Linux Lecture 1 PDF

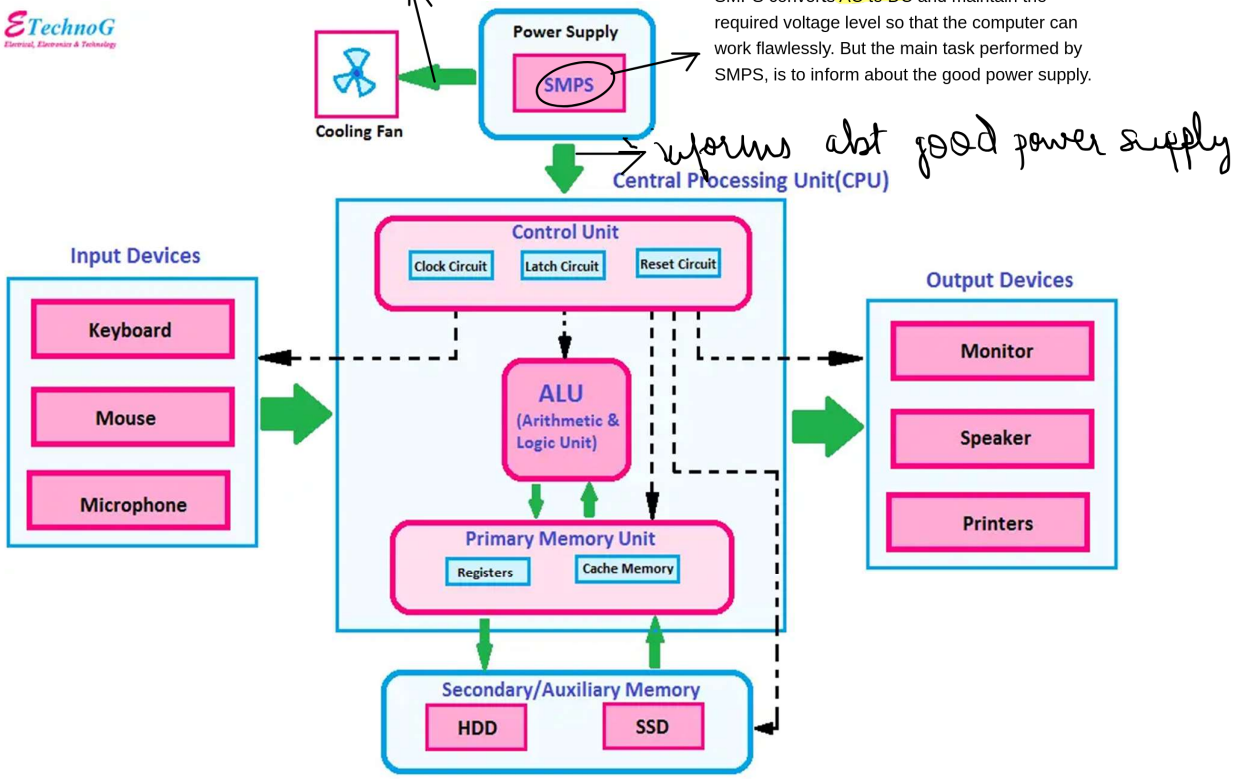
Clock circuit is used to keep track of current time to make precise time measurements

here, SMPS converts AC to DC

SMPS (Switched mode Power Supply)

