

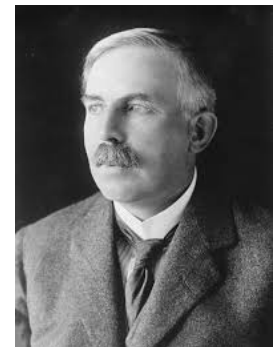
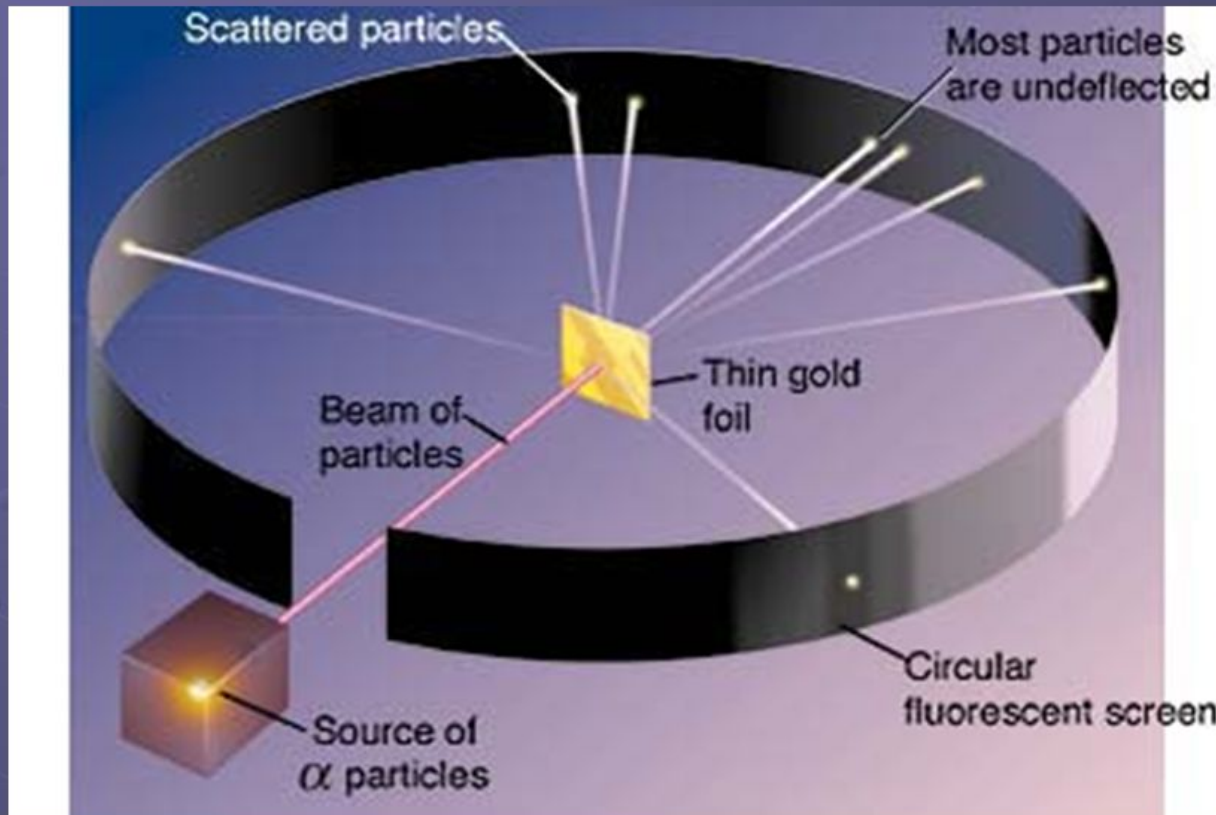
Fundamentals in command line interpreter

S. K. Prasad
Bose Institute, Kolkata

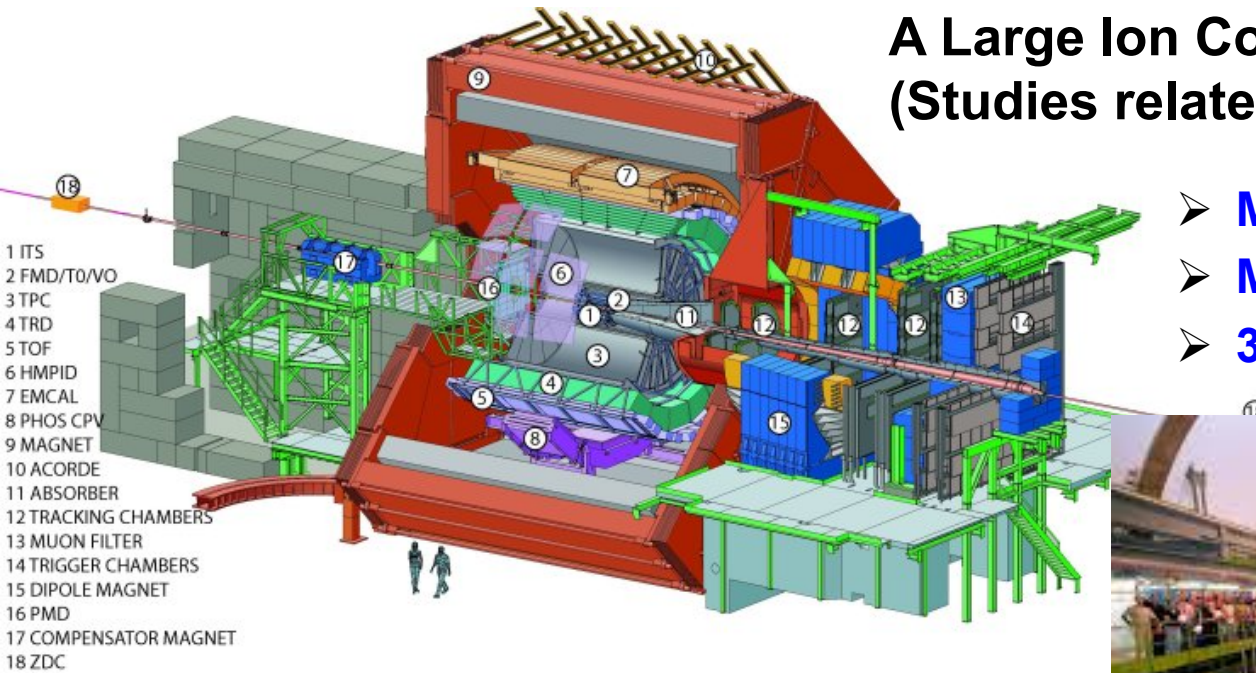
Rutherford experiment and discovery of Nucleus (1908 – 1913)

