3/25

controller

CPU

Fibre channel switch

DISK

network

drive

direct-attached

SATA

SCSI (SAS)

2 protocols of communication: Fibre Channel or iSCSI

Storage view: you recommend possibilities and see what respond.

File system: Many files access the same disk

Block devices are things you put on top of

What is a file? set of locations (blocks) with related data
ordered set of bytes
collection of data and metadata

A file in windows is different to a file in UNIX, VMS and ...
But the common properties for a file are: persistent content
read, written
structure, sequence of bytes
other
UNIX was the first to declare a file as a sequence of bytes. A file has attributes and is reachable by name.

A file has a namespace — strings → files → directories

For the application, we need to have an API. We have to have an internal implementation in Linux. VFS is an interface.

Namespace:

Any file system has a hierarchical namespace, where you either have a root or a set of devices.

Root: /usr/local...

Devices: C: \temp...

In UNIX hierarchy, we have:

There is some element that do not have the structure of files, but they still have basic "read" and "write" on them.

Another extension to this namespace is links. You can have 2 elements connecting to an element or a symbolic link, pointing to another place.

Namespace implementation:

Namespaces are typically organized into directories. A directory in general will map strings to objects (files).
In UNIX a file has no unique property which is determined by the path and hash tables are most often used in file systems.

**File operations**

- open(path) ➔ handle
- close(handle)
- read(handle,...) ➔ data
- write(handle,data)
- seek(handle,offset)
- write(handle,offset,...)

- rename
- delete
- create
- mkdir
- main

This is based on the fact that files have no internal structure. The structures are application dependents.

**Layout of file systems:**

- Typically file systems have the following structures:
  - Superblock: located at a known position on the disk, global parameters
  - Directory: map string ➔ object (length, attributes,)

Notice that every file is built on top of a block.

- File (inode): per file object on UNIX, contain properties ➔ block length
  - Directory: ➔ linear search ➔ ext
  - File ➔ tree
  - File ➔ binary search
  - File ➔

- NIFS
- XPS
- etc
- etc
- NIFS
file contents:  
linked list: direct

array of pointers

msdos

problem here is that
you have a limited number

indirect blocks

direct

indirect

direct