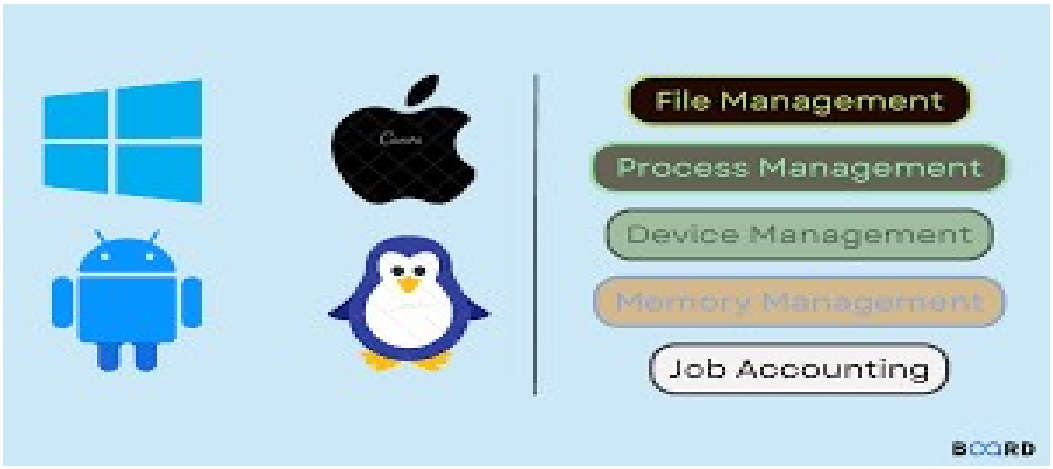
SMPS converts AC to DC and maintain the required voltage level so that the computer can work flawlessly. But the main task performed by SMPS, is to inform about the good power supply.

informs abt good power supply



Computer Block Diagram

Operating System Functions:



Latch circuit is a basic ckt that has a momentary input that turns on the output.

A reset circuit is a pin or circuit that helps a microprocessor reinitialize itself and resume normal operation when it encounters an error condition.

Registers are memory locations that a CPU can directly access.