- by **Hans Geiger** and **Ernest Marsden** under the direction of **Ernest Rutherford** at the Physical Laboratories of the University of Manchester

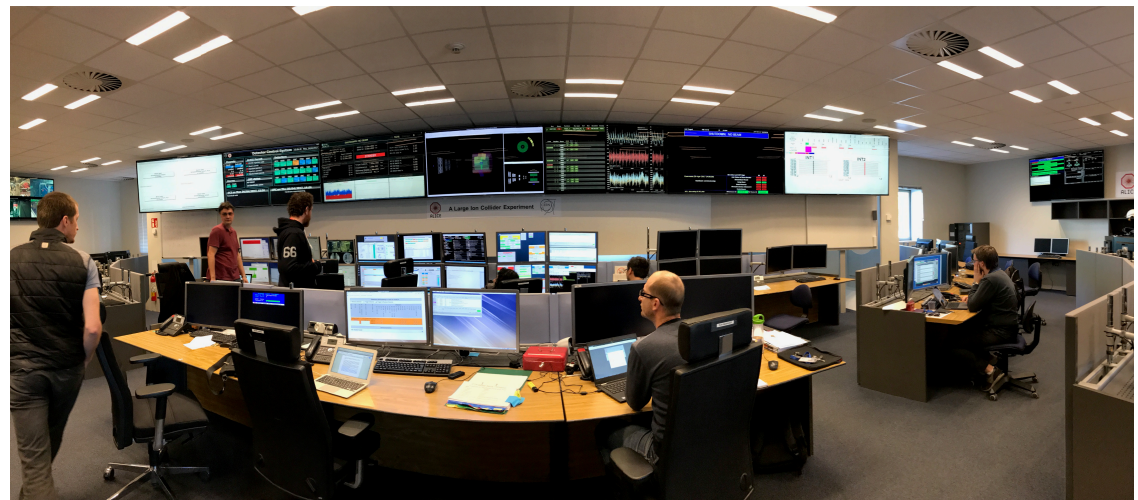
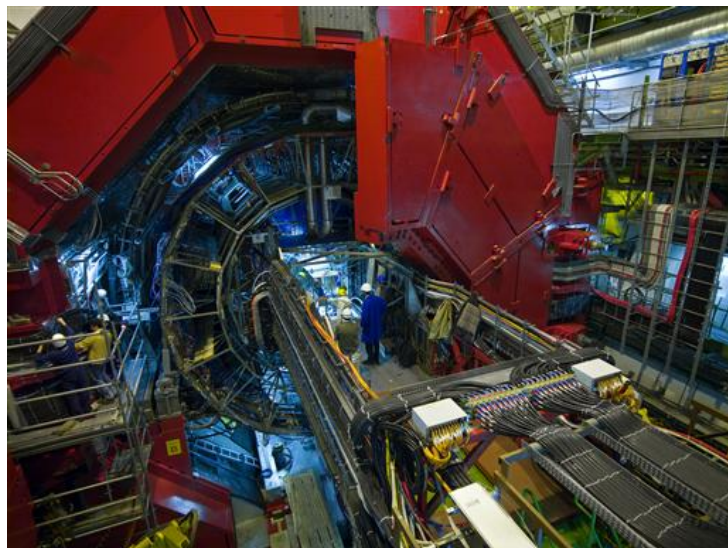
Rutherford's Plan to "See" Inside Atom



A Large Ion Collider Experiment (2009) (Studies related to quark gluon plasma)

