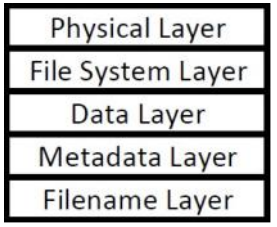
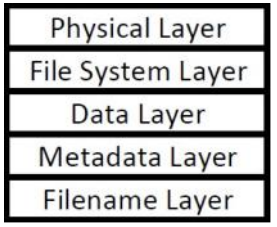
- To describe how data is stored and accessed on a physical and logical storage device

- To describe how to view NTFS (New Technology File System) information for a volume in windows

- File Systems
  - Physical Layer
  - File System Layer
  - Data Layer
    - Allocated/unallocated blocks/clusters
    - Contiguous/non-contiguous file storage
    - FAT: description & history
  - Metadata Layer
    - NTFS: ACL, Permissions, Security (Lab)
  - Filename Layer
  - BitLocker Encryption
  - Volume Sets & RAID



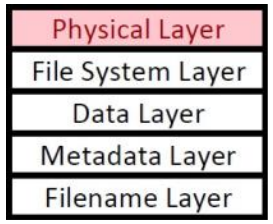
File System



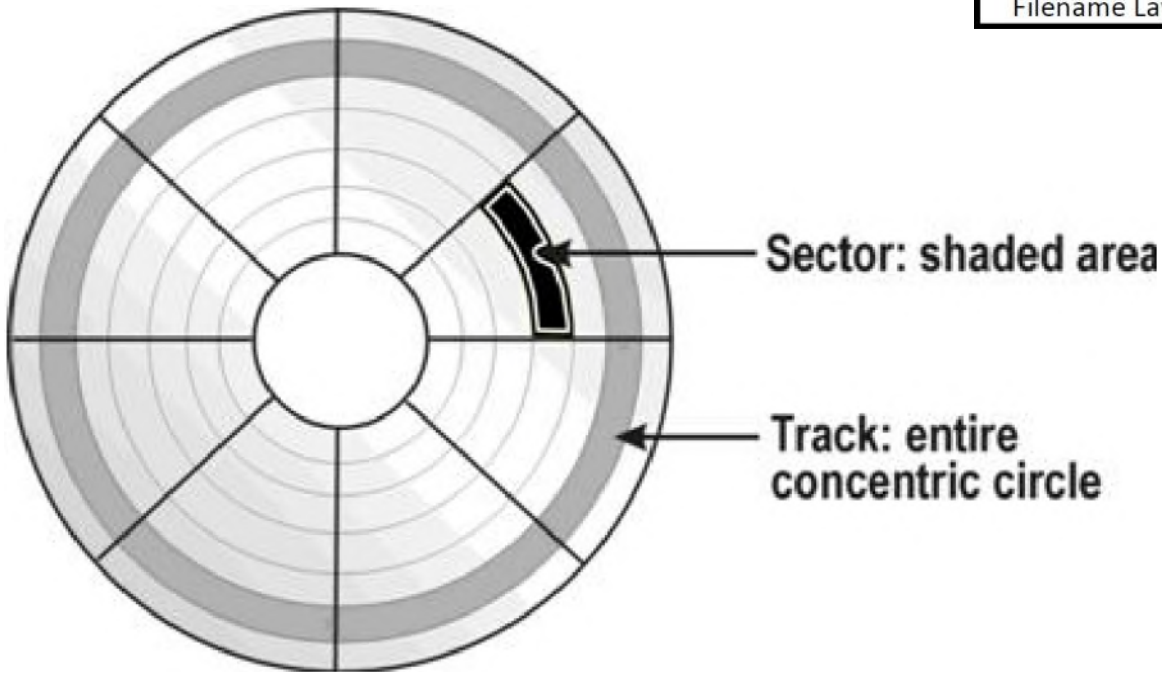
- Defines the way data is named, stored, organized, and accessed on a disk volume.
- Each system has its own properties and features
- Organized into five layers:

1) Physical layer
2) File System layer
3) Data layer
4) Metadata layer
5) Filename layer

Physical Layer

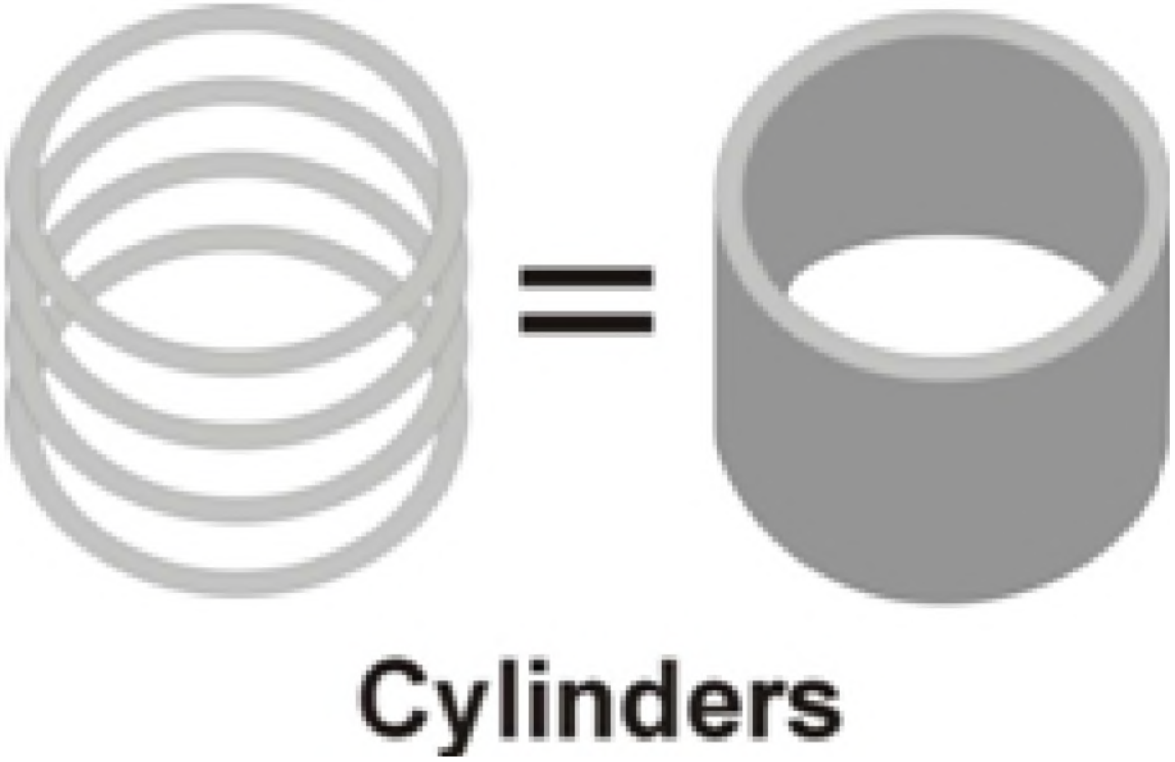


- Disk Geometry
  - Track
  - Sector
  - Head

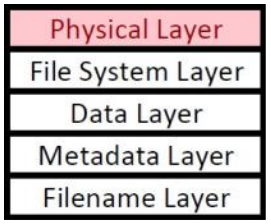


Physical Layer

- Disk Geometry (cont'd)

