LINUX COMMAND SHELL

WEEK 1 – 08/28/2019

LET'S PLAY

- https://create.kahoot.it/share/lecture-2/a0c39a05-1c8f-40f2-a894-b56a691a851f

BUILDING BLOCKS FOR SOFTWARE DEVELOPMENT

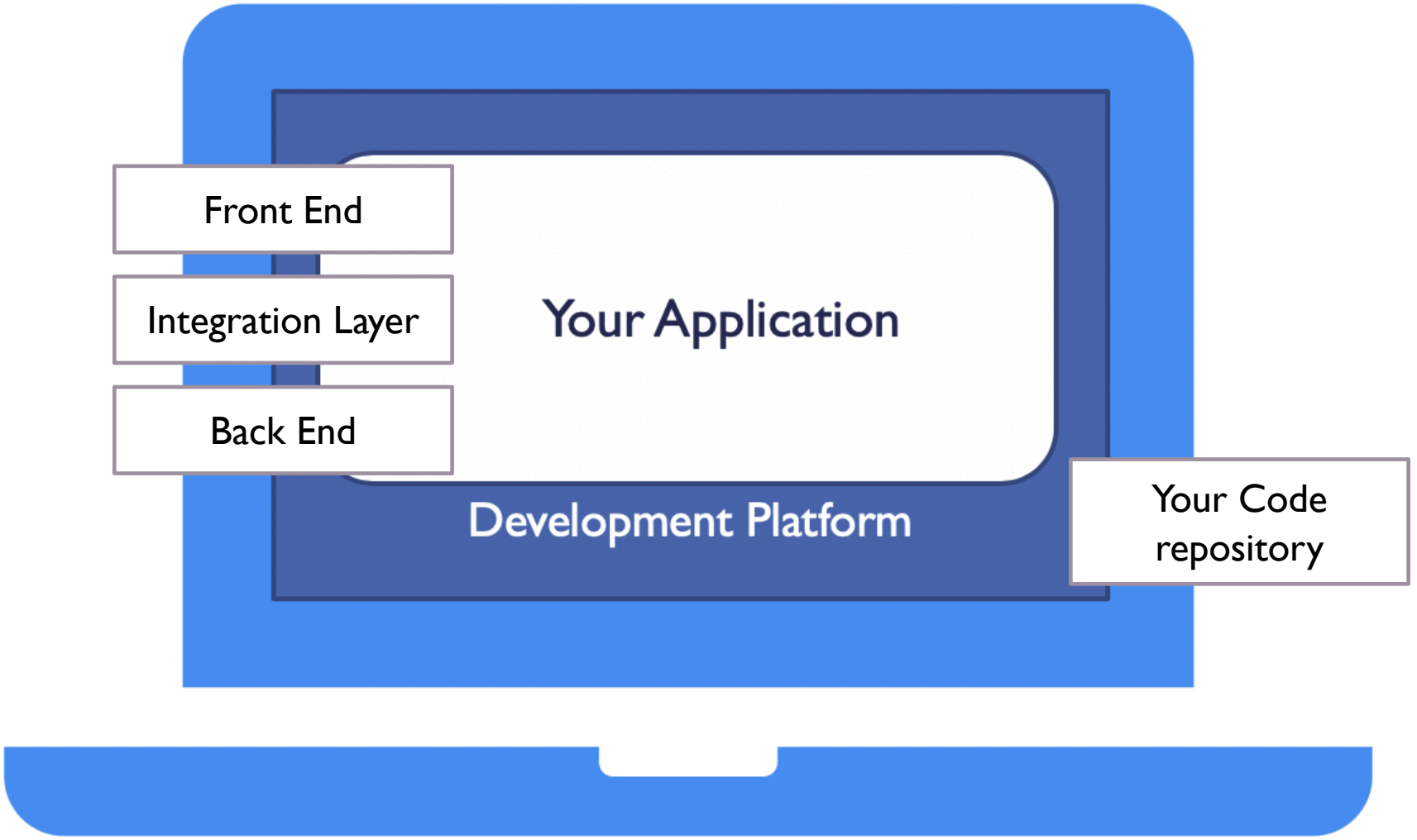
- Operating system
- A framework
- Programming languages
- Database
- Repository
- Web services

PLATFORM AND ARCHITECTURAL LAYERS

- Platform
  - Web Site, Mobile App, Embedded System
  - Hosted where? Local, cloud
- Layers
  - User Interface
  - Back End Database
  - Integration Layer

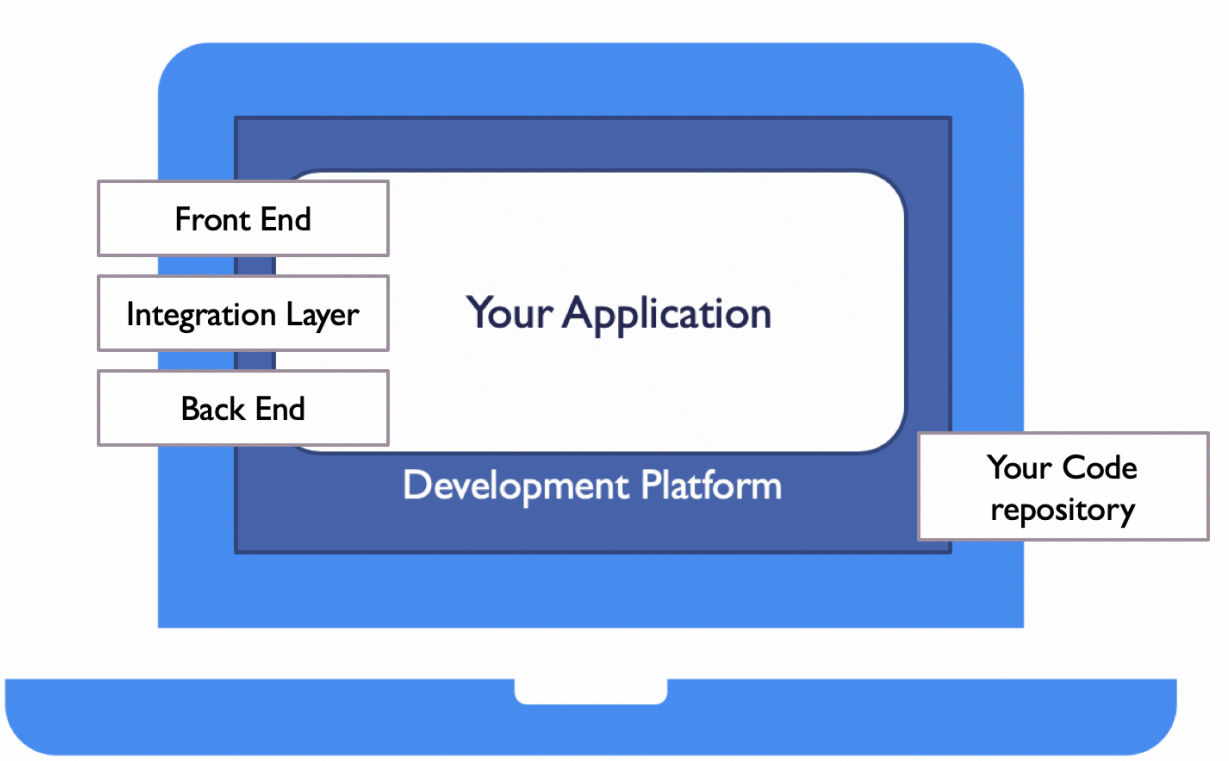
DIFFERENT ARCHITECTURES FOR HOSTING APPLICATION