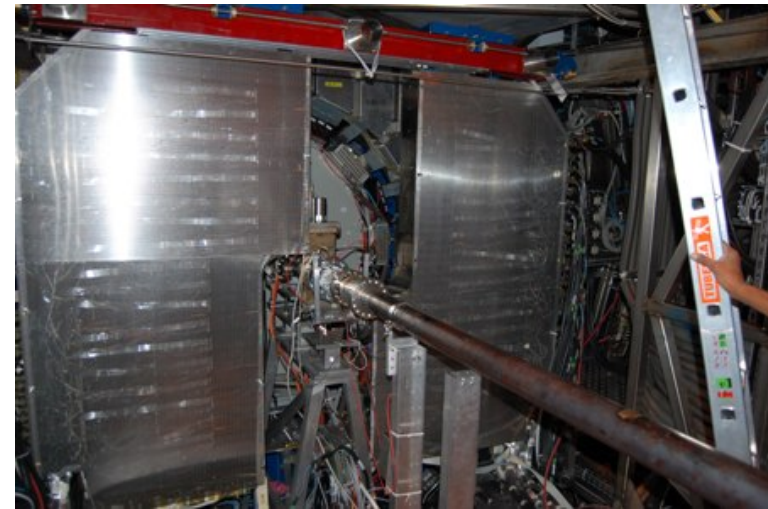
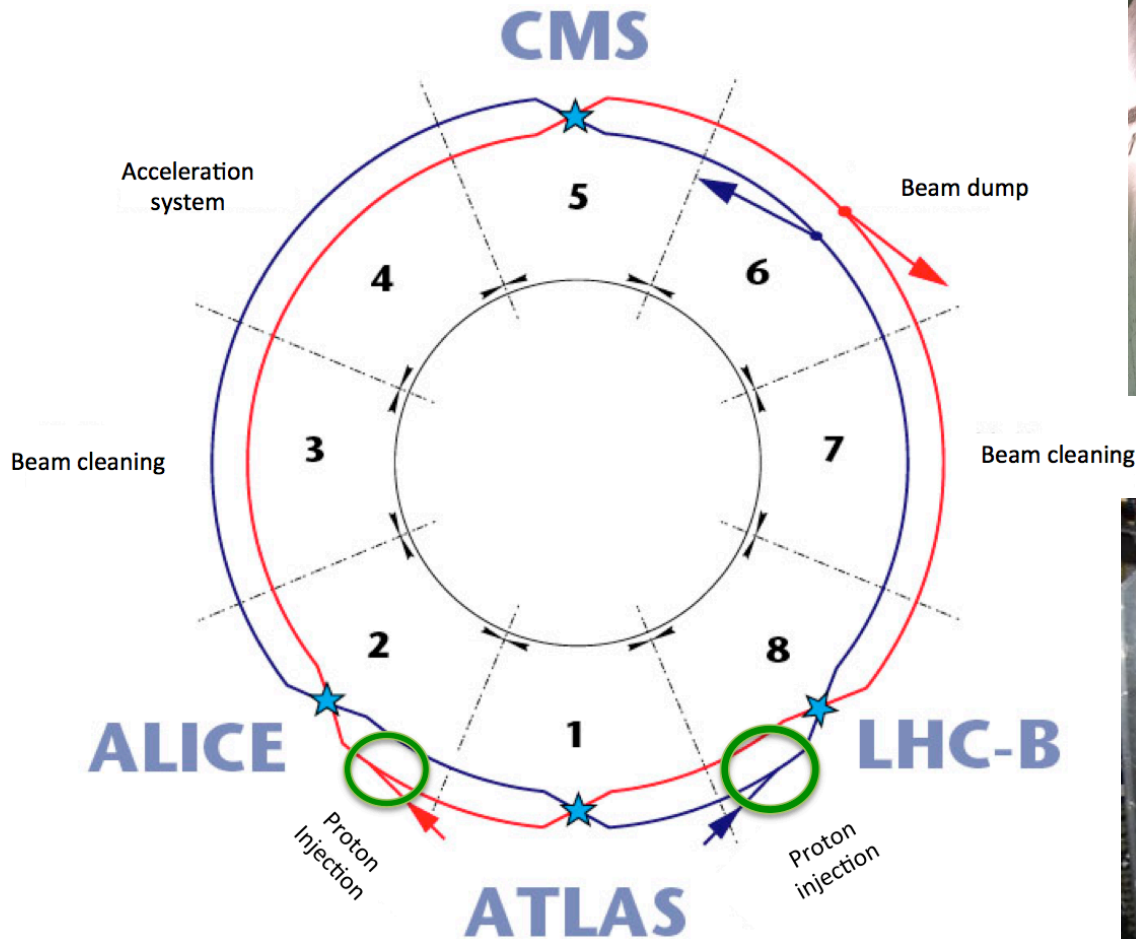


- More than 2000 Scientists
- More than 150 institutes
- 37 countries



Large Hadron Collider (Discovery of Higgs boson)

LHC rings (two rings in a 27 km tunnel)



The Data Challenge

- LHC experiments will produce **10-15 million Gigabytes** of data each year (about 20 million CDs!)
- LHC data analysis requires a computing power equivalent to **~ 100,000 of today's fastest PC processors.**
- Requires many cooperating computer centres, CERN providing only ~20% of the computing resources

