**Linux operating system**

**What is Linux OS:**

Open source operating system

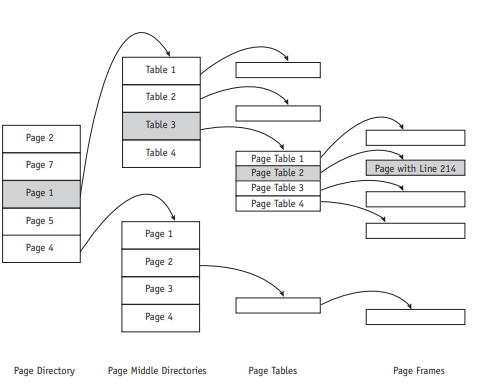
POSIX-compliant

CLI: Command Line Interface

**Memory management**

[**https://tldp.org/LDP/tlk/mm/memory.html**](https://tldp.org/LDP/tlk/mm/memory.html)

* **Memory Allocation in Linux**:
  + Linux reserves 1GB of high-order memory for the kernel and assigns 3GB to running processes.
  + The 3GB address space is allocated for process code, data, shared libraries, and process stack.
* **Memory Protection**:
  + Protection mechanisms are based on data types in each region of a process's address space.
  + The kernel adjusts protection settings if a process attempts unauthorized access.
* **Page Management**:
  + Linux loads requested memory pages into memory for processes.
  + Least Recently Used (LRU) algorithm is used to release pages when additional memory is needed.
* **Page Tables**:
  + Linux uses page tables to manage free and allocated memory pages.
  + Some chip architectures utilize memory access through segments.
* **Virtual Memory Management**:
  + Virtual memory is organized with a multi-level table hierarchy supporting 64-bit and 32-bit architectures.
  + Three levels of tables (Main, Middle, and Page Table Directories) are used to locate requested instruction or data within a job.



**Processor management**

[**https://tldp.org/LDP/tlk/kernel/processes.html#:~:text=Processes%20carry%20out%20tasks%20within,a%20computer%20program%20in%20action.&text=Linux%20is%20a%20multiprocessing%20operating%20system**](https://tldp.org/LDP/tlk/kernel/processes.html#:~:text=Processes%20carry%20out%20tasks%20within,a%20computer%20program%20in%20action.&text=Linux%20is%20a%20multiprocessing%20operating%20system)**.**

1. **Table of Contents Structure:**

Each process is referenced by a descriptor, which contains about 70 data fields describing the process's attributes along with information needed to manage it.

The kernel dynamically allocates these descriptors when processes start running.

1. **Data Type:**

All process descriptions are organized into a doubly linked list. This allows the operating system to move quickly between processes and perform operations such as inserting, deleting, or updating them efficiently.

1. **Additional List:**

Besides the main list containing all processes, there is another doubly linked list that stores "ready" or "running" processes.

Each element in this list has additional fields:

"next run": Points to the process that will be run next.

"previously run" (previously run): Points to a previously run process.

1. **List Management:**

To manage and update the list of process descriptions, the operating system uses a number of macro instructions. These macro instructions can be used to:

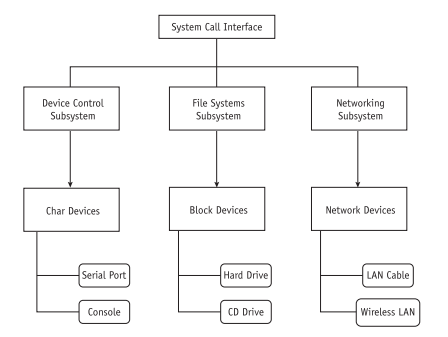
Add a new process to the list.

Remove a completed process from the list.

Update the status of a process (such as from "ready" to "running" or vice versa).

