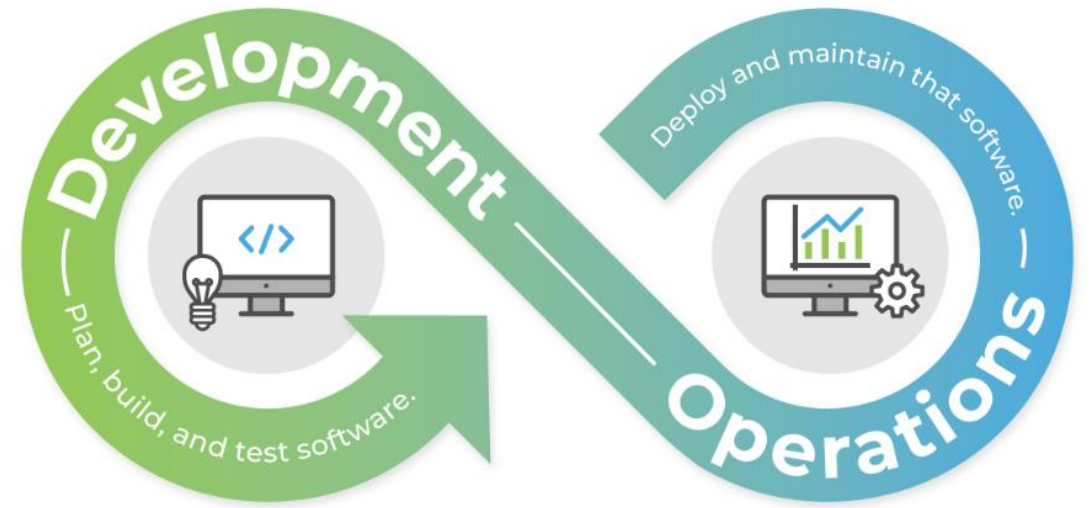


DevOps

01 Linux Basics

Introduction to UNIX & Linux



Agenda for this session

- Why do we learn 'Linux Basics' in DevOps elective?
- Introduction to OS, UNIX History & Evolution
- Introduction to Linux, Shells & File System
- Difference between UNIX and Linux
- Absolute & Relative Paths



Why 'Linux Basics' in DevOps Course?

- *In 2023, Linux runs on all of the [top 500 Supercomputers](#), again!.*
- *Linux is once again the most loved platform for development in 2019 on [StackOverflow developer survey results](#) 2019.*
- *96.3% of the world's top 1 million web servers run on Linux.*
- *Every Facebook post you make, every YouTube video you watch, every Google search you run, is [done on Linux](#).*
- *95% of the servers that run the world's top 1 million domains are powered by Linux.*
- *In 2018, Android dominated the mobile OS market with 75.16%. 85% of all smartphones are based on Linux. Smart TVs and other similar devices use Linux.*
- *Even the famous DevOps tools like Docker, Ansible and Kubernetes etc. run with the help of Linux. For most of the years initially, Docker was only available on Linux based systems. A Linux system is required to be the Ansible controller. Also, Master nodes in Kubernetes can only be Linux systems*

Source: <https://kodekloud.com/blog/why-is-linux-so-popular>

Knowing Linux is a big plus

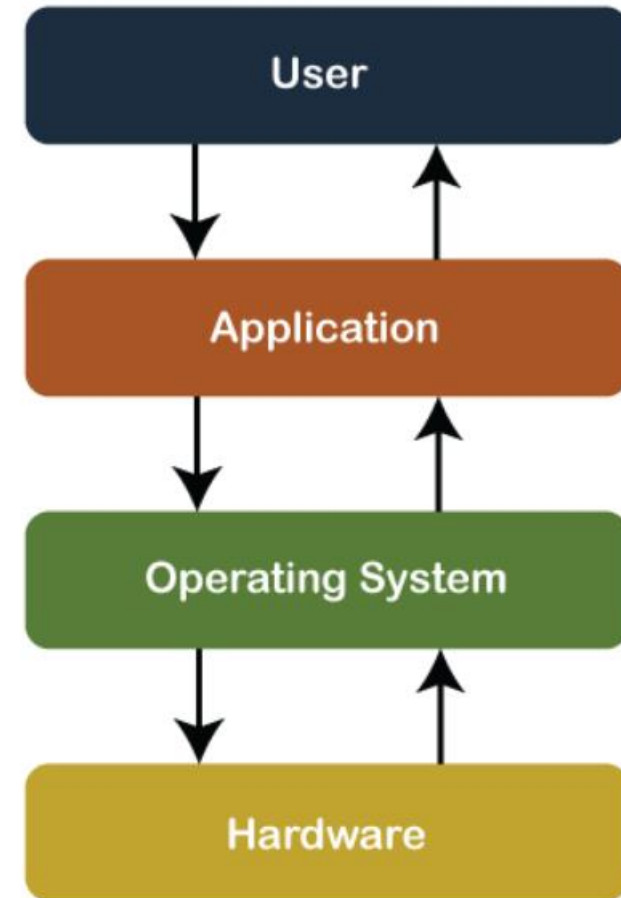
- Industry - (IT services, start-ups, captive centres and others) consider candidates who know Linux in addition to Windows
- Higher Studies / Research – It is an advantage to know Linux. Many international universities and research institutes consider Linux.
- When you learn DevOps and Cloud Computing, knowing Linux basics is a value addition.

Operating System – A Quick Glimpse



Operating System – What is it?

- Software or program that runs when you turn on your computer. All other programs (such as browsers, word processors, and so on) run on top of operating system.
- Operating system manages all resources on a computer (CPU, memory, hard disk, and so on.)



Operating System – What does it do?

- **Controls Hardware** - The operating system controls all parts of the computer in order to get everything working together.
- **Runs Applications** – It runs application software such as word processors, web browsers, games, etc...
- **Manages Data and Files** – It helps you manage data and files. It helps you create, modify, delete and organize data and files. It helps you organize data and files in a certain way.