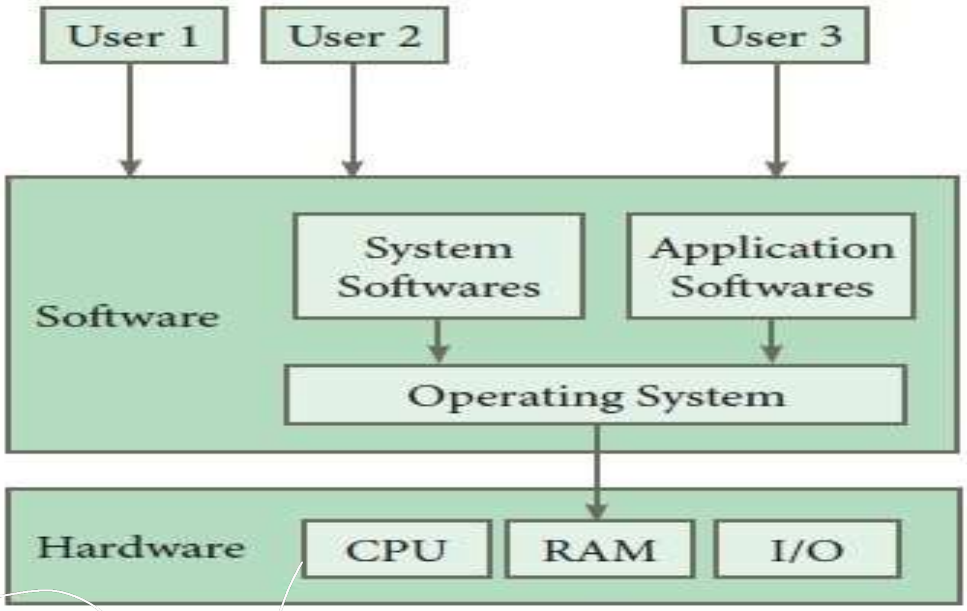
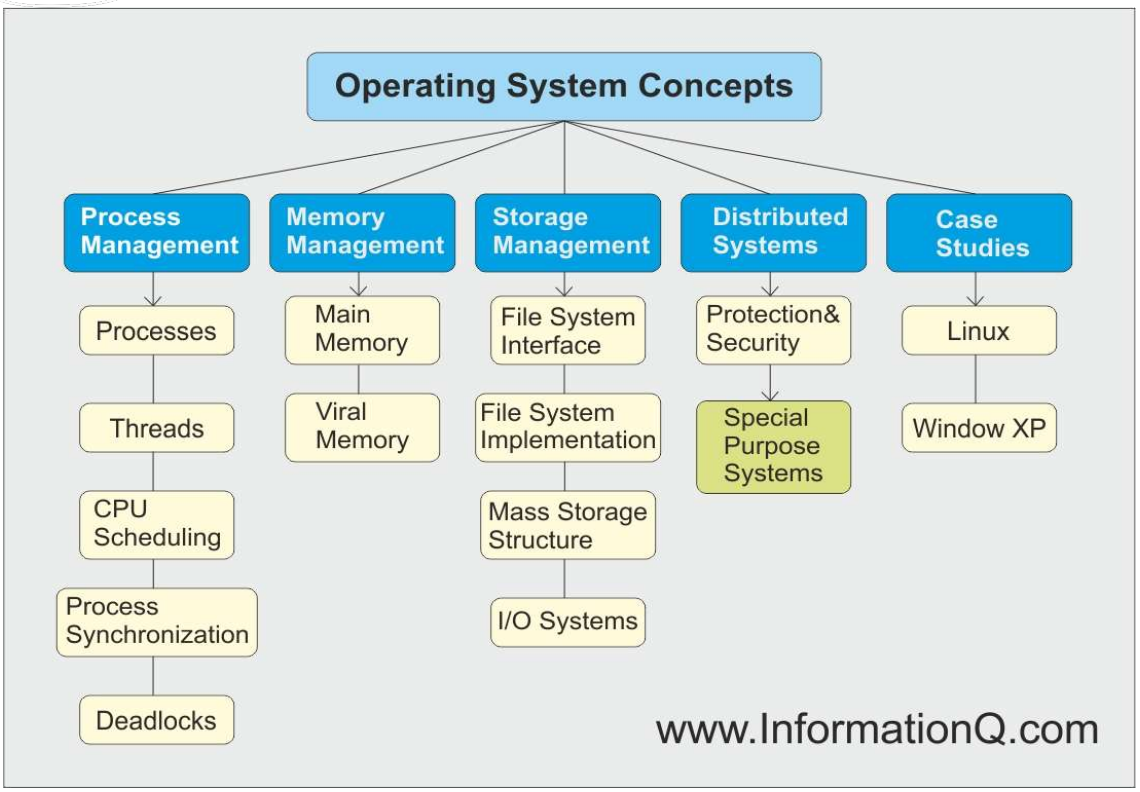


Figure 5.1. Overview of an Operating System



Linux

He developed the Kernel, not OS

History of Linux:

- In 1991, Linus Torvalds, a Finnish computer science student, developed the Linux kernel.

- Linus released the kernel as free software under the GNU General Public License (GPL).

- The kernel's development was open and collaborative, allowing contributors worldwide to enhance and extend it.

- Over time, a vibrant Linux community evolved, resulting in numerous distributions and applications.

Linux Foundation:

- The Linux Foundation is a consortium founded to support and promote Linux and open-source technologies.

- It sponsors the development of the Linux kernel and other important open-source projects.

- The Linux Foundation provides training, certification, and collaboration opportunities for developers and organizations.

- It hosts events, including the Linux Plumbers Conference and the Open Source Summit.

