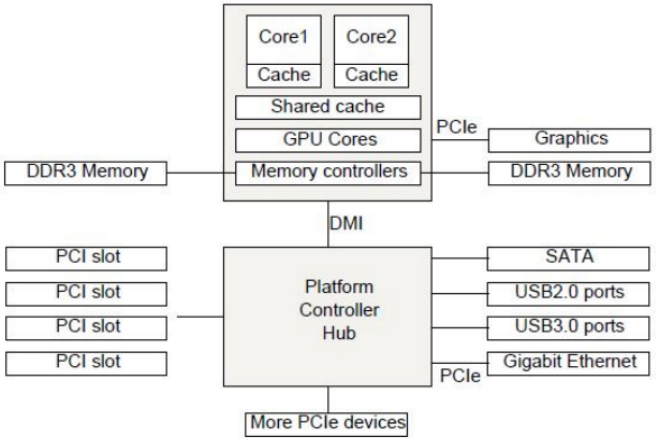
Input/Output Huge amount of I/O Devices, Hard Drives, Network Interfaces, Serial Ports, Keyboard, Mouse, Graphics, Cameras, etc

Devices connected with CPU via a Bus System (SW), I/O Interfaces (SW) and I/O Controllers (HW)

Software that talks to the I/O Controller (Commands/Responses), called Device Driver Runs in Kernel or User Mode. Built-into Kernel or modular and loadable at boot-/run-time



IO and Bus System X86 System with different Bus standards:



Operating System Variants

- Mainframe OS: IBM z/OS, IBM z/VM
- Server OS: Windows Server, Linux, Solaris
- Multiprocessor OS: Windows, Linux, Solaris
- Personal Computer OS: Windows, MacOS, Linux
- Handheld Computer OS: Android, iOS
- Embedded OS: VxWorks, QNX
- Real-Time OS: VxWorks, QNX
- Sensor Node OS: TinyOS, Contiki
- Cloud OS: OpenStack, OpenNebula
- Smart Card OS: JavaCard, MULTOS

Operating Systems vs. Distributions

- Operating System: Kernel, System Libraries, System Utilities
- Distribution: OS + Applications, Tools, Documentation, etc.

Example: Linux Kernel + GNU Tools + X11 + Gnome + Firefox + LibreOffice = Ubuntu evtl. add more info from slides

Operation System Concepts

Basics

Interacting with the OS:

-