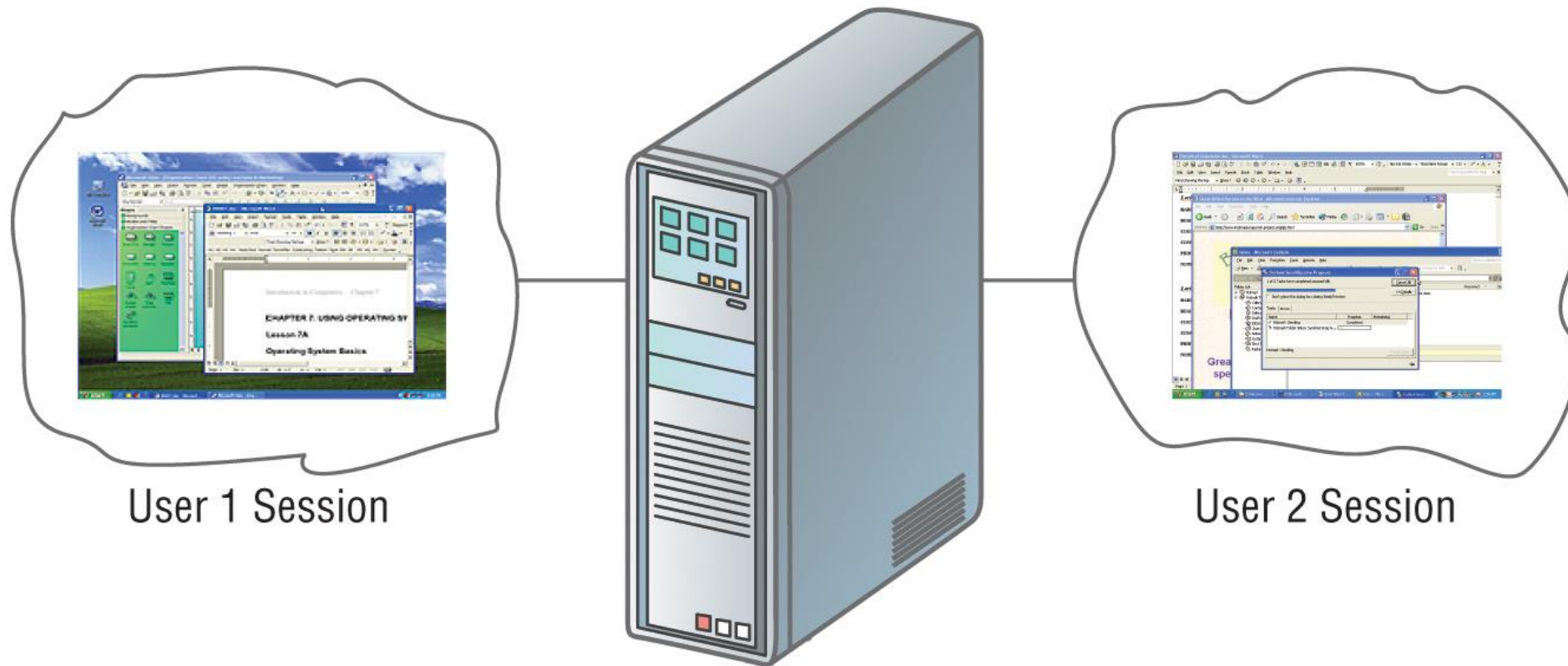
Operating System - Functions

- Memory Management
- Processor Management
- File Management
- Device Management
- I/O Management
- Security
- Command Interpretation
- Storage Management

Types of Operating Systems

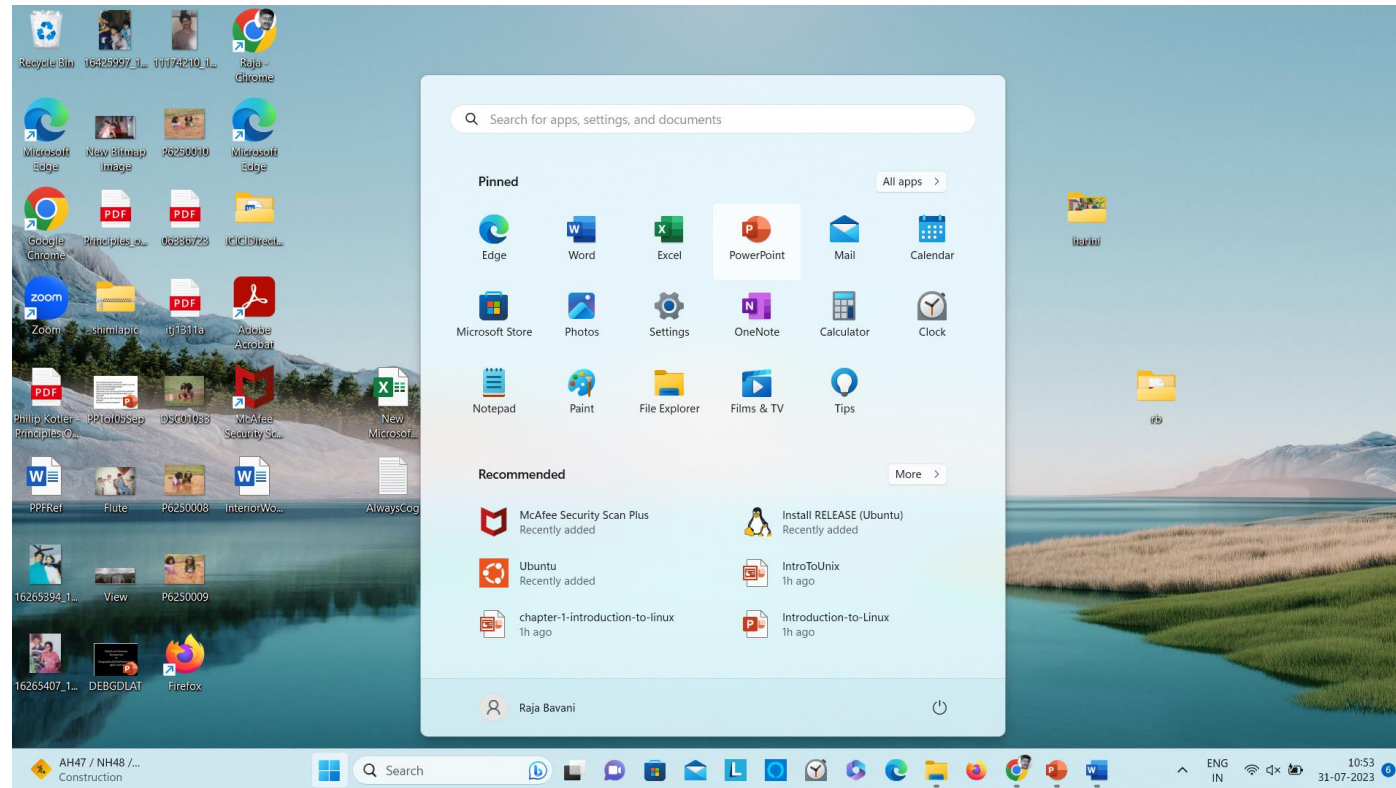
1. **Real Time OS:** Very small in size and very fast in execution, built into a device, respond quickly to inputs from users (Example: OS in MP3 player, medical devices)
2. **Single User/Single Tasking:** Takes up some space on the hard disk, runs on personal computers (Example: MS-DOS, Palm OS, iPad OS)
3. **Single User/Multi Tasking:** Most common OS in personal computers with higher configuration (expensive ones) - (Windows XP, Mac OS)
4. **Multi User/Multi Tasking:** Many users connect to one computer, One session per user (Example: UNIX, Linux)

