

# Overview of Computer Workshop

## Unit-3, Lecture – 1

*By*

Dr. Priyambada Subudhi

Assistant Professor

IIIT Sri City

# What is an Operation System?

- An **operating system** is the low-level software that schedules tasks, allocates storage, and handles the interfaces to peripheral hardware, such as printers, disk drives, the screen, keyboard, and mouse.
- An operating system has two main parts: the **kernel** and the **system programs**.
- The **kernel** allocates machine resources—including **memory** and **disk space**, and **CPU cycles** to all other programs that run on the computer.
- The system programs include device drivers, libraries, utility programs, shells (command interpreters), configuration scripts and files etc.
- Common contemporary OSs include **Microsoft Windows**, **Mac OS X**, and **Linux**.

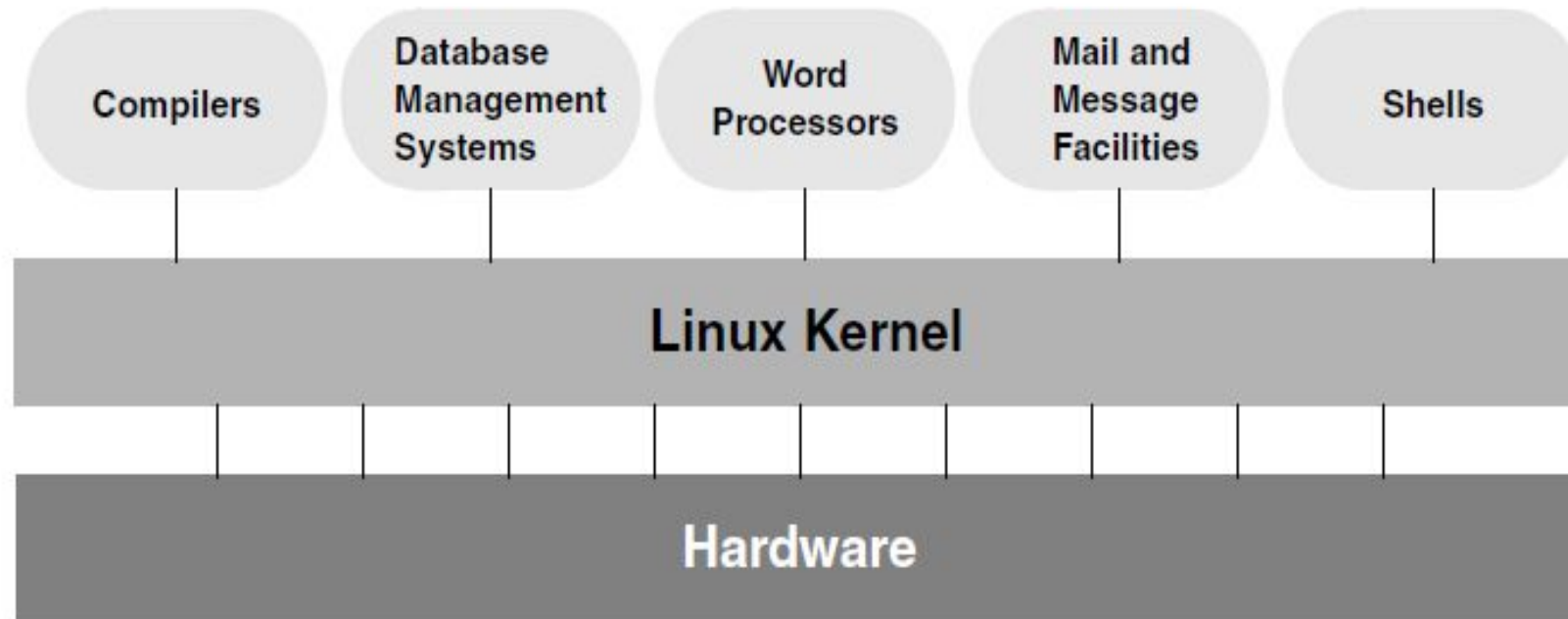
# Linux

- Linux is a member of the large family of **Unix-like OS**.
- The *Linux kernel* was developed by Finnish undergraduate student **Linus Torvalds**, who used the Internet to make the source code immediately available to others for free.
- Torvalds released Linux version 0.01 in September 1991.
- Programmers around the world were quick to extend the kernel and develop other tools, adding functionality to match that already found in both BSD UNIX and System V UNIX (SVR4) as well as new functionality.
- The name *Linux* is a combination of *Linus* and *UNIX*.