Everything resides on your computer

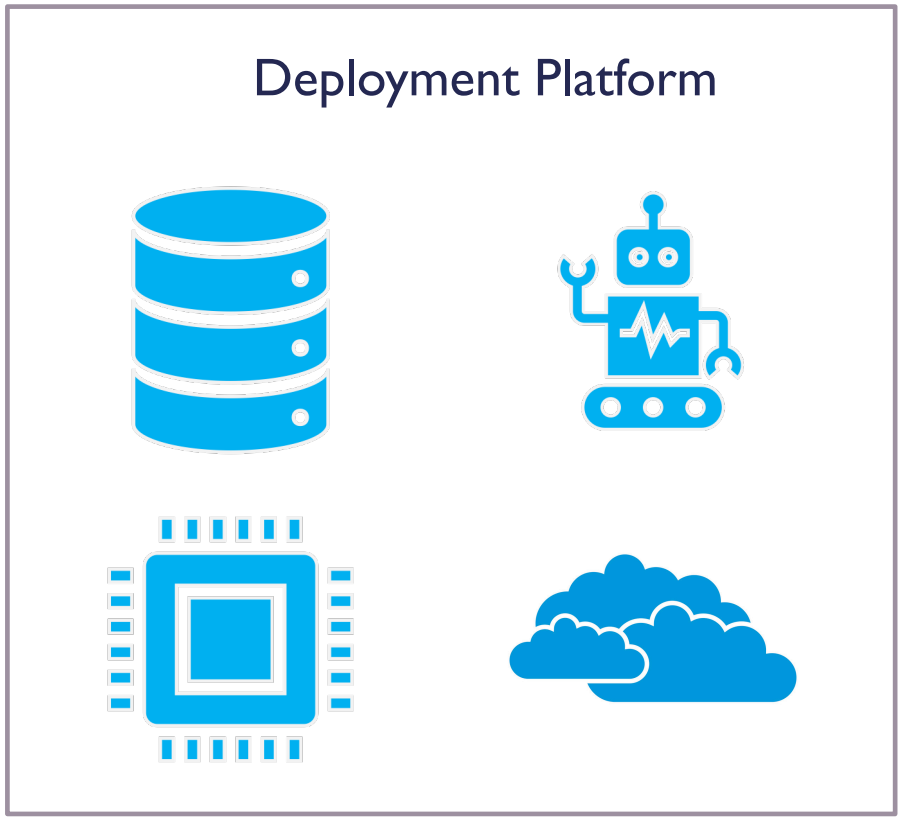


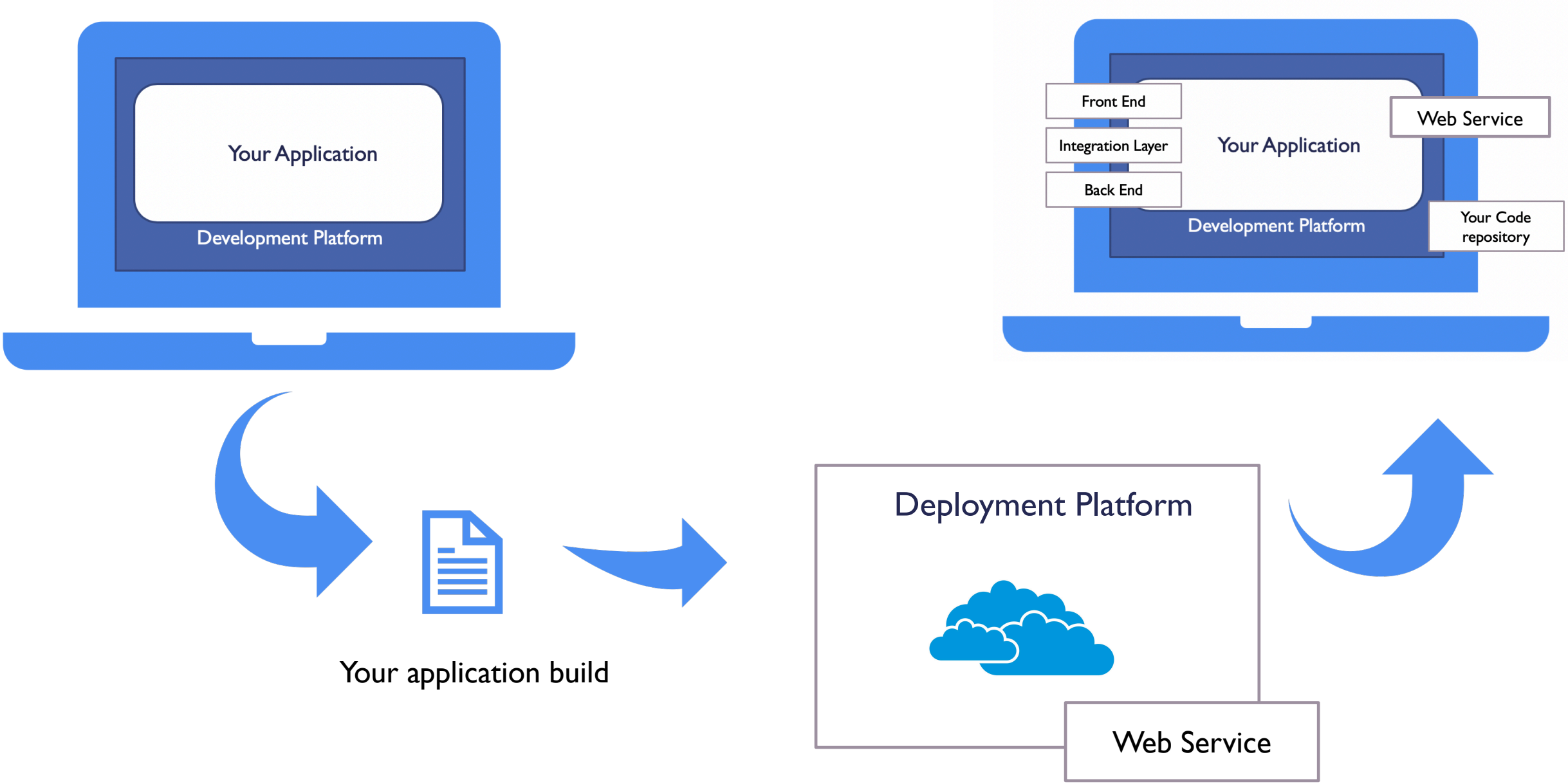
What makes up your application?