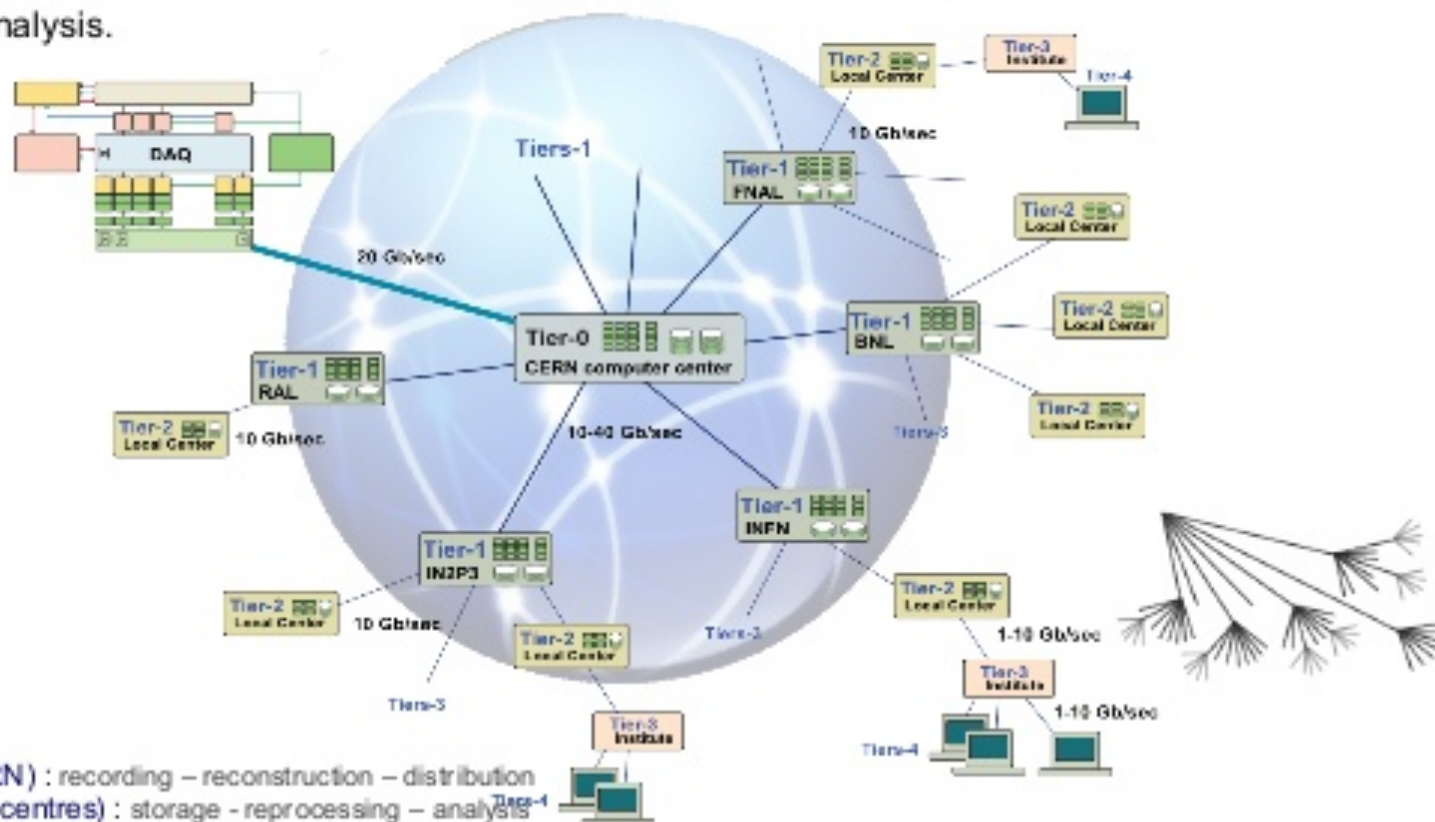




LHC Offline Computing. The GRID



The GRID. A distributed computing infrastructure (~150 kCores), uniting resources of HEP institutes around the world to provide seamless access to CPU and storage for the LHC experiments. A common solution for an unprecedented demand (in HEP) of computing power for physics analysis.



Tier-0 (CERN) : recording – reconstruction – distribution
Tier-1 (~10 centres) : storage – reprocessing – analysis
Tier-2 (~140 centres) : simulation – end-user analysis

Fundamentals of Linux/Unix

Syllabus:

An introduction to Linux/Unix operating system, fundamentals in command line interpreter, Remote access, a hands-on study of basic commands, file system, text editor, and shell scripting/programming

Fundamentals of Linux/Unix

Linux: A multi-user, multi-tasking, open-source operating system (OS)

Multi-user:

- Can have multiple user accounts with their own passwords
- Multiple users can login and work simultaneously
- Multiple users can have their own environments, own home directory, own desktop interface

Multi-tasking:

- Multiple programs running at same time
- Programs from users and system programs in the background

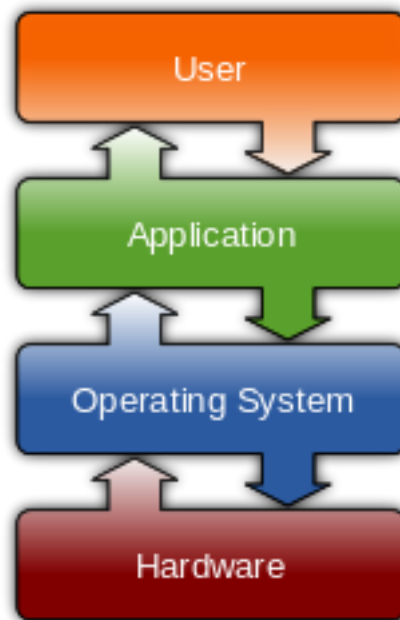
Open source:

- Source code is available to anyone for any purpose (free software)
- Use it, change it, study it, modify it, distribute it
- The concept started in 1990

Unix: A multi-user, multi-tasking operating system (OS)

Operating System

OS (Operating System): An operating system is made up of software instructions that lie between the computer hardware (disks, memory, ports) and the application programs (word processors, web browsers, etc.)

