

Computer Systems and Architecture

Computer Systems

Stephen Pauwels

Academic Year 2020-2021

Overview of the Semester

UNIX

Introduction

Regular
Expressions

Scripting

Data Representation

Integers,
Fixed point,
Floating
point, IEEE-
754, ASCII,
Unicode

Assembler - MIPS

Basics,
Memory

Stacks,
Subroutines

Advanced

Computer Systems and Architecture

Introduction to UNIX

Stephen Pauwels

Academic Year 2020-2021

Outline

- What is UNIX?
- Getting started
 - UNIX Shell
 - Files and Processes
 - Basic UNIX Commands
- Input/Output



UNIX

- Operating System (OS)
- Servers, desktops, laptops, tablets, smartphones



- Different types
 - Sun Solaris
 - FreeBSD
 - GNU/Linux
 - MacOS



UNIX

- Three parts

Shell

Interface between user and kernel
Command Line Interpreter (CLI)

Files and
Processes

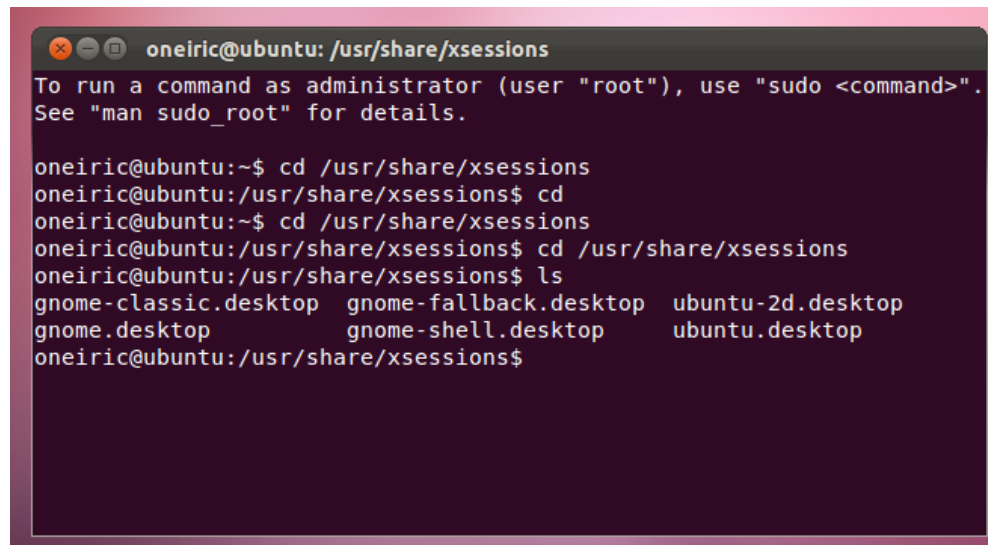
Everything in UNIX is a file or a
process (programs are processes)

Kernel

Central component, manages the
computer's hardware

UNIX Shell

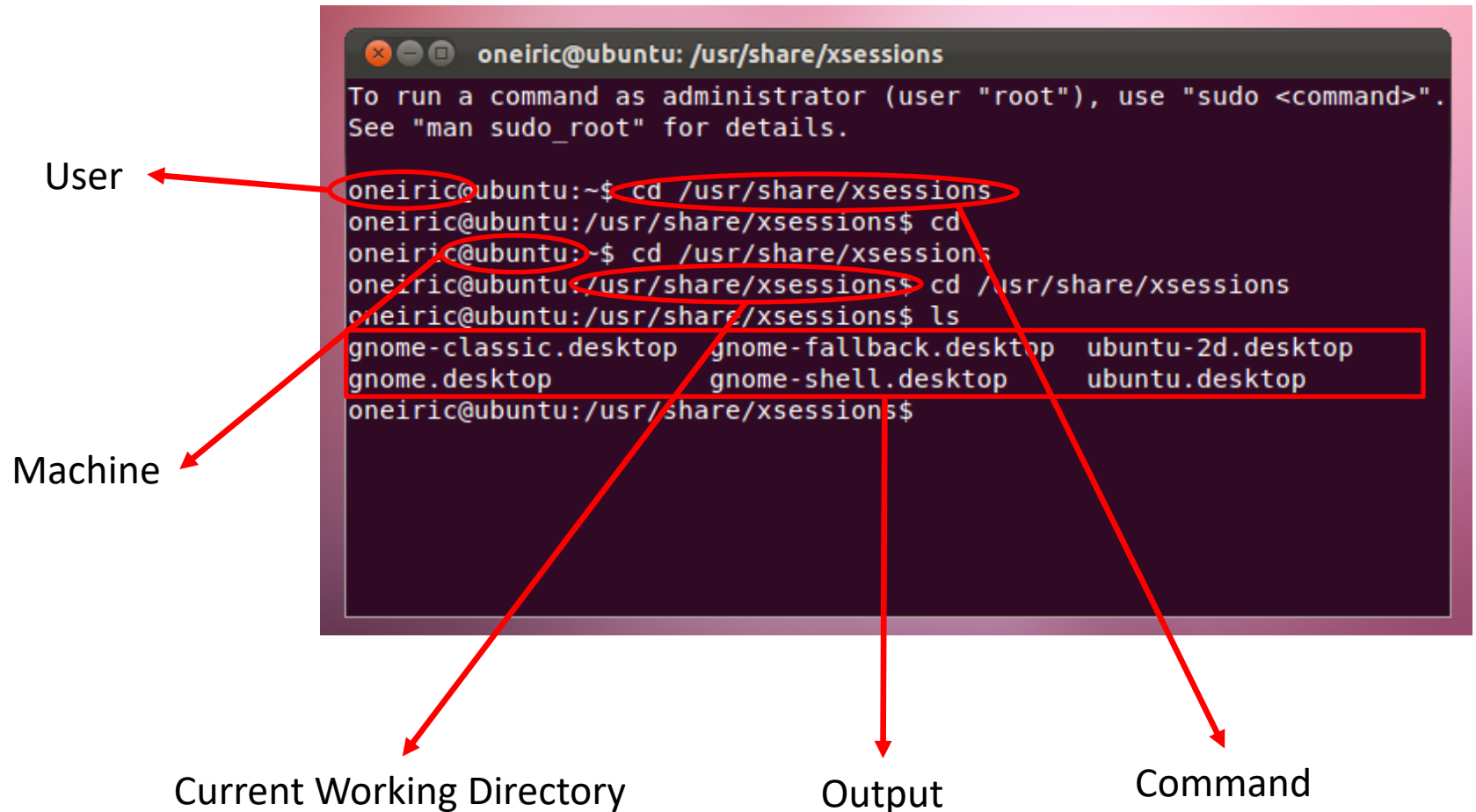
- Interprets commands
 - Commands are themselves programs
- Filename and command completion (use the [tab] key)
- History (use the ↑ and ↓ keys)
- Current working directory



```
oneiric@ubuntu: /usr/share/xsessions
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

oneiric@ubuntu:~$ cd /usr/share/xsessions
oneiric@ubuntu:/usr/share/xsessions$ cd
oneiric@ubuntu:~$ cd /usr/share/xsessions
oneiric@ubuntu:/usr/share/xsessions$ cd /usr/share/xsessions
oneiric@ubuntu:/usr/share/xsessions$ ls
gnome-classic.desktop  gnome-fallback.desktop  ubuntu-2d.desktop
gnome.desktop          gnome-shell.desktop     ubuntu.desktop
oneiric@ubuntu:/usr/share/xsessions$
```

