4.2 Storage Technologies

Suraj Tiwari

Operating System

Masters in Computer Engineering

Solid State Drive (SSD)

- Storage device that uses **flash memory** to store data.
- Widely used in laptops, desktops, and servers due to their **high speed, reliability, and efficiency**.
- Store data in **NAND flash memory chips**, which are **non-volatile**.

Solid State Drive (SSD)

Working Mechanism:

- **Flash Memory Storage-** Data is stored in microchips instead of a spinning disk.
- **Controller**-A processor is the SSD that manages data storage, retrieval, and error correction.
- **Data Access-** SSDs use electrical circuits to access data instantly instead of moving parts.

Why is an SSD faster than an HDD?

- **No moving parts**- HDDs use spinning platters and a mechanical arm, which take time to read/write data. SSDs use electronic circuits, making data access almost instantaneous.
- **Parallel Data Access**- SSDs use multiple NAND chips and can read/write data in parallel, increasing speed.
- **Low Latency**- SSDs have much lower access times (0.1ms) compared to HDDs (10ms).
- **Higher Bandwidth**- SSDs (especially NVMe SSDs) use **PCIe interfaces**, providing much more faster data transfer rates compared to SATA-based HDDs.

Speed Comparison (Average Speeds)

Storage Type	Read Speed (Mb/s)	Write Speed (Mb/s)
HDD (5400 RPM)	80-120	50-100
HDD (7200 RPM)	120-160	80-140
SATA SSD	500-550	450-500
NVMe SSD	2000-7000	2000-7000

Non-Volatile Memory Express (NVMe)

- High-speed **storage protocol** designed for **SSDs** that use the **PCIe** (Peripheral Component Interconnect Express) interface.
- Allows SSDs to communicate directly with the **CPU and RAM**, reducing latency and improving data transfer speeds.

Non-Volatile Memory Express (NVMe)

Working Mechanism:

- **Direct Communication with CPU-** NVMe connects to the PCIe bus, bypassing the limitations of older SATA controllers.
- **Parallel Processing**- Supports **multiple queues (up to 64K)**, each capable of handling 64K commands, unlike SATA, which has only **one queue with 32 commands**.
- **Low Latency-** Reduces the time taken to process data requests.
- **Higher Bandwidth**-Uses PCIe Gen 3, Gen 4, or even Gen 5, achieving speeds up to **14000 Mb/s** in the latest models.

How is NVMe Faster than SATA SSDs?

Feature	SATA SSD	NVMe SSD
Interface	80-120	50-100
Commands per Queue	120-160	80-140
Maximum Speed	500-550	450-500
Latency	2000-7000	2000-7000
Parallel Processing	Limited	Highly Optimized

Persistent Memory

- **Non-volatile** memory that combines the speed of RAM with the data retention capabilities of storage devices like SSDs and HDDs.
- Allows data to persist even after power loss, unlike traditional RAM, which loses data when powered off.
- PMEM is often used in **databases, AI/ML applications, and high-performance computing**, where fast access and data persistence are critical.

Persistent Memory

Working Mechanism: Operates in a **memory module form factor (DIMM)** and connects via the DRAM slots on a motherboard. It works in two primary modes:

1. Memory Mode

- Functions like traditional RAM (DRAM) but uses a small amount of actual DRAM as a cache.
- Data is **not persistent** in this mode (works like volatile memory).

2. App Direct Mode

- Applications can access PMEM directly, allowing **persistent storage at near-RAM speeds**.
- Data is **retained even after power loss**.

PMEM uses **byte-addressable access**, meaning it can modify small chunks of data efficiently, unlike SSDs and HDDs that rely on block-based access.

Why is Persistent Memory Faster?

- **Near-RAM Speeds-** PMEM is **much faster than SSDs** and HDDs because it is **connected directly to the memory bus**.
- **Low Latency-** Access times are measured in nanoseconds, compared to SSDs (microseconds) and HDDs (milliseconds).
- **No Need for Data Transfers** - Unlike SSDs, which require data to be moved to RAM before processing, PMEM allows direct access, reducing **I/O bottlenecks**.
- **Byte-Addressable Reads/Writes** – Unlike SSDs, that write in blocks, PMEM enables **fine-grained modifications**, improving performance.

Speed Comparison

Storage Type	Read Latency	Write Latency	Speed (mb/s)
HDD	5-10 ms	5-10 ms	100 mb/s
SATA SSD	50-100 µs	50-100 µs	500 mb/s
NVMe SSD	10-50 µs	10-50 µs	2000-7000
Persistent Memory (PMEM)	100 ns	100 ns	10000+ mb/s
DRAM (RAM)	10 ns	10 ns	50000+ mb/s

4.3 Modern File Systems: ZFS, Btrfs, Ext4, APES

- File systems manage how data is stored and retrieved on storage devices.
- A file system is like your computer's way of organizing and storing all its data. Think of it as a huge digital filing cabinet where:

  Folders and Files: Just as you might sort papers into folders, a file system organizes files into directories (folders) so you can find them easily.

  Tracking data: It keeps track of where each file is stored on your device- whether it's on an SSD, HDD, or another storage medium.