Deploy your Application on an external resource

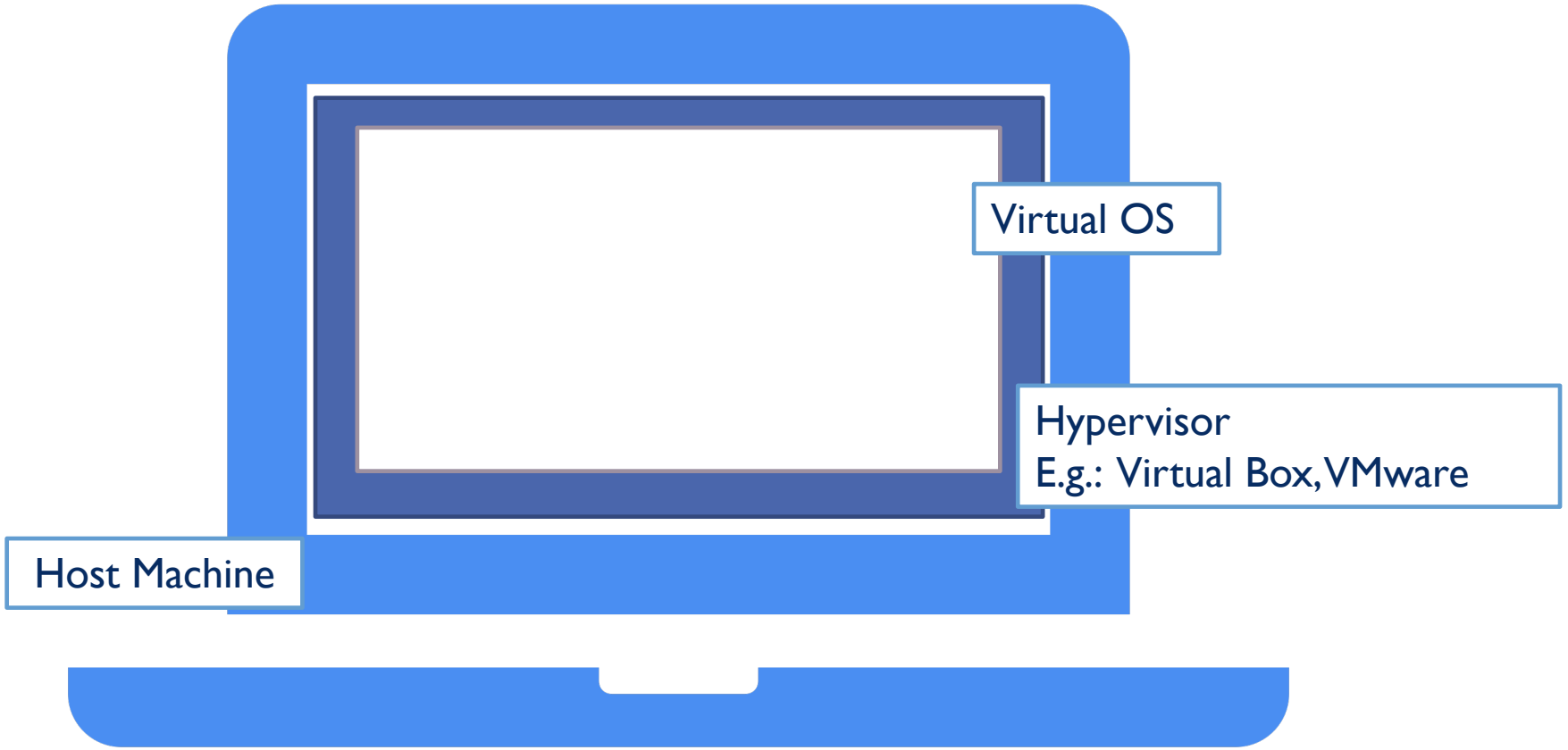


Your application build




Your Application
Development Platform
Front End
Integration Layer
Back End
Your Application
Web Service
Development Platform
Your Code repository
Your application build
Deployment Platform
Web Service

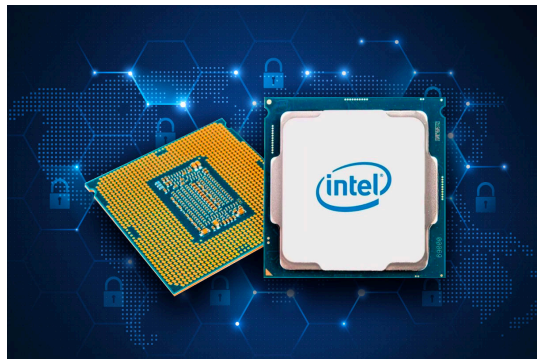
VIRTUAL MACHINES (VM)



Can I have more than 1 virtual Machine on a single host?

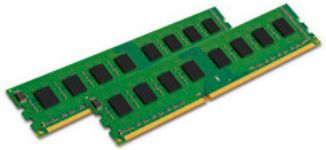
COMPUTER ARCHITECTURE

CPU



shorturl.at/emMQ9

RAM



shorturl.at/eNT06

I/O devices

shorturl.at/dkpy4

Storage devices

shorturl.at/lwKOP

shorturl.at/adsEW

OPERATING SYSTEM

Hardware Components are only able to control themselves

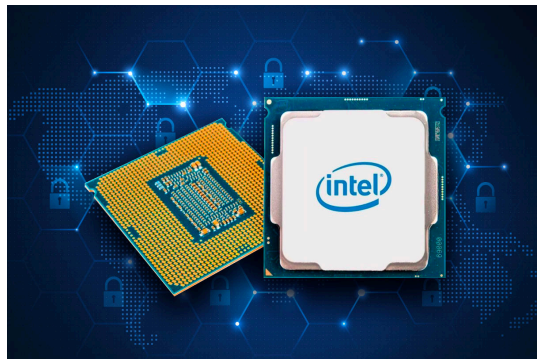
Components do not know how to interact with each other

How do we get the components to operate together?

