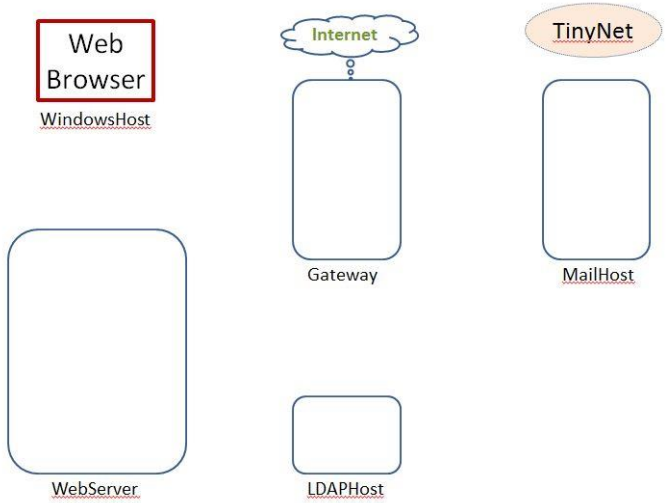
System and Network Administration

Operating System Essentials

Getting Started

TinyNet

- Comes in two parts
  - Base.iso (base image)
  - Config.iso (configuration and applications)



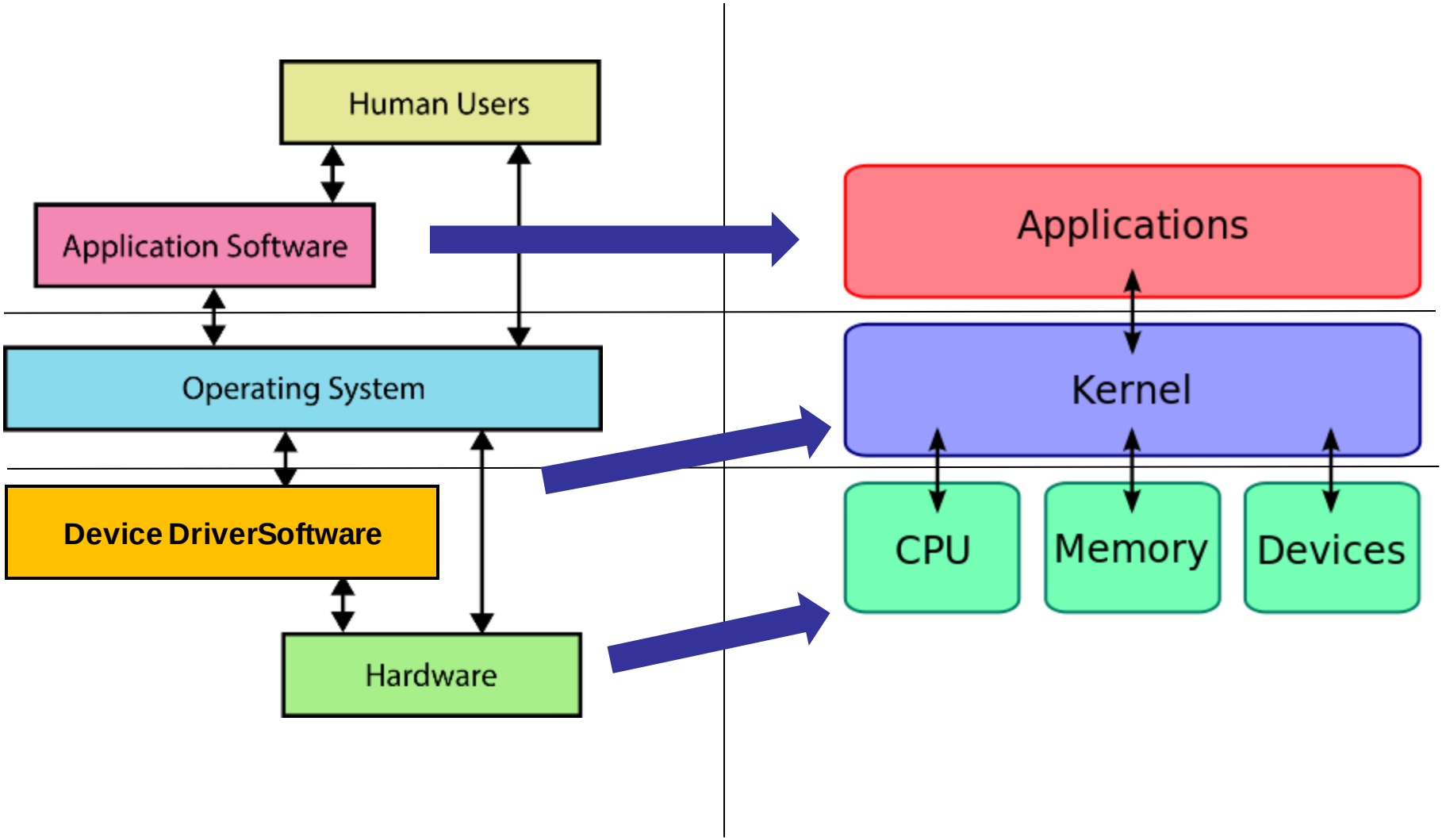
After installing VirtualBox, create VMs!

www.my-tiny.net

1. Create a virtual machine
2. Partition the disk
3. Create a filesystem
4. Copy the OS
5. Install the bootloader
6. Configure common services
7. Clone!