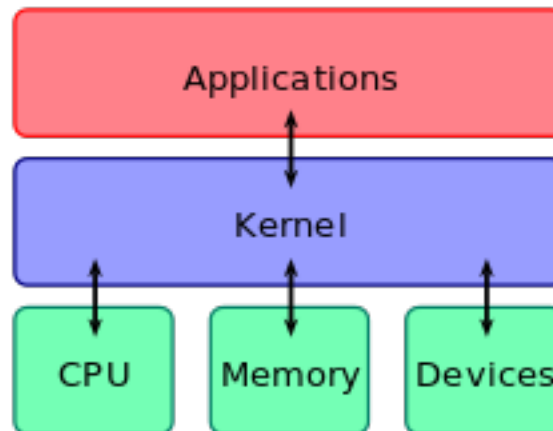


Kernel

OS provides kernel:

Kernel is a computer program that is the core of the OS with complete control of everything in the system (memory management, device management, system calls: when a process makes requests of the kernel it is called system calls).

- It connects the application software to the hardware of a computer.
- It is the first program loaded to startup
- Handles rest of the programs
- Handles memory and peripherals (keyboard, mouse, monitors, printers,

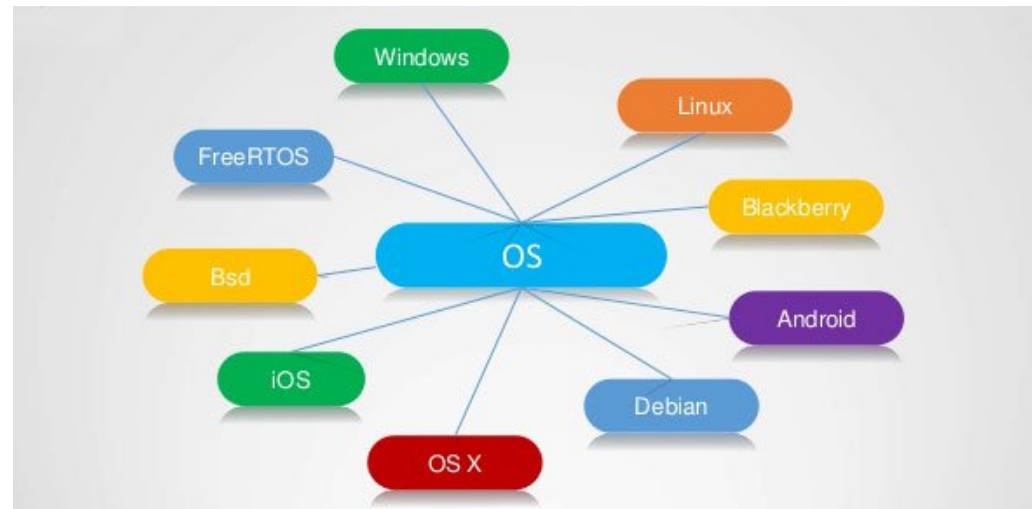
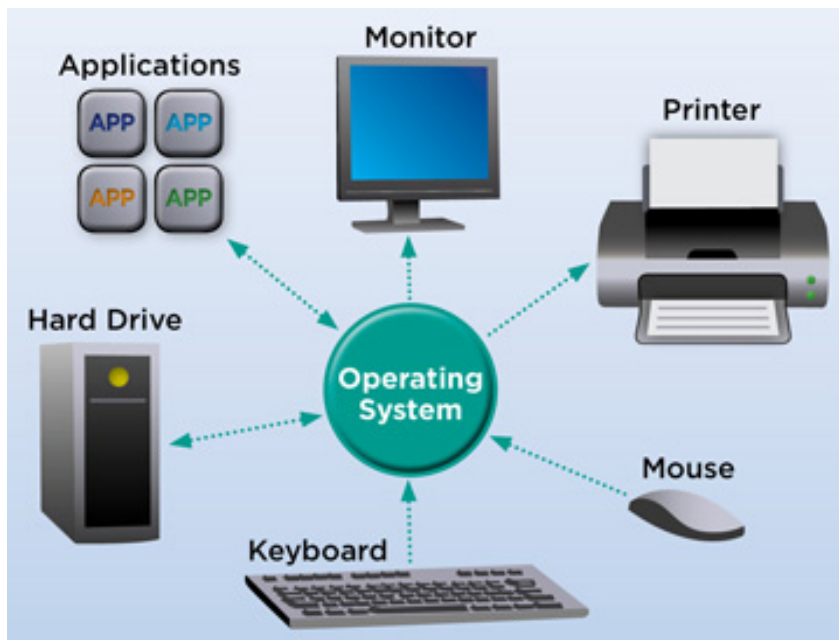


Operating System

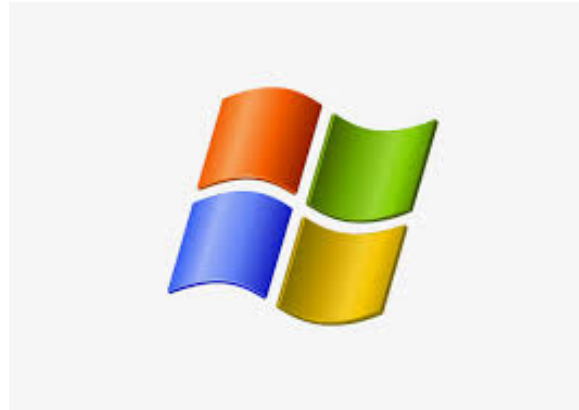
Beside kernel OS provides:

- File system: Structure to store information
- Device drivers: interfaces for hardware devices connected to the system
- User interfaces: ways to run program and access files (command line interfaces, graphical user interfaces (GUI))
- Security and protection
- Input/Output operations

Examples of OS: Windows, Linux, MacOS



Operating Systems



Mac OS or OSX: GUI based OS introduced by Apple for Apple computers only in 1984 (secure, reliable, user friendly, easy to connected peripheral components, ...)