Multi-user/multi-tasking OS



Users interact through a User Interface

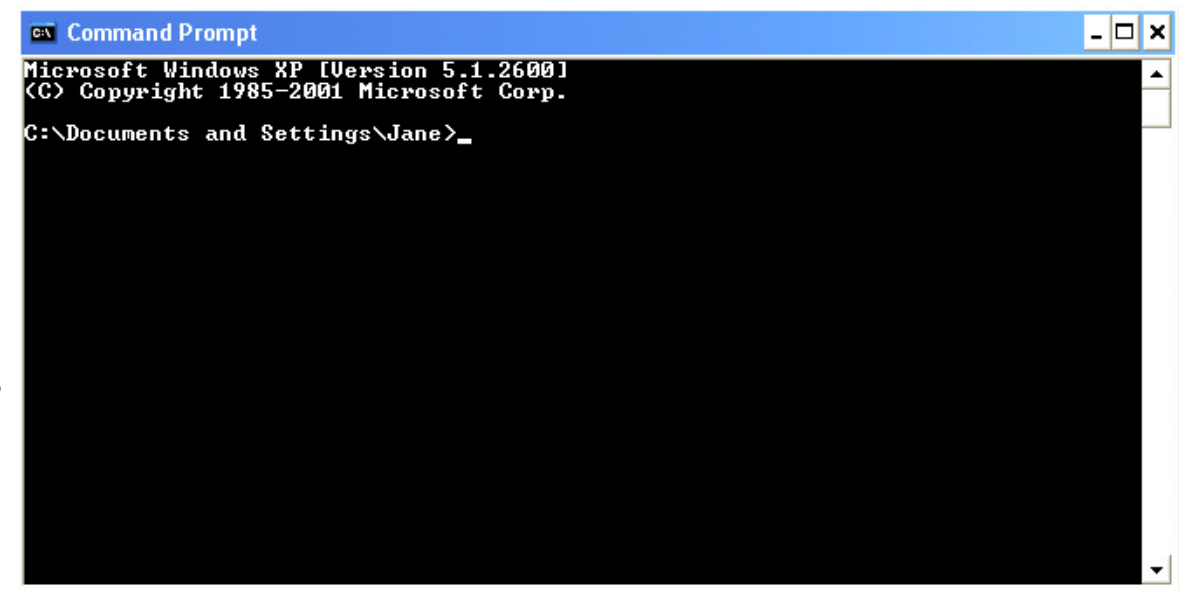
Also known as Graphical User Interface or GUI.



Another (old) way is a command line interface

Command Line Interface or CLI

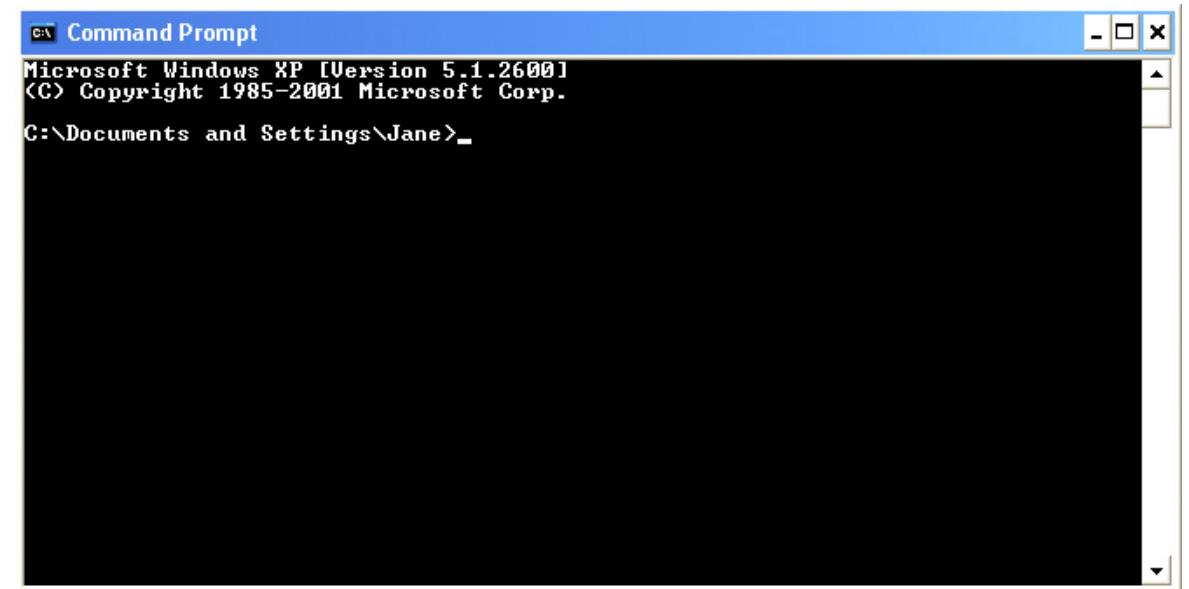
- User has to type a command at the prompt in order to perform an operation (such as creating a file, renaming a file, listing all files in a folder, ...)
- User has to remember all commands
- Unix, Linux, and DOS support CLI
- Example: Command prompt in Windows



Command Line Interface (CLI)

It is helpful in many ways...

1. Getting access to OS features and making system calls
2. Linking files
3. Installing device drivers
4. Trouble shooting
5. Running utilities



UNIX History

**UNiplexed Information
and
Computing System**

UNIX was invented in Bell Labs

The UNIX operating system was born in the late 1960s. It originally began as a one man project led by Ken Thompson of Bell Labs, and has since grown to become the most widely used operating system.

UNIX Evolution

- Since its inception, UNIX has gone through several different generations and even mutations.
 - Some differ substantially from the original version, like Berkeley Software Distribution (BSD) or Linux.
 - Others, still contain major portions that are based on the original source code.
- Detailed information on these variations of UNIX is available at <http://www.levenez.com/unix/history.html>.