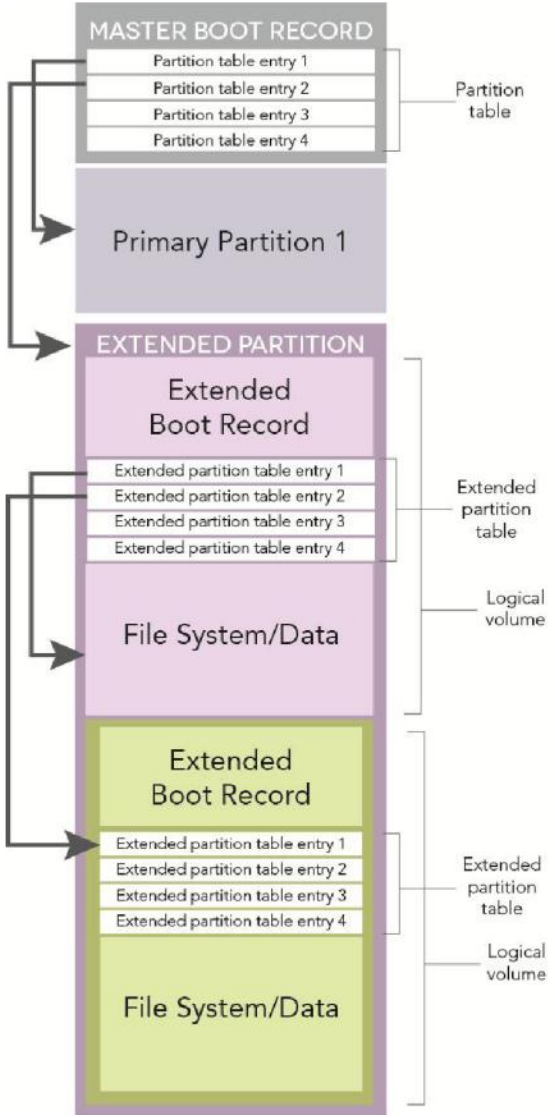


Partitioning



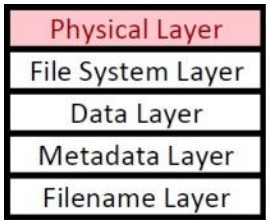
- Partitioning:
  - the process of identifying space on the hard disk to be used by the file system
- Partitioning Types
  - Master Boot Record
    - Primary Partition
    - Extended Partition
  - Globally Unique Identifier (GUID) Partitioning

Partitioning



Physical Layer
File System Layer
Data Layer
Metadata Layer
Filename Layer

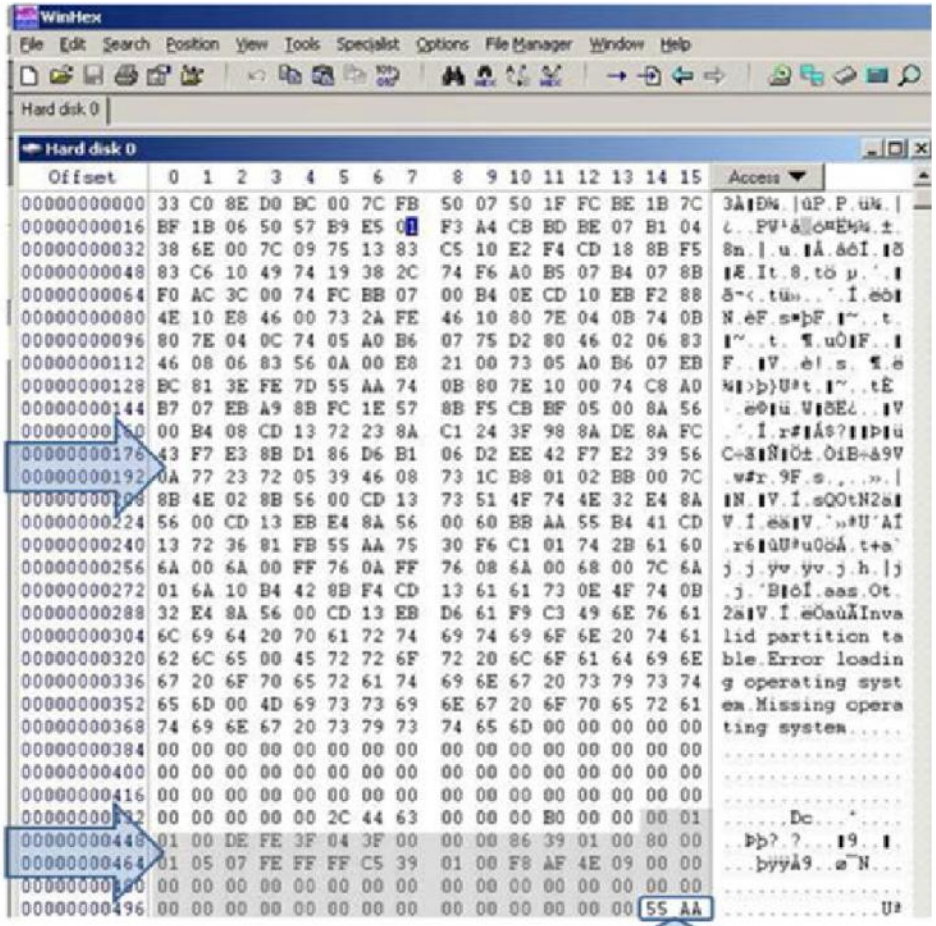
Master Boot Record



- Master Boot Record (MBR):
  - one of the most significant structures on a hard disk
  - resides at the first physical sector of the drive (sector 0) and is not part of any partition
  - is the first sector read from the boot device

Master Boot Record

- Physical Layer
- File System Layer
- Data Layer
- Metadata Layer
- Filename Layer

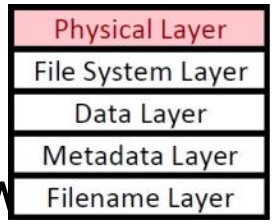


First Stage Boot Loader

Partition Table

Magic Number

Master Boot Record



- contains the partition table and instructions on how to continue the boot process
  - the first stage boot loader (first 446 bytes)
  - the partition table (next 64 bytes), and
  - a two-byte magic number (0x55AA).

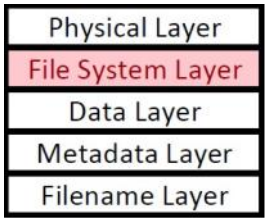
Partition Table

Physical Layer
File System Layer
Data Layer
Metadata Layer
Filename Layer

Offset (Dec)	Length Bytes	Content
0	1	State of Partition: 00 if not active, 80 if active
1	1	Head where the partition starts
2	2	Sector and cylinder where the partition starts
4	1	Type of partition
5	1	Head where the partition ends
6	2	Sector and cylinder where the partition ends
8	4	Distance, in sectors, from the partition sector to the first sector of the partition. (How far from the MBR sector is the starting sector?)
12	4	Number of sectors in the partition.

