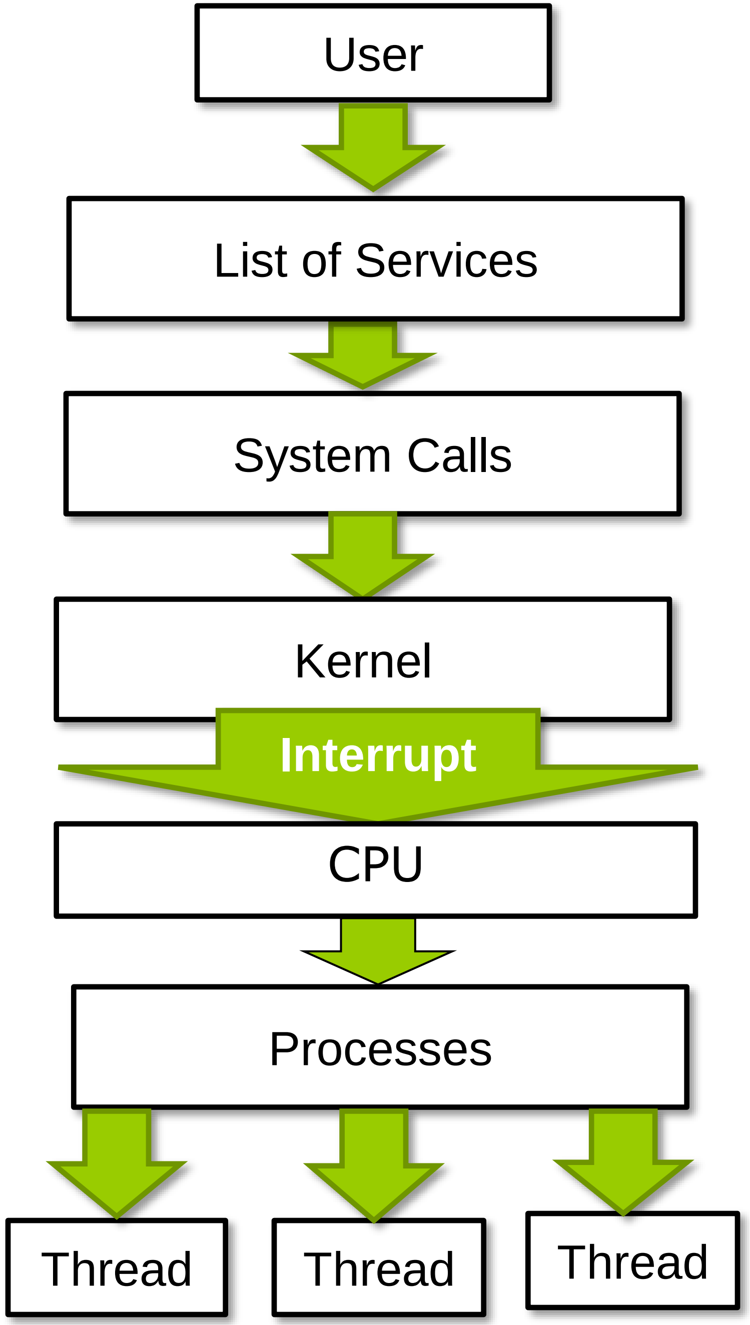
Part 3

The Big Pic of OS



User Mode vs. Kernel Mode

User Mode

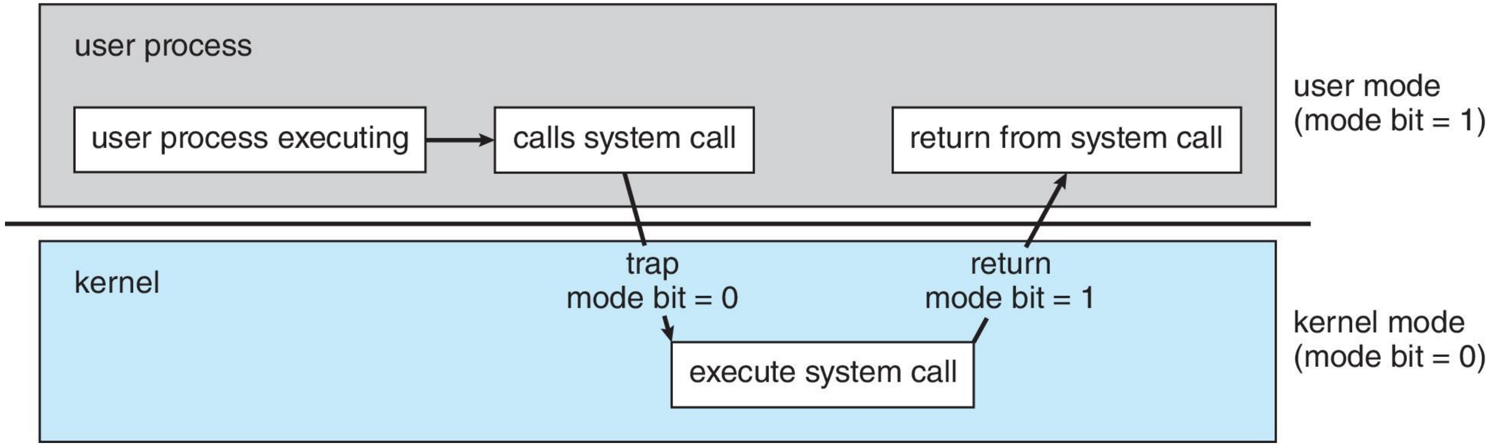
- The system is in user mode when the operating system is running a **user application such as handling a text editor.**
- The transition from user mode to kernel mode occurs when the application requests the help of operating system or an interrupt or a system call occurs.
- The mode bit is set to 1 in the user mode. It is changed from 1 to 0 when switching from user mode to kernel mode.

Kernel Mode

- The **system starts in kernel mode** when it boots and after the operating system is loaded, it executes applications in user mode.
- There are some **privileged** instructions that can only be executed in kernel mode.
- These are interrupt instructions, input output management etc. **If the privileged instructions are executed in user mode**, it is illegal and a trap is generated.
- The mode bit is set to 0 in the kernel mode. It is changed from 0 to 1 when switching from kernel mode to user mode.