Windows: MS-DOS based OS introduced by Microsoft in 1990 (easy to use therefore most popular, automatic file management, connecting peripheral components is very easy, ...)

Linux: Kernel based OS originally built at university of Helsinki in 1991. Unix-like (developed for servers, workstations). This is an open source OS.



Features:

- Open source: Freely available, anyone can modify, use and distribute it.
- Compatible with wide range of hardware (PCs, Macs, Server systems)
- Multi user capability
- Multi tasking
- Security
- GUI
- File system
- Hardware supports: printers, USBs, Hard-disks, CDs/DVDs, sound/video cards
- Application supports
- Network services: services to clients on LAN or to the entire internet, print server, file server, email server, web server

Linux Shell

Command line interpreter: Class of programs to read texts entered by a user. It provide an interface to interact with the system.

Linux Shell is a command line interpreter that allows you to access Linux tools

- Program used to interpret and manage commands
- It is required to run commands
- It provides ways to run programs, work with file systems, compile codes and manage computer
- Ex.: Terminal

How to get a shell:

- With Desktop: Right click on the desktop, then click open a terminal
- No desktop: In case of systems without GUI (server), one logs in from a text based prompt and immediately begins from a shell

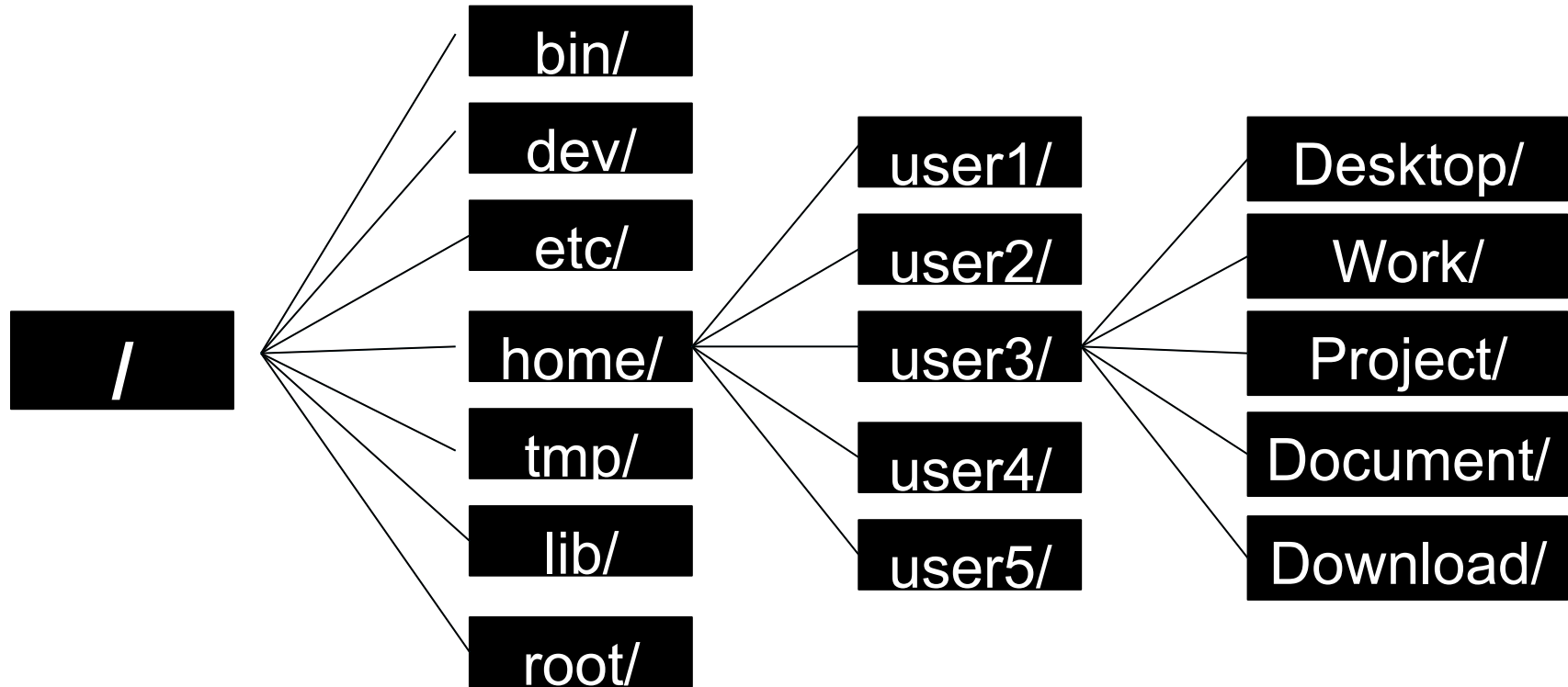
Shell prompt:

- Default prompt is dollar sign for normal user (\$)
- Pound sign (hash sign) for root user (#)

Linux File System

Hierarchy of directory and file systems in which all information on the computer are stored.