Hex Value	Type
0x01	FAT 12
0x0e	FAT 16
0x0c	FAT 32
0x83	Linux Native
0x82	Linux Swap
0xA5	BSD/386
0x05	Extended
0x07	NTFS
0xde	Unknown

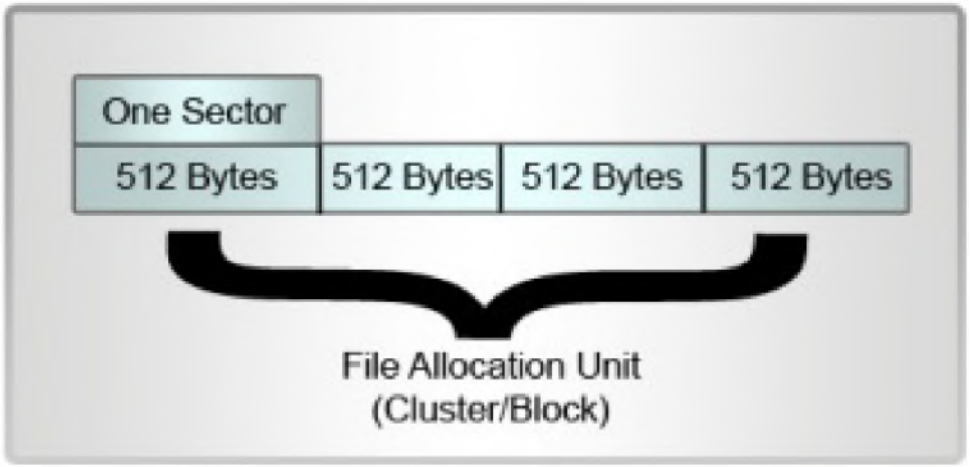
File System Layer



- Contains data that describes the file system structural details, such as:
  - file allocation unit sizes,
  - structure offsets, and
  - mounting information.
- This data is usually located in the first sector of the file system, typically in a file system data structure called a superblock or boot sector.

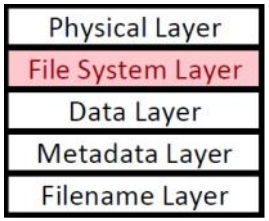
File Allocation Unit

Physical Layer
File System Layer
Data Layer
Metadata Layer
Filename Layer



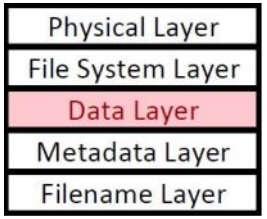
File Systems for Operating Systems

Type of Primary File System	Operating System	Includes Encryption	Max File Size	Max Volume Size	Supports Journaling
FAT32	Windows 9X, ME (DOS-based kernel)	No	4 GB	2 TB*	No
NTFS	Windows NT, 2000, XP, Vista, 7, 8, Server (NT-based kernel)	Yes	16 EB*	18 EB*	Yes (for system update files only)
HFS+, HFSX	Mac OS X	Yes	8 EB	8 EB	Yes
ext3	Linux	Yes	2 TB	32 TB	Yes
ext4	Linux	Yes	16 TB	1 EB	Yes



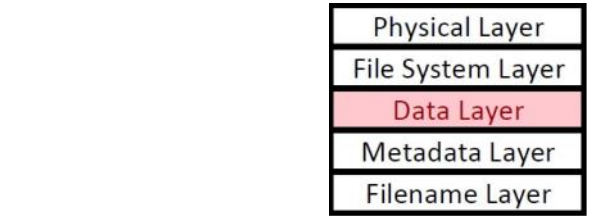
The (*) denoted entries have further limitations as defined by the OS. For example, while the code base for NTFS can address up to an 18-exabyte volume, Windows systems' programming supports only up to 256 TB volumes.

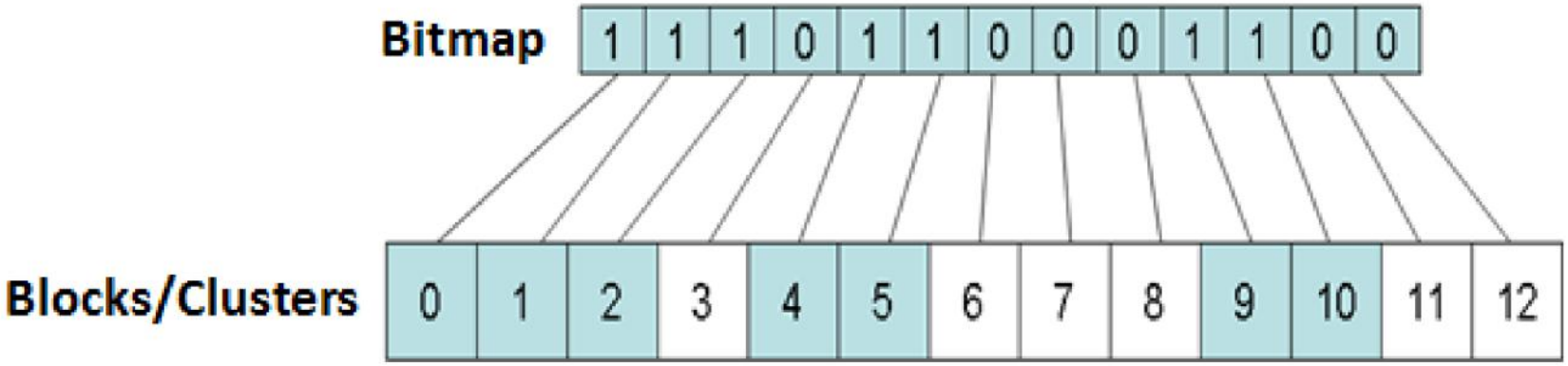
Data Layer



- Is where the actual file and directory data is stored
- Each cluster/block is given a logical address used by the file system to locate and store data on the disk
- Blocks/clusters are flagged in one of two states by the file system:
  - allocated
  - unallocated