# Features of Linux

- Linux is an **open-source** operating system: its source code is open and available to anyone to study.
- It is a **control program** for computers like any other OS.
- It has family of utility programs and a set of tools that allow users to connect and use these utilities to build systems and applications.
- Linux has a kernel programming interface.
- Linux can support many users simultaneously i.e. it is **multi-user OS**.
- Linux is a fully protected **multitasking** operating system, allowing each user to run more than one job at a time.
- Linux is portable.
- Linux is predominantly known for its use in servers.
- There are several Linux Distributions, such as: Ubuntu Linux, Red Hat Enterprise Linux, Linux Mint, Debian, Fedora etc.

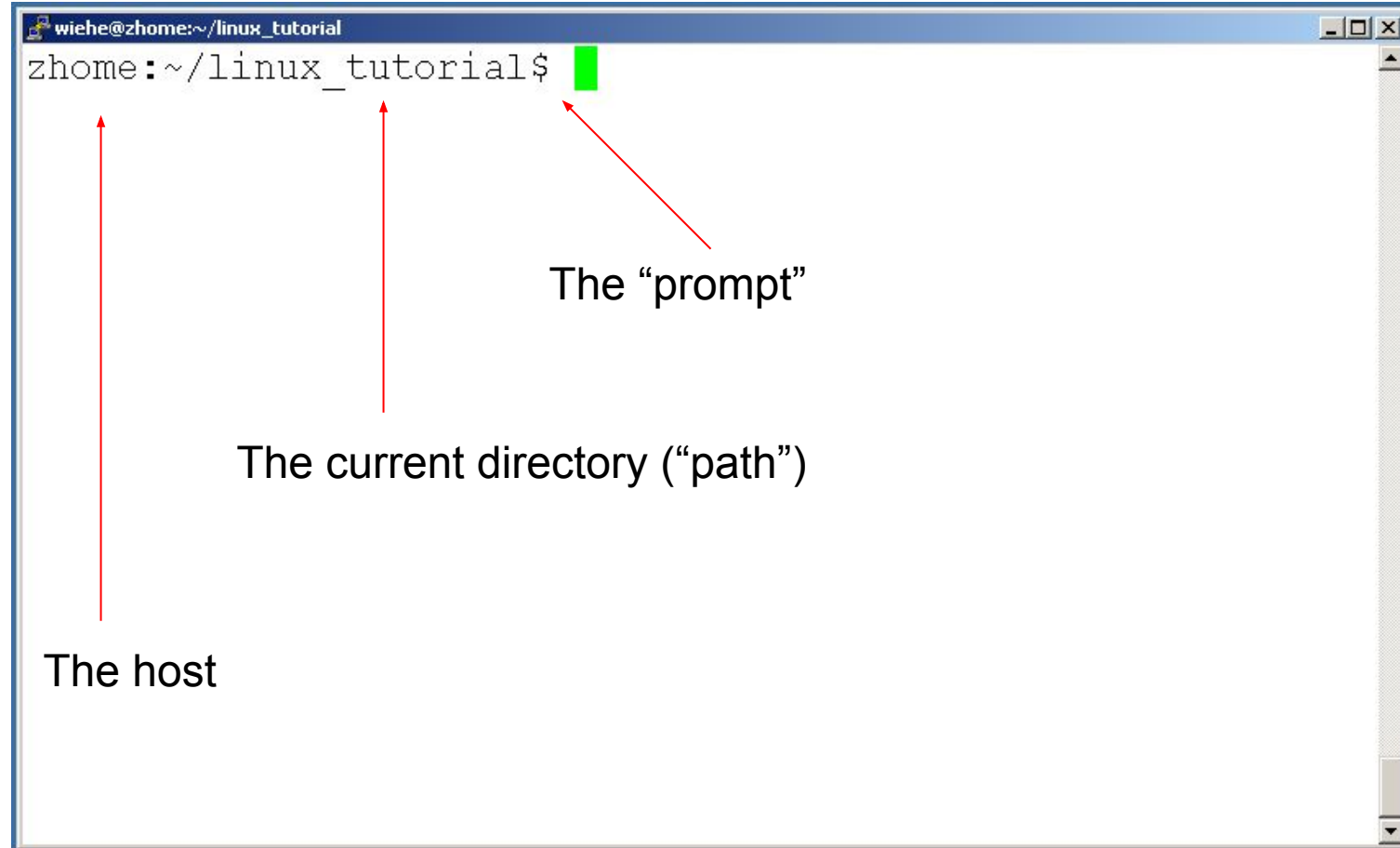
# A layered view of Linux OS



# User interface in Linux

- There are two different ways in which the user can interact with the Linux OS
  - Through a **graphical user interface (GUI)** in which the user uses a mouse to manipulate windows
  - Through the **command line interface (CLI)**, in which the user types commands at a prompt.
- **CLI**: It is a non-graphical, text-based interface to the computer system, where the user types in a command and the computer then successfully executes it.
- The Terminal is the platform or the IDE that provides the command line interface (CLI) environment to the user.
- The CLI terminal accepts the commands that the user types and passes to a shell or command interpreter.
- The shell interprets the command and calls the program you want.
- If the output is produced by the specific command, then this text is displayed in the terminal.

# A typical Linux terminal



# Basic commands in Linux

- Commands are case sensitive. Lower case alphabets are used in the commands without using any unnecessary characters.

- **date** : It is used to display date, time, time zone, and more.

**Syntax:** **date**

- **cal** : It is used to display the current month's calendar with the current date highlighted.

**Syntax:** **cal**

- **who**: It gives the information about the users logged on to the system. The first column displays the user name, second column displays the system's name used for connection and the rest displays the details when the users logged in.

**Syntax:** **who**



- **pwd** : It is used to display the location of the current working directory.

**Syntax:** **pwd**

- **man** :man stands for manual which is a reference book of a Linux operating system. It is similar to HELP file found in popular software.