At the top is the root directory which is represented by single slash (/)



/bin: contains linux commands

/dev: contains device files for hardware devices attached to the system

/etc: local system configuration files

/home: directory for users

/lib: contains shared libraries required to boot the system

/root: home directory for the root user

Linux Questionnaire

- (1) Name five important applications of computers in your research work ?
- (2) What is Operating System (OS) ?
- (3) Name three most commonly used OS in computer world ?
- (4) Name one of the widely used OS in Smart Phones ? Can you find OS used in your mobile ?
- (5) What the the main functions of Operating System ?
- (6) What do you mean by Open Source Code ?
- (7) What is Linux ?
- (8) What is Linux Kernel ?
- (9) What is the main difference between Unix and Linux ?
- (10) What are main advantages of Linux ?
- (11) In the file hierarchy, the top root directory in Linux is represented by ?
- (12) Does Linux support Graphical User Interface (GUI) ?
- (13) What is Linux Shell ? Which signs of Shell prompts are used for normal and root users in Linux ?

Linux commands

Linux command: It is a utility of the Linux OS that allows execution of all basic and advanced tasks in Linux. Linux commands are case sensitive.

To learn more about any command (press Q to exit):

\$man command (\$man ls, \$man cp)

Accessing/Managing directories/files:

a) Print name of current working directory

\$pwd

b) Create a new directory under any director

\$mkdir [option] directory-name

c) Delete an existing directory

\$rmdir [option] directory-name

\$rm [option] directory-name

d) Display list of contents of a directory

\$ls [option] file-name/directory-name

(if no file/directory is given then it lists current directory)

(options: -l (detailed option), -a (hidden files))

e) Change the current directory

\$cd /Location/directory-name

Linux commands

Accessing/Managing directories/files:

Creating empty files

\$touch file-name

\$touch file-name-1 file-name-2

Display content of a file, copy content of one file to another file

\$cat fileName (display content of a file)

\$cat -n fileName (display content of a file + line number)

\$cat file_a file_b > file_ab (add content of file_a and file_b to file_ab)

\$cat file_a >> file_b (append content of file_a to file_b)

\$more fileName (show one page at a time)

\$more -10 fileName (show only 10 lines at a time)

\$less filename (show one page at time)

\$less -10 filename (show 10 lines at a time)

(use spacebar to move to next page, 'q' to quit, Note: 'less' is more flexible, one can navigate up/down using pageup/pagedown key, one can search keyword using '/')

\$head fileName (display top 10 lines of a file)

\$head -N fileName (display top N lines of a file)

\$tail fileName (display last 10 lines of a file)

\$tail -Number fileName (display last N lines of a file)

Linux commands

Accessing/Managing directories/files:

Copying content of a file/directory to another file

Copying files/directories from one location to other

\$cp file-name new-file-name (use option -r for directories)

(content of first file is copied to second file, same as renaming), both exist

\$cp /Location/file-name /Location/new-file-name

(first file is copied to other location with new name, both files exist)

\$cp /Location/file-name /Location/.

(first file is copied to other location with same name, both exist)

Moving content of a file/directory to another file,

Moving files/directories from one location to other

\$mv file-name new-file-name

(content of first file is moved to second file, same as renaming, only second file exist)

\$mv /Location/file-name /Location/new-file-name

(first file is moved to other location with new name, only second file exist)

\$mv /Location/file-name /Location/.

(first file is copied to other location with same name, only second file exist)

Linux commands

Accessing/Managing directories/files:

Removing/deleting files/directories

\$rm [option] file-name/dir-name

\$rm -i file-name

\$rm -i /Location/file-name

\$rm -r dir-name

\$rm -r /Location/dir-name

Linux commands

File permissions: 9 bits

\$chmod [option] **mode** file/directory

Changes the access permissions of a file/directory

\$ls -l

d rwx rwx rwx

- rwx rwx rwx

Owner/user, group, others

r -> read => 4

w -> write => 2

x -> execute => 1

mode:

0 => no permission

4 => read only, 2 => write only, 1 => execute only

6 => read & write, 5 => read & execute, 7 => read, write, execute

\$chmod 746 file_name

- rwx r-- rw-

File permission to a directory (recursive)

\$chmod -R 777 DirectoryName/

Linux commands

Login related commands:

To identify and get information about user and login session

\$id

Identity: user-name (user-id), group-name (group-id),

Information about login session: when, how-long, idle, from where

\$who (info. About current login)

\$whoami (tell user-name of current login)

\$w (info. About all logged-in users)

\$su user-name (to run a shell using another user with same shell environment)

\$su - user-name (to run a shell using another user with target users' shell environment)

\$su root (to run a shell as root user with same shell environment)

\$su - root (to run a shell as root user with root's shell environment)

\$sudo (running command as root without using root password)

Changing passwords

\$passwd (change password for current user)

\$passwd user-name (change password for given user)

\$chage -l (to know information about the password)

Linux commands

System commands:

\$df

(display information about total disk space, disk space currently in use, free disk space on all mounted drives)

\$df -h

(size in human readable format (giga/mega/kilo bites))