Bitmap

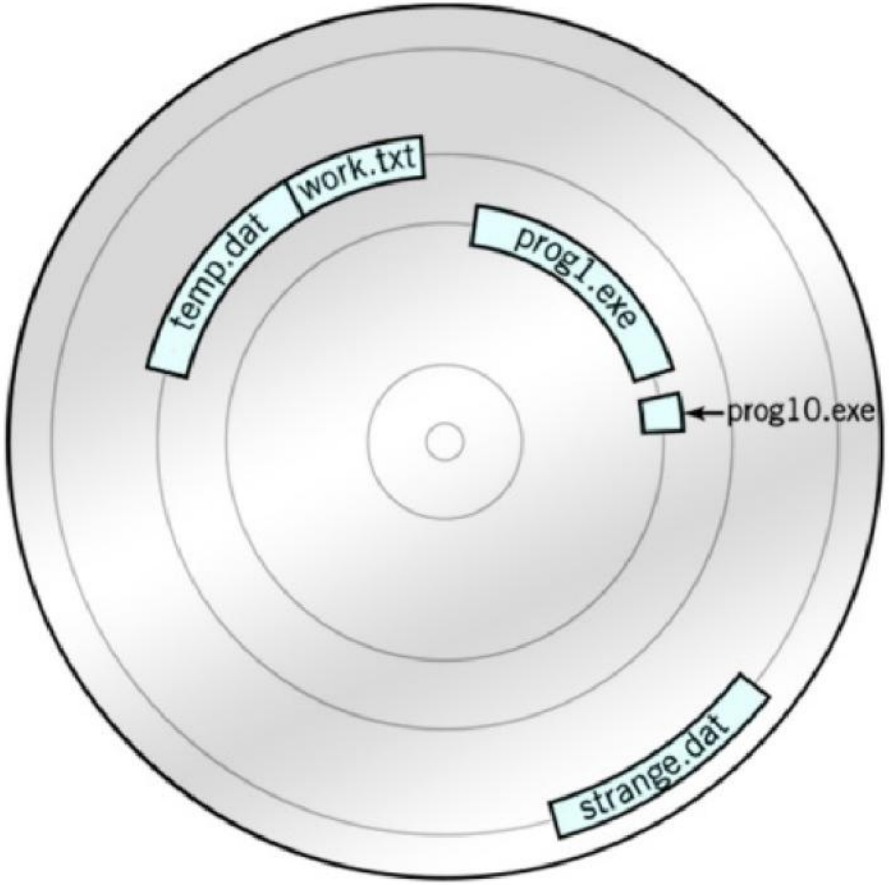

Physical Layer
File System Layer
Data Layer
Metadata Layer
Filename Layer


Bitmap
1 1 1 0 1 1 0 0 0 1 1 0 0
Blocks/Clusters
0 1 2 3 4 5 6 7 8 9 10 11 12

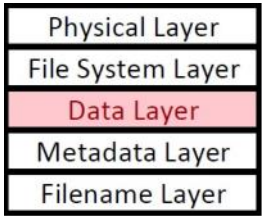
File Allocation

Physical Layer
File System Layer
Data Layer
Metadata Layer
Filename Layer

Contiguous File Storage



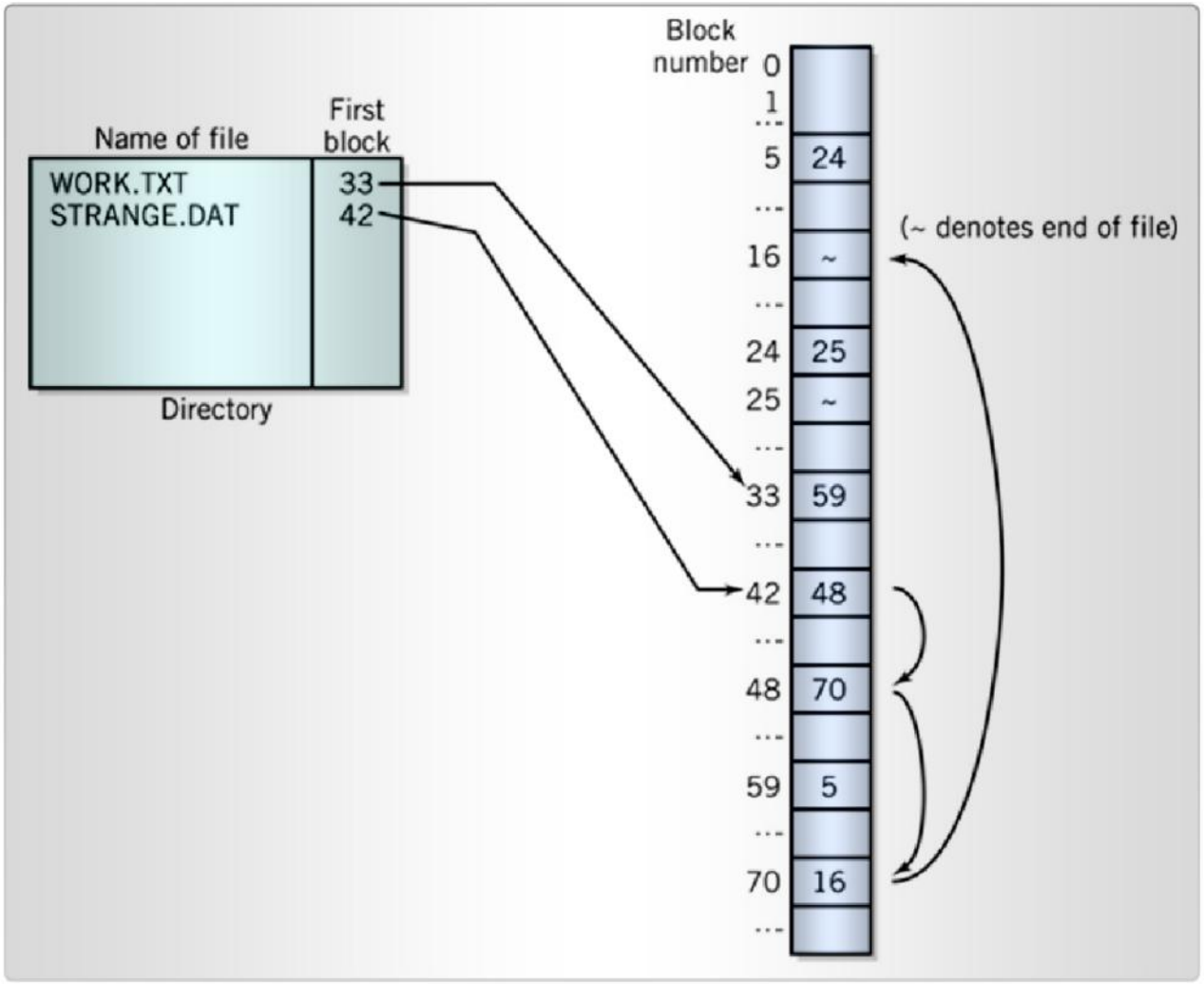
File Allocation (2)



Non-contiguous File Storage

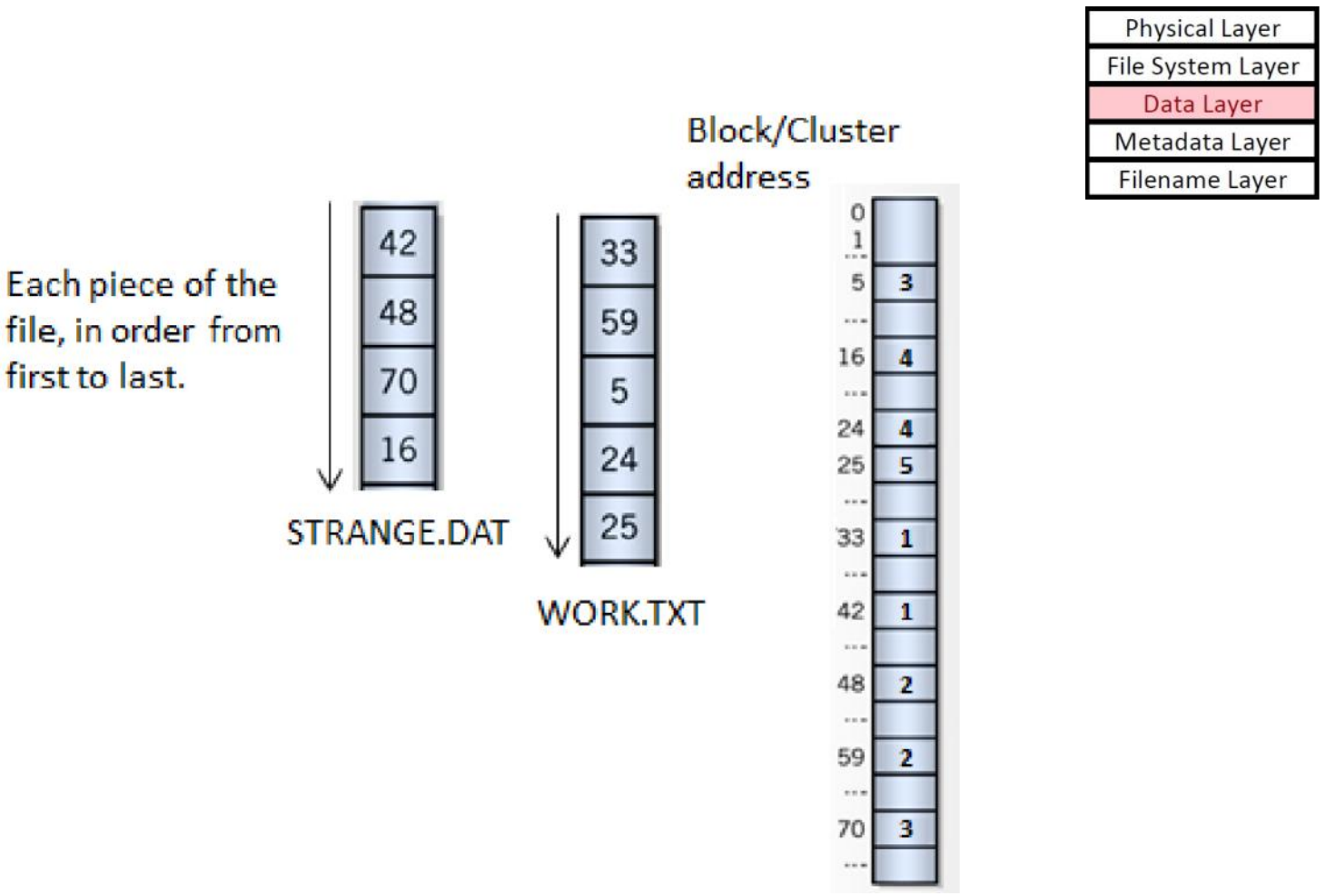
- Used only when contiguous blocks/clusters are unavailable to fit an entire file, will the file system fragment it.
- This noncontiguous file allocation occurs when data is stored in non-sequential block/cluster addresses.
- There are two methods for keeping track of where all the parts of a file are addressed and stored:
  - linked allocation, and
  - indexed allocation.