Transition from User to Kernel Mode

- Timer to prevent infinite loop / process hogging resources
  - Timer is set to interrupt the computer after some time period
  - Keep a counter that is decremented by the physical clock
  - Operating system set the counter (privileged instruction)
  - When counter zero generate an interrupt
  - Set up before scheduling process to regain control or terminate program that exceeds allotted time



Process Management

- A process is a program in execution. It is a unit of work within the system. **Program** is a ***passive entity***, **process** is an ***active entity***.
- Process needs resources to accomplish its task
  - CPU, memory, I/O, files
  - Initialization data
- Process termination requires reclaim of any reusable resources

Process Management

- Single-threaded process has one **program counter** specifying location of next instruction to execute
  - Process executes instructions sequentially, one at a time, until completion
- Multi-threaded process has one program counter per thread
- Typically system has many processes, some user, some operating system running concurrently on one or more CPUs
  - Concurrency by multiplexing the CPUs among the processes / threads

Process Management Activities

The operating system is responsible for the following activities in connection with process management:

- Creating and deleting both user and system processes
- Suspending and resuming processes
- Providing mechanisms for process synchronization
- Providing mechanisms for process communication
- Providing mechanisms for deadlock handling

Memory Management

- To execute a program all (or part) of the instructions must be in **memory**
- All (or part) of the data that is needed by the program must be in **memory**
- Memory management determines what is in memory and when
  - Optimizing CPU utilization and computer response to users
- **Memory management activities**
  - Keeping track of which parts of memory are currently being used and by whom
  - Deciding which processes (or parts thereof) and data to move into and out of memory
  - Allocating and deallocating memory space as needed

File-system Management

- OS **provides uniform, logical view of information storage**
  - Abstracts physical properties to logical storage unit - **file**
  - Each medium is controlled by device (i.e., disk drive, tape drive)
    - Varying properties include access speed, capacity, data-transfer rate, access method (sequential or random)
- File-System management
  - Files usually organized into **directories**
  - **Access control** on most systems to determine who can access what
  - OS activities include
    - Creating and deleting files and directories
    - Primitives to manipulate files and directories
    - Mapping files onto secondary storage
    - Backup files onto stable (non-volatile) storage media

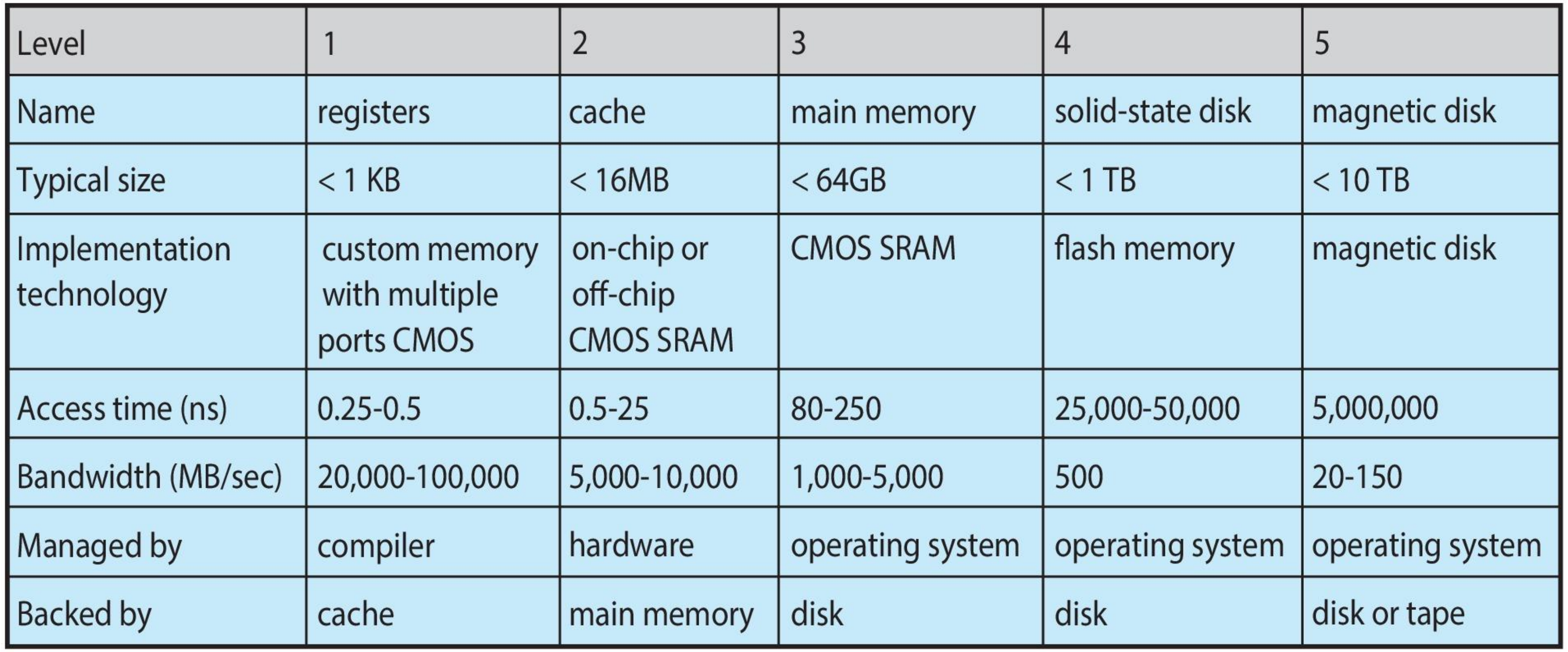
Mass-Storage Management

- Usually disks used to store data that does not fit in main memory or data that must be kept for a “long” period of time
- Proper management is of central importance
- Entire speed of computer operation hinges on disk subsystem and its algorithms
- **OS activities**
  - Mounting and unmounting
  - Free-space management
  - Storage allocation
  - Disk scheduling
  - Partitioning
  - Protection
- Some storage need not be fast
  - Tertiary storage includes optical storage, magnetic tape
  - Still must be managed – by OS or applications

Caching

- Important principle, performed at **many levels** in a computer (in hardware, operating system, software)
- Information in use copied from slower to faster storage temporarily
- **Faster storage (cache) checked first to determine if information is there**
  - If it is, information used directly from the cache (fast)
  - If not, data copied to cache and used there
- Cache smaller than storage being cached
  - Cache management important design problem
  - Cache size and replacement policy