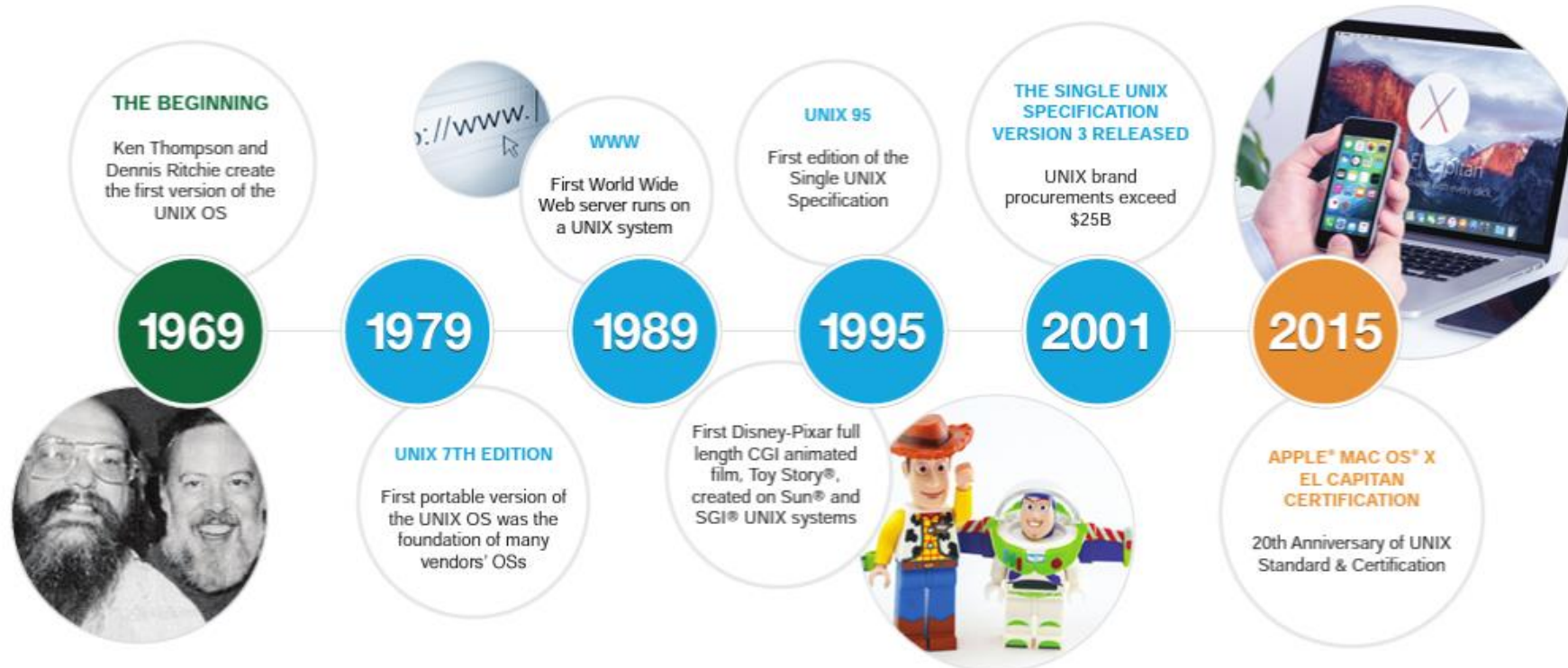
UNIX®

An Open Group Standard



The UNIX platform demonstrates the value of being open, since as a truly open standard it allows all to focus on driving innovation of the ecosystem around the platform rather than competing at the core OS level. The open standard makes portability easier for software developers, provides integrators with choice in the building blocks for solutions, and enables customers to focus on solving business problems rather than integration issues.

Steve Nunn, President and CEO, The Open Group

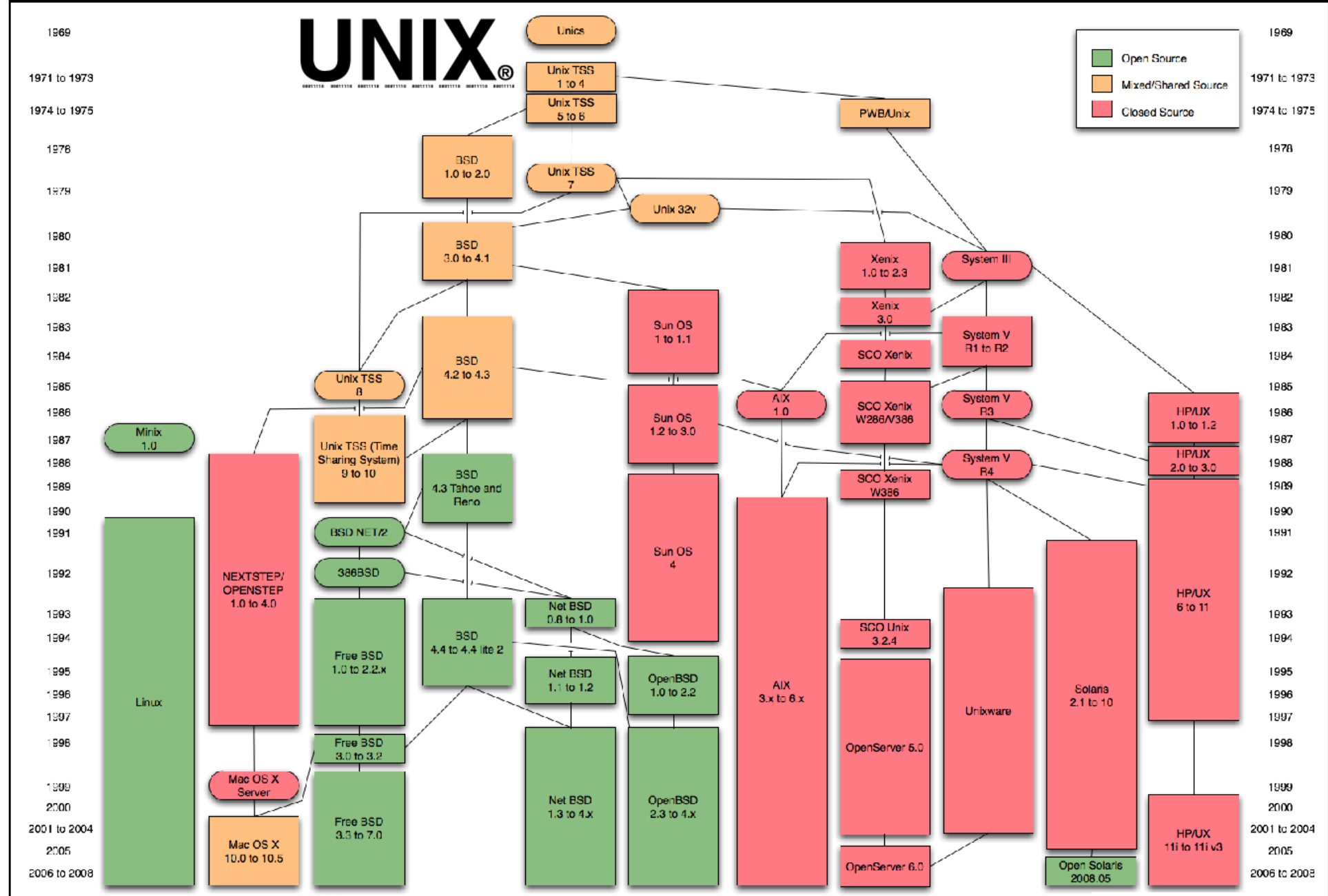


THE
Open
GROUP

UNIX® and The Open Group® are registered trademarks of The Open Group. Sun® is a registered trademark of Oracle Corporation. SGI® is a registered trademark of Silicon Graphics International Corporation. Apple® and Mac OS® are registered trademarks of Apple, Inc. Toy Story® is a registered trademark of Disney Enterprises, Inc.