Q
&
A

OS in detail


Human Users
Application Software
Operating System
Device DriverSoftware
Hardware
Applications
Kernel
CPU
Memory
Devices

Services and Facilities

- Command processor (shell)
- I/O control system
- Process control management and interprocess communication
- Memory management
- Storage management
- File management
- Network management, communication support, and communication interfaces
- System protection management and security

Operating Systems

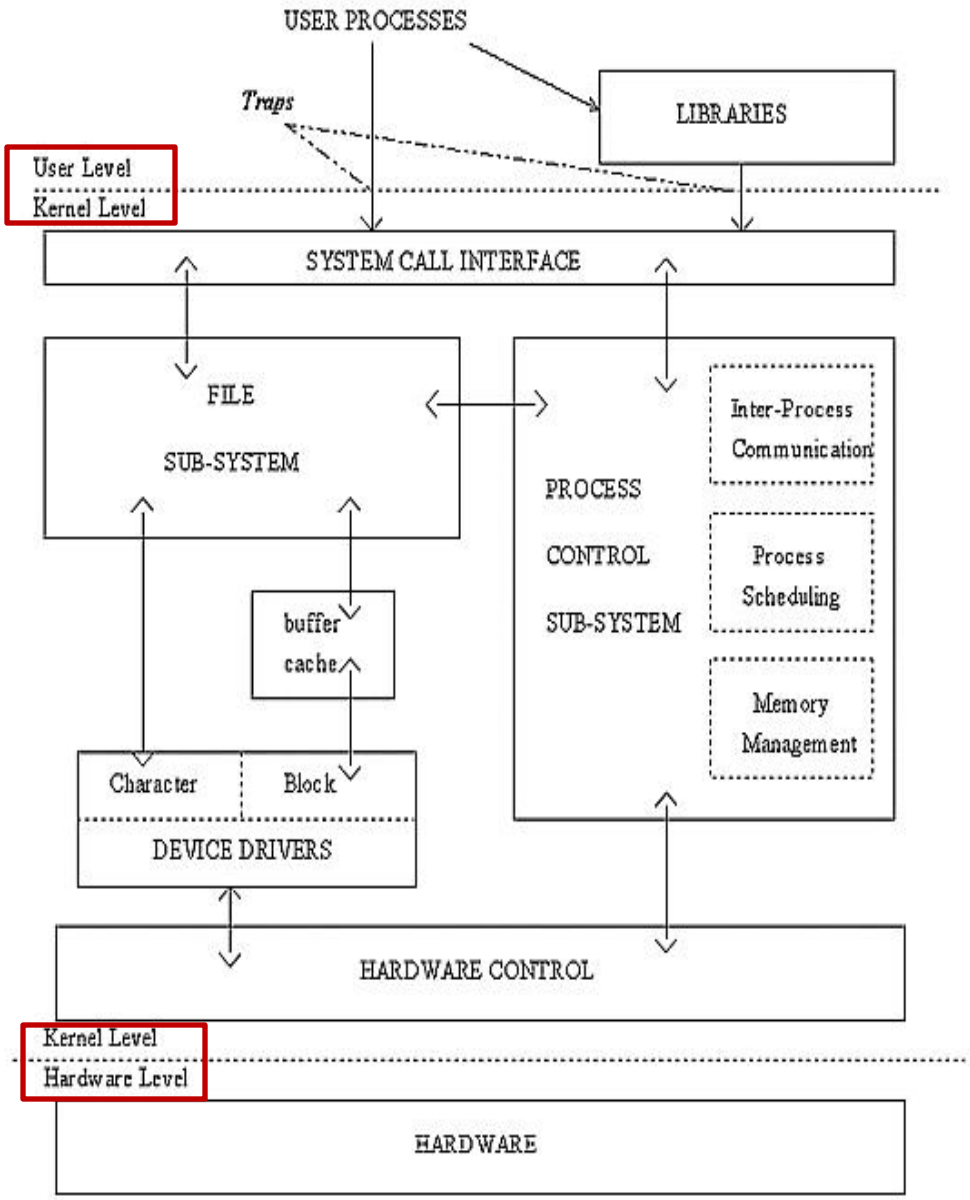
- The operating system itself consists of hundreds or thousands of programs, each specialized for particular OS tasks.
- Provides access to frequently needed facilities
- Shields user from low-level details
  - **but not system and network administrators!**
  - **User Interface: Command Line, Curses, GUI**

  The most complete administrative interface for Linux is the command line. GUI and Web based administration are possible, but don't always offer the same number of configuration options.

  Windows: powershell

Operating Systems

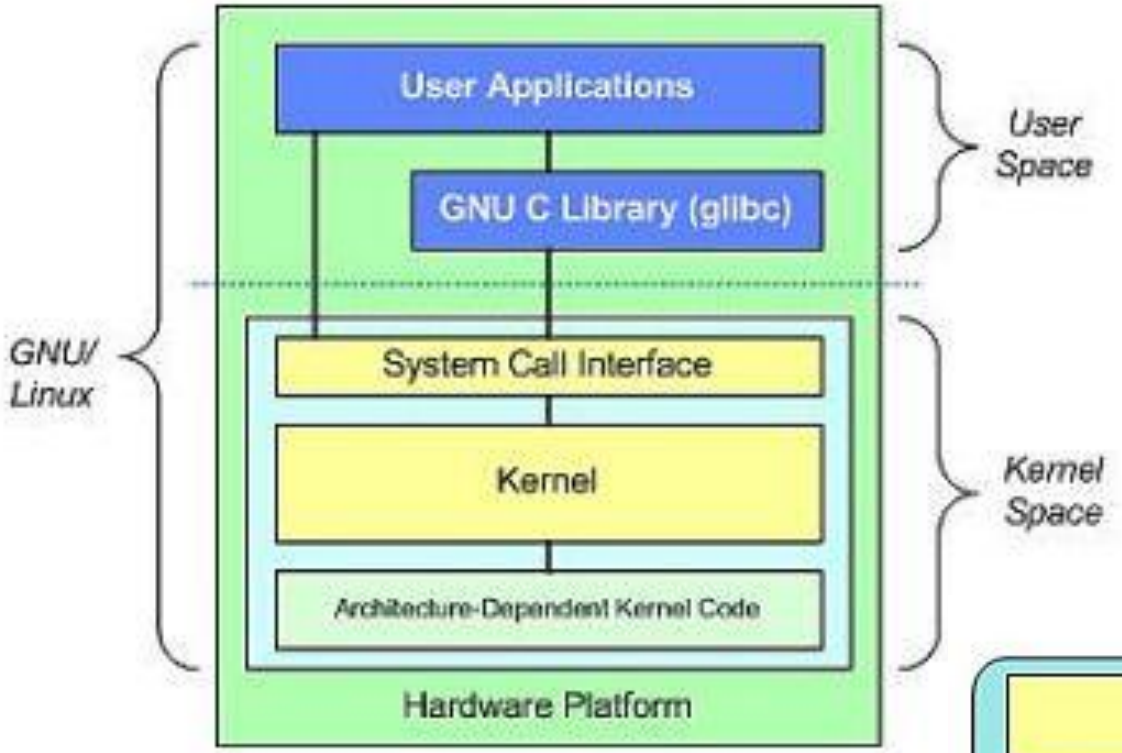
- *Application programs*: how to perform computing task
- OS allocates access to the hardware (CPU, memory, etc.) and controls their use
- *Like a government, the OS performs no useful function by itself – it simply provides an environment for other programs to do useful work.*
- Processes/threads: logical concurrency
  - multi-user and event-driven
  - client/server architecture
- Multi-core processors – real-time concurrency