Characteristics of Various Types of Storage

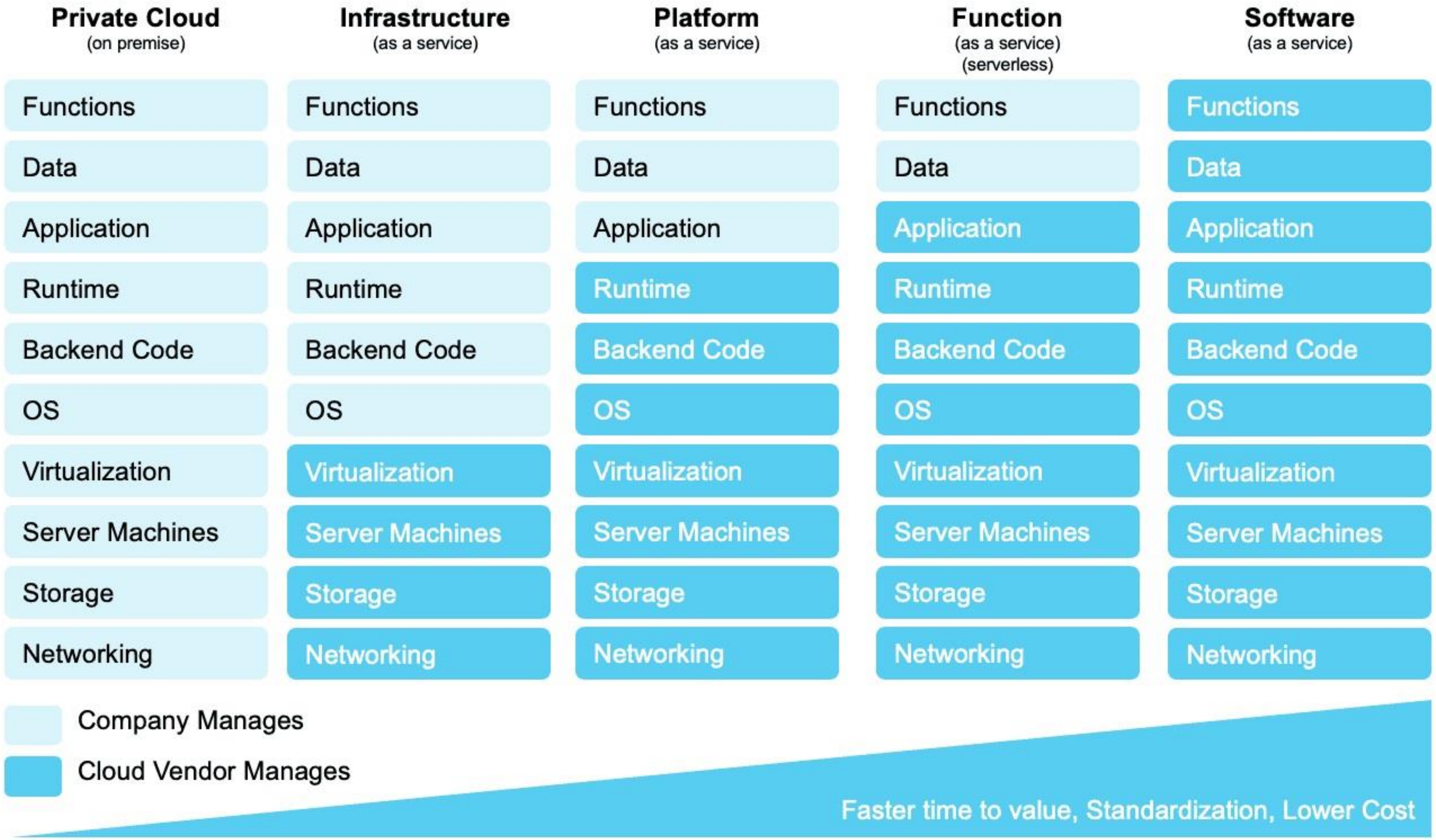


Level	1	2	3	4	5
Name	registers	cache	main memory	solid-state disk	magnetic disk
Typical size	< 1 KB	< 16MB	< 64GB	< 1 TB	< 10 TB
Implementation technology	custom memory with multiple ports CMOS	on-chip or off-chip CMOS SRAM	CMOS SRAM	flash memory	magnetic disk
Access time (ns)	0.25-0.5	0.5-25	80-250	25,000-50,000	5,000,000
Bandwidth (MB/sec)	20,000-100,000	5,000-10,000	1,000-5,000	500	20-150
Managed by	compiler	hardware	operating system	operating system	operating system
Backed by	cache	main memory	disk	disk	disk or tape

Movement between levels of storage hierarchy can be explicit or implicit

Computing Environments – Cloud Computing

- Delivers computing, storage, even apps as a service across a network
- Logical extension of virtualization because it uses virtualization as the base for it functionality.
  - Amazon **EC2** has thousands of servers, millions of virtual machines, petabytes of storage available across the Internet, pay based on usage
- Many types
  - **Public cloud** – available via Internet to anyone willing to pay
  - **Private cloud** – run by a company for the company’s own use
  - **Hybrid cloud** – includes both public and private cloud components
  - Software as a Service (**SaaS**) – one or more applications available via the Internet (i.e., word processor)
  - Platform as a Service (**PaaS**) – software stack ready for application use via the Internet (i.e., a database server)
  - Infrastructure as a Service (**IaaS**) – servers or storage available over Internet (i.e., storage available for backup use)
  - OpenStack OS for the Cloud



Private Cloud (on premise)	**Infrastructure** (as a service)	**Platform** (as a service)	**Function** (as a service) (serverless)	**Software** (as a service)
○ Functions	○ Functions	○ Functions	○ Functions	● Functions
○ Data	○ Data	○ Data	○ Data	● Data
○ Application	○ Application	○ Application	● Application	● Application
○ Runtime	○ Runtime	● Runtime	● Runtime	● Runtime
○ Backend Code	○ Backend Code	● Backend Code	● Backend Code	● Backend Code
○ OS	○ OS	● OS	● OS	● OS
○ Virtualization	● Virtualization	● Virtualization	● Virtualization	● Virtualization
○ Server Machines	● Server Machines	● Server Machines	● Server Machines	● Server Machines
○ Storage	● Storage	● Storage	● Storage	● Storage
○ Networking	● Networking	● Networking	● Networking	● Networking