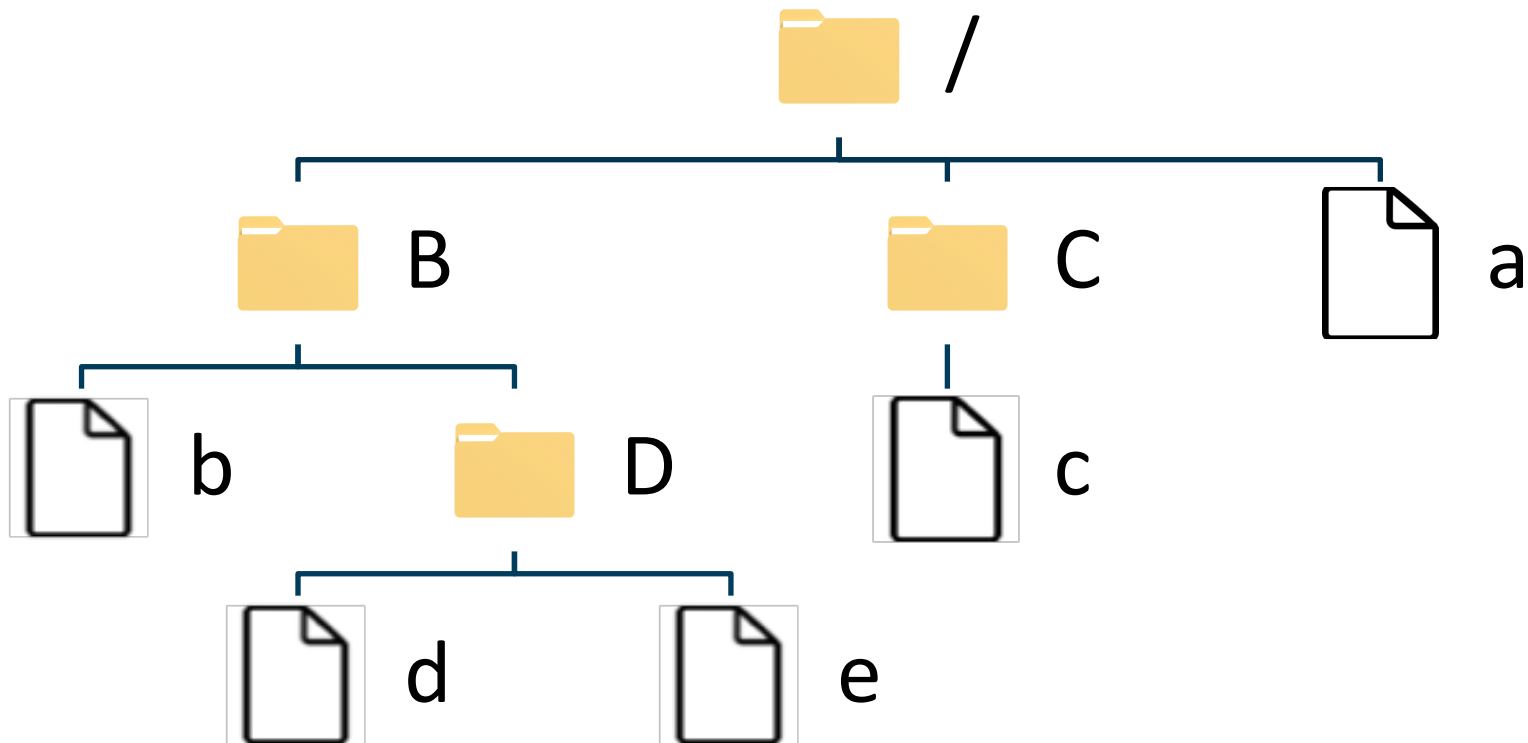
UNIX Shell: The terminal



Files and Processes

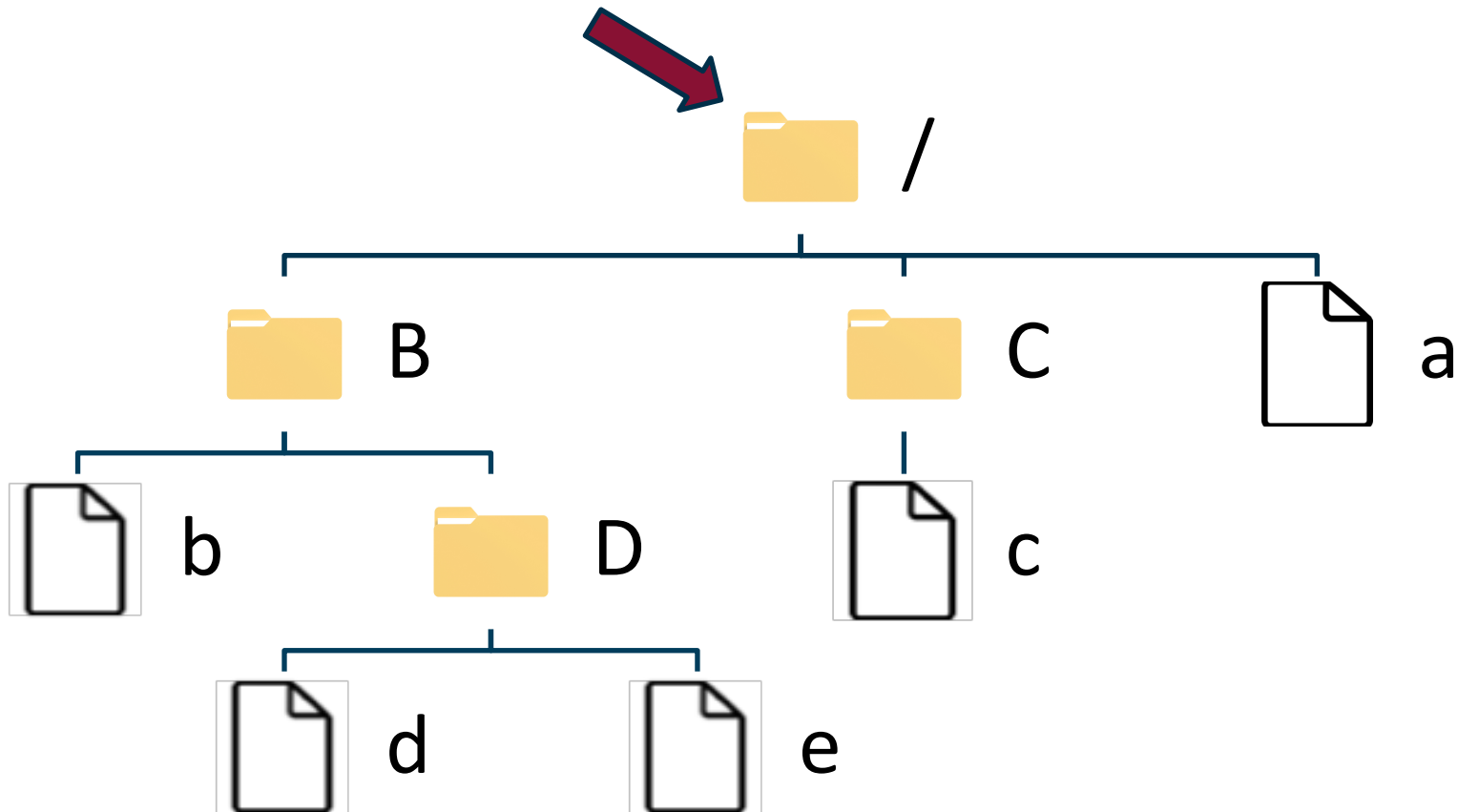
- Everything in UNIX is a file or a process
 - A process is an executing program (with a unique PID)
 - A file is a collection of data
- Directory structure
 - Root: /
 - Home dir: ~/
 - Current dir: ./
 - Parent dir: ../
 - Absolute path: /home/p10/p101234/oefeningen.html
 - Relative path: ./oefeningen.tml