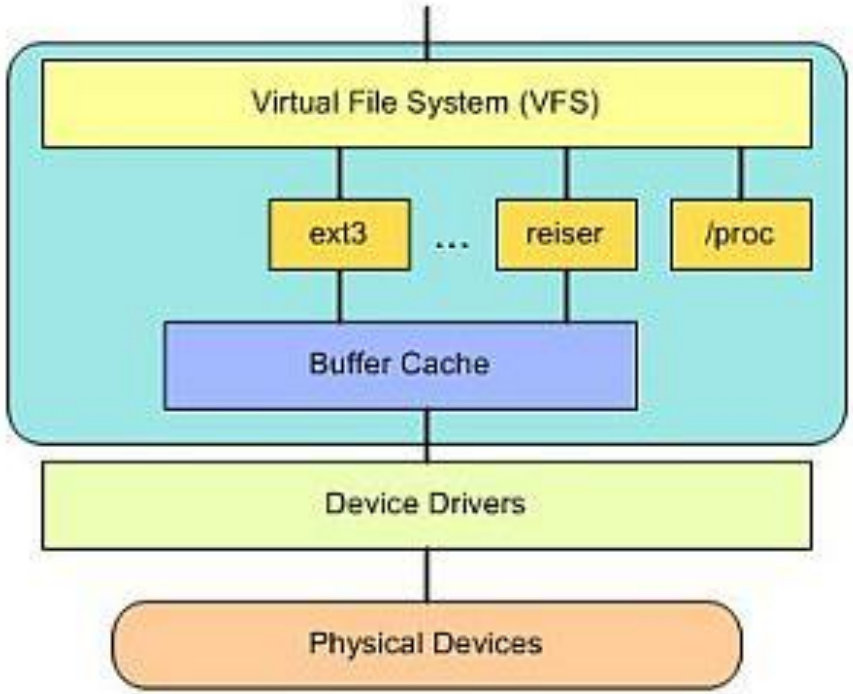


Block Diagram of UNIX Kernel

Device Drivers

- In kernel space, the virtual file system (VFS) decodes the file type and transfers the file operations to the appropriate channel, like a filesystem module in case of a regular file or directory, or a device driver in case of a device file.
- All of the functions used to access a specific device are jointly referred to as the device driver.
- Device drivers must be statically compiled into the kernel or stored on disk as a dynamically loadable module.


User Applications
GNU C Library (glibc)
System Call Interface
Kernel
Architecture-Dependent Kernel Code
Hardware Platform
GNU/ Linux
User Space
Kernel Space


Virtual File System (VFS)
ext3
...
reiser
/proc
Buffer Cache
Device Drivers
Physical Devices


Q
&
A

Processes

- Each process is a collection of resources:
  - Instance of a running program in RAM, current directory, open files with current position, space and time limits, etc…
- Processes are “time-sliced” by OS scheduler
  - Various policies and queues of waiting processes
- A process may contain concurrent paths of execution called *threads*
- To see the process hierarchy
  - Unix: ps or top
  - Windows: taskmanager
- Shutdown terminates all processes

Process Environment

Chains of processes share an *environment*

- Environment is inherited by new processes.
  - changes made in local environment are not returned back to parent process environment.

- Environment Variables
  - text strings
  - usually set by user or script
  - this is how command parameters are passed from parent to child.

daemon

A daemon is a background process that performs a specific function or system related task. (in Windows, a “service”)

