This chapter describes the internal data structures and algorithms used by the operating system to implement file systems. It also discusses the lowest level of the file system, the secondary storage structure.

**Disk**

- The surface of a platter is logically divided into circular **tracks**, which is subdivided into **sectors** (hundreds of them). The set of tracks (thousands of) that at one arm position (multiple plates) makes up a **cylinder**
- **Transfer rate** is rate at which data flow between drive and computer, typically several megabytes of data per second
- **Positioning time** (random-access time) is time to move disk arm to the desired cylinder (**seek time**) and time for desired sector to rotate under the disk head (**rotational latency**), usually several milliseconds.

**Disk Structure and Attachment**

- Disk is organized into *logical blocks*, which is the *smallest data transfer unit*. The typical size of a block is 512 bytes. A logical block can be represented by a vector [cylinder, surface, sector], or by a *logical block address* (LBA), each sector is assigned a number (from 0 to the max. number of the sector)
- **Host-attached storage** (I/O port): This can be through IDE (typical desktop computer), or SCSI and fiber channel (FC). **Network-attached storage** (NAS) is accessed remotely over a network using conventional networking protocols (e.g., IP protocol or the Internet protocol), which usually has a large latency and its performance is subject to the network traffic condition
- **Storage-area network** (SAN): a private network using storage protocols. This is flexible in configuration. FC is the most common SAN interconnect.

**Disk Scheduling**

- **FCFS**: this is intrinsically fair, but the performance can be bad depending on the request arrival orders
- **SSTF**: the shortest-seek-time-first. Better performance in term of seek time than FCFS, but can result in starvation
- **SCAN** and **LOOK** service requests in both direction, while **C-SCAN** and **C-LOOK** service requests in one direction.
- There could be a variety of factors affecting the selection of disk scheduling algorithms such as directory and file allocation. Under a given list of requests, an optimal scheduling can be derived but could be computationally expensive; also this can be useless since requests are coming dynamically. Typically, SSFT and LOOK are default choices.

**RAID**

- This provides high reliability via *redundancy* and performance improvement in term of high data-transfer rate through *parallelism*.
- Bit-level or block-level stripping for performance improvement (parallelism)