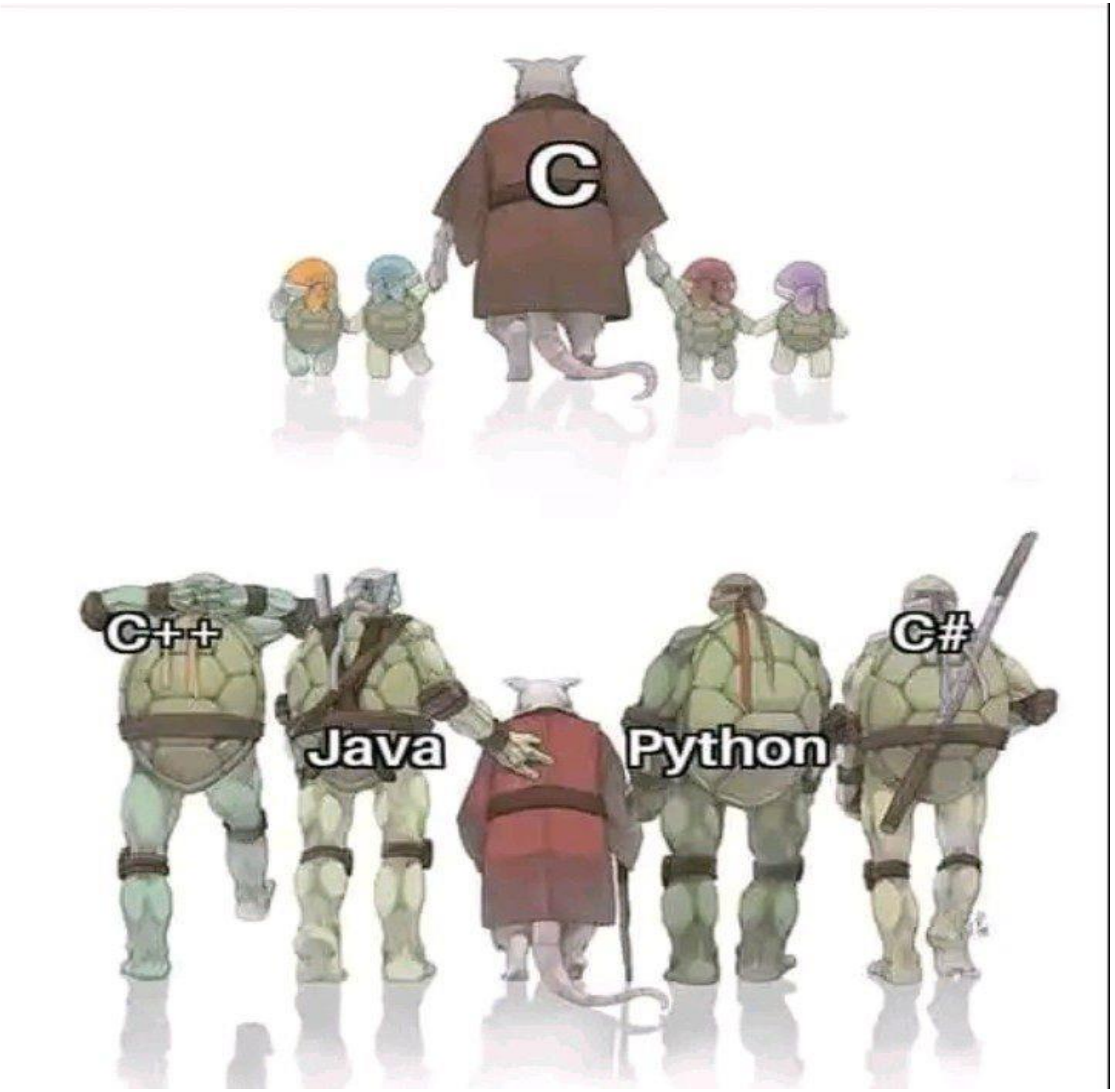
○ Company Manages

● Cloud Vendor Manages

Faster time to value, Standardization, Lower Cost

Operating-System Structures

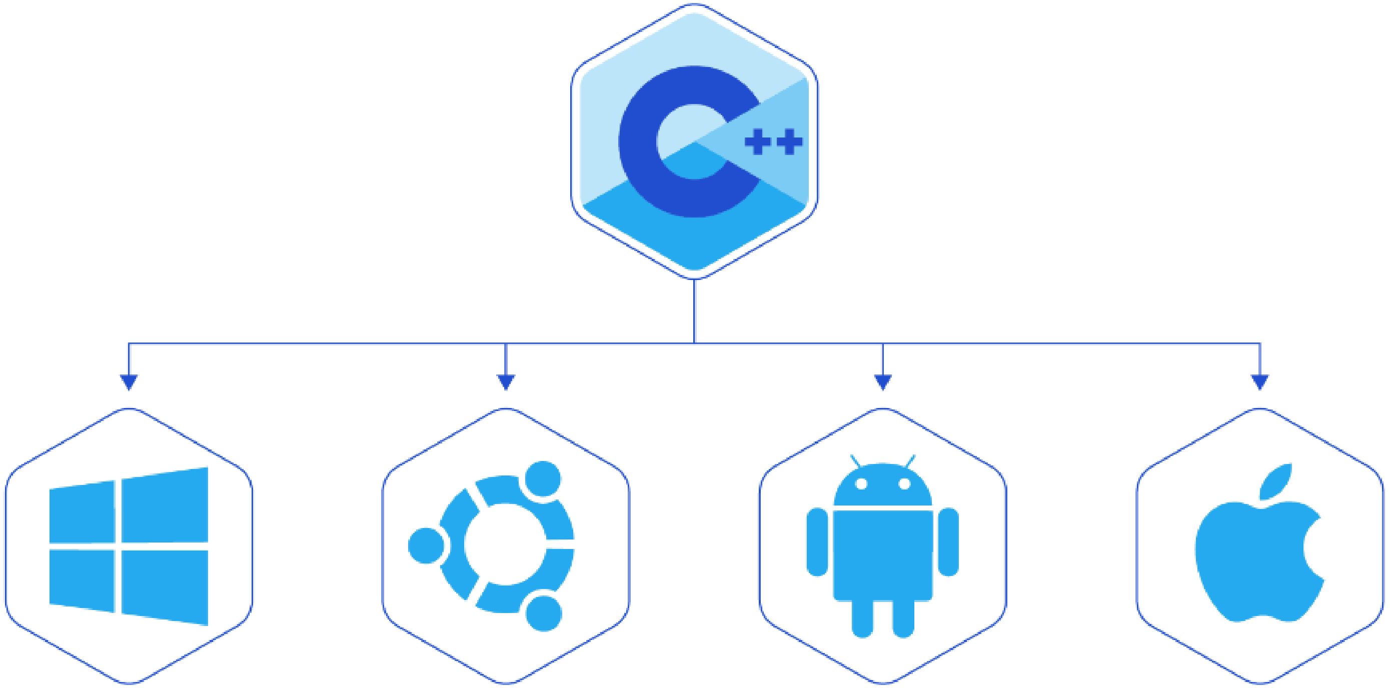
Programming Languages of OS

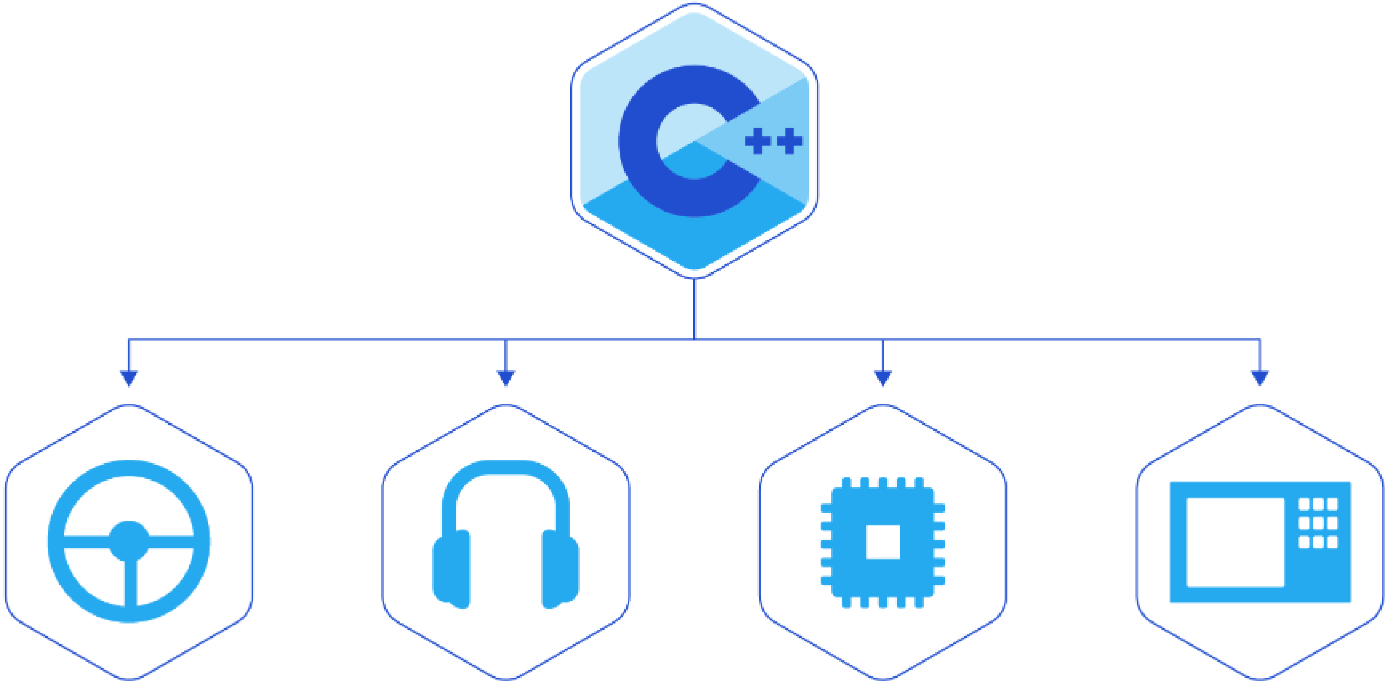


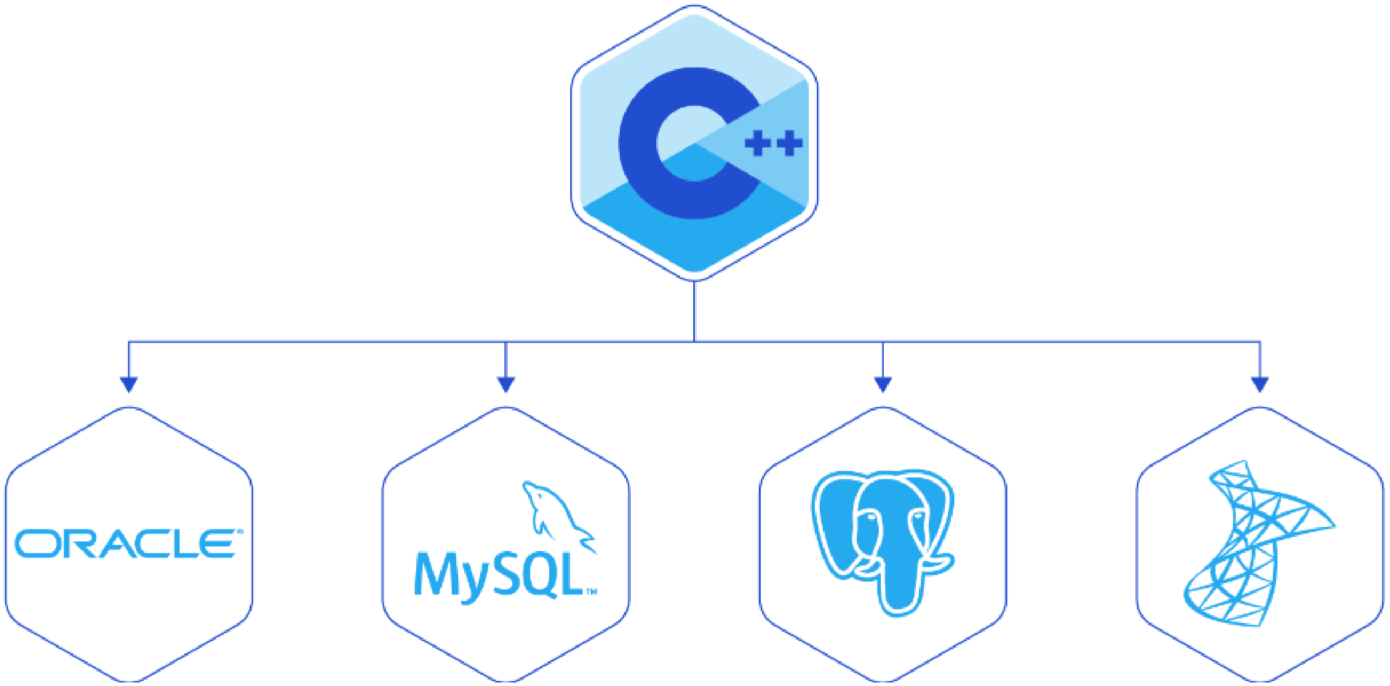
C
C++
Java
Python
C#

HOW OLD IS YOUR LANGUAGE ?

C	1972
C++	1980
C#	2001
Javascript	1995
NodeJs	2009
TypeScript	2012
Objective-C	1986
Swift	2015
Java	1995
Kotlin	2011
Scala	2003
Clojure	2007
Haskell	1990
PHP	1995
Python	1990
Ruby	1995
Erlang	1986
Rust	2010
Go	2009
HTML	1991
CSS	1996
Lisp	1958
Perl	1987






C++
ORACLE
MySQL

Lines of Code!

Number of Lines of Code

Windows XP: 40 million
Windows Vista : 50 million
Windows 7: 40 million
Windows 8: 50 million

Windows 10 : 50 million

Android : 12 Million

Facebook : 100 Million

Mac OS 10.4 : 86 Million

New Car: 100 Million

Google : 2 Billion

- Boeing 787 : 14 Million
- F35 Fighter Jet : 24 Million

- Since, the software industry changes with every new update or release. “What is the best programming language of the world?” is hard. If you want to start your career as a developer, it is only you who can decide which one is best suited as per your interest in varying fields.
- For example:
- **Front-end web development – JavaScript**
- **Back-end web development – JavaScript, Python, Java, PHP, Ruby, C#**
- **Desktop application – Java, C++, Python**
- **Mobile development – Swift, Java, C#**
- **Game development – C++, C#**

Operating-System Structures

- Operating System **Services**
- User and Operating System-**Interface**
- System **Calls**
- **System Services**
- **Linkers and Loaders**
- Why Applications are **Operating System Specific**