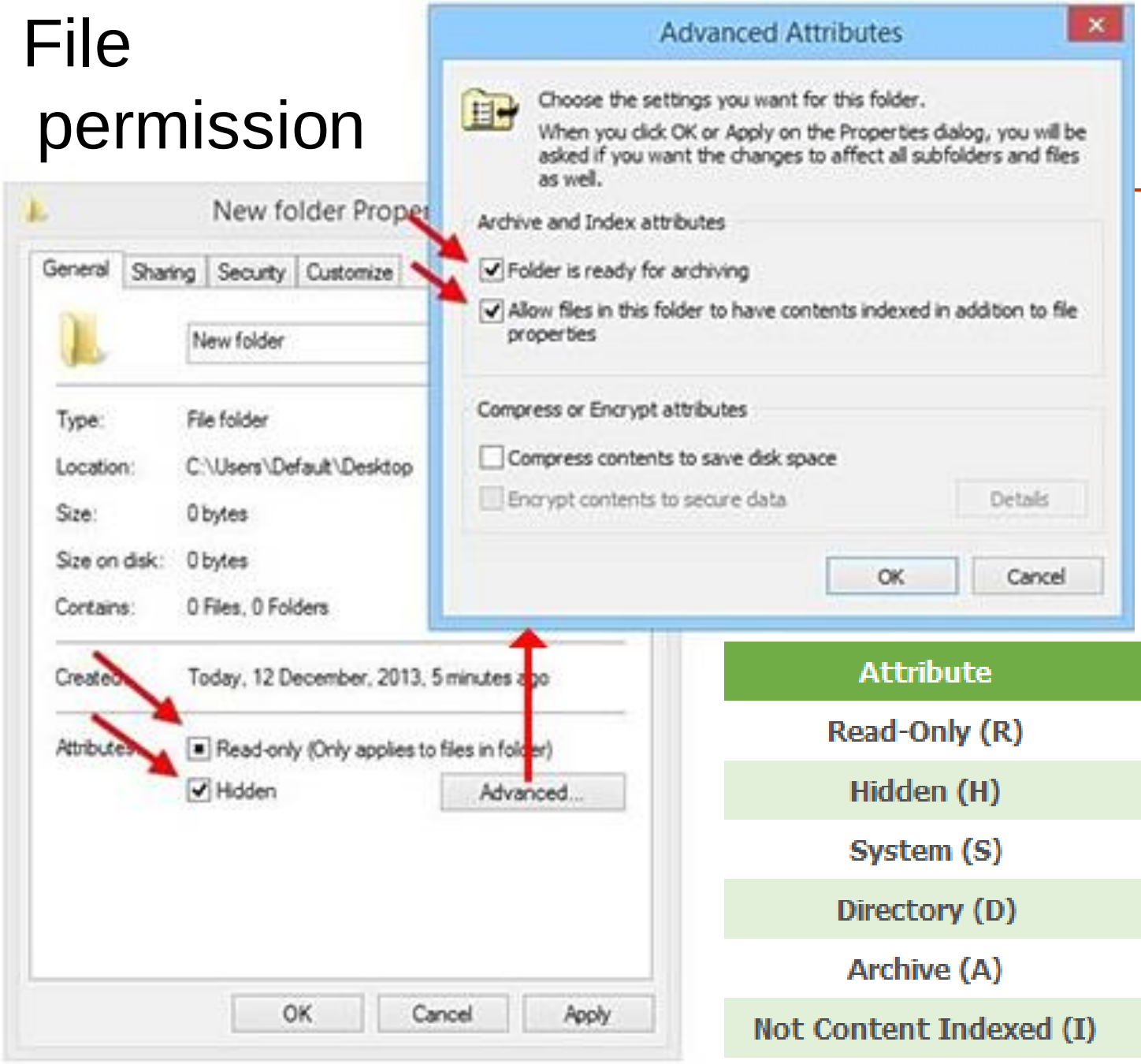
- Started at boot time – runs as long as the system is up
  - or started and stopped as needed


Q
&
A

Access Controls

- Every process and every system resource has a single owner
- Every owner has a unique User ID (UID) and belongs to one or more Groups identified by a Group ID (GID)
  - Unix groups are domains in Windows
- The owner can give a a group privilege to access the resource. Anyone else is just the world at large.
- Permissions limit reading, writing, execution
- Superuser privileges override all explicit privileges
  - root (unix), Administrator (Windows)

File permission



Attribute	Bit Value
Read-Only (R)	00000001
Hidden (H)	00000010
System (S)	00000100
Directory (D)	00010000
Archive (A)	00100000
Not Content Indexed (I)	10000000000000