UNIX®, a robust open standard managed by The Open Group, enables increased IT agility, scalability, and interoperability across diverse operating and computing environments driving business, technology, and innovation around the globe.

For more information visit: www.opengroup.org/UNIX



UNIX - General Characteristics

- **Multi-user & Multi-tasking** - UNIX (most versions) allow multiple users to log onto the system, and have each user run multiple tasks.
- **Over 30 Years Old** – It is more than 30 years old. It continues to have high popularity and usage. Many variations have spawned off and many have died off, but most modern UNIX systems can be traced back to the original versions. UNIX has endured the test of time. (Windows 1.0 released in the mid 80s was not stable or very complete. Windows 3.x family was released in the early 90s).
- **Large Number of Applications** – There are numerous applications available on UNIX ranging from commercial applications such as CAD, Maya, WordPerfect, and others. Many are free.

UNIX - General Characteristics (Contd..)

- **Free Applications and Even a Free Operating System** - Many of the UNIX applications are free. And many of the flavors of the OS are free.
- **Less Resource Intensive** - Most UNIX installations tend to be much less demanding on system resources (CPU, Memory). In many cases, a PC that can run Windows is more than sufficient to run the latest version of Linux.
- **Internet Development** - Much of the backbone of the Internet is run by UNIX servers. Many of the more general web servers run UNIX with the Apache web server (it is free too).

Parts of the UNIX OS

- **The Kernel** - Technically, the kernel is the OS. It handles memory management, input and output requests, and program scheduling. It provides the basic software connection to the hardware.
- **The Shell and Graphical User Interfaces (GUIs)** - UNIX shell provides a “command line” interface which allows the user to type in commands. These commands are translated by the shell into something the kernel can comprehend, and then executed by the kernel.
- **The Built-in System Utilities** - Programs that allow a user to perform tasks which involve complex actions. Utilities provide user interface functions that are basic to an operating system, but which are too complex to be built into the shell. Examples of utilities are programs that let us see the contents of a directory, move & copy files, remove files, etc.
- **Application Software & Other Utilities** – They are additional programs that are bundled with the OS distribution, or available separately. These can range from additional or different versions of basic utilities, to full scale commercial applications.

Flavors of UNIX

1. PROPRIETARY: (redistribution and modification are prohibited or restricted; license is not free)
 - Solaris
 - IRIX
 - Mac OS X
 - and many others...
2. OPEN SOURCE: (source code is readily available and free to modify)
 - Free BSD
 - Linux Distributions
 - RedHat
 - Fedora Project (maintained by RedHat)
 - Mandrake
 - Debian
 - Ubuntu
 - SUSE
 - Slackware
 - and many others...

Linux is a open source UNIX-based OS developed in 1991 by Linus Torvalds, a Finnish undergraduate student.

Linux Basics

**Lovable Intellect
Not Using XP**



Linux

- Linux is a clone of UNIX written from scratch by Linus Torvalds with assistance from a loosely-knit team of hackers across the Net.
- Linux and Unix strive to be POSIX (Portable Operating System Compliance) compliant. POSIX is a family of standards specified by IEEE for maintaining compatibility among operating systems.
- 64% of the world's servers run some variant of Unix or Linux. The Android phone and the Kindle run Linux.



Linux has many distributions



Difference between Linux and UNIX

#	Factor	Linux	UNIX
	Cost	Linux is freely distributed. It can be downloaded for installation. There are paid versions also available for Linux.	Different flavors of Unix have different pricing depending upon the type of vendor.
	Development	Linux is Open Source, and thousands of programmer collaborate online and contribute to its development.	Unix versions are primarily developed by Bell Labs as well as other commercial vendors.
	Users	Anyone can use Linux whether a home user, developer or a student.	Developed mainly for servers, workstations and mainframes.
	CLI	BASH is the Linux default shell. It offers support for multiple command interpreters.	Originally made to work in Bourne Shell. However, it is now compatible with many others software.
	Usage	Linux OS can be installed on various types of devices like mobile, tablet computers.	UNIX is used to run internet servers, workstations & PCs.
	Source Code	Available to the general public	Not available