Hardware Requirements:

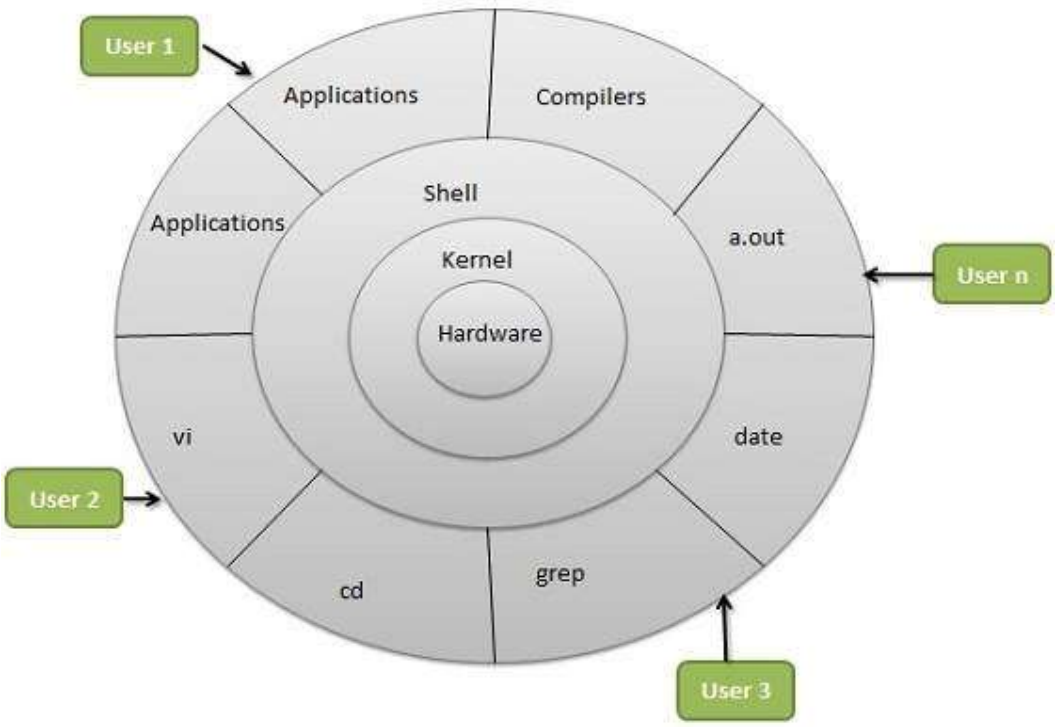
- Hardware requirements for Linux can vary significantly based on the distribution and usage scenario.

- Server installations may require more memory and processing power compared to desktop systems.

- Lightweight distributions, like Puppy Linux, can run on older hardware with minimal resources.

- Embedded systems might have specialized hardware requirements depending on the target platform.

Linux Components:



Linux is a popular operating system that is known for its open-source nature and flexibility. It consists of several key components that work together to provide a fully functional computing environment. Here are the major components of a typical Linux operating system:

1. Linux Kernel: The core of the Linux operating system is the Linux kernel. It is responsible for managing hardware resources, scheduling processes, and handling essential system functions. The kernel is the bridge between the hardware and software.

2. Shell: The shell is the user interface that allows users to interact with the Linux system. Common shells include Bash (Bourne-Again Shell), Zsh, and others. The shell interprets user commands and provides a command-line interface for running programs and managing the system.

3. System Libraries: Linux includes a set of system libraries that provide essential functions and services to applications. These libraries contain reusable code for tasks like file I/O, memory management, and network communication. The GNU C Library (glibc) is a critical system library used in most Linux distributions.

4. File System: Linux supports various file systems, including ext4, XFS, and Btrfs. The file system manages files, directories, and storage devices. It handles file creation, reading, writing, and organization.

5. Device Drivers: Device drivers are essential software components that enable the kernel to communicate with hardware devices, such as graphics cards, network adapters, and printers. Linux supports a wide range of hardware through its extensive collection of device drivers.

6. Process Management: Linux manages processes, which are running programs or tasks. This includes process creation, scheduling, termination, and resource allocation. The kernel ensures efficient utilization of the CPU and memory.

7. Memory Management: Memory management is responsible for allocating and managing system memory, including physical and virtual memory. It ensures that processes can access and use memory resources efficiently.

8. Networking Stack: Linux offers a robust networking stack with support for various protocols, such as TCP/IP. It enables network communication, including network configuration, socket management, and data transmission.