Q
&
A

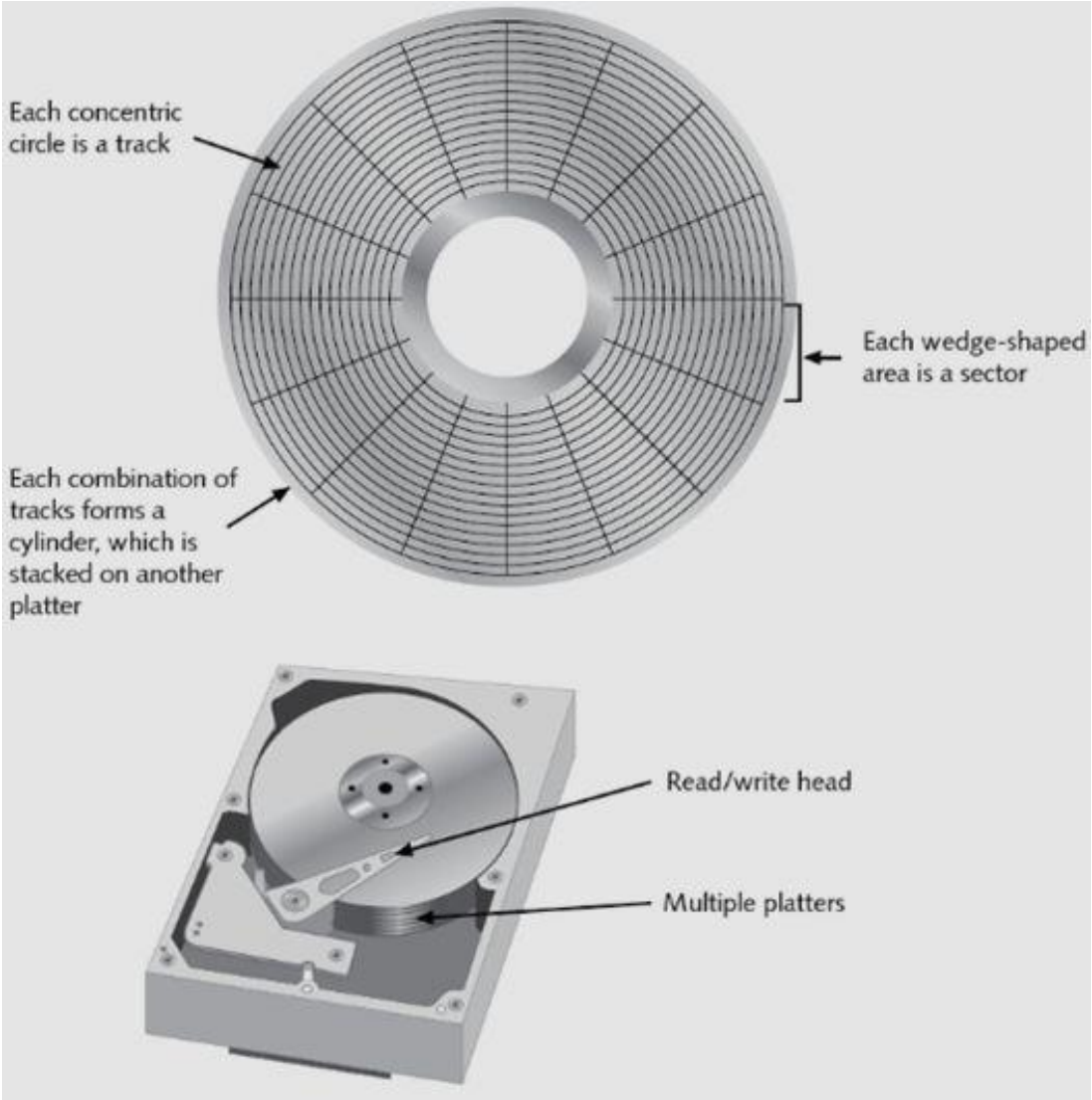
Hard Disk and Filesystem Fundamentals

Physical:

- Track
- Sector
- Cylinder

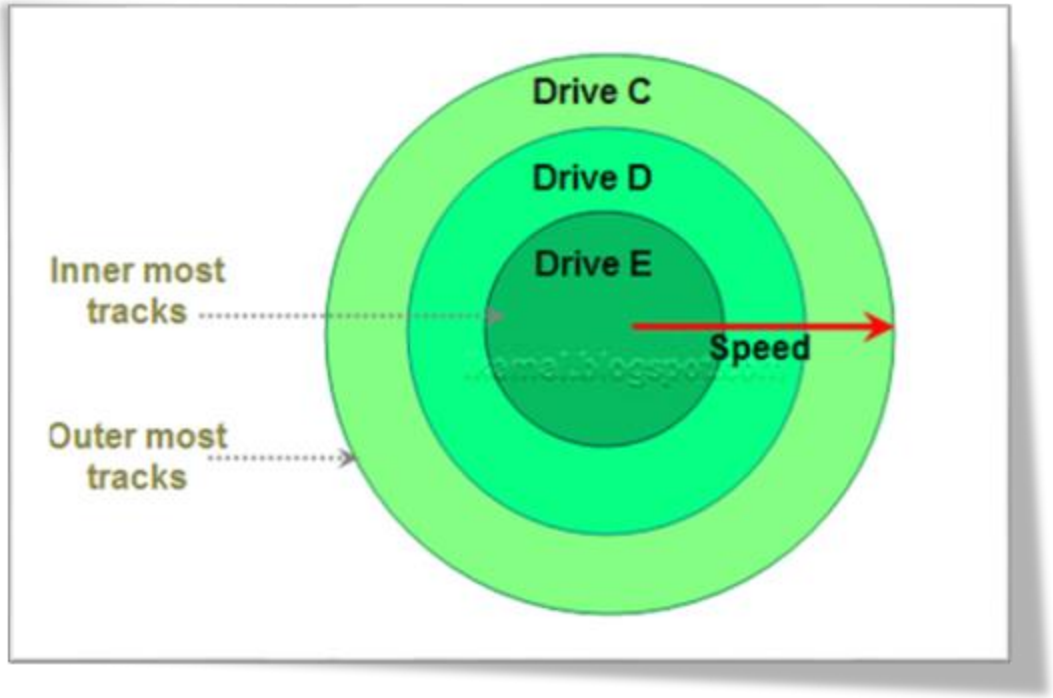
Logical:

- Partition
- Cluster



Disk Partition

- A partition separates a disk into logical “drives”
- A “DOS” (32-bit OS) partition can be one of two types:
- Primary : can hold information regarding the operating system which is required for booting.
- Extended: only for data files



Disk partitioning: `fdisk`

- Reserve space for a particular function
  swap space, software, user directories, shares
- Each partition is given a logical device name
  C:\ D:\ E:\ *etc*
  /dev/sda1 /dev/dsk/c0t0d0s0
- Each partition can use a different file system
- Logical volumes seamlessly span multiple partitions