Linked File Allocation

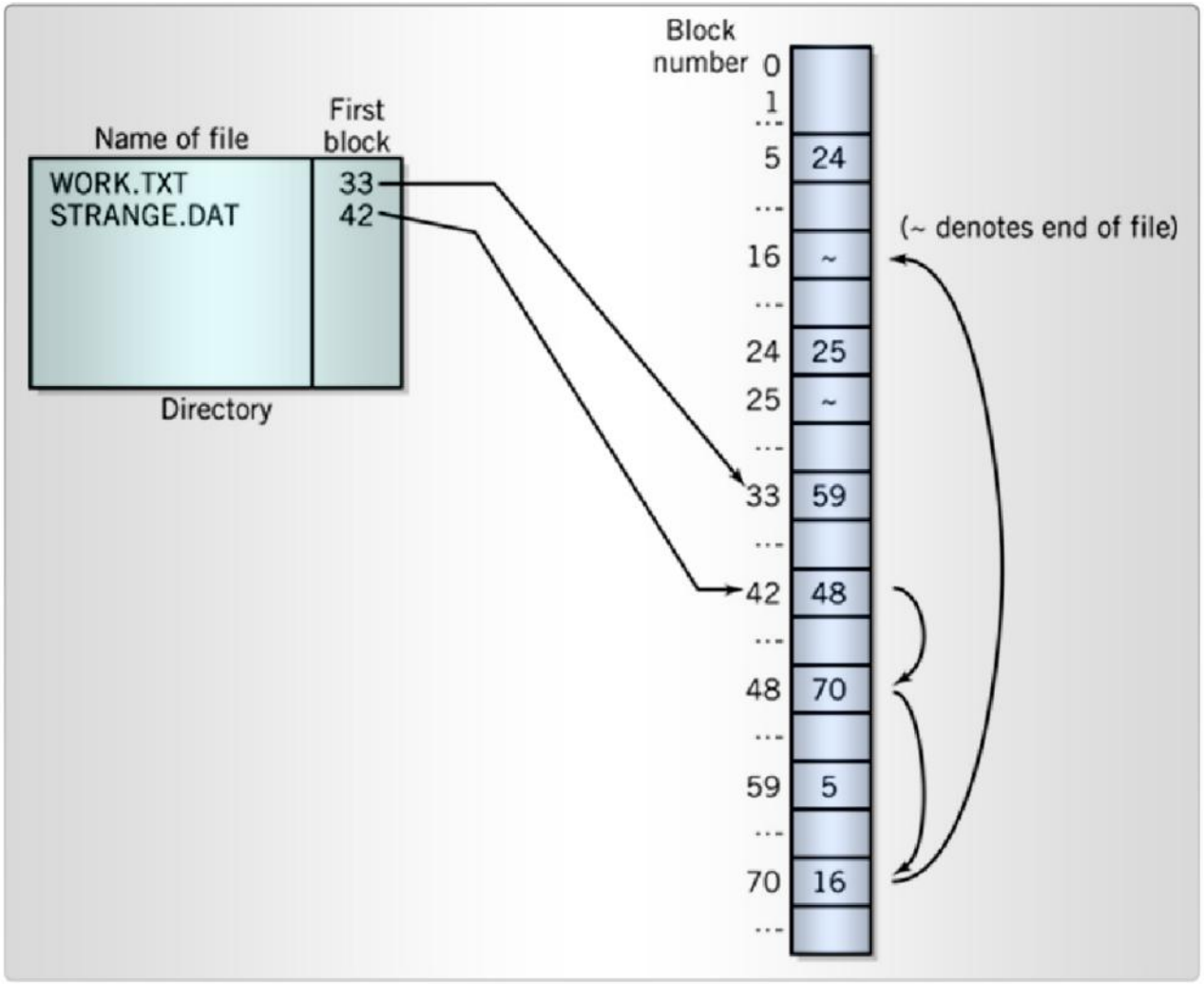


Physical Layer
File System Layer
Data Layer
Metadata Layer
Filename Layer

Index File Allocation

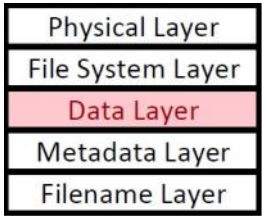


File Allocation Table (FAT)



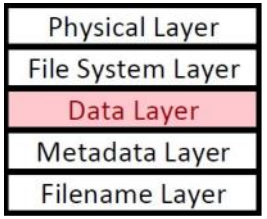
Physical Layer
File System Layer
Data Layer
Metadata Layer
Filename Layer

FAT (cont)



- Non-proprietary industry standard since 1977
- Originally designed for small disks and simple folder structures
- Used in MS-DOS and Windows 9x
- Still found in USB sticks, flash drives, memory cards:
  - Used on many portable/embedded devices
  - Digital cameras, etc.

