Introduction

Madiba Hudson-Quansah

Contents

Chapter 1

Introduction

Definition 1.0.1: Operating System

Central software component that manages all hardware and software, controlling
- Files, devices, section of main memory, and CPU time
- Who can use the system and how the system is used

Definition 1.0.2: Subsystem

A component of an operating system that manages a specific resource.

Operating systems include for essential subsystem managers, where each manager works with other manager, and performs a unique role.

- Memory Manager
- Processor Manager
- Device Manager
- File Manager
- Optionally, Network Manager

Each manager performs the following classes of tasks:

- Monitor the system's resources continuously
- Enforce the system's security and protection mechanisms, i.e. who gets what and how much
- Allocate a resource when appropriate
- Deallocate a resource when it is no longer needed

Definition 1.0.3: User Interface

Allows the user to issue commands to the operating system.

1.1 Main Memory Manager

In charge of main memory (RAM) and read-only memory (ROM). The memory manager is responsible for:

- Checking validity and legality of memory space request
- Reallocating memory to make more useable space available
- Deallocating memory to reclaim it

- Protecting space in main memory occupied by the operating system

1.1.1 ROM

The ROM holds firmware, and is non-volatile.

Definition 1.1.1: Firmware

Determines when and how to load each piece of the operating system after the power is turned on. For example, loading the kernel, BIOS, etc.

1.2 Processor Manager

In charge of allocating the CPU.

- Tracks process status, including which process is currently using the CPU
- Reclaims the CPU when a process if finished or reached the maximum computation time.

Processes are identified by a unique process ID (PID), which is stored in a Process Control Block (PCB).

1.3 Device Manager

In charge of connecting with every available device, responsible for:

- Choosing the most efficient resource allocation method, i.e. scheduling processor.
- Identifying each device uniquely
- Starting device operation when appropriate.
- Monitoring device operation and progress
- Deallocating the device when the operation is complete

1.4 File Manager

In charge of tracking every file in the system, data files, program files, compilers, application programs, etc. is responsible for:

- Enforcing user/program resource access restrictions
- Controlling user/program modification restrictions
- Allocating space for a file on secondary storage
- Retrieving files efficiently

1.5 Network Manager

In change of sharing resources between multiple computers, responsible for:

- Authorizing users to share resources
- Controlling access to shared resources
- Monitoring network performance
- Essentially every aspect of network connectivity

1.6 User Interface

Definition 1.6.1: User Interface

The section of the operating system that allows fir direct interaction with users

There are two primary types of user interfaces:

- Graphical User Interface (GUI)
  - Input from pointing devices
  - Menu options, desktop metaphors, and windows
- Command-Line Interface (CLI)
  - Input from keyboard
  - Commands with options and arguments

1.7 Cooperation Issues

As no single manager con work in isolation, cooperation between managers is essential. Each manager performs its tasks and interacts with other managers as needed. This requires incredible precision for the operating system to work correctly, and becomes even more complex when networking is involved.

1.8 Types of Operating Systems

Operating Systems fall into five categories:

- Batch
- Interactive
- Real-time
- Hybrid
- Embedded

These categories are distinguished by two features:

- Response time
- Method of data entry

1.8.1 Batch Systems

- Jobs entered as a whole and in sequence
- Input relied no punched cards or tape
- Efficiency measured in throughput

1.8.2 Interactive Systems

- Allows multiple jobs
- Faster turnaround than batch systems
- Slower than real-time systems
- Introduced to provide fast turnaround when debugging programs
- Requires complex algorithms as jobs share processing power

1.8.3 Hybrid Systems

- Combination of batch and interactive systems
- Light interactive load
- Accepts and runs batch programs in the background

1.8.4 Real-time Systems

- Used when reliability is critical
- Used in time-critical environments (Spacecraft, Airport traffic control, Fly-by-wire aircraft, etc.)
- Two types of real-time systems

  Hard real-time systems - Risk total system failure if a single predict time deadline is missed

  Soft real-time systems - Suffer performance degradation as a consequence if a deadline is missed

1.8.5 Network Systems

- Special class of software, where users perform tasks using few, if ant, local resources, solely relying on network resources, e.g. cloud computing
- Requires Wireless Networking Capability, which is a standard feature in many computing devices (cell phone, tables, and other handheld web browsers)

1.8.6 Embedded Systems

- Computers placed inside other products (automobiles, digital music players, elevators, pacemakers, etc.)
- Adds features and capabilities
- Performs a specific set of programs/functions
- Embedded systems are non-interchangeable among systems, i.e. they are purpose-built
- Small kernel supporing simple and flexible function capabilities