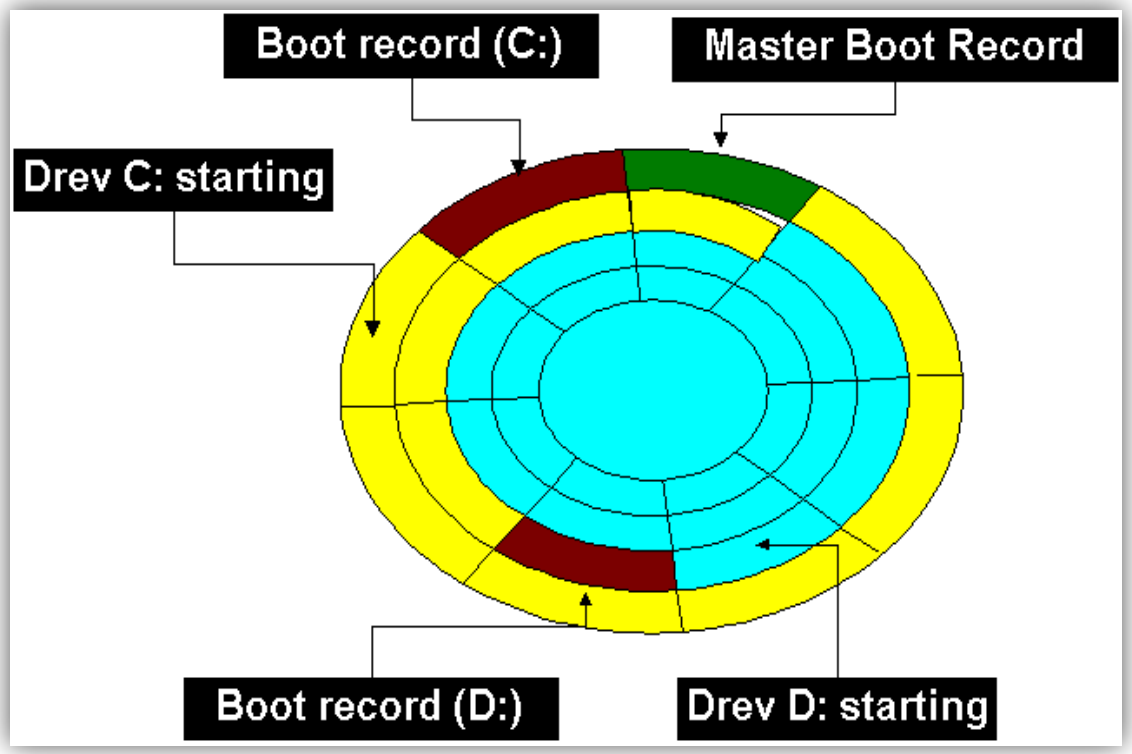
Use different partitions when

- They have different functions
- They are owned/maintained differently
- They change at different rates
- Backup policy is different for each

Master Boot Record

- The master boot record (MBR) is the first **sector** of a hard disk

*It includes a table that contains information about each **partition***



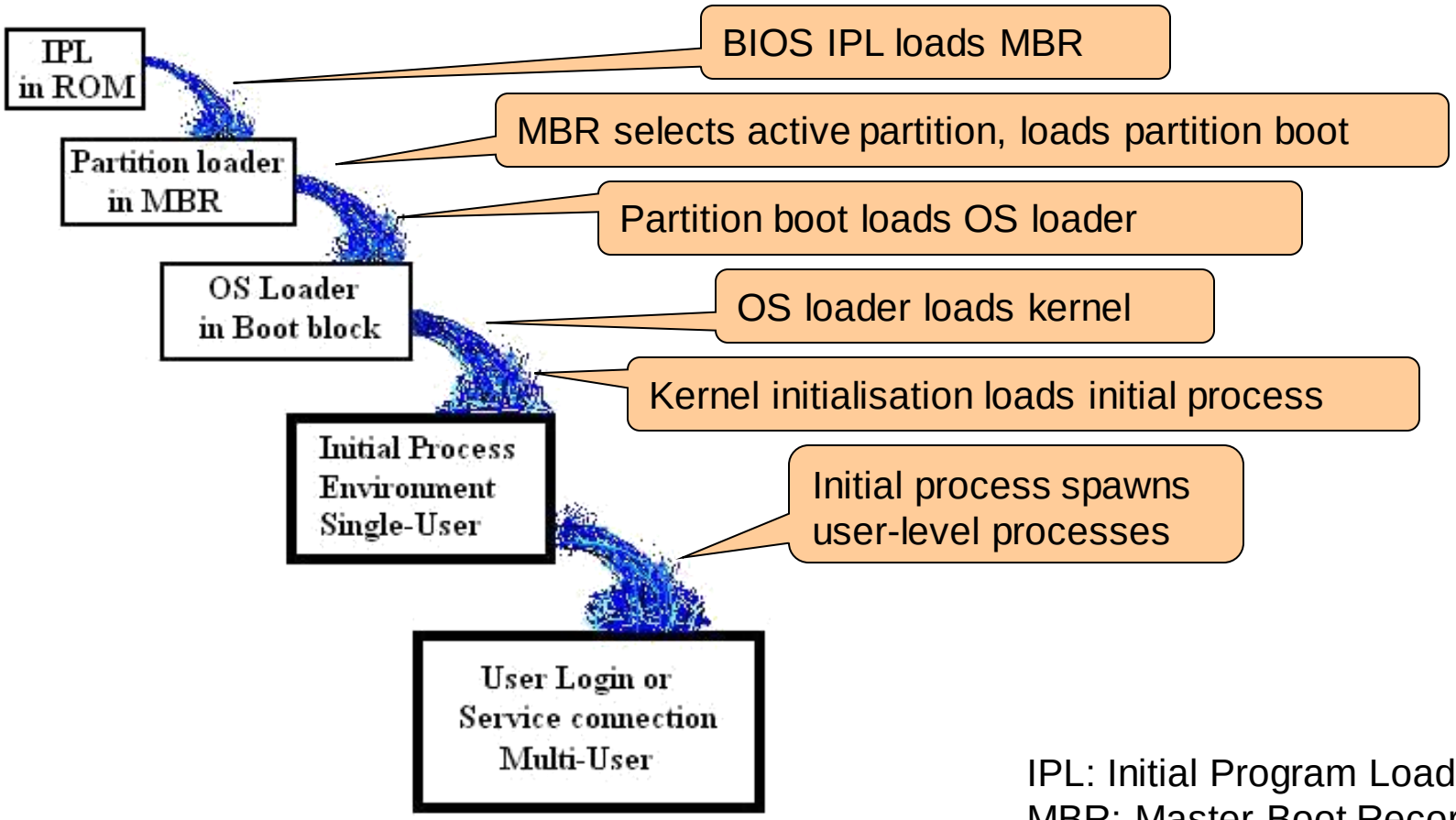
The diagram shows a disk with two primary partitions

The MBR also includes a program that reads the boot sector record of the partition containing the operating system into RAM.