- Operating-System **Design and Implementation**
- **Operating System Structure**
- **Building and Booting** an Operating System
- Operating System **Debugging**

Operating System Services

- Operating systems provide an environment **for execution of programs and services to programs and users**
- **One set** of operating-system services provides **functions that are helpful to the user:**
  - **User interface** - Almost all operating systems have a user interface (UI).
    - Varies between **Command-Line** **(CLI)**, **Graphics User Interface** **(GUI)**, **touch-screen, Batch**
  - **Program execution** - The system must be able to load a program into memory and to run that program, end execution, either normally or abnormally (**indicating error**)
  - **I/O operations** - A running program may require I/O, which may involve a file or an I/O device

- **Command-line interfaces**, where the user provides the input by typing a command string with the computer keyboard and the system provide output by printing text on the computer monitor. Used for system administration tasks etc.
- **Batch interfaces** are non-interactive user interfaces, where the user specifies all the details of the *batch job* in advance to batch processing, and receives the output when all the processing is done. The computer does not prompt for further input after the processing has started.
- **Graphical user interfaces** (GUI), which accept input via devices such as computer keyboard and mouse and provide articulated [graphical] output on the [computer monitor]. There are at least two different principles widely used in GUI design: object-oriented interfaces and application]] oriented interfaces.
- **Touch-screen interface**