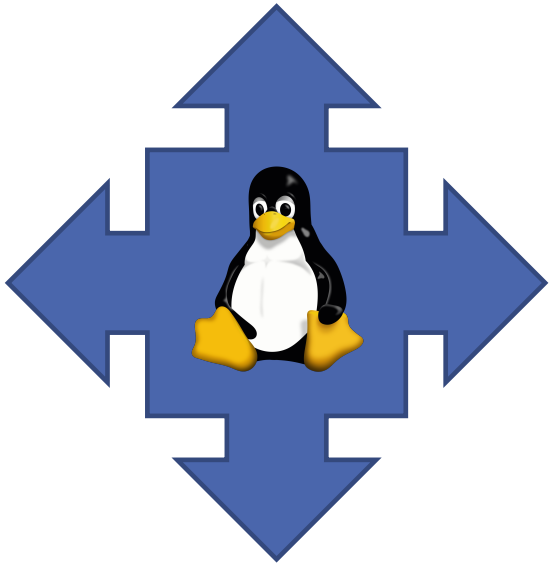
Make way, the Operating System is here!

COMPUTER ARCHITECTURE

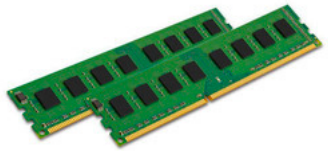
CPU



shorturl.at/emMQ9



RAM



shorturl.at/eNT06

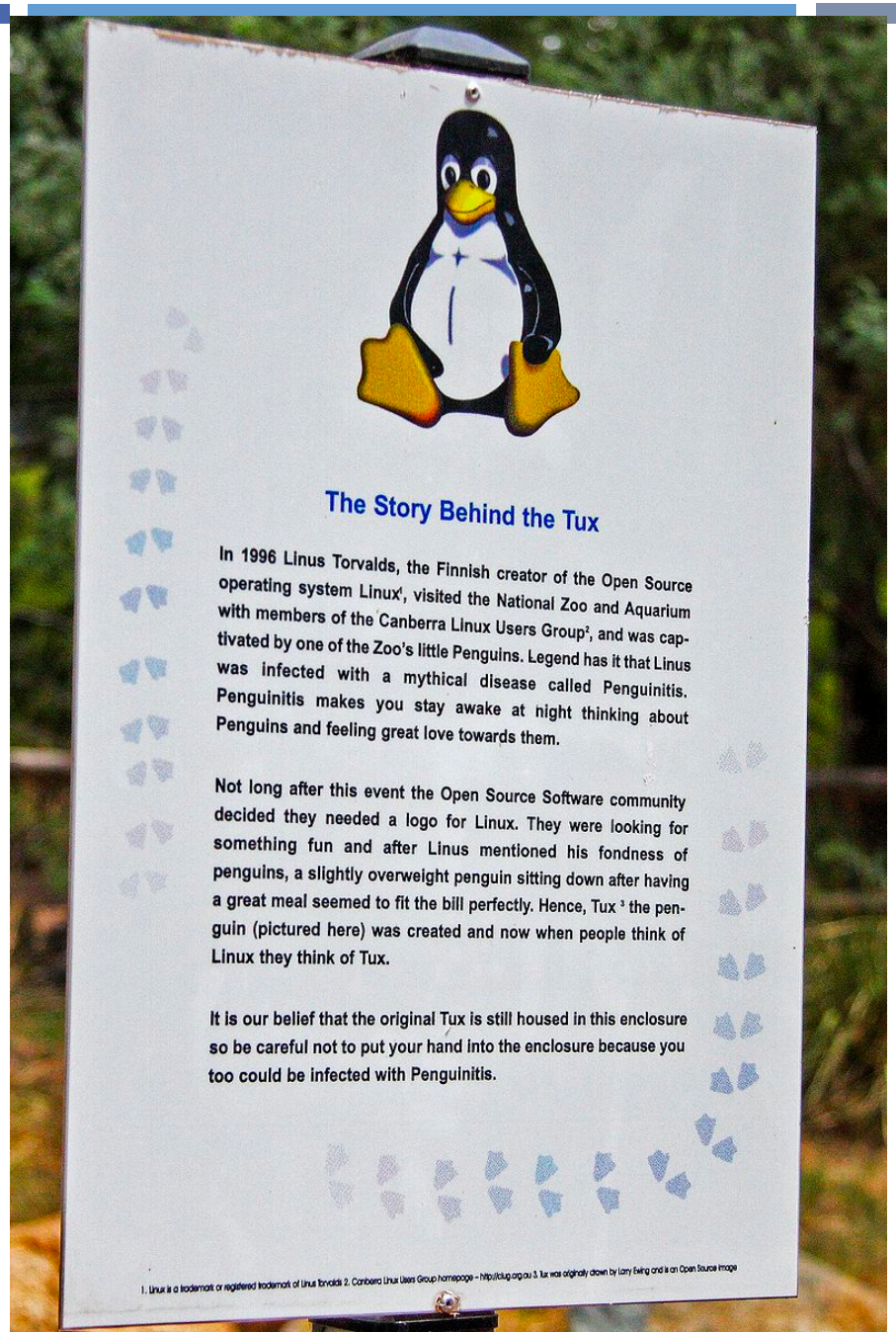
I/O devices

shorturl.at/dkpy4

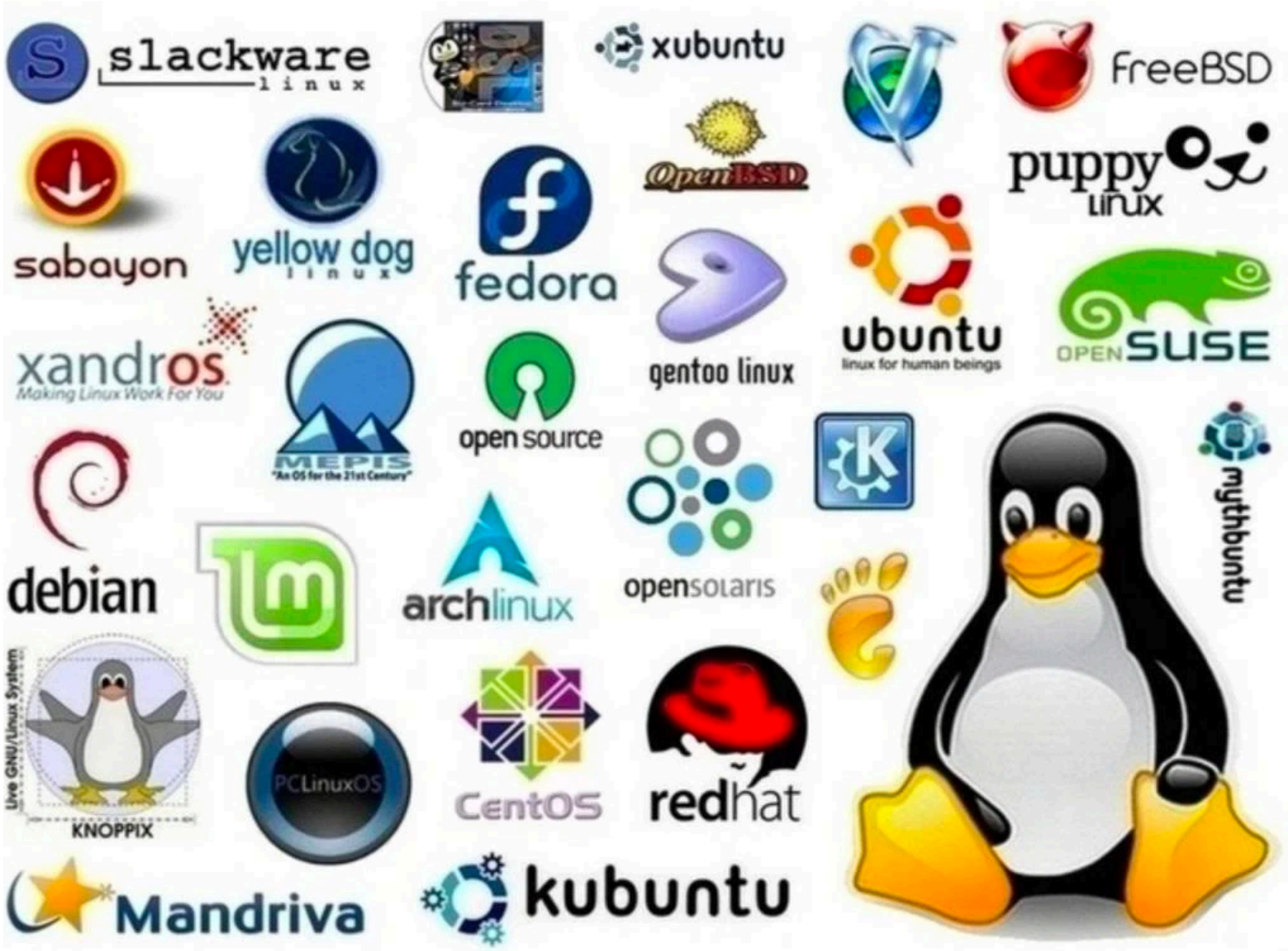
Storage devices