Multiple Operating Systems – Boot Managers

- With multi-use machines and big disks it is possible to have several different OSs
- Each OS has its own boot manager in the MBR
- Some are generalised, some not
  - For example, Windows relies on files accessed from drive `C:`
  - so install Windows first, then install Linux
- Unix loaders:
  - LILO (LInux LOader),
  - syslinux (common for iso)
  - GRUB (GRand Unified Bootloader)

PC Bootstrap Sequence -- An *Avalanche* boot

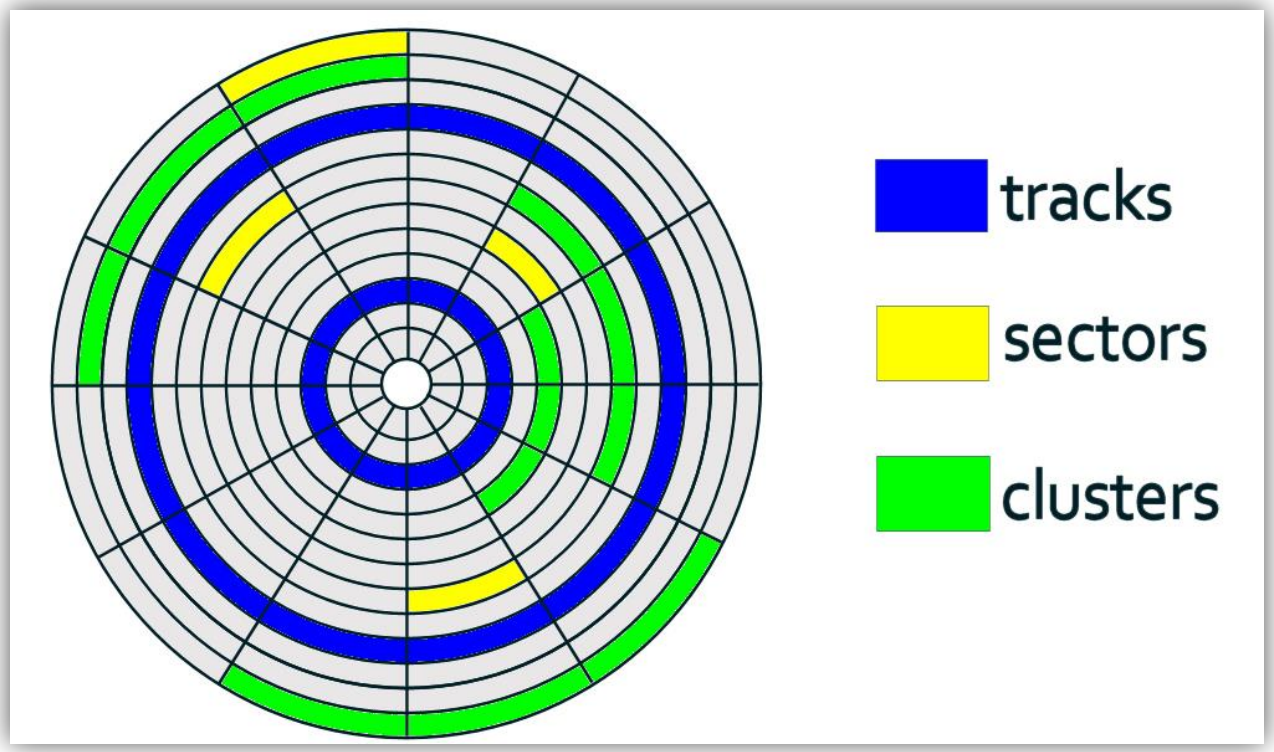


IPL: Initial Program Loader
MBR: Master Boot Record


Q
&
A

Cluster

- A group of **sectors** is called a **cluster**.
- The file system divides the storage on a disk volume into **clusters** for efficient disk usage and performance.



