ExcelShore®
Operational Excellence

Cybage®
Delivering Value. Scientifically.

Title : Introduction to UNIX

Presented by : Bamacharan Kundu

This document and all its contents contain information from Cybage Software Private Limited which may be privileged, confidential, or otherwise protected from disclosure. The information is intended to be for the addressee(s) only. Any unauthorized disclosure, copy, distribution, or use of the contents of this message is strictly prohibited.

Goal

After the conclusion of this section you should be able to

- log onto the Unix system

- understand the concept of current working directory

- traverse and manipulate the UNIX file system

- describe the role of the shell within the UNIX environment

- use simple commands to manipulate files (cd, ls, cp, rm, cat)

- use standard I/O, piping, and redirection from the UNIX shell

What is UNIX?

What is UNIX?

What is UNIX?

- A Computer Operating System
- A Software Development Environment
  - management of hardware resources
  - directory and file system
  - loading / execution / suspension of programs
- Originally written in assembler, later rewritten in C(allowing greater portability)

What is an operating System?

- Interacts with:
  - Applications
  - Users, through a command language interpreter

- OS offers Services:
  - Scheduling of multiple programs
  - Memory management
  - Access to hardware
  - Reports errors to applications

How it is build (Philosophy behind)?

- Make each program do one thing well
  - Reusable software tools: 1 tool = 1 function
- Expect the output of every program to become the input of another, yet unknown, program to combine simple tools to perform complex tasks
- Every thing seen as a file

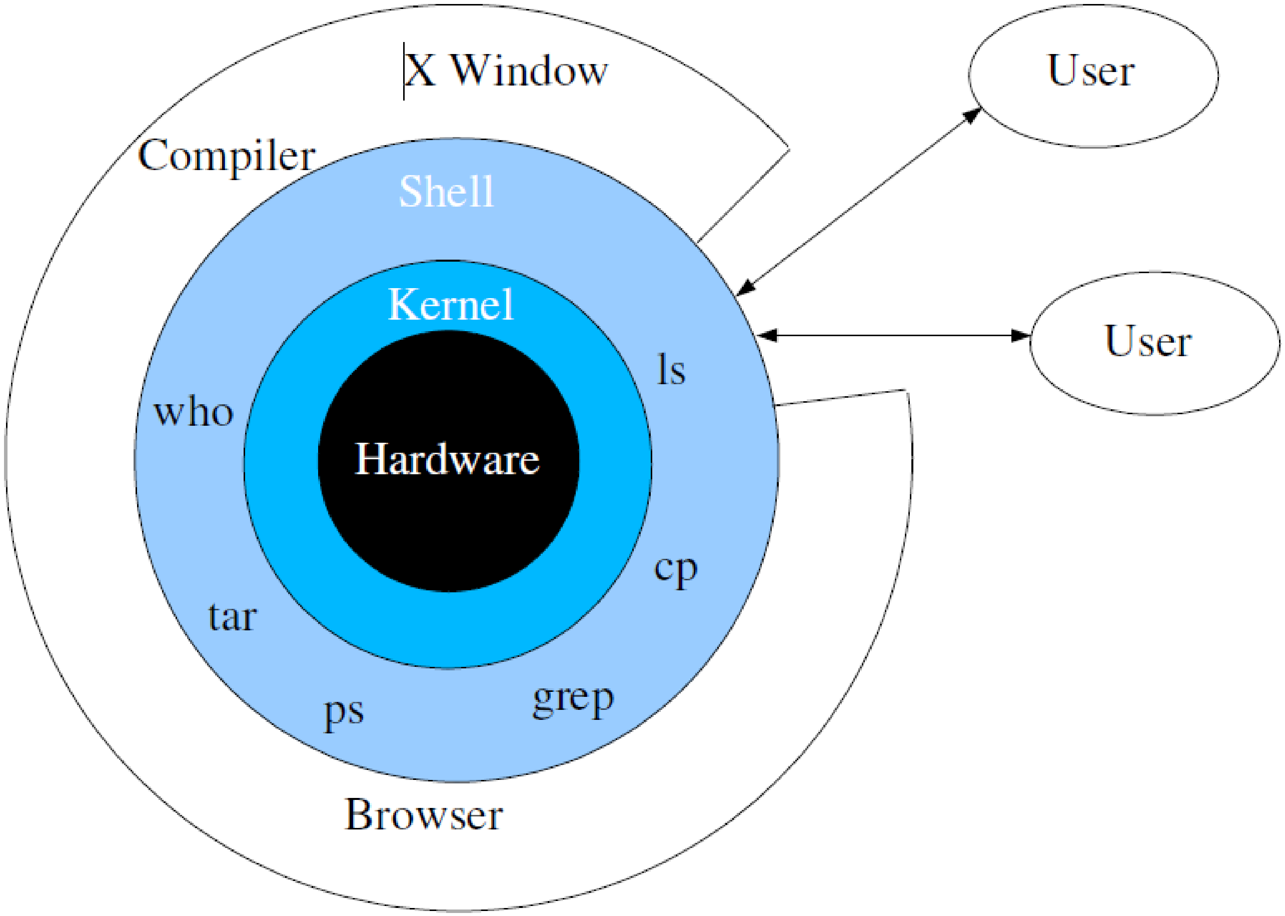
Why Use Unix?