shorturl.at/lwKOP

shorturl.at/adsEW

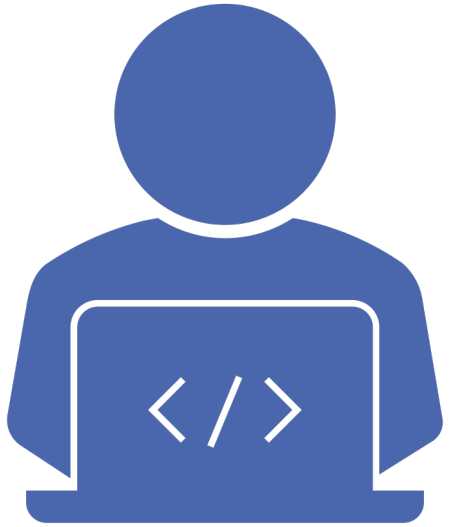


shorturl.at/fxGW8



FLAVORS OF LINUX OS

shorturl.at/ayGS5



WHO IS THE FATHER OF LINUX?

LINUS TORVALDS

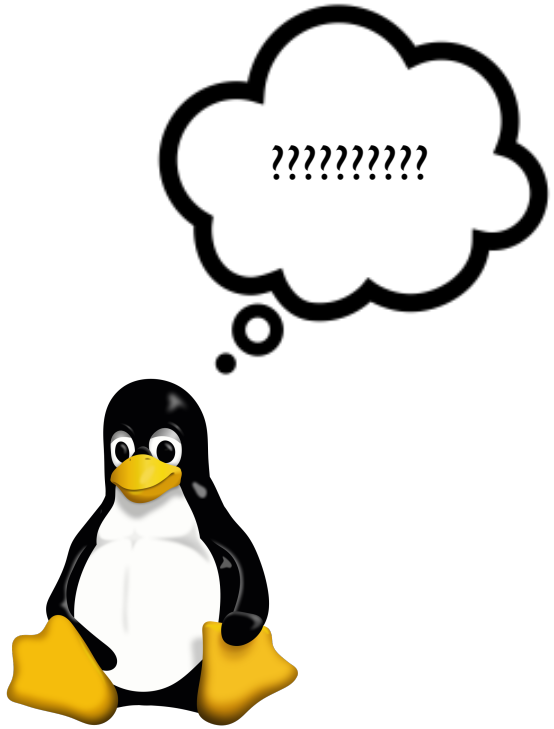
shorturl.at/lrH02

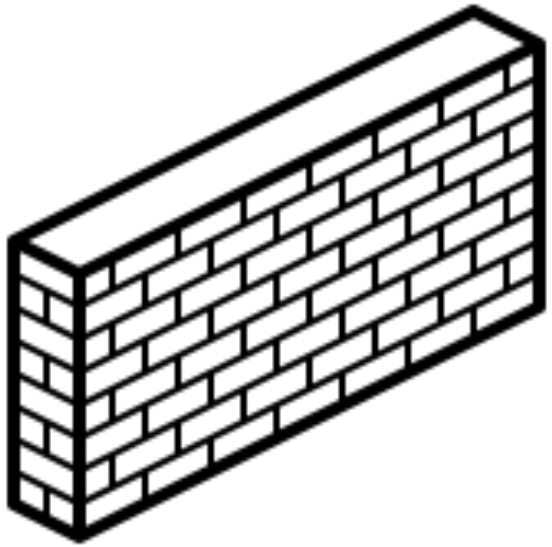
UNIX OR LINUX? WHICH IS IT?