**Syntax:** **man command\_name**

- **clear**: This command clears all the clutter on the terminal and gives you a clean window to work on, just like when you launch the terminal.

**Syntax:** **clear**

- **echo**: It is used to display a line of **string/text** that is passed as the argument.

**Syntax:** **echo [options] [string]**

Using '>>' with 'echo' command appends a line to a file.

**Syntax:** **echo [string] >>filename**

# Directory Related commands

- **mkdir** : It is used to create a directory in your present working directory.

**Syntax:** `mkdir dir_name`

- **rmdir** : It is used to remove a directory (Ensure that the directory is empty).

**Syntax:** `rmdir dir_name`

- **cd**: It is used to change the current directory.

**Syntax:** `cd dir_name`

## Options:

- **cd** change directory to the home directory.
- **cd ..** move to the parent directory of current directory one level up from the current directory.

# File related commands

- **touch** : It is used to create empty files.

**Syntax:** `touch file_name` or `touch file_name1`

`file_name2`

- **cat** : It is used to create a new file, displaying the content of the file or adding new content to the file.

**Syntax:** `cat > file_name` (for creating new file named `file_name`)

**Options:**

- `cat filename` (display content of the file)
- `cat filename1 filename2 filename3` (display content of multiple files)
- `cat filename1 > filename2` (Create a newfile `filename2` and redirect the content of `filename1` to `filename2`. If `filename2` is an existing file then its content is overwritten by content of `filename1`)
- `cat filename1 >> filename2` (Append the content of `filename1` to the end of content of `filename2`)

- **tac**: It is the reverse of the cat command. It displays the contents of the file in reverse order.

**Syntax:** **tac filename**

- **rm** : It is used to delete a file.

**Syntax:** **rm filename**

- **mv** :It is used to rename a file or to move a file from one directory to other directory.

**Syntax:** **mv oldfilename newfilename**

- **cp**: It is used to copy the content of one file to another file.

**Syntax:** **cp source\_filename destination\_filename**

- **ls**: It makes a list of file names or directory present in the current working directory.

**Syntax:** **ls**

**Option:**

**ls -l** (Long listing of files)

**ls -a** (lists all files including hidden files starting with '.')

**ls -i** (lists files along with the inode number)

**ls -s** (lists files along with their size)

**ls -S** (lists by sorting with file size)