FAT (cont)

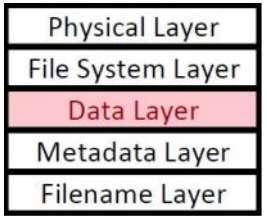


FAT32

- Introduced in 1996
- Increased max file size to 4GB
- Increased max volume size to 2TB

exFAT

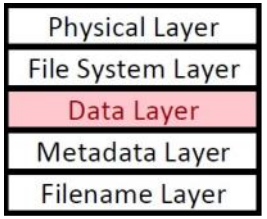
- Introduced in 2006
- Max file size 16EB (ExaBytes)
- Max volume size 128PB (PetaBytes)



New Technology File System (NTFS)

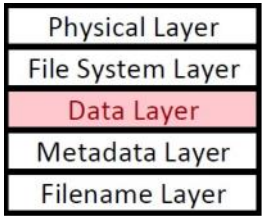
- Proprietary Microsoft file system
- Used in all modern versions of Windows
  - 2000, XP, 2003, Vista, 7, 8
  - Server 2003, 2008
- Supports file-level security, compression & auditing
- Designed to be high-performance and self-healing

NTFS (cont)



- Basic entity in NTFS is a volume
- Volumes may be a fraction of a disk (partition) or span multiple disks (RAID)
- Every file is described in a Master File Table (MFT)
  - Record range from 1-4KB in size (set at volume creation)
  - First 16 entries are attributes of the MFT
- NTFS deals with clusters
  - A number of sectors that is a power of 2
  - Cluster size is set when the file system is formatted
  - Allows for very large volumes (2^{64} clusters, each cluster is less than or equal to 256TB)

Format Command

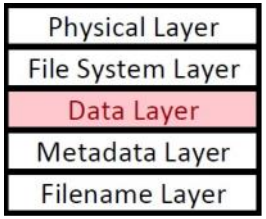


- Installs the boot record on the disk along with the root directory

- To format a partition, type the format command following by the letter of the drive containing the partition to be formatted a colon

- Example: format F:

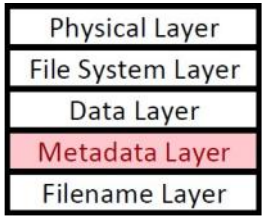
Format (cont.)



- Common switches:
  - **/FS:** filesystem – Specifies the type of file system (FAT, FAT32, or NTFS)
  - **/V:** label – Specifies the volume label
  - **/Q** – Quick Format: Creates the file system structures for the volume but does not scan for bad areas
  - **/C** – NTFS only: Files created on the new volume are compressed by default
  - **/P** – Windows Vista and later only: This switch zeros, or wipes, a partition. For example, format /P:4 D:

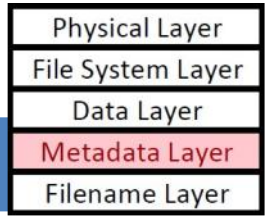
    → wipes the D: drive with zeros four times

Metadata Layer



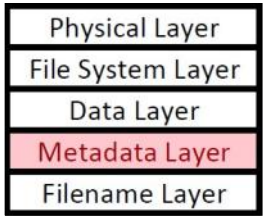
Metadata Field	Description
Type	Needed if system supports different file types; also used for special attributes such as *read-only*, *system*, *hidden*, *archive*; alphanumeric character or binary; sequential or random access required and so on.
Size	Size of the file in bytes, words, or blocks/clusters.
Maximum Allowable Size	Maximum size file is allowed to be.
Location	Pointer to where file is stored on disk.
Protection	Access control data limiting who has access to a file.

Metadata Layer (cont)



Metadata Field	Description
Name of Owner	File owner's user ID; used for protection.
Name of Group	Name of group with privileges
Creation Timestamp	When file was created
Modification Timestamp	When file was modified. Sometimes user identification is also maintained for audit purposes.
Access Timestamp	When file was last accessed

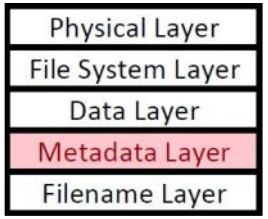
NTFS (cont)



Access Control List and Permissions

- Unlike FAT file systems, NTFS has the ability to track file ownership and control permissions – the level of access a user has to a file or folder.

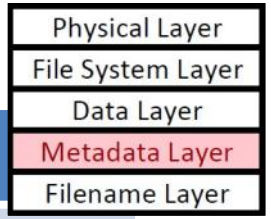
- An Access Control List (ACL) stores the permissions that pertain to a given file or folder.



File Permissions

Permission	Allows the User to …
Read	Read the file and view file attributes, ownership and permissions
Write	Overwrite the file, change file attributes, and view file ownership and permissions
Read & Execute	Run applications, plus perform the actions permitted by the Read permission
Modify	Modify and delete the file, plus perform the actions permitted by the Write, Read and Execute permissions
Full Control	Change permissions and take ownership, plus perform the actions permitted by all other NTFS file permissions

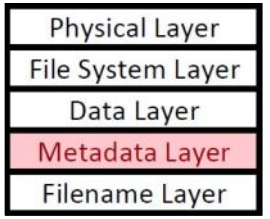
NTFS (cont)



Folder Permissions

Permission	Allows the User to …
Read	See files and subfolders in the folder and view folder ownership, permissions and attributes
Write	Create new files and subfolders with the folder, change folder attributes, and view folder ownership and permissions
List Folder Contents	See the names of files and subfolders contained within a folder
Read & Execute	Navigate through folders to reach other files and folders, even if the users do not have permission for those folders; perform actions permitted by the Read and List Folder Contents permissions
Modify	Delete the folder, and perform actions permitted by the Write and Read and Execute permissions
Full Control	Change permissions, take ownership, delete subfolders and files, and perform actions permitted by all other NTFS file permissions

NTFS (cont)

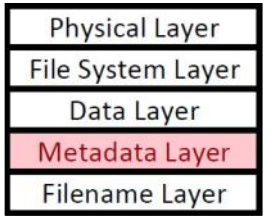


Permission Inheritance

- Permissions assigned to a specific folder are inherited by the subfolders and files contained within that folder
- Put another way, all permissions assigned to the parent folder affect any existing files and subfolders, as well as all subsequently created new files and subfolders

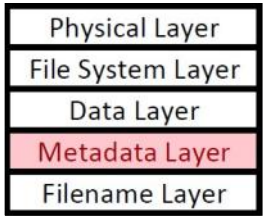
→ However, **this rule has exceptions** because permissions can be set on a single document that can override the permissions set on the folder in which the document resides.