- Unix was originally developed at Bell Labs by Ken Thompson, Dennis Ritchie and others for use inside the Bell system
- It is trademarked as **UNIX**
- UNIX variants were then developed at UC Berkeley, Microsoft, IBM, Sun Microsystems etc.
- Linux is an open source variant of UNIX

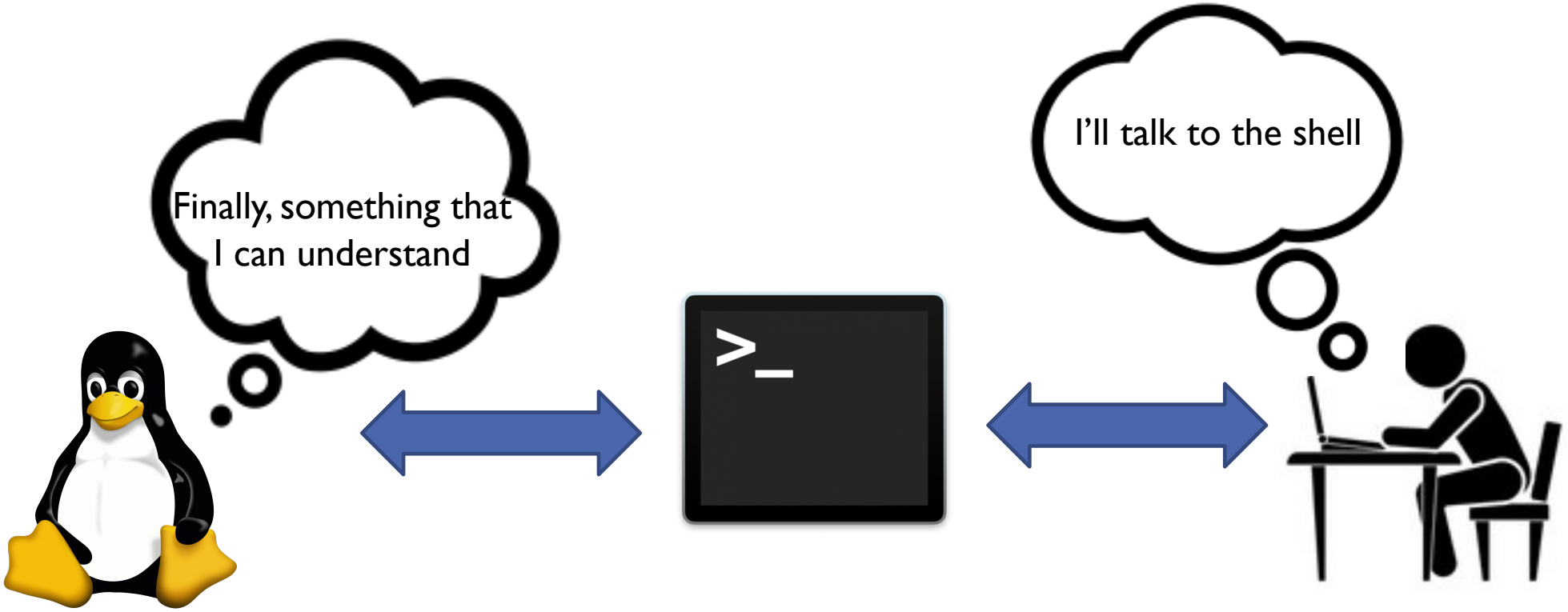
Can you guess which version of UNIX has the largest installed base ?

HOW DO WE TALK TO THE OS?





INTRODUCING THE SHELL



HOW TO USE THE TERMINAL WINDOW?

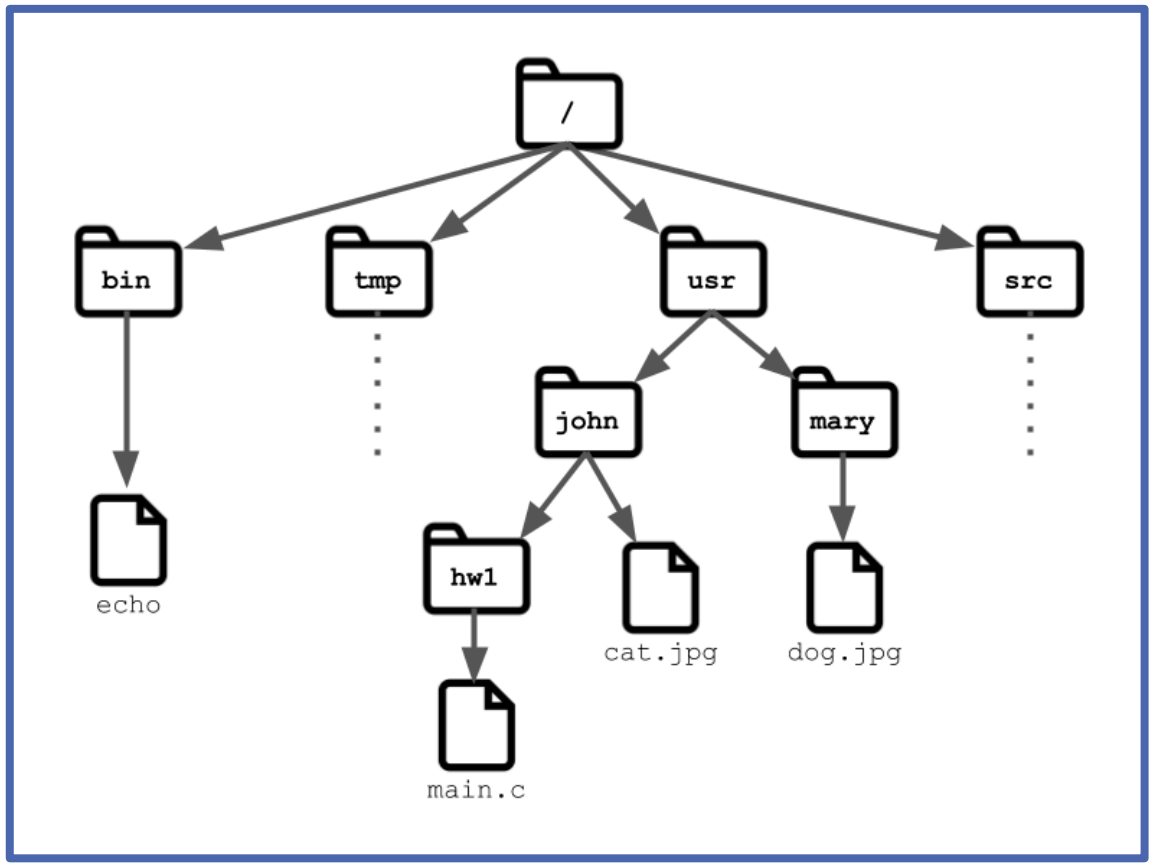
- When you select the terminal icon on your desktop, you can create an interface with a shell
- Everything you type on the terminal window is sent back to the shell and the response from the OS is printed on your terminal window
- DEMO

GETTING HELP WITH THE COMMANDS

- Most programs will understand the "--help" option and print information about the command's usage
- The shell can also lookup the usage of a command by using the "man" command.
- Let's see an example.

Are the commands case sensitive?