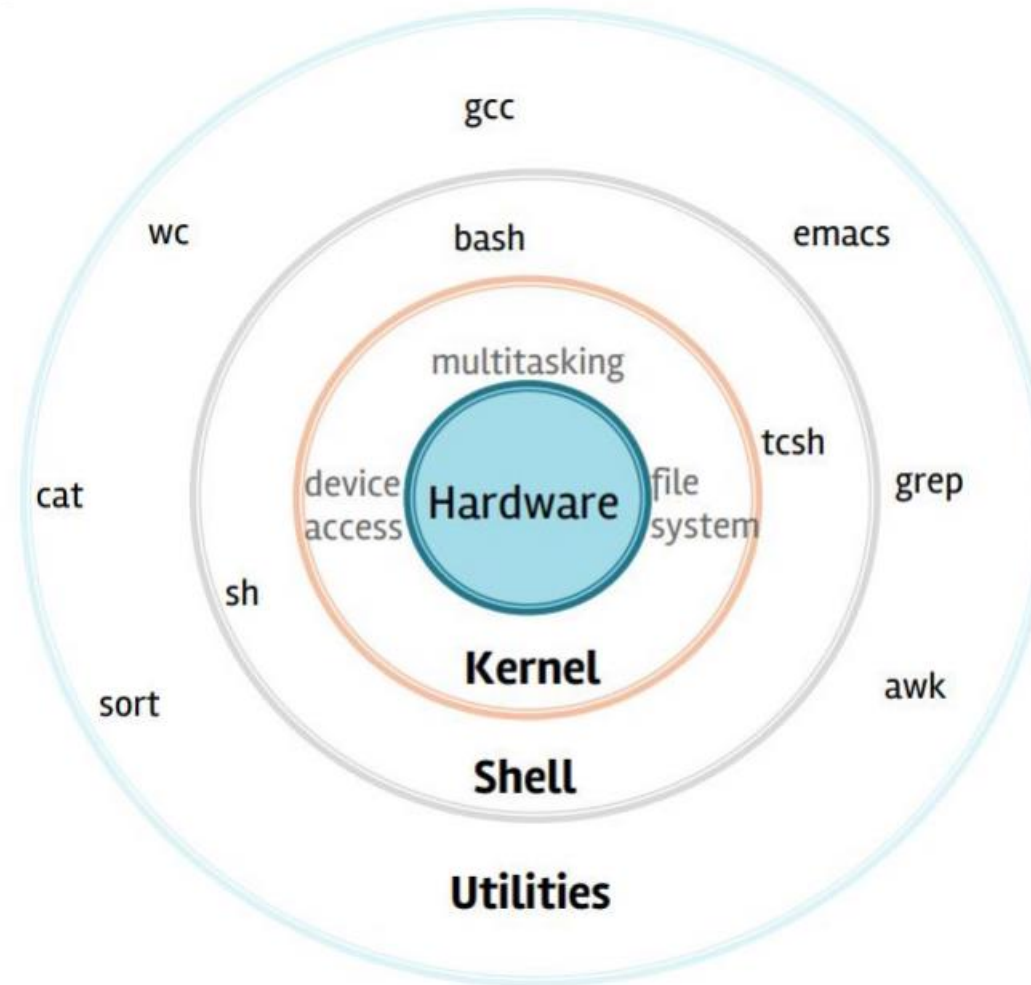
Pros and Cons of Linux



Why Linux



Linux – High level view



Shell

- Shells are how command-line interfaces are implemented in Linux/Unix. The command line interface is the primary interface to Linux/Unix operating systems.
- There are different types of shells. Each shell has varying capabilities and features and the user should choose the shell that best suits their needs.
- The shell is simply an application running on top of the kernel and provides a powerful interface to the system.

Different types of shells

Bourne Shell	sh	Developed by Stephen Bourne at AT&T Bell Labs. The Bourne shell was the first default shell on Unix systems, released in 1979. The shell program name is sh, and the traditional location is <i>/bin/sh</i> . The prompt switches to \$, while the root prompt is #. Very popular because it is compact and fast. However lacks features such as autocomplete, command history and, logical and arithmetic expansion.
C Shell	csh	Developed by Bill Joy at University of California, Berkeley in the late 1970s. The main objective was to improve interactive use and mimic the C language. The path to the C shell executable is <i>/bin/csh</i> . The prompt uses % for regular users and # for the root user. Drawbacks: a) syntax inconsistencies, b) no support for standard input/output (stdio) file handles or functions, and c) Not fully recursive, which limits complex command handling.
Korn Shell	ksh	Developed by David Korn at AT&T Bell Labs in the early 1980s.. It is based on the Bourne shell (sh). Backward-compatible with the Bourne shell and includes many features of the C shell. The location is in <i>/bin/ksh</i> or <i>/bin/ksh93</i> . The prompt is the same as the Bourne shell (\$ for a user and # for root).

Different types of shells

Bourne Again Shell	bash	Developed by Brian Fox in 1989 for the GNU Project as a free software replacement for the Bourne shell (sh). Default Shell on Linux and Mac OSX. The location is /bin/bash. Like the Bourne shell, the bash prompt is \$ for a regular user and # for root. Bash introduces features not found in the Bourne shell, some of which include brace expansion, command completion, basic debugging and signal handling, command history, conditional commands such as the <u>bash if</u> and <u>bash case</u> statements, and <u>Heredoc</u> support.
TENEX C Shell	tcsh	Developed by Ken Greer in the early 1980s at Carnegie Mellon University. It is essentially the C shell with programmable command line completion, command-line editing, and a few other features. The executable path is in /bin/tcsh. The user prompt is hostname:directory> while the root prompt is hostname:directory#. Early versions of Mac OS and the default root shell of FreeBSD use tcsh. Additional features of the shell include, advanced command history, programmable autocomplete, wildcard matching, job control and built-in where command.