NTFS Cumulative Permissions



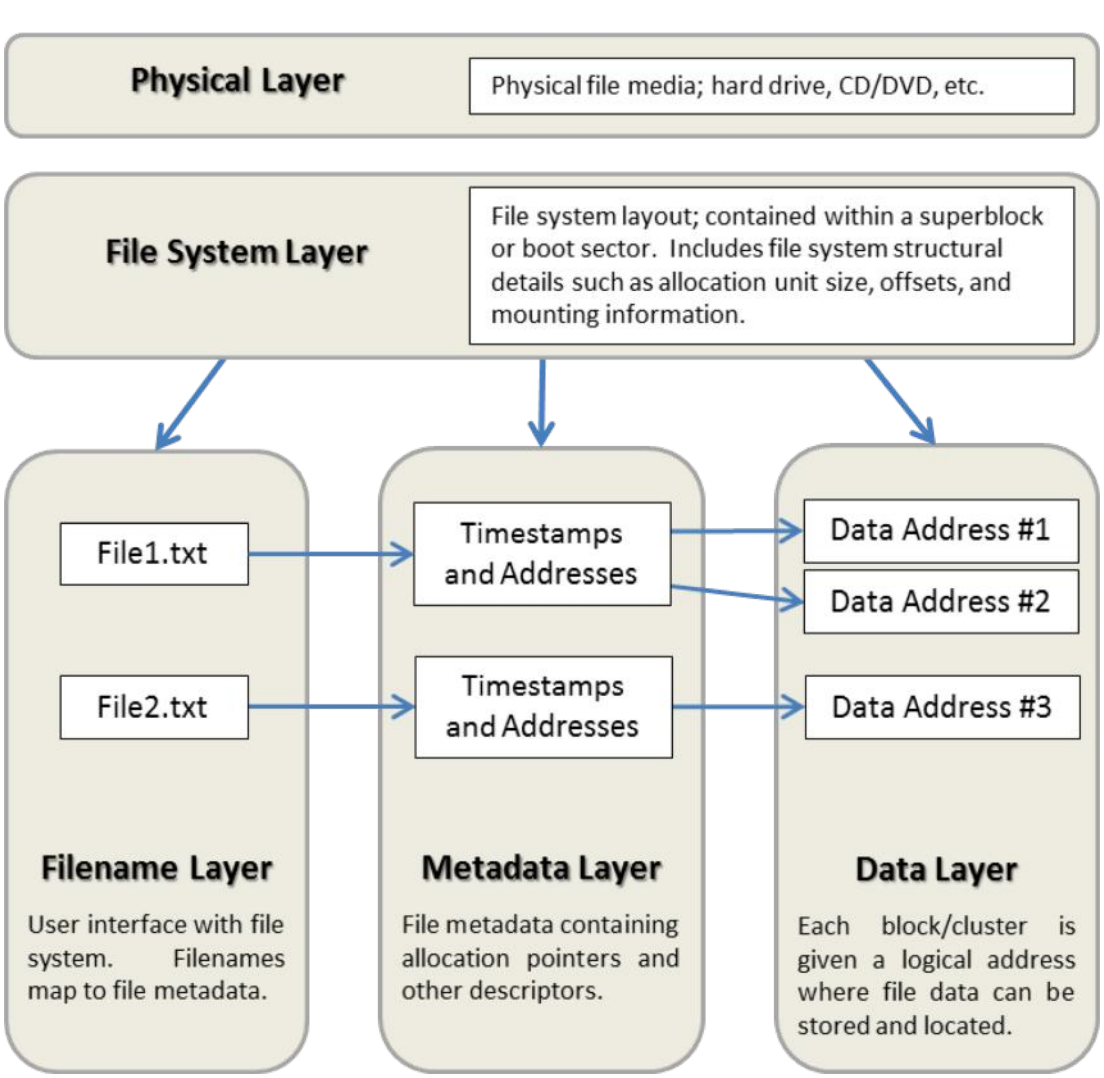
- A specific user's permissions are the sum of that user's individual permissions and group permissions.
- Example: if a user has Read permission for a folder and is a member of a group with Write permission for the same folder, the user has both Read and Write permissions for that folder.
- A Deny permission supersedes all other permissions set for that user across all groups.
- Example: if the user became a member of a second group with a Deny permission set for that folder, the user would be unable to access the folder.

NTFS Security



- Each file references a security descriptor
  - Owner of the file
  - Access Control List (ACL)
- Supports encryption via Windows BitLocker feature
- Optionally checks permissions on directory traversal
  - Not activated by default

Filename Layer



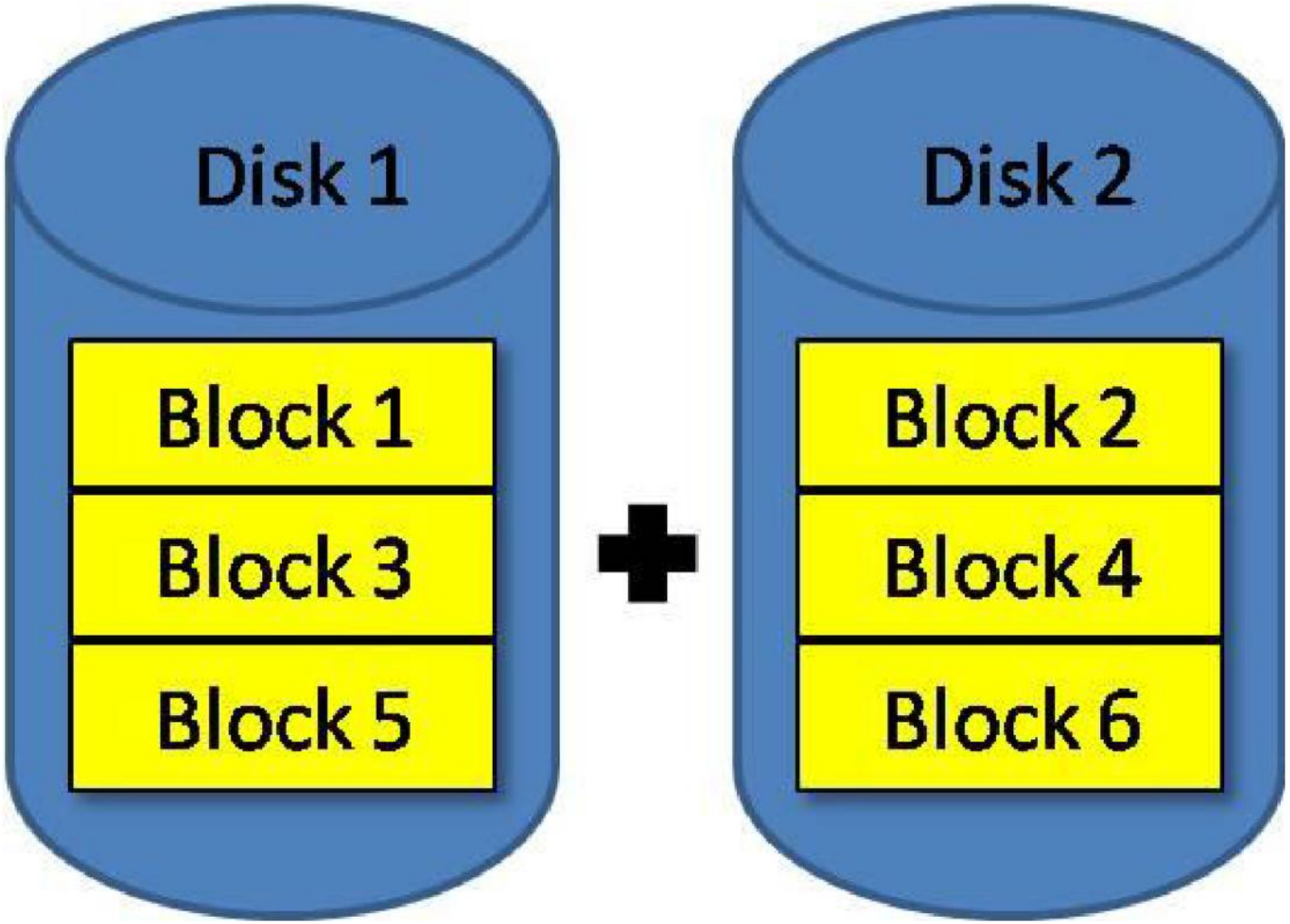
Windows BitLocker

- A full disk encryption feature included with select editions of Windows Vista and later (Ultimate & Enterprise) in January 2007
- Designed to protect data by providing encryption for entire volumes
- Uses the AES encryption algorithm in cipher block chaining (CBC) with a 128-bit or 256-bit key
- Uses Trusted Platform Module (TPM) or USB drive
- For encrypting removable media, use BitLocker To Go. If media is lost or stolen, the device is unreadable without the recovery password.