Why UNIX?

Why Use Unix?

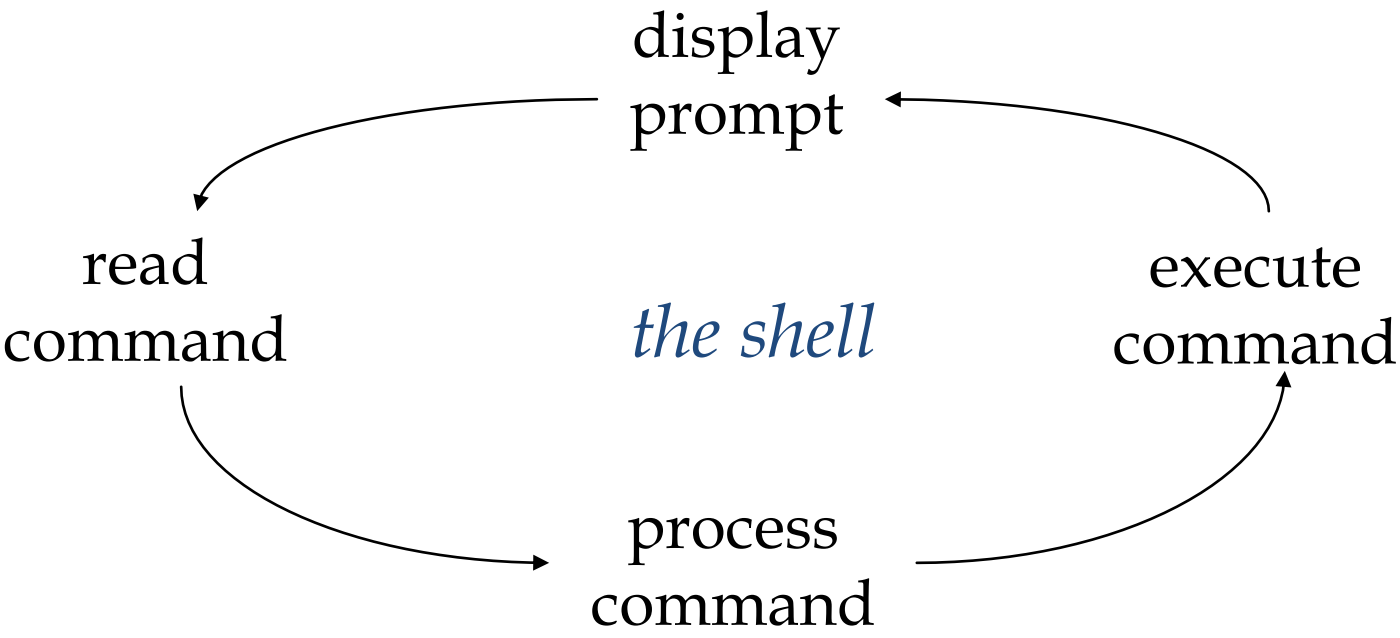
- Multi tasking/Multi-user
- Extensive set of utilities
- Built-in Networking
  - Portable (PCs, Mainframes, super-computers)
  - Graphical (with command line)

UNIX: Closer View



The Shell

- The UNIX user interface is called the *shell*.
- The shell does 4 jobs repeatedly:



display prompt

read command

the shell

execute command

process command

Session Startup

- Once you log in, your shell will be started and it will display a prompt.
- By default current working directory is home directory
- When the shell is started it looks in your home directory for some customization files.

You can change the shell prompt, your PATH, and a bunch of other things by creating customization files.

Interacting with the Shell

- The shell prints a prompt and waits for you to type in a command.
- The shell can deal with a couple of types of commands:
  - shell internals - commands that the shell handles directly.
  - External programs - the shell runs a program for you.

