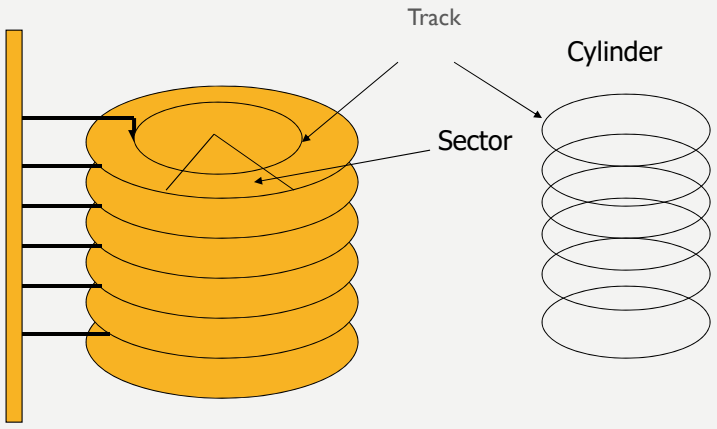
CPSC332 PART 2 FILE STRUCTURES

SHAWN WANG, PH.D., PROFESSOR
TELEPHONE: (657)278-7258
EMAIL: XWANG@FULLERTON.EDU

SECONDARY STORAGE



SECONDARY STORAGE

- Seek time: read/write head on track.
- Rotational delay (latency): first sector (block) under head.
- Block transfer time: read/write one block.

AN EXAMPLE (SEAGATE CHEETAH)

- Capacity: 9 GB.
- Average seek time: 8 msec
- Spindle speed: 10,000 rpm.
- Average rotational delay?
  Half of one track => 3 msec.
- Transfer time per track?
  Rotation time per track=>6 msec
- Size of sectors: 512 bytes.
- Number of sectors per track: 170.
- Number of tracks per cylinder: 16.
- Number of cylinders: 6,526.

SOME CALCULATIONS

Assume we have 34, 000 records, each has 256 bytes.

Contiguous storage (Best scenario):

- Total number of tracks?
  2 records/per sector => 17,000 sectors => 100 tracks.
- Total time for reading one track?
  seek time + rotational delay + transfer time.
- seek time: 8 msec
- rotational delay: 3 msec
- transfer time per track: 6 msec
- Total time for reading one track = 17 msec
- Total time for reading all records = 17 x 100 = 1.7 sec.

SOME CALCULATIONS

Random storage (Worst Scenario):

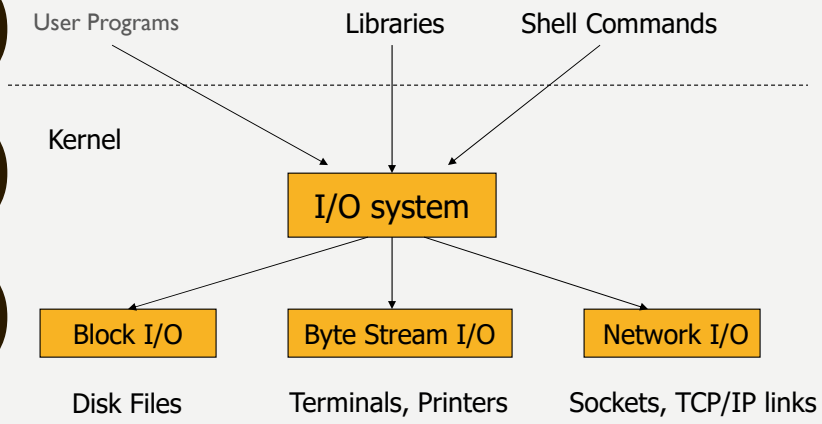
- Minimum reading unit => cluster = 8 sectors.
- Transfer time per cluster = 8 x 6/170 => 0.28 msec.
- Total time for reading one record = 8 + 3 + 0.28 = 11.28 msec
- Total time for reading all records = 34,000x11.28=383520 msec = 6 min. 23.52 sec.