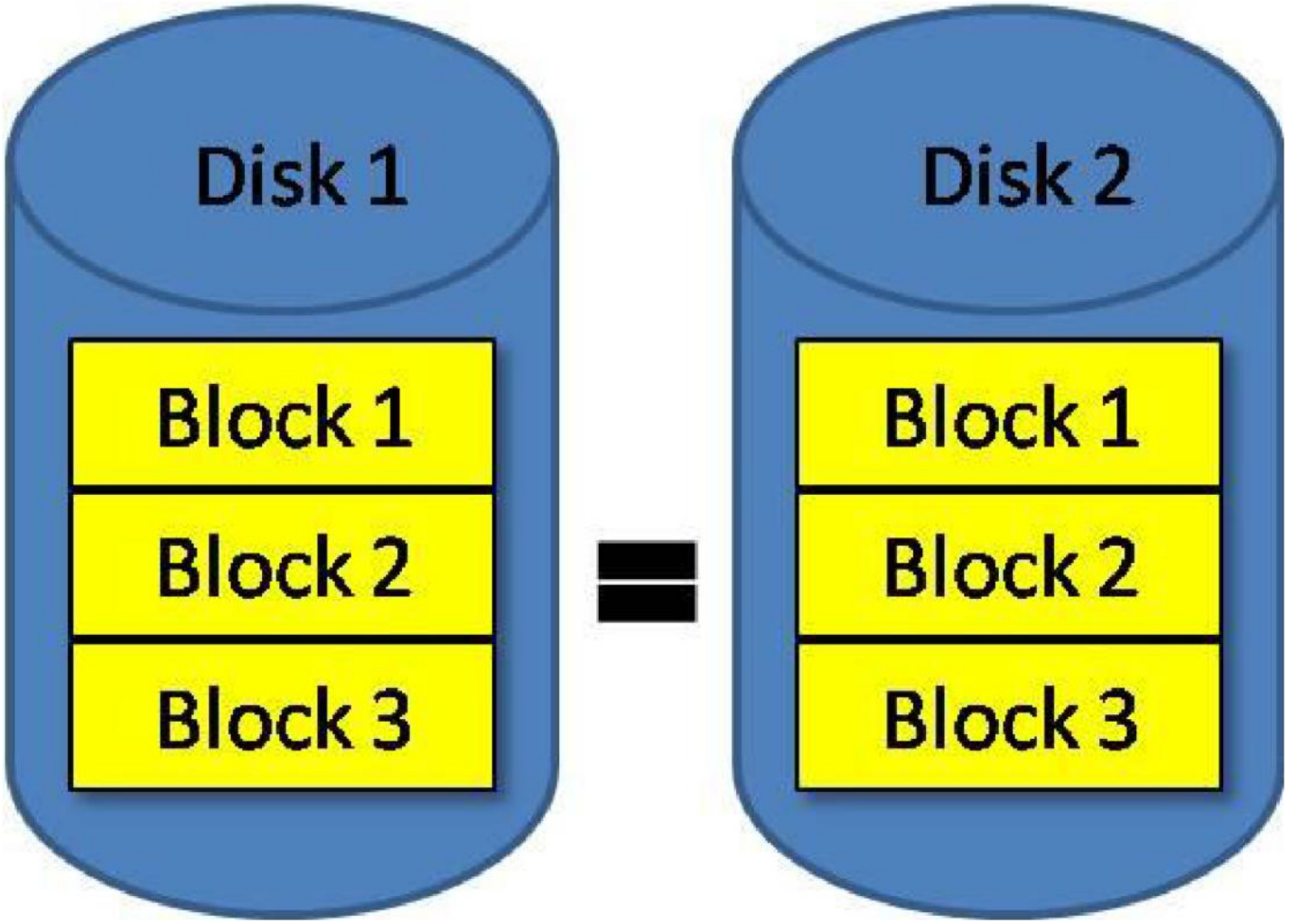
Volume Sets & RAID

- *FtDisk* is a fault-tolerant disk driver for Windows
  - Provides several ways to combine multiple drives into one logical volume (called a **volume set**)
- Volumes can also be resized dynamically
  - Including adding multiple disks to an existing volume
- Natively supports RAID
  - Level 0: Striping
  - Level 1: Mirroring
  - Level 5: Stripe set w/distributed parity blocks

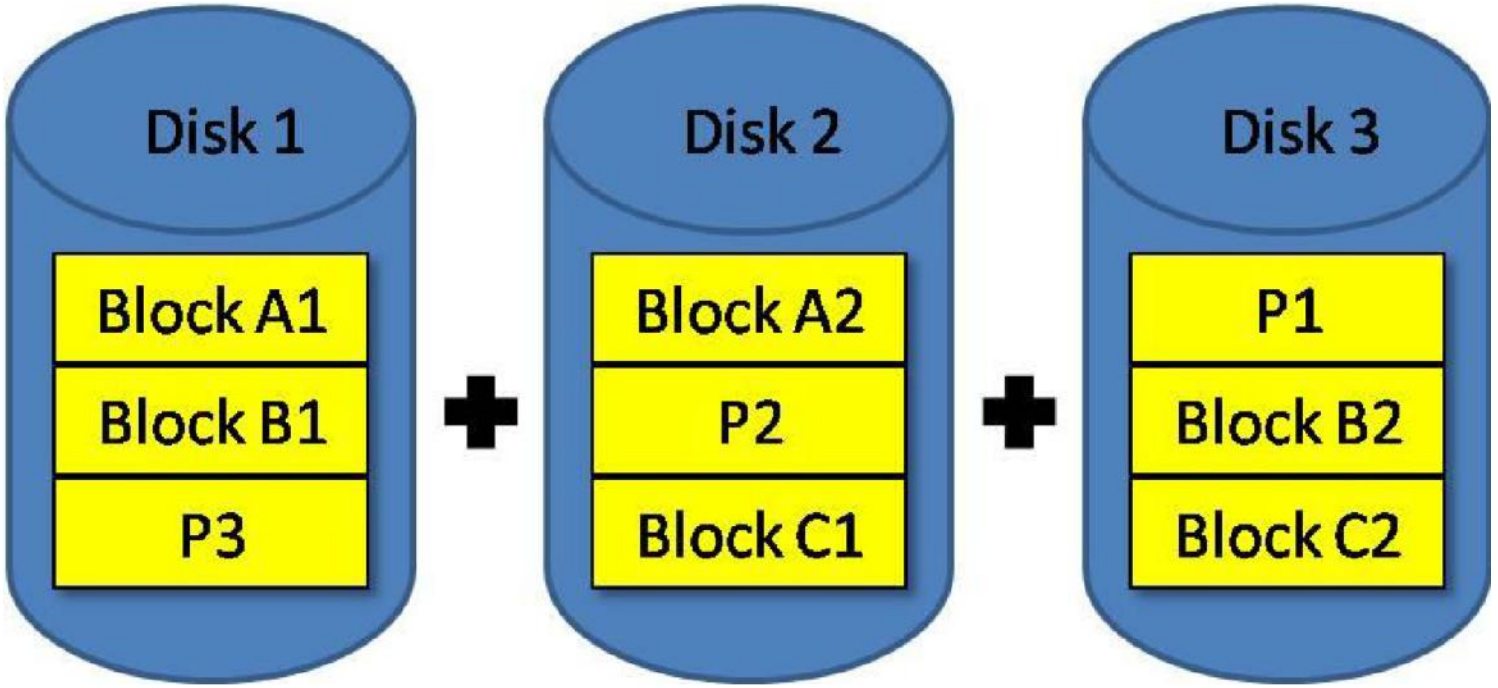
RAID 0: Striping



RAID 1: Mirroring



RAID 5: Fault Tolerance



Summary

- To describe how data is stored and accessed on a physical and logical storage device 
- To describe how to view NTFS (New Technology File System) information for a volume in windows