Relative Vs Absolute Paths



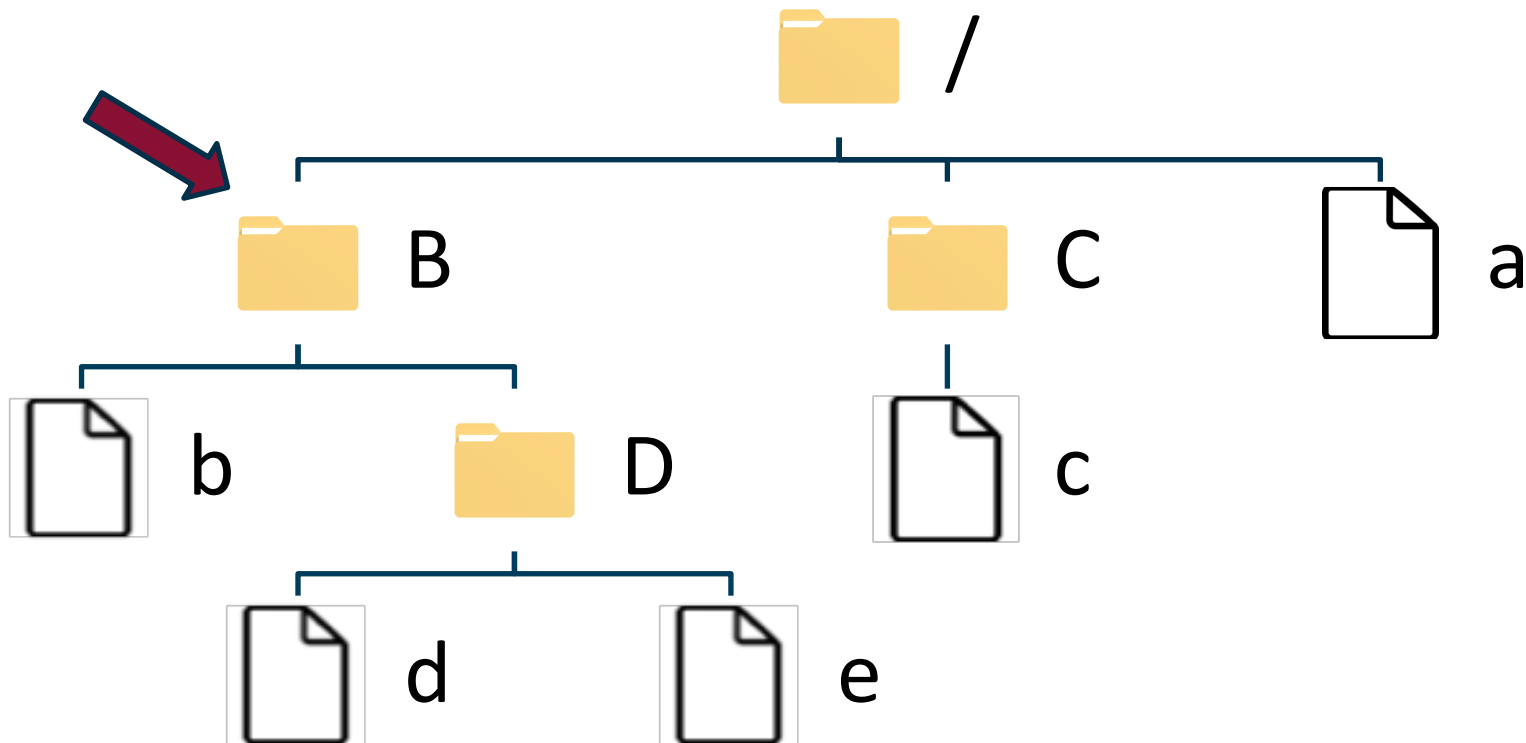
Relative Vs Absolute Paths

Absolute Path: /



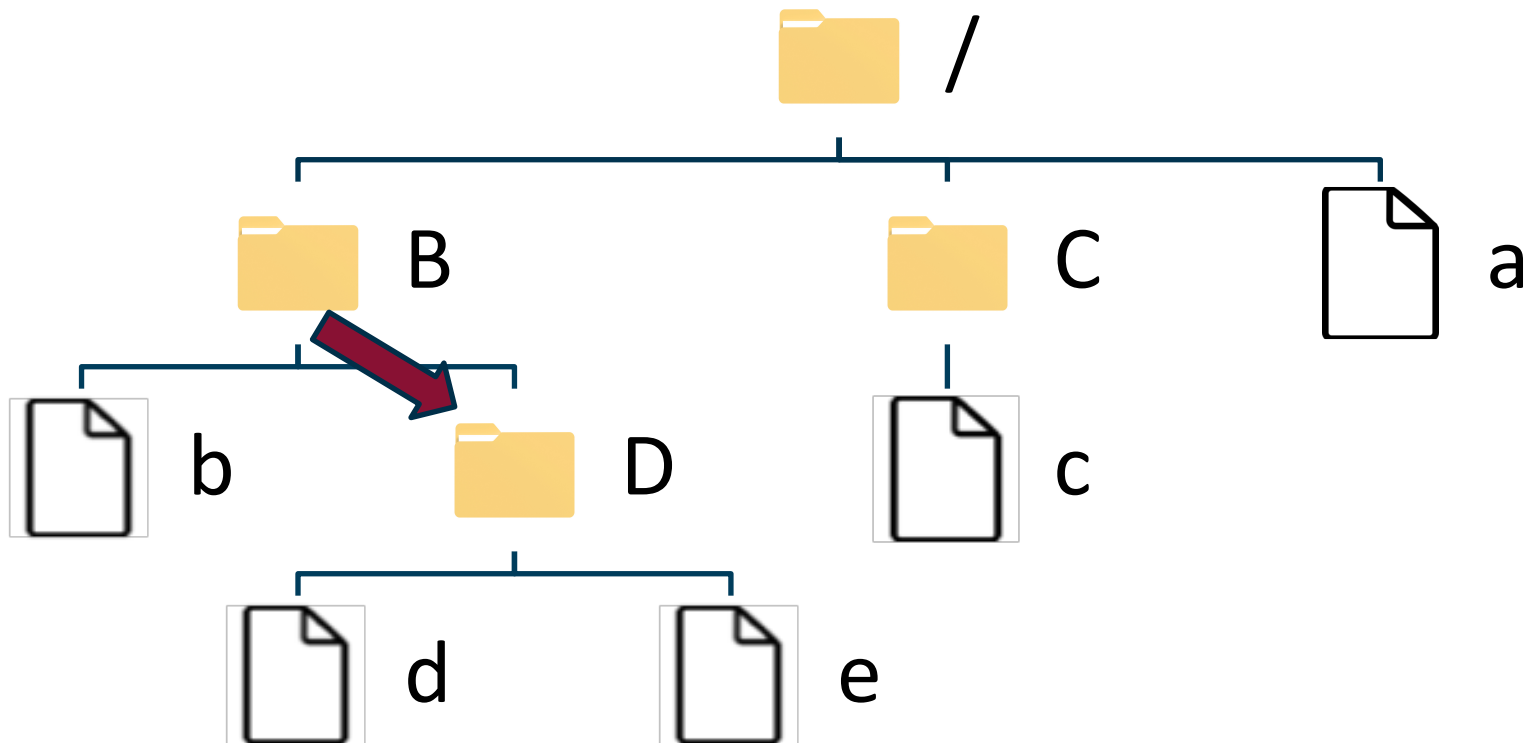
Relative Vs Absolute Paths