Basic Commands

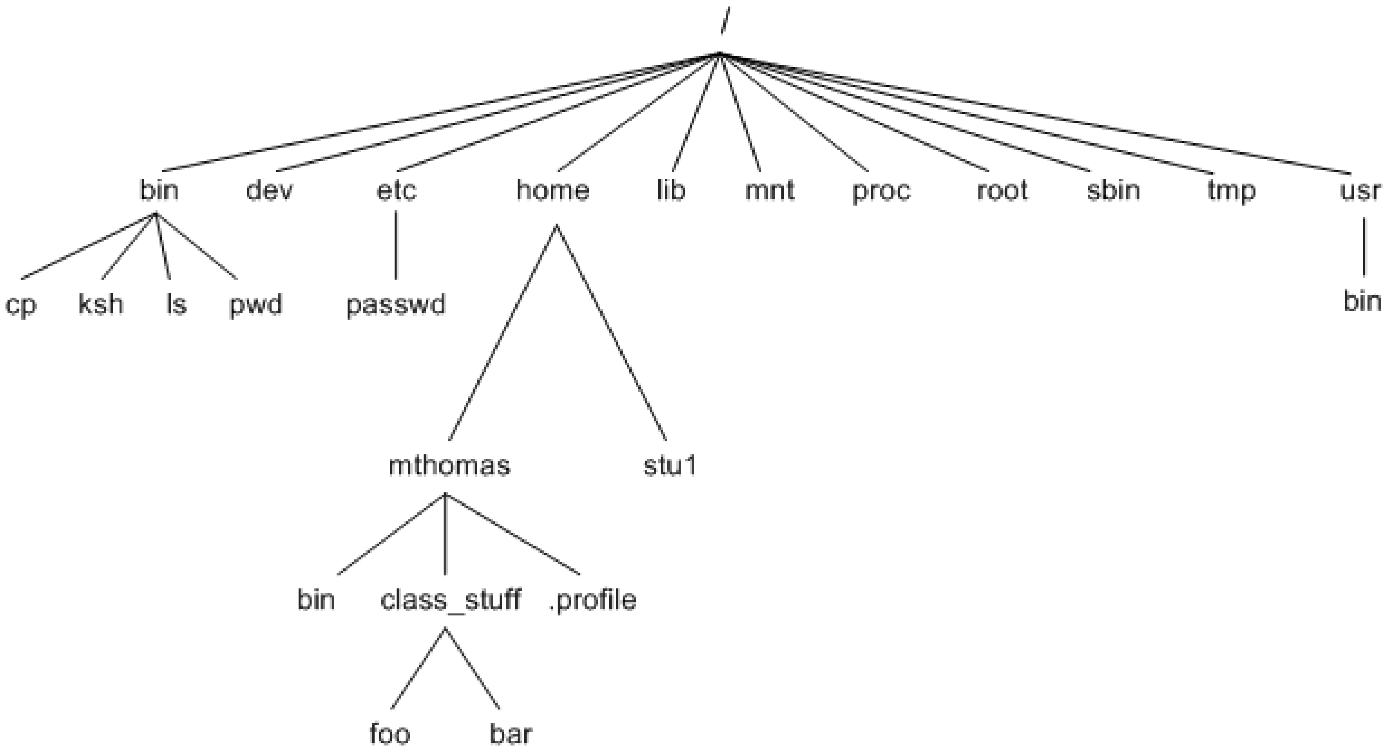
- Shell internal Commands
  - date
  - echo
  - clear
  - man
  - ps
  - top

- External commands
  - Commands created by a user (show example)

File System

- Organized as a tree
  - Each node is a directory
  - Each directory can contain other files or directories or both
  - Root: "/“
- Each file in a given directory must be unique
- UNIX is cAsE sEnSiTiVe

File System


/
bin
dev
etc
home
lib
mnt
proc
root
sbin
tmp
usr
cp
ksh
ls
pwd
passwd
bin
mthomas
stu1
bin
class_stuff
.profile
foo
bar

File System

- Files are referenced by name
  - absolute reference: beginning with "/“
  - relative reference: based on current directory
- Notation
- "..": parent directory
- ".": current directory
- "~": home directory

Commands

- ls (-l a t r R)
- pwd
- df
- cd
- cp
- mv
- rm

File attributes

- Every file has some attributes:
  - Access Times:
    - when the file was created
    - when the file was last changed
    - when the file was last read
- Size
- Owners (user and group)
- Permissions

File System Security

- Each file has three sets of permission bits
  - User
  - Group
  - Other
- Each set has three bits that represent:
  - Read
  - Write
  - execute

Other filesystem and file commands

- **mkdir** make directory
- **rmdir** Remove directory
- **touch** change file timestamp
- **cat** concatenate files and print out to terminal

Thank You!