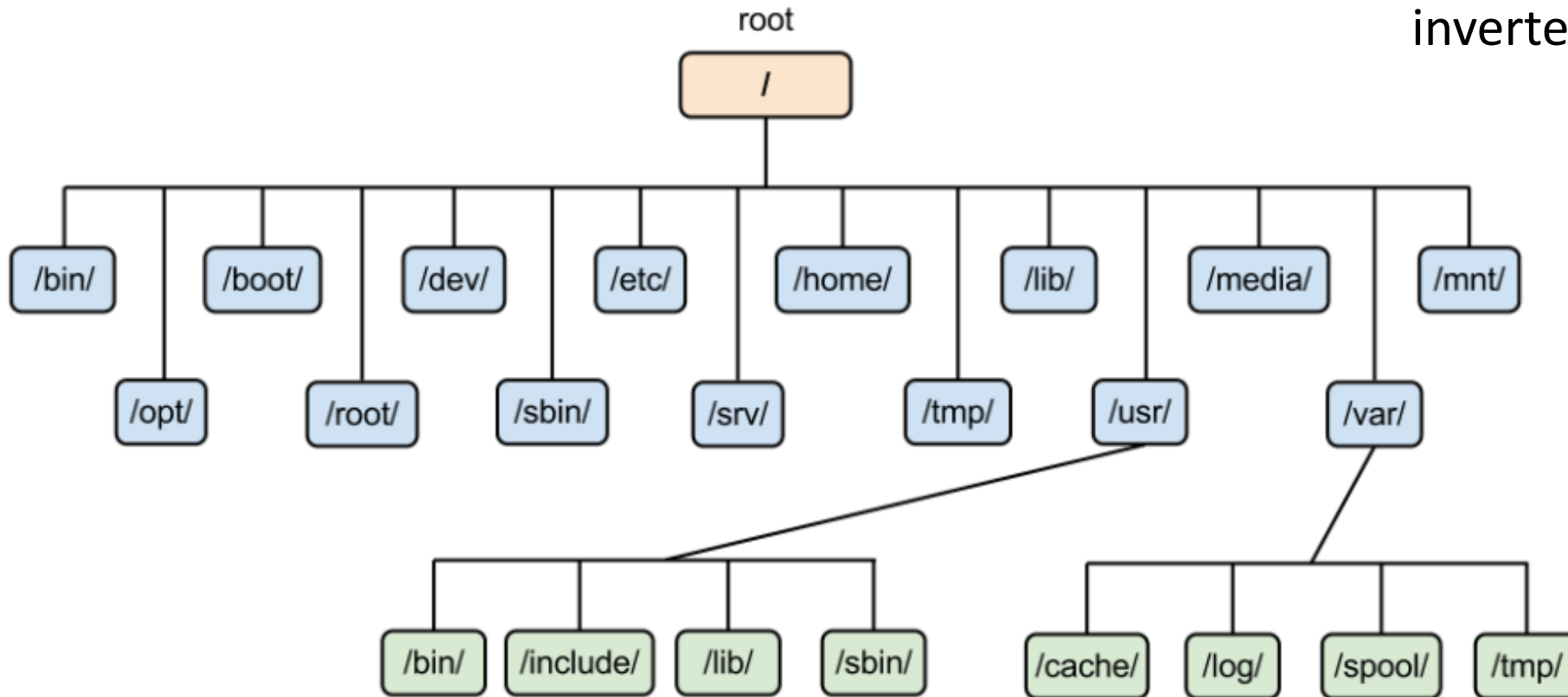
Linux File System

**Lovable Intellect
Not Using XP**

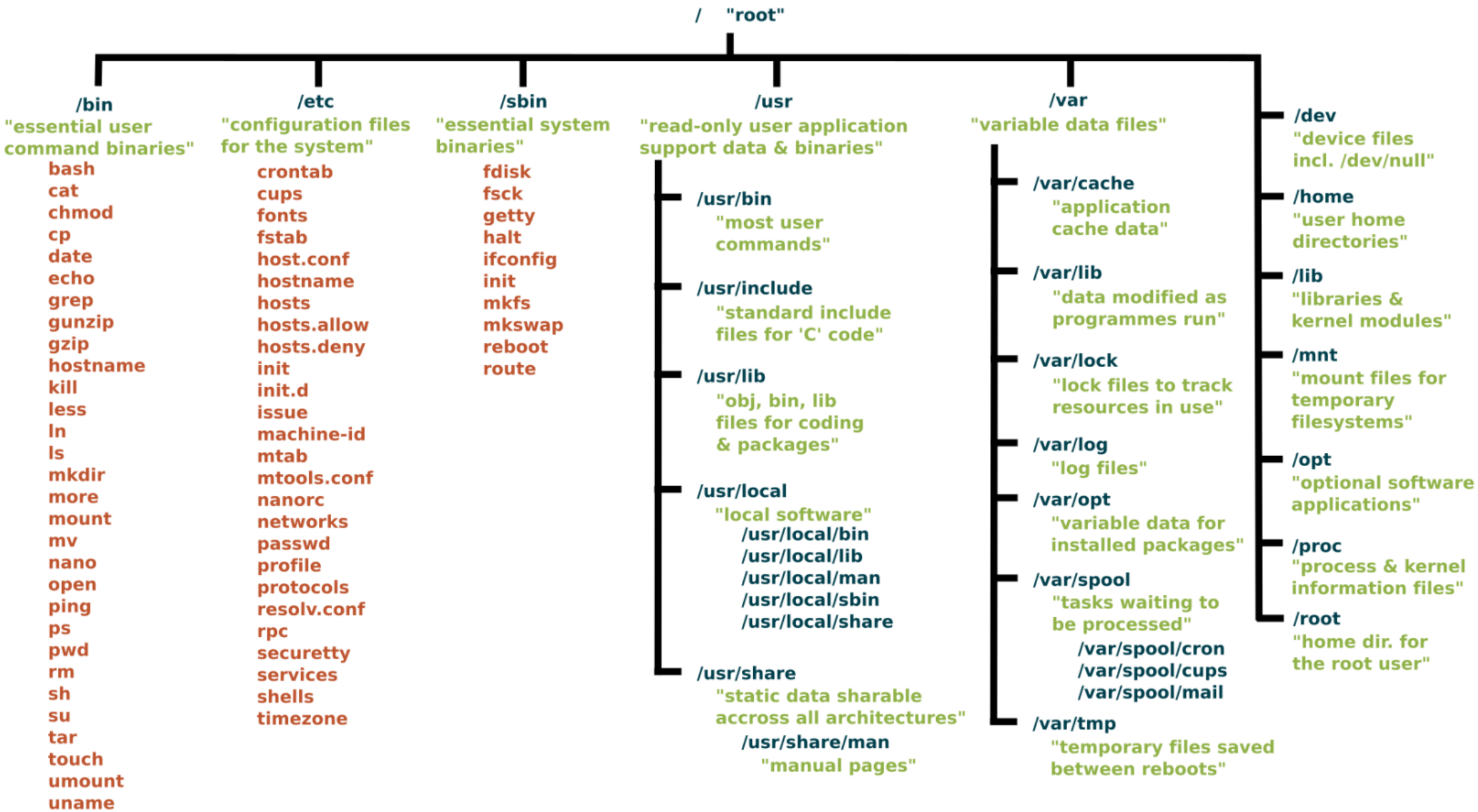


Linux File System

Linux file
system
resembles an
inverted tree



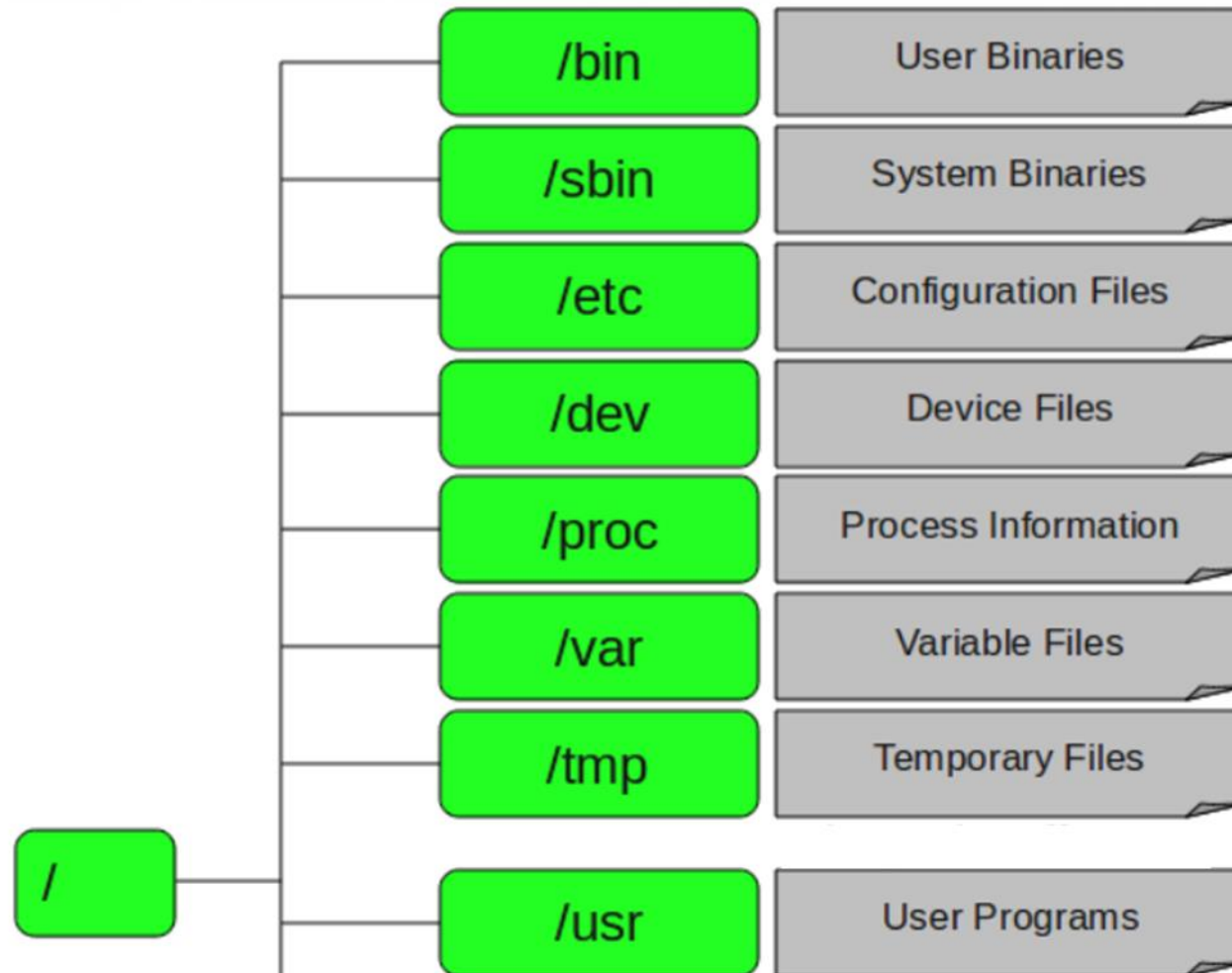
Source: <https://www.devopsschool.com/blog/linux-tutorials-root-file-systems-in-linux/>