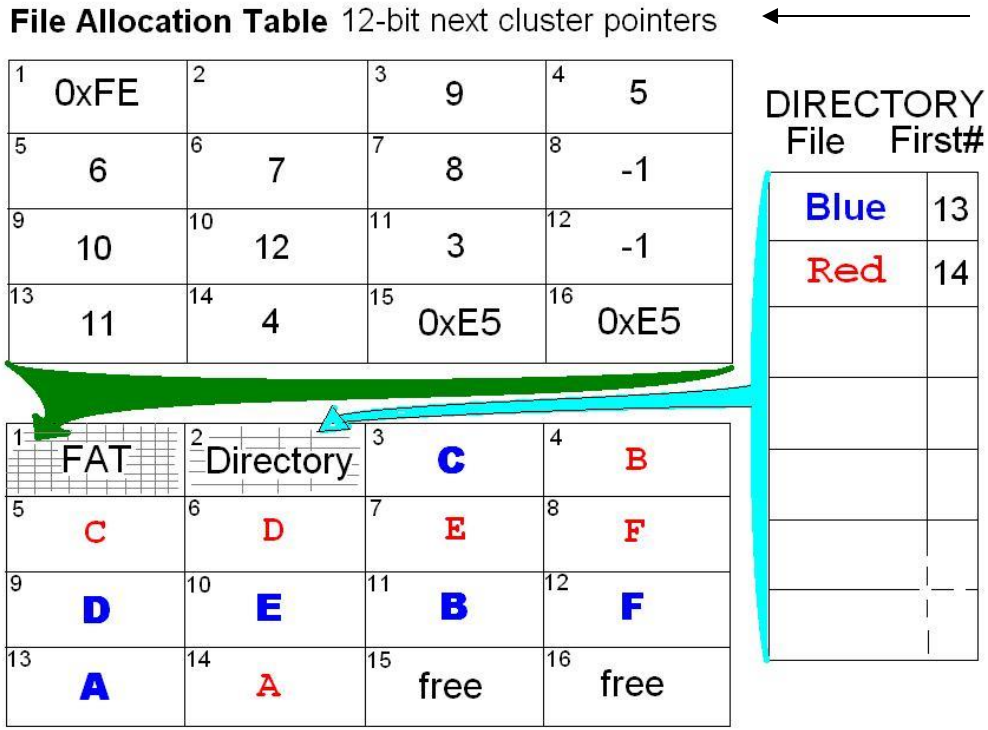
File Systems

- Structures disk area for addressable access
  - "like painting car spaces in a carpark"
- Usually unique to OS
  windows: FAT, FAT32, NTFS
  linux: ext2, ext4,
  linux is one huge computer science project!
- Physical *sectors* grouped into allocation units called *blocks* in Unix and *clusters* in Windows
- Building File System - `mkfs` or `format`
  boot sector, data area, free list

Swap Space

- *Swapping* frees RAM by storing an image of an idle process on disc - few modern OSs actually do swapping
- The *swap file* is now used for *paging* – virtual memory stored on disc
  - In Unix the swap file is traditionally a partition
  - In Windows the swap file is always a file
- Recommended size 2.5 * RAM
  - Any more will probably never be used

Example: MS-DOS File Allocation Table



Original FAT only allowed “8.3” filenames

Executable files recognised by .EXE, .COM, .BAT

FAT32 allows long filenames

NTFS - more efficient use of clusters

Unix filesystems use an analogous scheme of blocks with an inode table

Windows file Systems

File system	Maximum filename length	MBR	Allowable characters in directory entries	Maximum pathname length	Maximum file size	Maximum volume size
NTFS	255 characters	0x07	Depends on namespace used	32,767 Unicode characters with each path up to 255 characters	16 EB	16 EB
FAT32	8.3	0x0B 0x0C	Any byte except values 0-31, 127 and: " * / : < > ? \ | + , . ; = [] (lowcase a-z are stored as A-Z).	No limit defined	2 GB	2 TB

Linux File Systems

File system	Maximum filename length	MBR	Allowable characters in directory entries	Maximum pathname length	Maximum file size	Maximum volume size
ext3	255 bytes	0x83	Any byte except NUL and /	No limit defined	2 TB	32 TB
ext4	255 bytes	0x83	Any byte except NUL and /	No limit defined	16 TB	1 EB

- **Most of the Linux is using ext* family**


Q
&
A

Mounting Disks / Partitions

- Naming conventions for devices
  - Disk: `/dev/sda` `/dev/sda1` …
    (usb will be recognised as this)
  - CD/USB: `/dev/sr0` …

- `mount /dev/sda1 /mnt/sda1`
  access everything on the drive through the directory instructions in /etc/fstab

- `umount /mnt/hostshare`
  - cannot unmount a busy filesystem
  - show open files on a filesystem: `lsof`

Paths

`/` *root directory* `.` *here* `..` *up one level*

`/tmp/foo` *absolute* `tmp/foo` *relative* `../../foo`

`cd` *change directory* **`mv`** *move or rename*

```
cd /etc
cd foo
cd ..
mv foo ofo          rename
mv ofo /var/opt     move
cd /var/opt/ofo
./myscript.sh       execute a file stored here
```

Symlink

- Soft link or symbolic link (symlink): a small file that is a pointer to another file.
  - contains the path to the target file instead of a physical location on the hard disk; can point across partitions.
  - Note that removing the target file for a symbolic link makes the link useless.
  - Windows: shortcut (right click : file : properties)
- used to create a new name to satisfy dependencies
  - a program that expects a file to be in another location
- Symbolic links can generally save a lot of work


Q
&
A

Shutdown

Complex systems need safe shutdown sequence to avoid damage

Quiescent state difficult to predict in multi-tasking systems

- Complete all operations in progress
- Prevent new operations from starting
- Close files
- Terminate processes and services
- Synchronise and Flush buffers/caches
- Dismount/park/eject disks
- Power off !

Caution!

Running VMs on a laptop:

Check your power settings – close lid, low battery, etc.

Never **Hibernate**, only **Sleep**

("Hibernate" suspends too many host processes, and your VM will get corrupted – "Sleep" works well enough)

(or … just poweroff the VM)


Q
&
A

Shared Libraries (.so)
Dynamic Link Libraries (.dll)

- Loaded into RAM on demand
- Managed by some kernel routines which use an "index" to locate a required module
  - Special command used to do this – `ldconfig`
- **Must have the right libraries on your system**
  - **dependencies** – missing or wrong version, cannot start
  - **packages** – bundle libraries, but often depend on others
  - **package managers** – help sort out dependencies

It is very likely that you will actually need to configure these, for one application or another

Source and Packages

- A Linux distribution is a collection of utilities bundled around the Linux kernel.
- Source code is the program in text file format, usually written in the language C
- A binary file is the result of compiled source code.
- A dependency is a component of the system that must already be installed before another program will function.
  - Some, but not all, compilation scripts will attempt a **dependency check** prior to installation.

Source and Packages

Packages are pre-configured binary files for specific distributions.

- Usually a compressed archive:

  `.tar.gz  .tgz  .lzip  .bzip  .rpm  .deb  {.cab}`

Package managers keep track of which packages have been installed, and perform a dependency check when you install software

Some distributions offer a tracking service that will notify you when new versions of installed packages are available.

Others automatically download packages from an official repository

Debian/Ubuntu	**RedHat**	**Slackware**
Local – dpkg	Local – rpm	Local – pkgtool
Auto – apt-get	Auto – yum	Auto – *(none)*

Celebrate Diversity!


Q
&
A