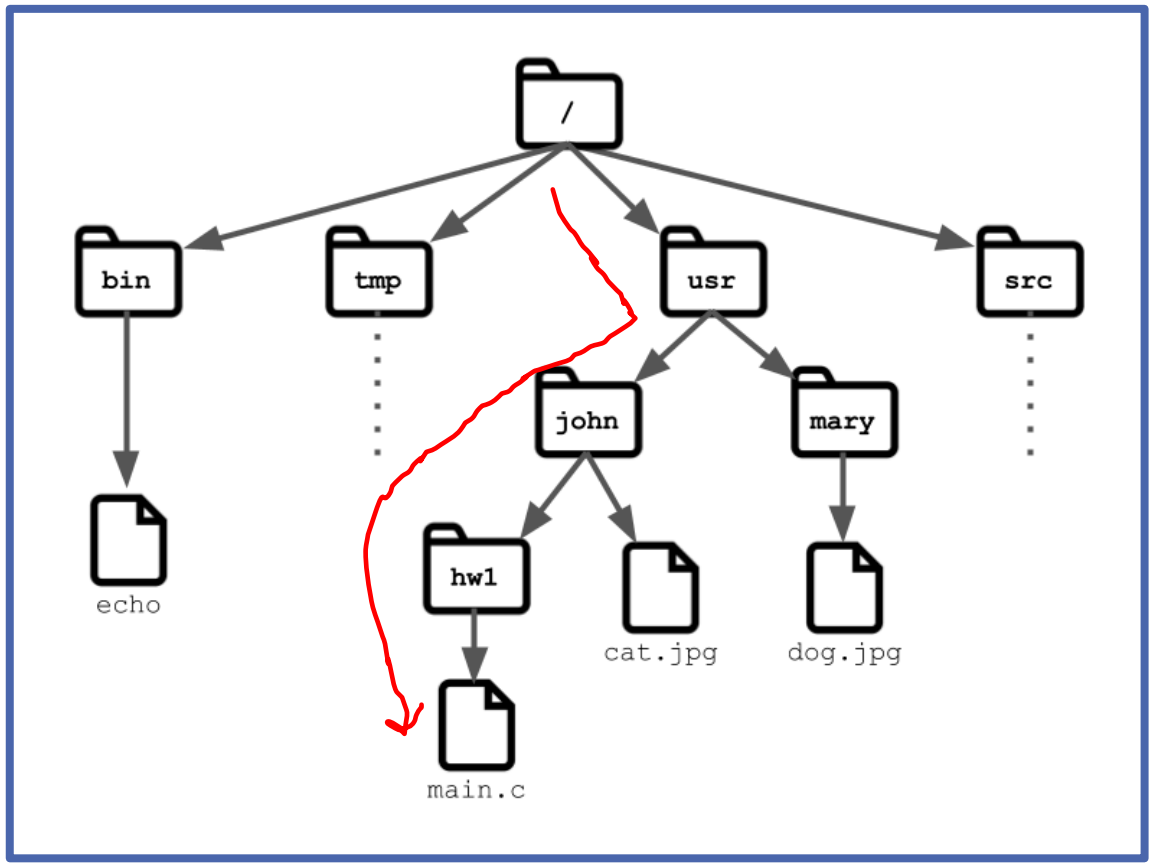
NAVIGATING THE FILE SYSTEM



- Current working directory
  - **pwd** – print working directory
  - **cd** – change directory
  - Relative path vs full path
- File System Commands
  - **ls** – list directory contents
  - **cp** – copy files
  - **rm** – remove files
  - **mv** – move files
  - **mkdir** – make directory
  - **rmdir** – remove directory

shorturl.at/cW467

REFERRING TO FILES: ABSOLUTE PATHS

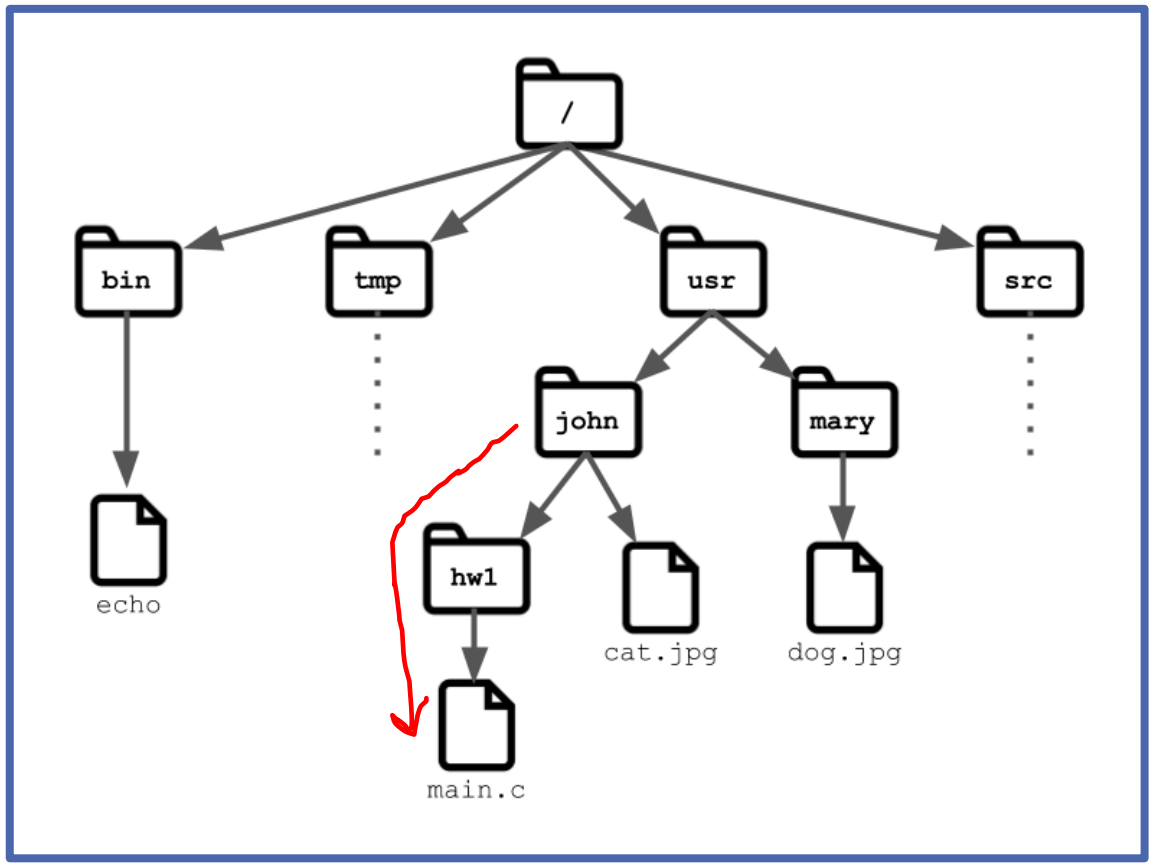


- List the directories on the path from the root ("/")
- Separated by "/"