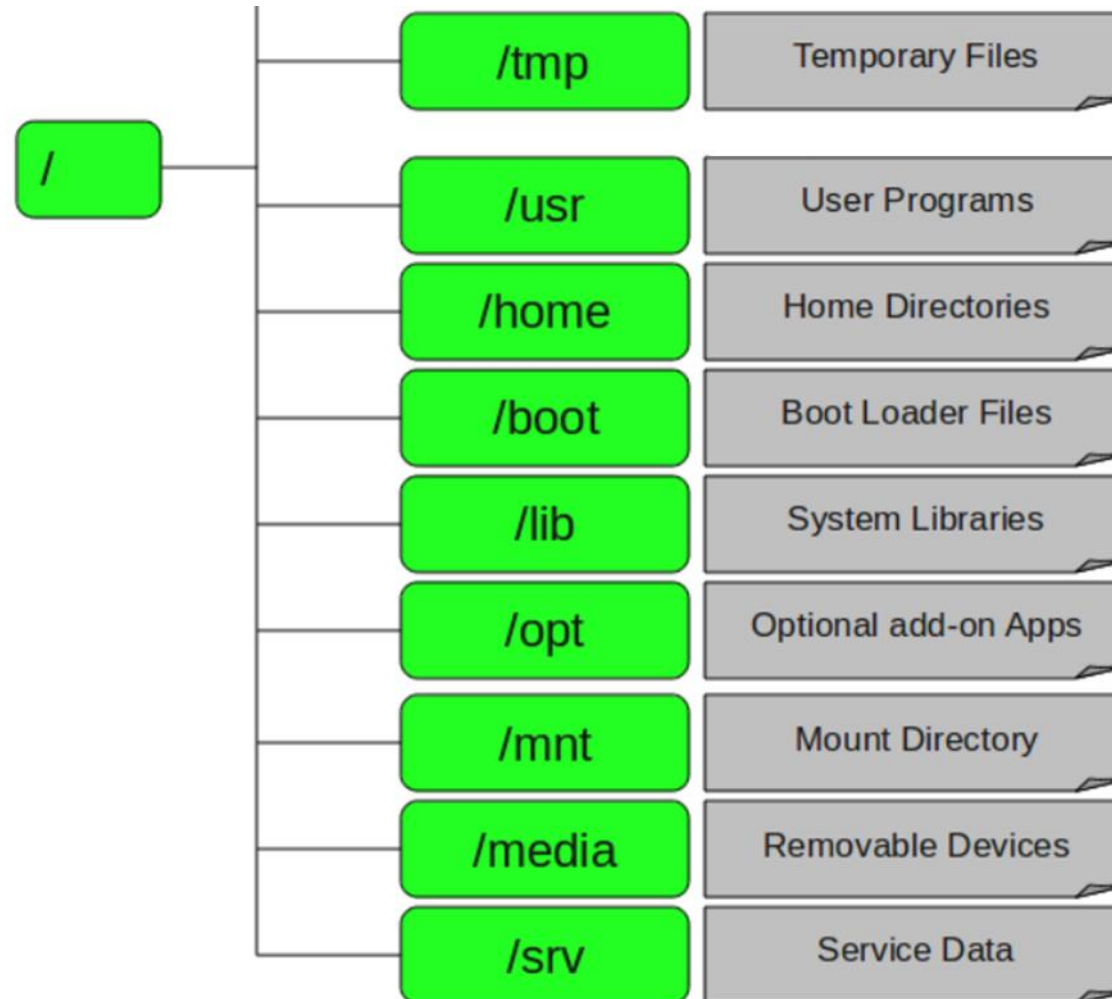
Linux File System

- The structure resembles an upside-down tree
- Directories (also know as folders) are collections of files and other directories.
- Every directory has a parent except for the root directory.
- Many directories have subdirectories.
- File representation holds good for everything including devices (/dev folder and files under that)
- Hidden files start with '.' in file name

Directories under '/' (root)



Directories under '/' (root)



CLI is used to access the file system

- CLI is very powerful
- It is case sensitive. For CLI, 'T' and 't' are not the same.
- In Windows we have C: , D: etc. In Unix or Linux CLI we have tree structured file system. There is no C: or D:
- In Windows you can 'Undo' something you performed. In Unix or Linux you cannot do that. So, be careful.

Absolute path and relative path

- Path points to a specific position in the directory tree.
- You can use either the relative path or absolute path

Absolute path

- The path is defined uniquely and does not depend on the current path

Examples:

/tmp is unique since /tmp is the absolute path (Linux or Unix)

C:/Users/Raja/Documents/MITWPU/DevOps (Windows)

Relative path

. (one dot or period) is the current working directory

.. (two dots or periods) is one directory up

(Try `cd .<enter>` at command prompt and check your current directory.
Try `cd ..<enter>` at command prompt and check your current directory.)

You can combine . and .. to navigate the file system hierarchy.

Relative path depends on your current position (or current directory) in the hierarchy.

Example: `./documents` or `./mitwpu/devops`

Summary

- Why do we learn 'Linux Basics' in DevOps elective?
- Introduction to OS, UNIX History & Evolution
- Introduction to Linux, Shells & File System
- Difference between UNIX and Linux
- Absolute & Relative Paths

Any Questions?



Thank You