\$du (show disk space occupied by file/directories; block size/number of disk blocks)

\$du -s

(total disk space used in current directory)

\$du -h

(disk space in human readable format for all files and directories in the current working directory)

\$date [option] (display current system time)

Monitoring system performances and other commands

Information about running processes (which programs are running, resources they are using, who is running them)

\$ps au

(options a: processes of all users, u: show user names)

Username, PID (process ID), TTY (terminal device used), %CPU, %MEM,

Information about all processes including those which are running in background)

\$ps aux

\$ps aux | less (to show page-by-page)

Linux commands

To locate path of a command

\$which ls

\$top (display process activity) (press q to quit)

\$kill [option] PID (to kill/terminate a process)

To clear the current shell

\$clear

To create aliases for commands

\$alias alias-name=command

To remove alias

\$unalias alias-name

Print the output as received from the shell

\$echo text

\$echo “text”

\$echo -e “text” (line break/tab can be used, \n: line break, \t: tab)

To list last commands used in shell

\$history

Linux commands

To reboot the system

\$reboot

To shutdown the system

\$shutdown -h now

To logout a user

\$logout

Installing software

\$sudo apt-get install <package-name>

\$sudo yum install <package-name>

Compressing in linux

\$tar cvf target-file-name.tar source-file/source-dir (making tar file)

\$tar xvf file-name.tar

Mount/unmount devices

\$mount [option] [device] mountpoint

\$umount [option] mountpoint

Text editors

Text Editors:

A text editor is a software/program that provides users a facility to type some texts into it, edit/change it and store it into the computer.

Few text editors are provided by the OS.

One can install additional text editors compatible to the OS.

Ex.: vi, emacs, pico, kedit, gedit, notepad (Windows)

Text Editor: vi

`$vi file-name`

It has two modes:

command mode (move the cursors, save changes, exit, it understands the commands)

input mode (allow to enter texts)

By default a file is opened in command mode.

Type 'i' to go from command mode to input mode

In input mode you can insert the texts you want

Press 'Esc' to go back to command mode from input mode

Editor vi

Saving and Quitting vi editor:

Go to command mode by pressing 'Esc'

Type: ':w' save the file but keep it open

Type ":q" exit without saving

Type ":q!" exit without saving the changes

Type ":wq" exit after saving the changes

Shift+zz: Save the file and quit

Undo last changes (all changes to entire line): u (U)

Delete character at the cursor: x

Delete word: dw

Delete N-word: Ndw (N:1,2,3, ...)

Delete content of line after cursor: D

Delete a line: dd

Delete N-lines: Ndd (N: 1, 2, 3, ...)

Shell script

Shell script:

Shell script is a file containing texts.

It contains series of commands.

While running a shell script the shell/terminal reads this file and execute the commands as if they are entered directly on the command line interpreter.

Most of the things that can be done in command line can be done by shell scripts.

Shell script

Steps for writing a simple shell script:

- Open a file in a dir you want with .sh extension
- Enter the texts (some commands)
- Save it
- Provide executable permission to the file
- Run/Execute it

`#!/bin/bash`

This line indicates the program to be used to interpret the script. In this case it is bash shell (/bin/bash)

`# Program to print texts`

This line is comments and is not execute

`echo "This is my first shell script"`

This line prints the texts

Steps for writing a simple shell script:

- `vi print_text.sh`
- Write inside file

`#!/bin/bash`

`# Program to print texts (P1)`

`echo "This is my first shell script"`

- `chmod 700 print_text.sh`
- `./print_text.sh`

`chmod 700 print_text.sh`

Command to make the program executable

`./print_text.sh`

Executes the program

Shell script

For loop

```
#!/bin/bash
#Program using for loop (P2)
for num in 1 2 3 4
do
echo $num
done
```

```
#!/bin/bash
# Program using for loop (P3)
for num in {1..5}
do
echo $num
done
```

For loop

```
#!/bin/bash
#Program using conditional statement (P4)
echo enter a number
read y
if [[ $y -lt 10 ]]; then
    echo "The number is less than 10 "
else
    echo "The number is less than 10 "
fi
```

- eq (Equal to)
- ne (Not equal to)
- lt (Less than)
- le (Less than or equal to)
- lt (Less than)
- ge (Greater than or equal to)

Remote login

Remote Access

Establishing connection between two machines located remotely

Allowing access and permissions to remotely manage other computers, transfer files, and do virtually anything you can do while physically sitting in front of the machine.

One is **host/server** other is **user/client**

Two most widely used methods/protocols:

Secure Shell (SSH) for Linux-based machines

Remote Desktop Protocol (RDP) for Windows-based machines

Requirements:

- The remote computer must be turned on have a network connection
- The host and user applications need to be installed
- IP address or the name of the remote machine you want to connect to
- You need to have the necessary permissions to access the remote computer
- Firewall settings need to allow the remote connection

SSH commands

Login to remote a machine:

```
$ssh user_name@ip_address ($ssh sidharth@14.139.199.51)
```

Copying file from a remote machine:

```
$scp user_name@ip_address:/full_path/file_name file_name  
($scp sidharth@14.139.199.5:/home/sidharth/test.sh test.sh)
```

Copying file to remote a machine:

```
$scp file_name user_name@ip_address:/full_path/file_name  
($scp test.sh sidharth@14.139.199.5:/home/sidharth/test.sh)
```

THANK YOU AND BEST OF LUCK