- The difference: 383520/1700 = 225.6

WRITING A BYTE

1. Program => operating system (system call)
2. OS => file manager
3. File manager checks file pointer
4. File manager checks FAT for physical address (virtual memory)
5. Block buffered? Yes. Write to buffer. No. Read the block from disk to buffer then write to buffer.
6. File manager => I/O processor
7. I/O processor decides whether it should wait or write immediately
8. I/O processor => disk controller
9. Disk controller sends data to disk

I/O IN UNIX



STREAM FILE (AN EXAMPLE)

In C:

```
struct Person
   {
   char LastName [16];
   char FirstName [16];
   char Address [30];
   char City [16];
   char State [3];
   char ZipCode [10];
   };
```

In C++:

```
class Person
  {
  private:
    char LastName [16];
    char FirstName [16];
    char Address [30];
    char City [16];
    char State [3];
    char ZipCode [10];
  public:
   // methods  ... ...
  };
```

INPUT: OVERLOADING >>

```
istream & operator >> (istream & is, Person & p)
  {
  char Delim='|';
  is.getline (p.LastName, 16, Delim);
  if (strlen(p.LastName)==0) return is;
  is.getline(p.FirstName, 16, Delim);
  is.getline(p.Address, 30, Delim);
  is.getline(p.City, 16, Delim);
  is.getline(p.State, 3, Delim);
  is.getline(p.ZipCode,10, Delim);
  return is;
  };
```

OUTPUT: OVERLOADING <<

```
ostream & operator << (ostream & os, Person & p)
{
os  << p.LastName << endl
    << p.FirstName << endl
    << p.Address << endl
    << p.City << endl
    << p.State << endl
   << p.ZipCode << endl;
return os;
};
```

FIELD STRUCTURES

- Fixed length

```
Ames   Mary  123 Maple     Stillwater   OK74075
Mason  Alan   90 Eastgate  Ada          OK74820
```

- Length indicators

```
04Ames04Mary09123 Maple10Stillwater02OK0574075
05Mason04Alan1190 Eastgate03Ada02OK0574820
```

- Delimiters

```
Ames|Mary|123 Maple|Stillwater|OK|74075|
Mason|Alan|90 Eastgate|Ada|OK|74820|
```

- Keyword = value

```
lname=Ames|fname=Mary|address=123 Maple|
city=Stillwater|state=OK|zip=74075|
```

RECORD STRUCTURES

- Fixed length:

  1) with fixed length fields

```
Ames   Mary  123 Maple    Stillwater   OK74075
Mason  Alan   90 Eastgate  Ada          OK74820
```

2) with variable-length fields separated by delimiters

```
Ames|Mary|123 Maple|Stillwater|OK|74075|
Mason|Alan|90 Eastgate|Ada|OK|74820|
```

What are the differences? What other combinations are possible?

RECORD STRUCTURES

- Fixed number of fields

```
Ames|Mary|123 Maple|Stillwater|OK|74075|Mason|
Alan|90 Eastgate|Ada|OK|74820|
```

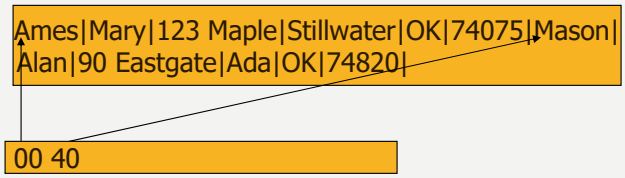
- Using length indicators

```
40Ames|Mary|123 Maple|Stillwater|OK|74075|36Ma
son|Alan|90 Eastgate|Ada|OK|74820|
```

What are the advantages? What other combinations are possible?

RECORD STRUCTURES

- Index file



- Using delimiters

Ames|Mary|123 Maple|Stillwater|OK|74075|#Mason|Alan|90 Eastgate|Ada|OK|74820|