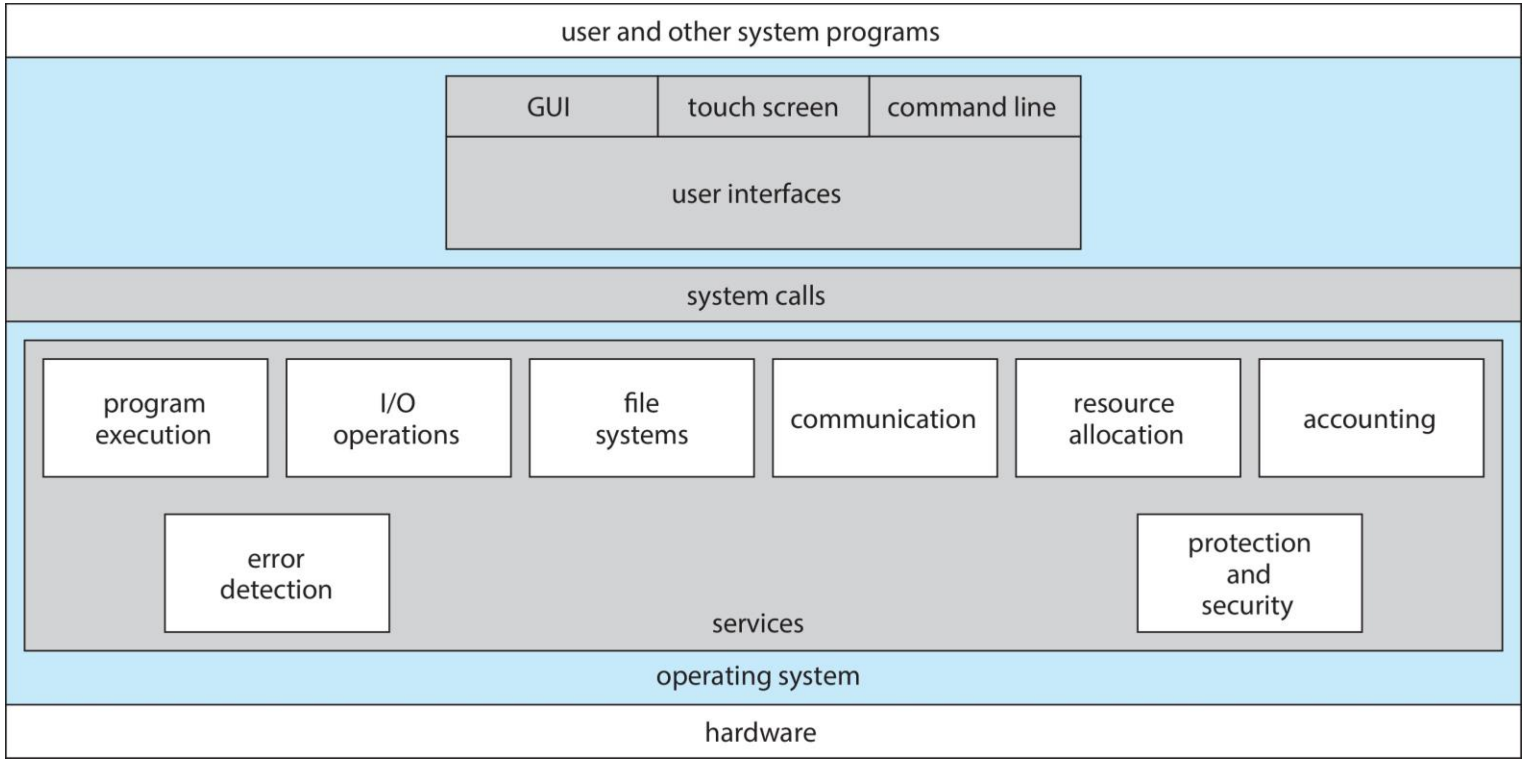
Operating System Services (Cont.)

- One set of operating-system services **provides functions** that are helpful to the user (Cont.):
  - **File-system manipulation** - The file system is of particular interest. Programs need to *read and write files and directories, create and delete them, search them, list file Information, permission management*.
  - **Communications** – Processes may exchange information, on the same computer or between computers over a network
    - Communications may be via shared memory or through message passing (packets moved by the OS)
  - **Error detection** – OS needs to be **constantly aware of possible errors**
    - May occur in the CPU and memory hardware, in I/O devices, in user program
    - For each type of error, OS should take the appropriate action to ensure correct and consistent computing
    - Debugging facilities can greatly enhance the user’s and programmer’s abilities to efficiently use the system

Operating System Services (Cont.)

- Another set of OS functions exists for ensuring the **efficient operation** of the system itself via **resource sharing**
  - **Resource allocation -** When **multiple users or multiple jobs running concurrently, resources must be allocated to each of them**
    - Many types of resources - CPU cycles, main memory, file storage, I/O devices.
  - **Logging -** To keep track of which users use how much and what kinds of computer resources
  - **Protection and security -** The owners of information stored in a multiuser or networked computer system may want to control use of that information, concurrent processes should not interfere with each other
    - **Protection** involves ensuring that all access to system resources is controlled
    - **Security** of the system from outsiders requires user authentication, extends to defending external I/O devices from invalid access attempts