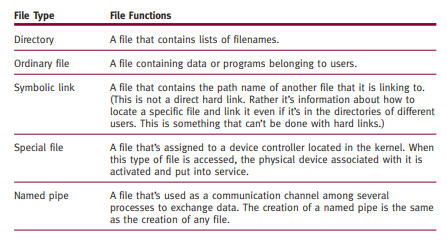
**Device management**

1. **Device Independence**:
   * Linux is device independent, enhancing its portability across different systems.
2. **Device Drivers**:
   * Device drivers manage data transmission between main memory and peripheral units.
   * Each device is assigned a name and descriptors stored in the device directory.
   * Devices are identified by a minor device number (used to access multiple physical devices) and a major device number (used to access device driver code).
3. **Device Classifications**:
   * Linux assigns each device a minor and major device number.
   * The major device number indexes an array of entry points in the Configuration Table, connecting system code to device drivers.
   * The Configuration Table facilitates the quick creation of new device drivers for different system configurations.
4. **Device Drivers in Linux**:
   * Linux supports standard device classes from UNIX and allows for the creation of new device classes to support new technology.
   * Notably, Linux can accept new device drivers while the system is running, without the need for rebooting, enabling administrators to load and test new drivers without disrupting system operation.
5. **Device Classes**:
   * Linux supports three standard classes of devices:
     + **Character Devices**: Handle data one character at a time (e.g., keyboards, mice).
     + **Block Devices**: Manage data in blocks or chunks (e.g., hard drives, SSDs).
     + **Network Devices**: Facilitate communication over a network (e.g., Ethernet adapters).

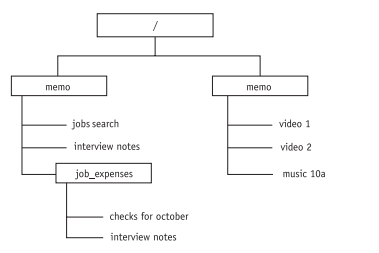


**File management**

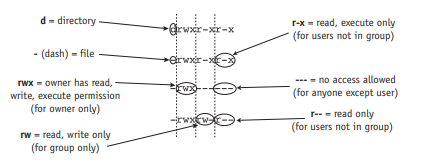
* **Data Structures:**
* All Linux files are organized in directories that are connected to each other in a treelike structure. Linux specifies five types of files:



* **Filename Conventions:**
* Filenames are case sensitive so Linux recognizes both uppercase and lowercase letters in filenames. For example, each of the following filenames are recognizable as four different files housed in a single directory: FIREWALL, firewall, FireWall, and fireWALL.



* **Directory Listings :**
* While directory listings can be created from Terminal mode using typed commands (**ls or ls -l**), many Linux users find that the easiest way to list files in directories is from the GUI desktop.



**User interface**

1. **Command-Driven Interfaces**:
   * Commands are interpreted and executed by the shell, such as the Bash shell.
   * The shell serves as both an interactive command interpreter and a coordinator of system programs.
2. **Graphical User Interfaces (GUI)**:
   * Linux distributions often include multiple GUI options, allowing users to choose the interface that best suits their needs.
   * Users can select from various GUI environments provided with the Linux operating system.
3. **System Monitor**:
   * The System Monitor window provides information about system status, including CPU, memory, and network usage.
   * It displays immediate history and usage statistics, along with information about supported file systems and running processes.
4. **Service Settings**:
   * Linux distributions offer administrators a range of services for system management.
   * Service options may vary between distributions, allowing administrators to configure and manage the system according to specific requirements.
5. **System Logs**:
   * System logs, found in the /var/log directory, provide detailed records of system activity.
   * Logs are essential for diagnosing issues such as system malfunctions, firewall failures, and disabled devices.
6. **Keyboard Shortcuts**:
   * Linux supports keyboard shortcuts to enhance user efficiency and productivity.
   * Many shortcuts are similar to those used in Windows operating systems, simplifying the transition between platforms.
   * For example, CTRL-V is commonly used for the PASTE command across Linux, UNIX, and Windows environments.