What is the absolute path of "main.c"?

shorturl.at/cW467

REFERRING TO FILES: RELATIVE PATHS

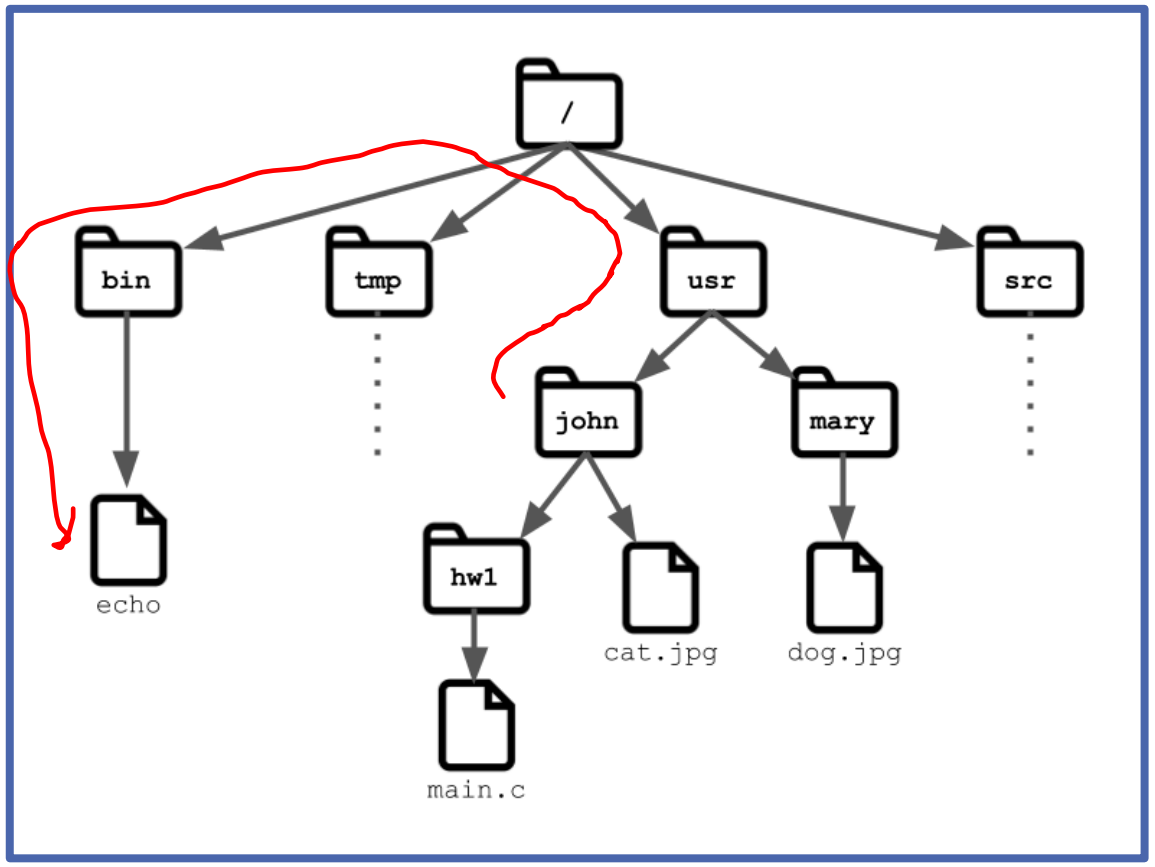


- Current directory (.)
- Parent directory (..)
- Relative path

If you're in the "john" folder, what is the relative path of "main.c" ?

shorturl.at/cW467

LET'S MAKE IT A BIT OF A CHALLENGE



If you're in the "john" folder, what is the relative path to "echo" ?

shorturl.at/cW467

ONCE YOU KNOW HOW TO GET TO YOUR FILES, HOW DO YOU WORK WITH THEM?

- **cat** – copy the contents to the screen
- **more (less)** – display file contents in a user friendly manner
- **head** - copy the first lines of a file to the screen
- **tail** – copy the last lines of a file to the screen
- **wc** – count the number of lines, words, characters in a file
- **grep** – globally search a regular expression and print

COMMANDS FOR CHECKING THE SYSTEM STATUS

- **who** – list of current users
- **whoami** – what is my user name
- **top** – list the processes using the most resources
- **ps** – process status
- **uptime** – how long has the system been running
- **date -** current date and time

PIPING

- Usually the shell takes an input from the terminal and prints the output back on the terminal
- There can be instances where we need to fetch information from one process and feed it into another process. This is called piping
- For eg: Read some portions of a file and send the filtered text to another file.

USEFUL COMMANDS IN PIPING

- **wc** – count the number of lines, words, characters in a file
- **grep –** output lines matching a pattern
- **sort –** sort lines of text
- **uniq –** filter out repeated lines in a file
- **cut –** cut out selected portions of each line of a file
- **tee –** sends the data to both a file and stdout

AUTO COMPLETION HACKS

From command line,

- Type part of name of a command or filename, press <tab> and it will auto complete the name for you
- Arrow keys (⬆⬇) to go back through history of commands you typed in.

SOME OTHER USEFUL COMMANDS

- wc
  - wc file
  - wc -l file
  - wc –w file
- grep *pattern [file]*
  - grep public *java
  - grep include controller.cpp
  - grep TODO src/*
  - ls | grep –i main

DIFF

- Compares the difference between 2 files
  - diff file1 file2
  - sdiff file1 file2

SORT

- sort data.txt
- sort –n data.txt
  - Why is n necessary?
- Sort by column
  - sort –k2 data.txt
  - sort –k2,3 data.txt
- sort –u data.txt

FIND

- Find a file in a directory tree
  - find . –name *filename* -print

VIM

- vim filename to edit a file, Vim starts out in command mode.
- To enter the insert mode, type i (for "insert")
  - To get out of insert mode, hit the Escape key.
  - Once you press Escape, you're in command mode again.
  - press ":" and Vim will switch to *last-line* mode. Enter a command like
    - :w to write the file, or
    - :q to exit the editor, or
    - :q! to quit without saving etc.

LET'S PRACTICE

- pwd
- ls
- ls –l
- cd (home)
- cd (down)
- mkdir
- rmdir
- rm
- touch
- cp (from) (to)
- mv (from) (to)
- echo "this" >> target
- cat
- vim
- grep "pattern" in-target

HELPFUL TIPS

- h moves the cursor one character to the left
- j moves the cursor down one line
- k moves the cursor up one line
- l moves the cursor one character to the right
- 0 moves the cursor to the beginning of the line
- $ moves the cursor to the end of the line
- w move forward one word
- b move backward one word
- G move to the end of the file
- gg move to the beginning of the file
- dd deletes a record
- x deletes a character