  Managing Space: It helps manage the available space and ensures that your data is stored safely and efficiently.
- In summary, a file system provides the structure that lets your computer know where to save, retrieve, and organize all your information.

1. ZFS (Zettabyte File System)

- **Developed by**: Sun Microsystems (now maintained by OpenZFS)
- **Best for**: High-end servers, NAS (Network Attached Storage) and enterprise storage.
- Imagine ZFS as a super-smart filing cabinet that not only stores your files but also checks them for errors, automatically fixes problems, and even keeps backup copies (snapshots). This means if something goes wrong, you can easily go back to an earlier version of your data.

1. ZFS (Zettabyte File System)

□ **Features:**

✔ **Copy-on-Write (COW)** – Prevents data corruption by writing changes to a new block before updating metadata.

✔ **Data Integrity & Self-Healing** – Uses checksums to detect and repair data corruption automatically.

✔ **RAID-Z Support** – Built-in RAID-like **redundancy** without requiring hardware RAID controllers.

✔ **Snapshots & Clones** – Instant backups and duplication of files without using extra space.

✔ **Scalability** – Supports zettabytes (1 billion terabytes) of data.

□ **Use Cases:**

- Large-scale **enterprise storage** (cloud storage, database)
- **NAS systems** (TrueNAS, FreeNAS)
- **Backup & archival storage**

2. Btrfs (B-Tree File System)

- **Developed by**: Oracle (Linux community-supported)
- **Best for**: Linux-based storage, data redundancy and snapshots.
- Think of Btrfs as a flexible, upgradeable digital filing system for Linux. It allows you to quickly save versions of your files (snapshots) and adjust storage settings on the fly, so you can recover old versions if something goes wrong.

2. Btrfs (B-Tree File System)

- **Features:**

✔ **Copy-on-Write (COW)** – Prevents data corruption and improves reliability.

✔ **Snapshots & Subvolumes** – Efficient backups and rollback features.

✔ **Data Deduplication & Compression** – Reduces disk usages and improves performance.

✔ **RAID Support** – Built-in support for RAID 0,1 and 10.

- **Use Cases:**

✔ Linux-based systems (Fedora, SUSE, openSUSE)

✔ **Docker & Kubernetes storage**

✔ **SSD optimization** (Btrfs handles SSD wear leveling better than Ext4)

3. Ext4 (Fourth Extended File System)

- **Developed by**: Linux community
- **Best for**: General-purpose Linux desktop & server storage.
- Ext4 is like the classic, well-organized filing system that many Linux computers have trusted for years. It’s fast, reliable, and efficient. While it might not have all the extra safety features like snapshots built in, it’s still modern compared to much older systems because it supports journaling and can handle large files and volumes well.

3. Ext4 (Fourth Extended File System)

- **Features:**

✔ **Journaling** –Prevents data corruption after crashes.

✔ **Supports Large Files** – Up to 16 TB file size, 1 EB (exabyte) filesystem size.

✔ **Backward Compatibility** – Can mount Ext3 and Ext2 file systems.

✔ **Less Overhead** – More stable than Btrfs but lacks advanced features.

Use Cases:

- Linux desktops & laptops (Ubuntu, Debian, Arch Linux)
- **Web & database servers**
- **Embedded systems** (Raspberry Pi, IoT devices)

4. APFS (Apple File System)

- **Developed by**: Apple
- **Best for**: macOS, iOS, iPadOS and modern Apple devices.
- APFS is Apple’s own modern filing system designed for their devices. It’s optimized for speed and security on flash storage (like SSDs) and includes features such as strong encryption and snapshots to help protect and quickly recover your data.

4. APFS (Apple File System)

- **Features:**

✔ **Optimized for SSDs** –Faster data access and efficient flash memory handling.

✔ **Snapshots & Cloning** – Allows instant backups and efficient data duplication.

✔ **Encryption & Security** – Built-in **full disk encryption** for security.

✔ **Space Sharing**– Multiple volumes can dynamically share available space.

Use Cases:

- MacBooks, iPhones,, iPads
- macOS Time Machine backups
- High-speed SSD-based storage

Comparison Table

File System	Best For	Snapshots	RAID Support	Encryption	Performance (SSD)	Max File Size
ZFS	Enterprise, NAS	Yes	Yes (RAID-Z)	Yes	High	16 Exabytes
Btrfs	Linux storage, servers	Yes	Yes (RAID 0,1,10)	Experimental	High	16 Exabytes
Ext4	Linux desktops, servers	No	No	No	Good	16 TB
APFS	Apple macOS, iOS	Yes	No	Yes	High	8 Exabytes

Why They’re Considered Modern:

Modern file systems include advanced features that older systems lack. They not only store files but also:

- Automatically check and repair errors.
- Create snapshots (like saving a "photo" of your data at a specific time) for easy recovery.
- Manage large volumes of data efficiently.
- Enhance security through encryption.
- Optimize performance, especially on faster storage devices like SSDs.
- These improvements mean your data is safer, more organized, and quicker to access, which is why these file systems are seen as modern solutions for today's computing needs.