Absolute Path: /B/



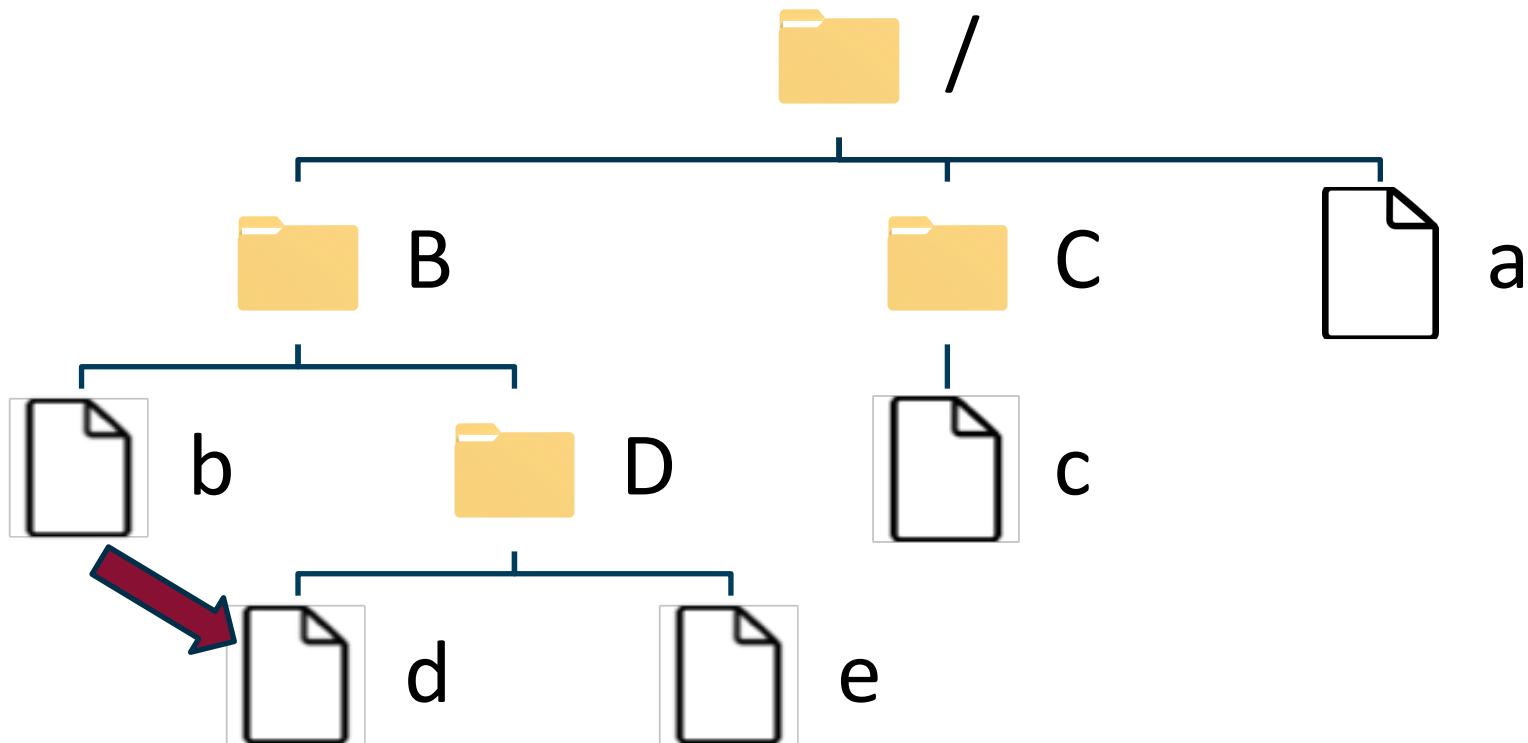
Relative Vs Absolute Paths

Absolute Path: /B/D



Relative Vs Absolute Paths

Absolute Path: /B/D/d

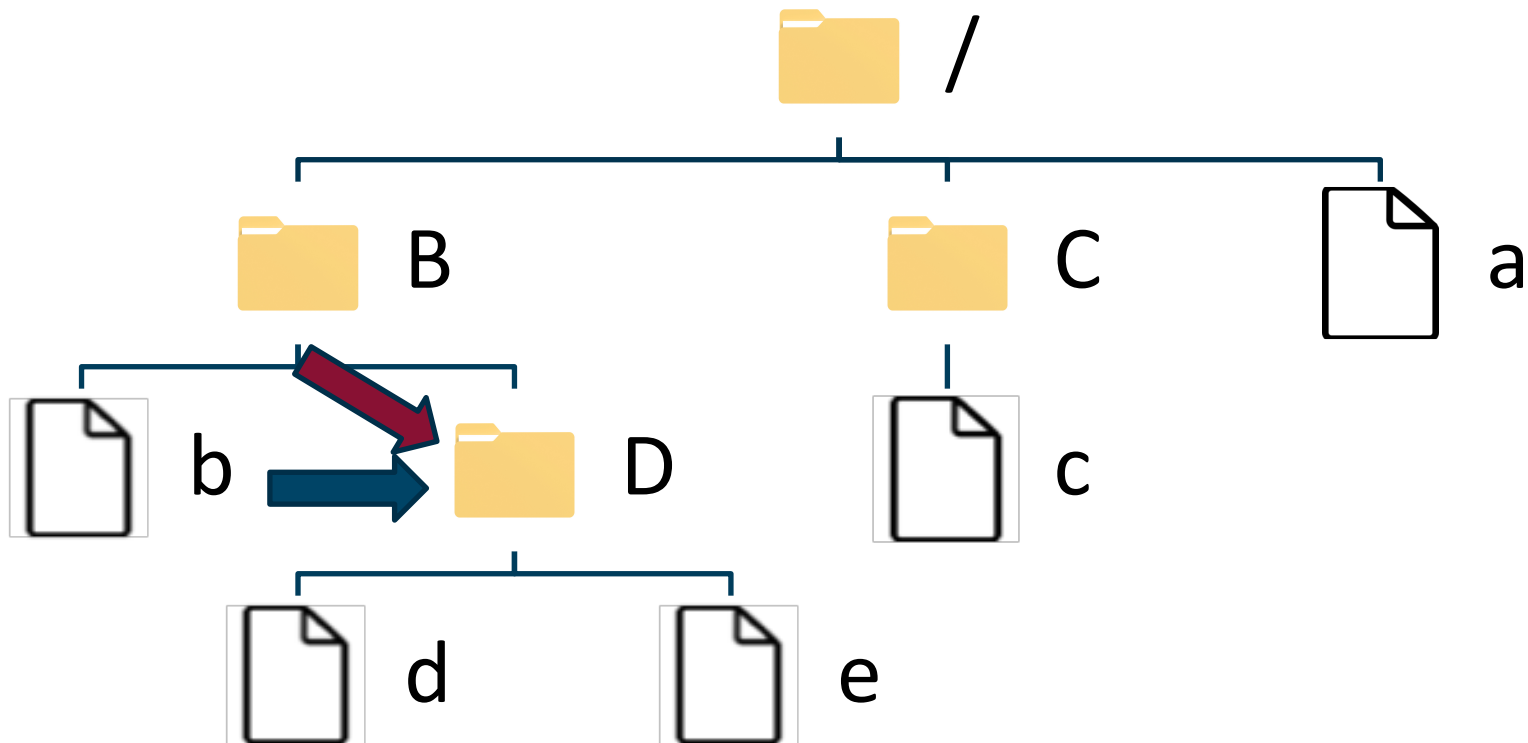