Commands

9. User Space Utilities: Linux includes a wide range of command-line utilities and system tools to perform tasks like file manipulation, process management, and system administration. Some common utilities include ls, cp, mv, ps, and top.

GUIs

10. Graphical User Interface (Optional): Many Linux distributions include a graphical user interface, such as GNOME, KDE, or Xfce, to provide a user-friendly desktop environment. Users can interact with the system using a mouse and graphical applications in addition to the command-line interface.

11. Package Management: Linux distributions often come with package management systems (e.g., APT, YUM, or DNF) to simplify software installation, updates, and removal. Users can easily install and update software packages from repositories maintained by the distribution.

12. Security and Authorization: Linux includes security features, such as user authentication, access control, and firewalls. It offers robust user and group management to control access to system resources.

13. Error Handling: Linux provides mechanisms for logging and handling system errors and exceptions to maintain system stability.

14. Printing Services: Linux supports printing services that enable users to configure and manage printers and print documents.

15. Desktop Environment (Optional): In addition to the graphical user interface, many Linux distributions offer complete desktop environments that provide a consistent look and feel and additional features for desktop users.

The specific components and their configurations can vary between different Linux distributions. Examples of popular Linux distributions include Ubuntu, Fedora, CentOS, Debian, and many others, each with its own package management system, default software selection, and system administration tools.

Linux Distributions (Distros):

- Linux distributions come in various flavors, each tailored for specific purposes and preferences.
- Ubuntu: User-friendly, suitable for desktops and servers.
- Debian: Known for stability and a large package repository.
- CentOS: Designed for enterprise use and known for long-term support.
- Fedora: Features the latest software and technologies.
- Arch Linux: Offers a minimalist, rolling-release system for advanced users.
- Red Hat Enterprise Linux (RHEL): Enterprise-grade distribution with paid support.

Features of Linux:

- Open Source: Linux is open-source, allowing users to access, modify, and distribute the source code.
- Stability: It's known for its reliability and long uptimes, making it suitable for critical tasks.
- Security: Strong security features, including user permissions, SELinux, and a robust firewall system.
- Multitasking: Linux handles multitasking efficiently, supporting multiple processes and users concurrently.
- Networking: Excellent networking capabilities, making it the foundation for many network devices and servers.
- Flexibility: It's highly customizable and can be tailored to specific needs and preferences.

Linux is open source, stable, secure & flexible with multitasking & networking capabilities.

Choosing a Suitable Linux Distribution:

- Use Case: Determine if you need a distribution for a desktop, server, embedded system, or a specific purpose like ethical hacking or multimedia production.

- Package Management: Choose between distributions with different package management systems (e.g., Debian-based, RPM-based, source-based).

- Community or Enterprise: Decide if you require community-driven support or enterprise-level support and maintenance.

- User-Friendliness: Consider the ease of use for both you and the intended end-users. Some distributions are more user-friendly than others.

- Hardware Requirements: Ensure the distribution matches your hardware capabilities, whether you're running it on a high-end server or an old laptop.

- Release Cycle: Some distributions have fixed release cycles, while others follow rolling releases. Choose based on your preference and need for stability.

- Support and Documentation: Assess the availability of community support, official documentation, and forums for troubleshooting and assistance.

- Software Ecosystem: Evaluate the software available in the distribution's repositories and ensure it meets your requirements.

- Security: Consider the distribution's security features, patching frequency, and overall reputation for security.

- Personal Preference: Finally, personal preference plays a role in selecting a distribution. Experiment with a few to find one that suits your workflow and preferences.

In conclusion, Linux is a diverse and robust operating system with a rich history, a wide range of distributions, and a multitude of features. To choose the right distribution, assess your specific needs and preferences, and consider factors such as use case, package management, support, hardware, release cycle, and personal comfort with the system.

Reference:

https://www.geeksforgeeks.org/linux-tutorial/

https://www.javatpoint.com/linux-tutorial