A View of Operating System Services

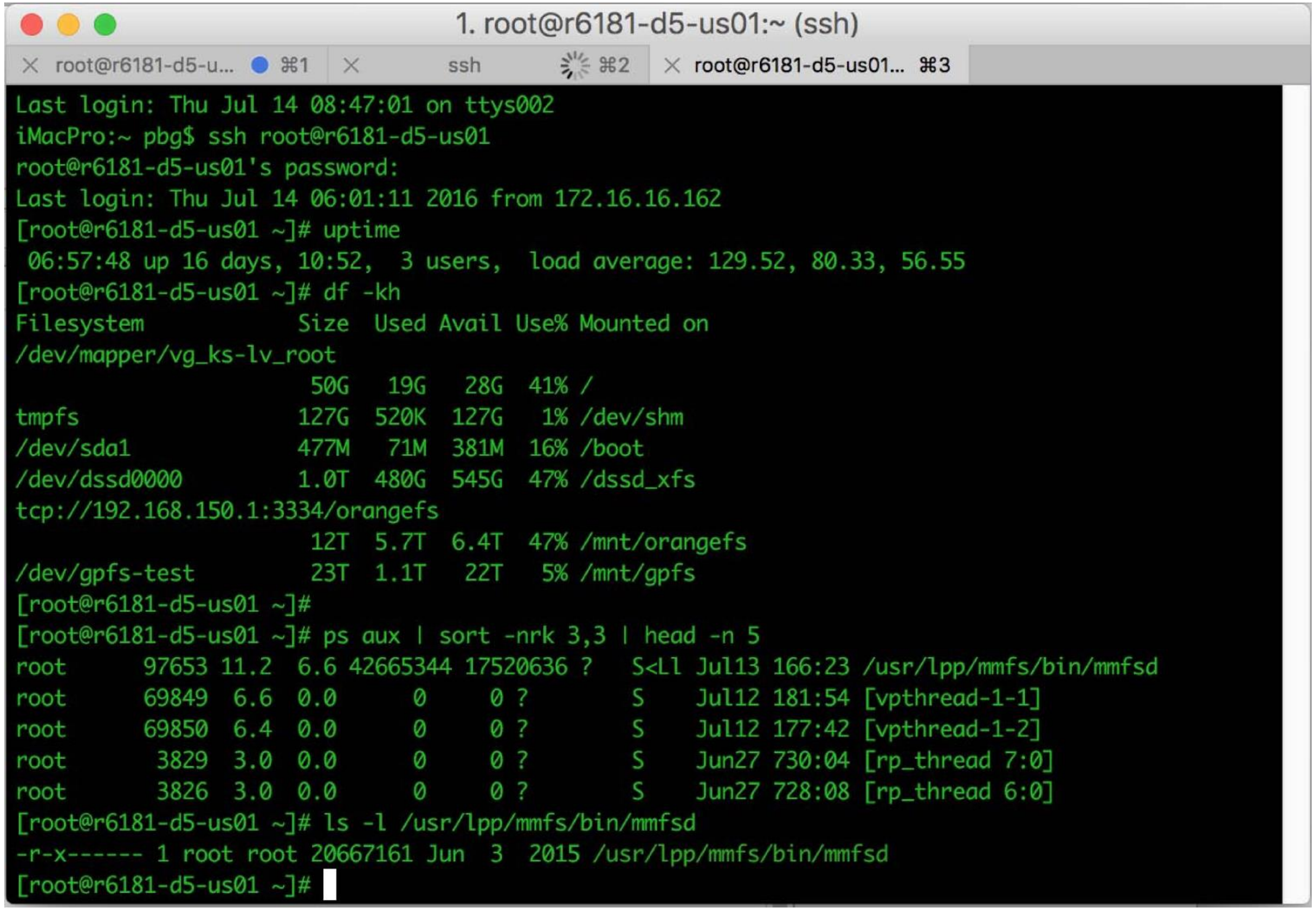


User Operating System Interface - CLI

CLI or **command interpreter** allows direct command entry

- Sometimes **implemented in kernel, sometimes by systems program**
- Sometimes multiple flavors implemented – **shells**
- Primarily fetches a command from user and executes it
- Sometimes commands built-in, sometimes just names of programs
  - If the latter, adding new features doesn’t require shell modification

Shell Command Interpreter



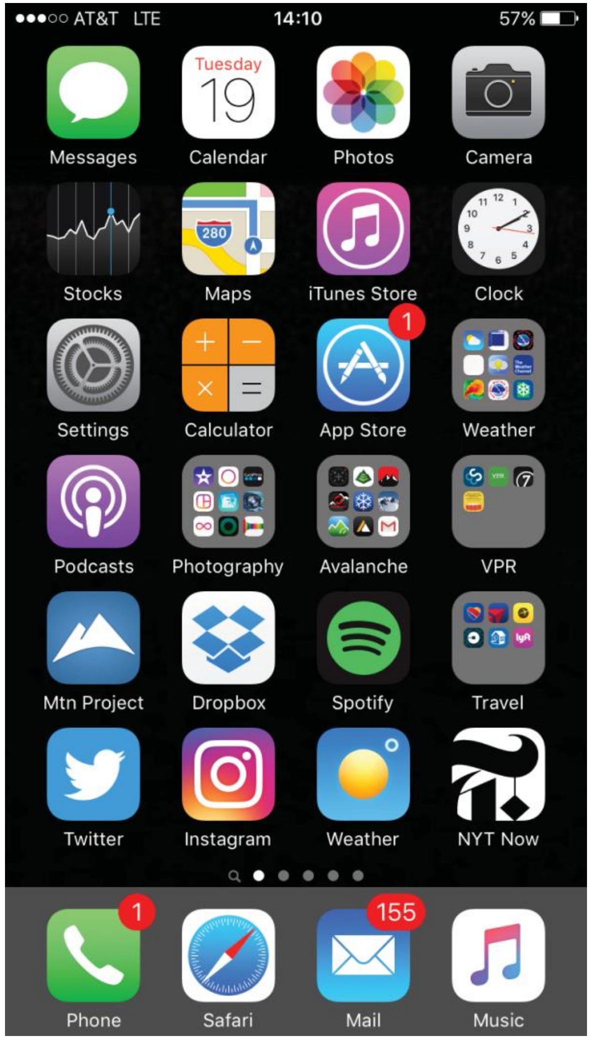
```
Last login: Thu Jul 14 08:47:01 on ttys002
iMacPro:~ pbg$ ssh root@r6181-d5-us01
root@r6181-d5-us01's password:
Last login: Thu Jul 14 06:01:11 2016 from 172.16.16.162
[root@r6181-d5-us01 ~]# uptime
 06:57:48 up 16 days, 10:52,  3 users,  load average: 129.52, 80.33, 56.55
[root@r6181-d5-us01 ~]# df -kh
Filesystem            Size  Used Avail Use% Mounted on
/dev/mapper/vg_ks-lv_root
                       50G   19G   28G  41% /
tmpfs                 127G  520K  127G   1% /dev/shm
/dev/sda1             477M   71M  381M  16% /boot
/dev/dssd0000         1.0T  480G  545G  47% /dssd_xfs
tcp://192.168.150.1:3334/orangefs
                       12T  5.7T  6.4T  47% /mnt/orangefs
/dev/gpfs-test         23T  1.1T   22T   5% /mnt/gpfs
[root@r6181-d5-us01 ~]#
[root@r6181-d5-us01 ~]# ps aux | sort -nrk 3,3 | head -n 5
root      97653 11.2  6.6 42665344 17520636 ?   S<Ll Jul13 166:23 /usr/lpp/mmfs/bin/mmfsd
root      69849  6.6  0.0      0     0 ?        S    Jul12 181:54 [vpthread-1-1]
root      69850  6.4  0.0      0     0 ?        S    Jul12 177:42 [vpthread-1-2]
root       3829  3.0  0.0      0     0 ?        S    Jun27 730:04 [rp_thread 7:0]
root       3826  3.0  0.0      0     0 ?        S    Jun27 728:08 [rp_thread 6:0]
[root@r6181-d5-us01 ~]# ls -l /usr/lpp/mmfs/bin/mmfsd
-r-x------ 1 root root 20667161 Jun  3  2015 /usr/lpp/mmfs/bin/mmfsd
[root@r6181-d5-us01 ~]# 
```