Relative Vs Absolute Paths

Relative Path: . /



Current Working Directory:

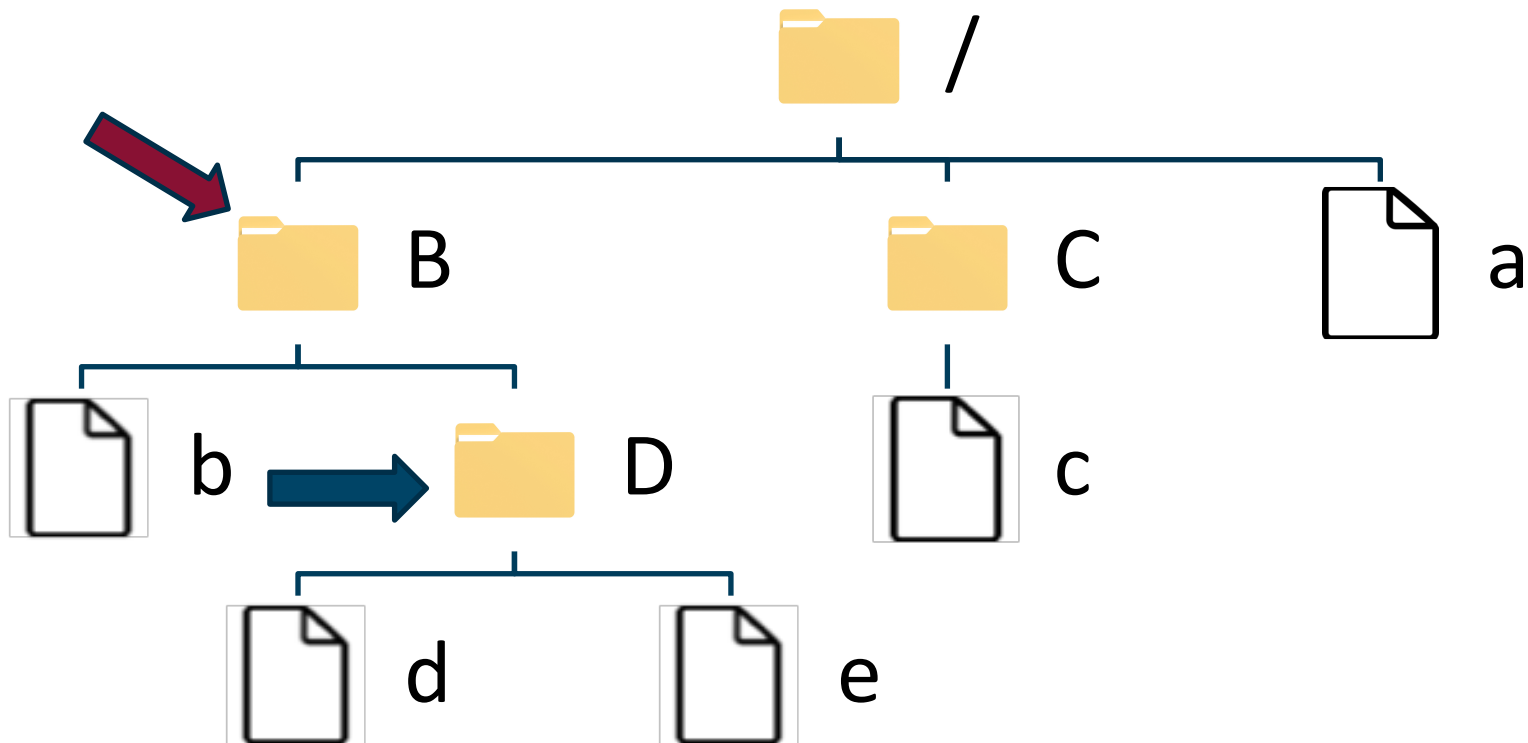


Relative Vs Absolute Paths

Relative Path: .. /



Current Working Directory:

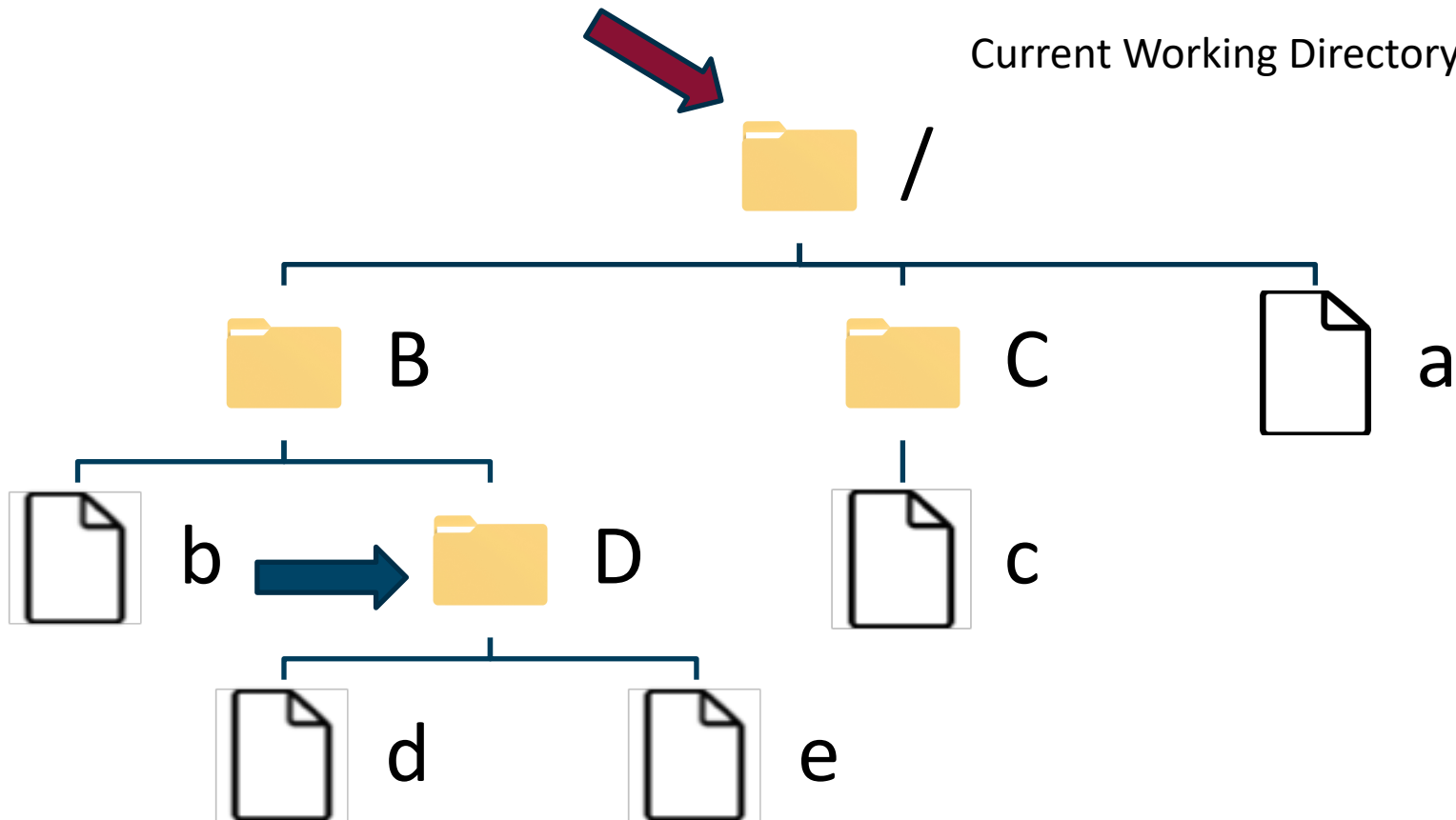


Relative Vs Absolute Paths

Relative Path: .. /.. /



Current Working Directory:

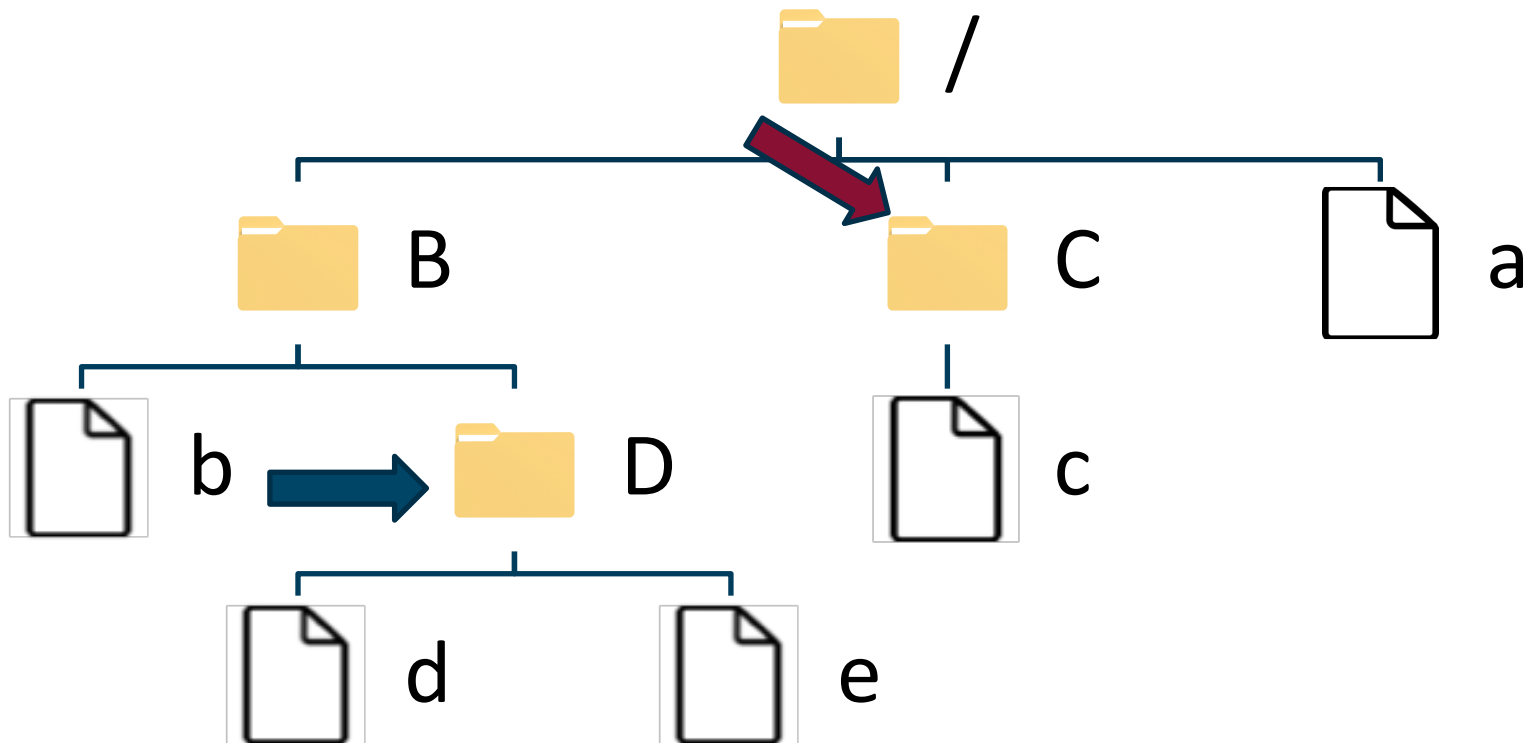


Relative Vs Absolute Paths

Relative Path: `../.. /C`



Current Working Directory:

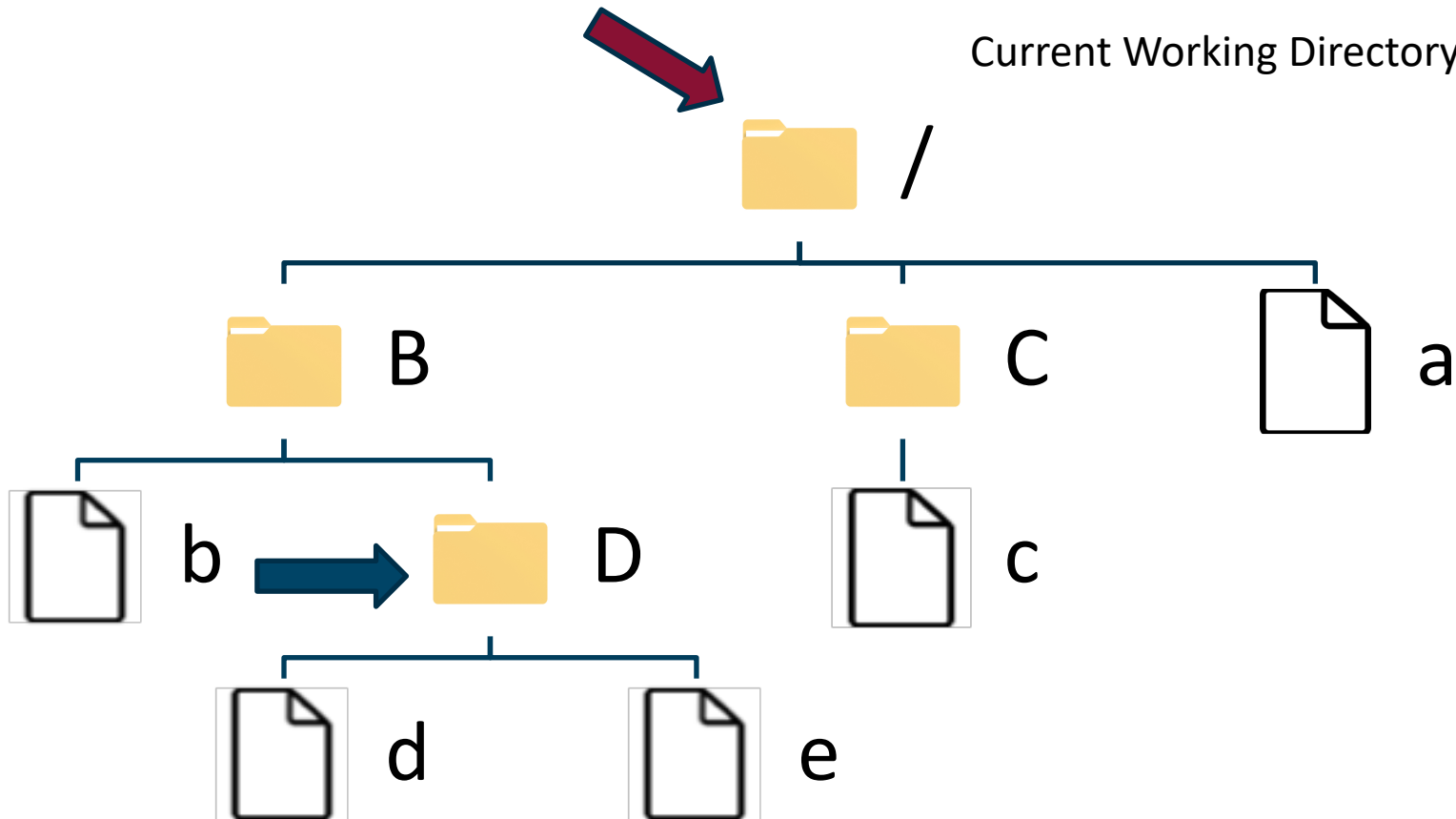


Relative Vs Absolute Paths

Relative Path: `../.. /C/..`



Current Working Directory:

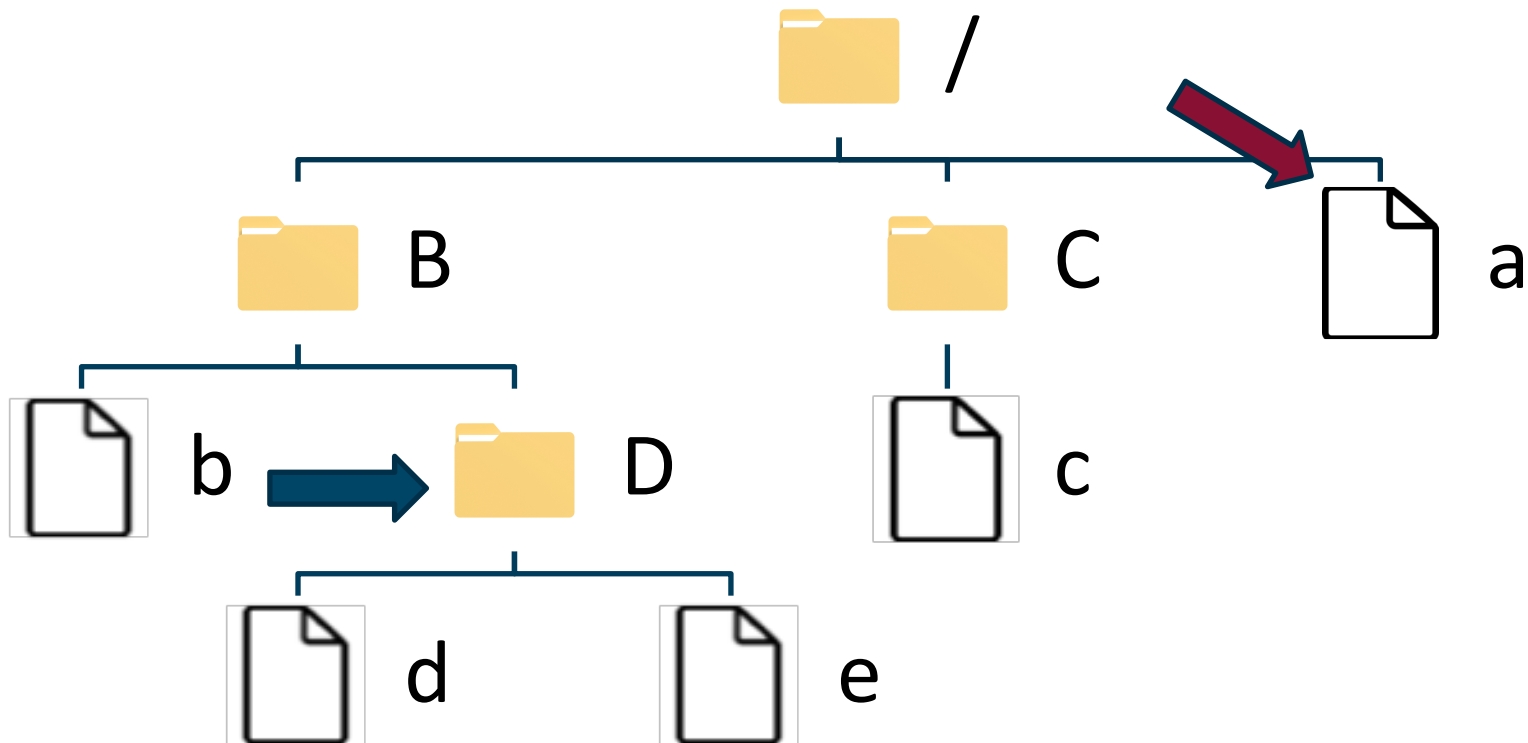


Relative Vs Absolute Paths

Relative Path: `../.. /C/.. /a`



Current Working Directory:

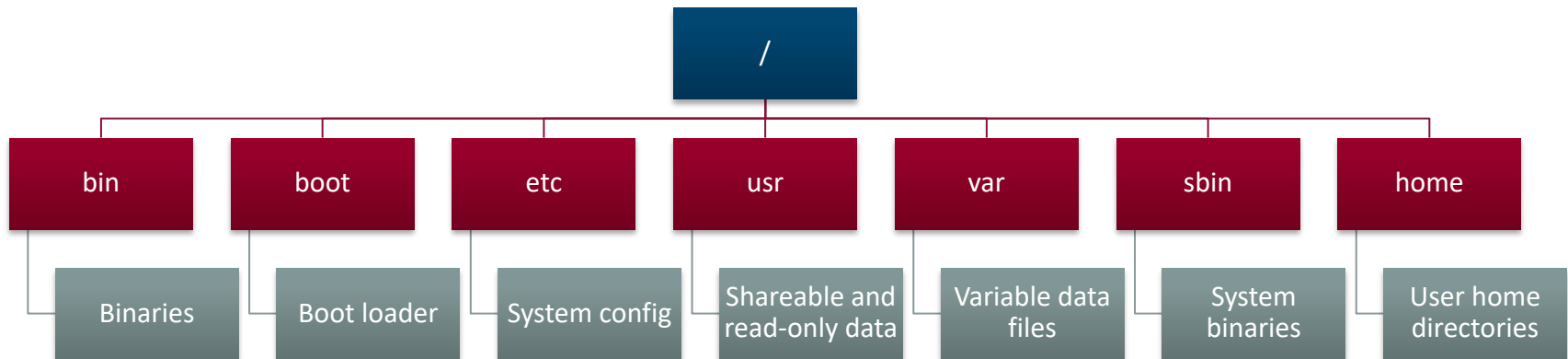


File Permissions

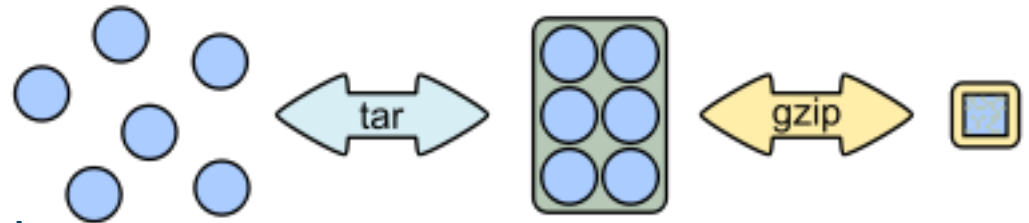
- Users
 - Unique username
 - Member of one or more groups
 - `/etc/passwd` and `/etc/group`
- Files
 - Owner (and associated group)
 - Set of permission flags: `r` (+4), `w` (+2), `x` (+1) for owner, group, other
 - Change with `chmod`, `chown`, `chgrp`
- Examples:
 - `chmod 755 file` – Owner can do everything, group and others can read/execute
 - `chmod 777 file` – Everyone can read, write and execute
 - `chmod 600 file` – Owner can read/write, others can't do anything
 - `chmod +x file` – Add execute permission for everyone



Directory Structure



Archiving



- Tar: Uncompressed
 - Create: `tar -cvf tarball.tar files`
 - Extract: `tar -xvf tarball.tar`
 - List files: `tar -tf tarball.tar`
 - Update files: `tar -uf tarball.tar files`
- Gzip: Compression
 - Create: `gzip tarball.tar`
 - Extract: `gunzip tarball.tar.gz`
- tar.gz or tgz?
 - Tar + Compression
 - Create: `tar -cvzf tarball.tgz files`
 - Extract: `tar -xvzf tarball.tgz`
 - List files: `tar -tzf tarball.tgz`

Standard in- and output

- Processes write to the standard output, and take their input from the standard input
- Keyboard and terminal
- Standard error (terminal)
- Redirection is possible
 - `>`, `>>`, `<`, `2>`

Redirection

<code>ls -alrF > listing.txt</code>	store ls output in listing.txt
<code>sort < listing.txt</code>	feed listing.txt to sort program
<code>echo HOI >> listing.txt</code>	append string HOI to listing.txt
<code>echo hello > /dev/null</code>	suppress output
<code>who 2> errors.txt</code>	store errors in file



Pipes

- Pipes redirect output from one process to the following one
- Without pipes:
who > names.txt
sort < names.txt
- With pipes:
who | sort



Exercises

- Course webpage
 - <http://msdl.cs.mcgill.ca/people/hv/teaching/ComputerSystemsArchitecture/#CS1>



Basic UNIX Commands - Browsing

<code>ls</code>	list files and directories
<code>ls -a</code>	list all files and directories
<code>ls f*</code>	list all files starting with 'f' (*: wildcard)
<code>mkdir <i>directory</i></code>	make a directory named <i>directory</i>
<code>cd <i>directory</i></code>	change to directory <i>directory</i>
<code>cd</code>	change to home directory
<code>cd ~</code>	change to home directory
<code>cd ..</code>	change to parent directory
<code>pwd</code>	display the path of the current directory
<code>find</code>	search through directory tree



Basic UNIX Commands - Files

<code>cp file1 file2</code>	copy file1 and call it file2
<code>mv file1 file2</code>	move or rename file1 to file2
<code>rm file</code>	remove a file
<code>rmdir directory</code>	remove a directory (directory must be empty!)
<code>cat file</code>	display a file
<code>less file</code>	display a file a page at a time
<code>head file</code>	display the first few lines of a file
<code>tail file</code>	display the last few lines of a file
<code>grep 'keyword' file</code>	search a file for keyword
<code>wc file</code>	count number of metrics in file
<code>ln -s to from</code>	make softlink from to
<code>uniq file</code>	report or filter out repeated lines in a file
<code>file file</code>	show the file type of a file
<code>du file</code>	show the disk size of a file or directory



Basic UNIX Commands – Varia

<code>man <i>command</i></code>	display the manual pages for a command
<code>date</code>	display date and time
<code>who</code>	info on all currently logged on users
<code>whoami</code>	info about yourself
<code>echo <i>hello world!</i></code>	display characters in the terminal
<code>sort <i>text</i></code>	sort its input
<code>finger</code>	lookup user info



Processes

- Jobs are connected to terminal which started them
- Foreground or background
- kill current job: Ctrl+C

<code>kill [-9] <i>process_id</i></code>	Kill job with pid <code>process_id</code>
<code>pkill <i>process_name</i></code>	Kill job with name <code>process_name</code>
<code><i>command</i> &</code>	Run command in the background
<code>ps [-ef]</code>	Display process info
<code>top</code>	Display process info interactively
<code>jobs</code>	Display users's jobs
<code>fg <i>pid</i></code>	Bring process to the foreground
<code>bg <i>pid</i></code>	Bring process to the background

Basic UNIX commands – Network

<code>ssh username@host</code>	Login with a remote shell
<code>scp from username@host:to</code>	Copy files over network
<code>wget http://url/file.jpg</code>	Download files from the web
<code>ping host</code>	Send a small packet to a host (test reachability)



Useful UNIX commands – Screen

- When connected to a remote server, running processes will be killed when connection is lost or terminated.
- The command screen will start a virtual terminal, this screen is not linked to a particular terminal
- Screen session has the same behavior as a normal terminal

<u>Outside a screen session</u>	
screen	Start a new screen session
screen -S <i>name</i>	Start a new screen session with given name
screen -r	Resume a screen session
screen -ls	List all available screen sessions
screen -X -S <i>sessionid</i> stuff <i>text</i>	Send text as input to screen with given id
screen -X -S <i>sessionid</i> quit	Removes a screen session
<u>Inside a screen session</u>	
Ctrl+a d	Detach screen from the terminal