User Operating System Interface - GUI

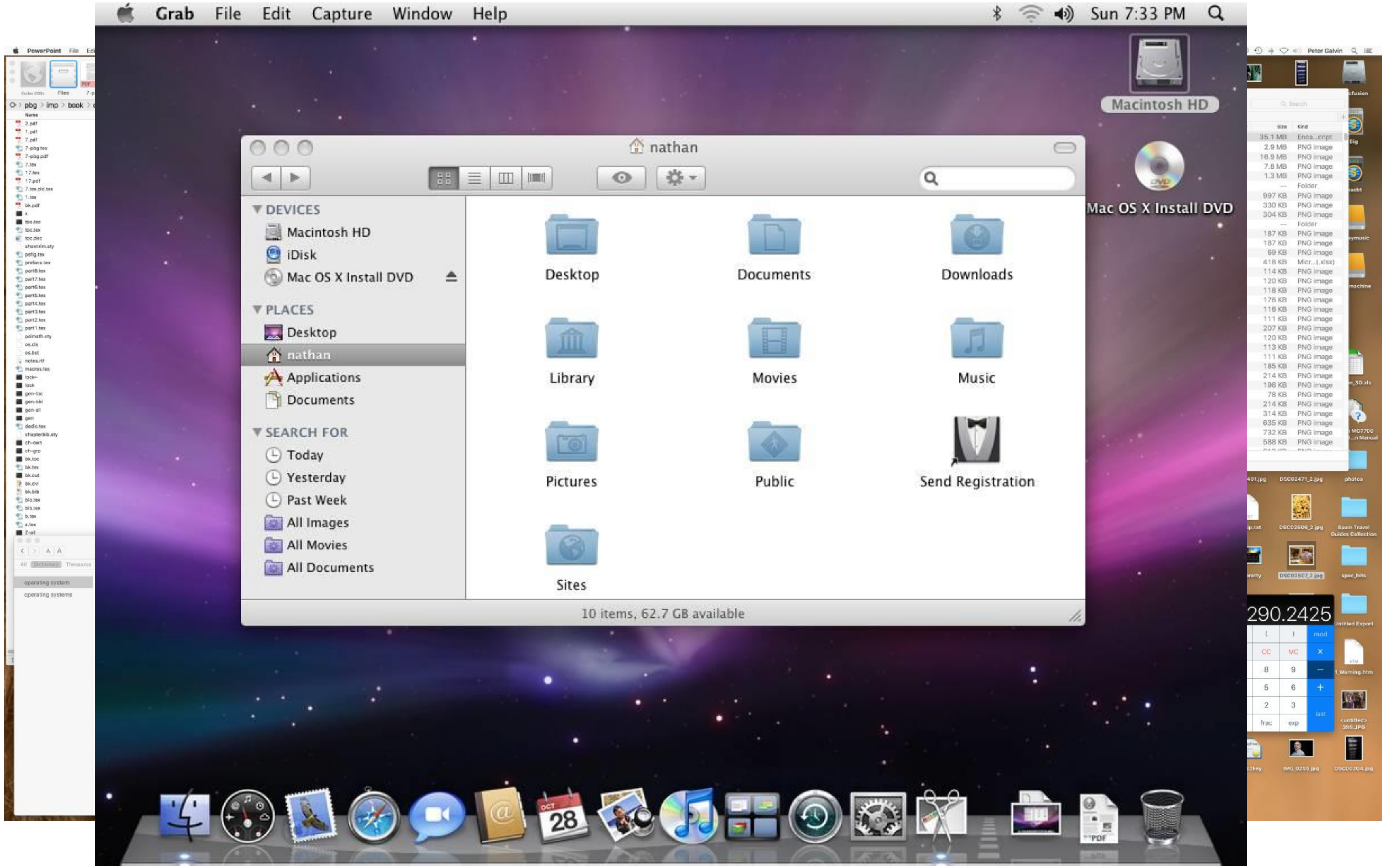
- User-friendly **desktop** metaphor interface
  - Usually mouse, keyboard, and monitor
  - **Icons** represent files, programs, actions, etc
  - Various mouse buttons over objects in the interface cause various actions (provide information, options, execute function, open directory (known as a **folder**)
  - **Invented at Xerox PARC**
- Many systems now include **both CLI and GUI** interfaces
  - Microsoft Windows is GUI with CLI “command” shell
  - Apple Mac OS GUI interface with UNIX kernel underneath and shells available
  - Unix and Linux have CLI with optional GUI interfaces (CDE, KDE, GNOME)

Touchscreen Interfaces

- Touchscreen devices require new interfaces
  - Mouse not possible or not desired
  - Actions and selection based on gestures
  - Virtual keyboard for text entry
- Voice commands



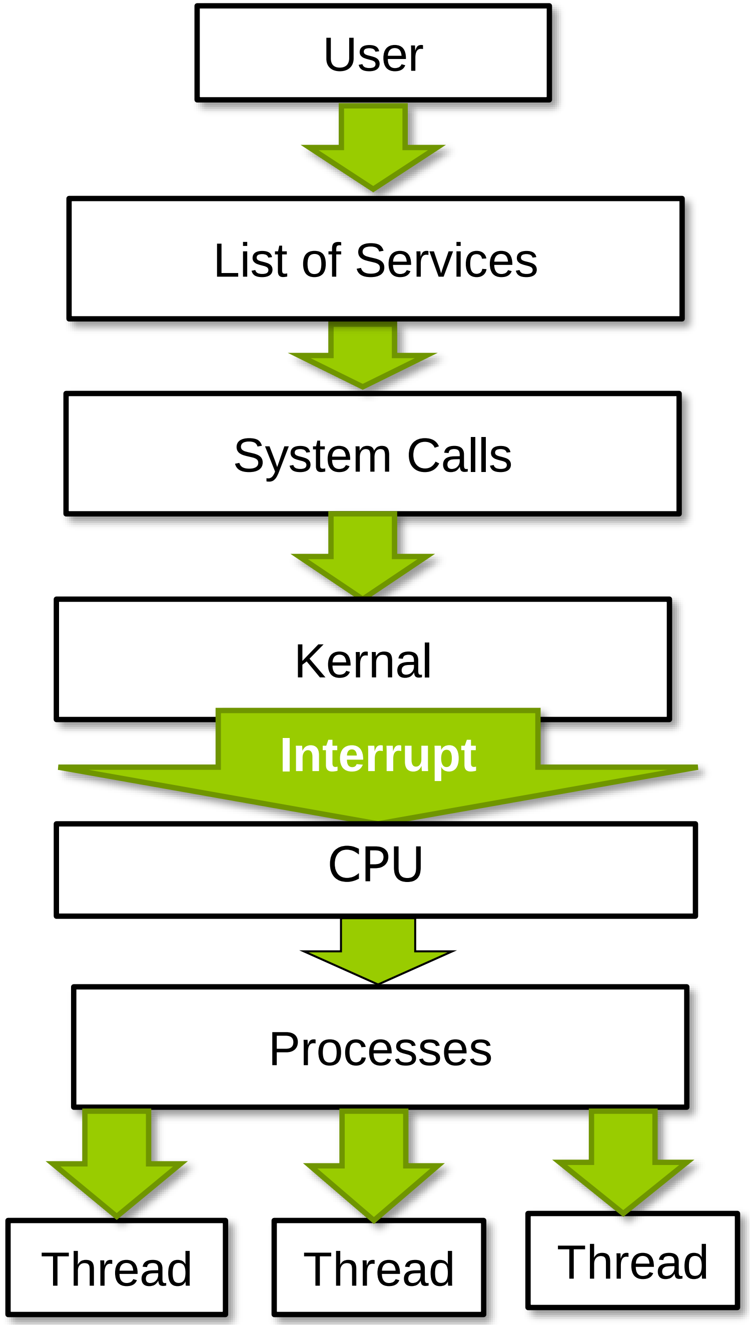
The Mac OS X GUI



Best new features in iPhone history

iOS 1: Safari
iOS 2: App Store
iOS 3: video recording
iOS 4: multitasking
iOS 5: iCloud
iOS 6: Passbook
iOS 7: Touch ID
iOS 8: Apple Pay
iOS 9: Low Power Mode
iOS 10:
iOS 11:
iOS 12:
iOS 13: Dark Mode

The Big Pic of OS



User

List of Services

System Calls

Kernal

Interrupt

CPU

Processes

Thread

Thread

Thread