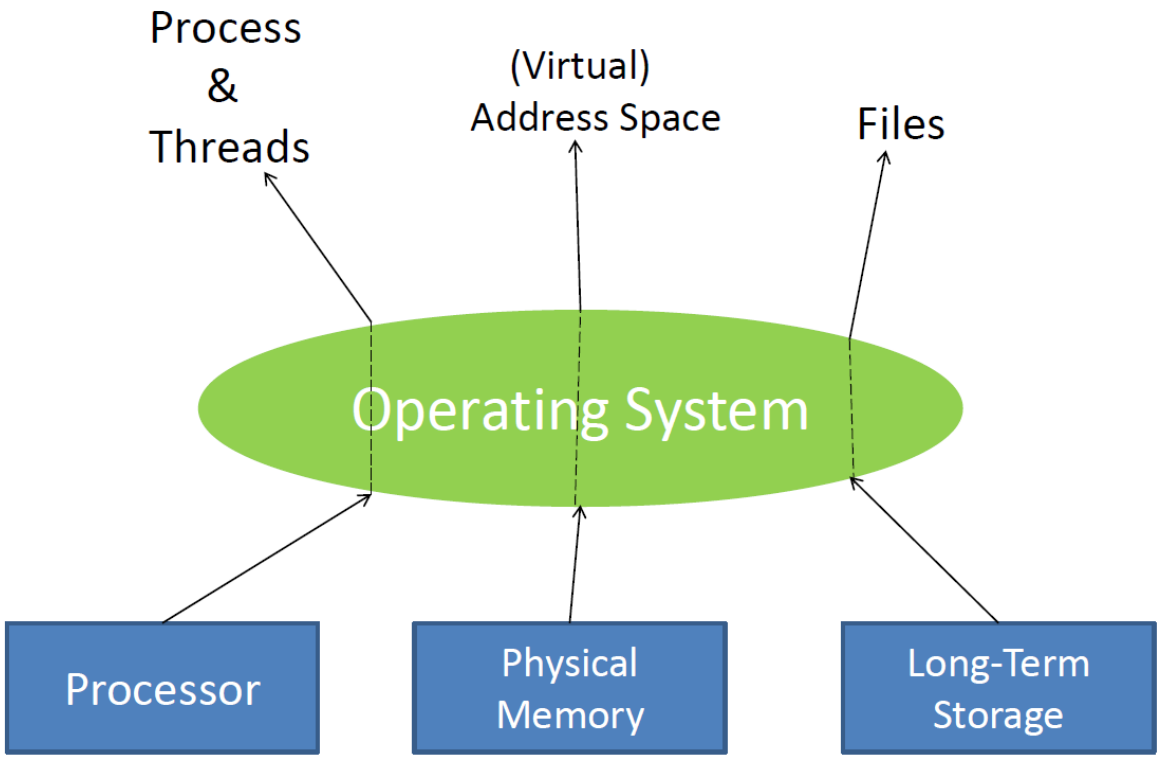
Operating Systems W12L1 - File Systems I

Class	Operating Systems
Created	@Nov 16, 2020 12:36 PM
Materials	09 - File Systems I.pdf
Reviewed	☐
Type	Lecture

Initial Discussion

- This is all about how the OS stores things efficiently on the disk
- How is it that we can use the same files from different machiens (and thus different operating systems)?
- Machines may have the same processor, but different operating systems and thus different binaries
    - Libraries exist for these reasons → OS executables all different and thus portable files

▼ A Way to Abstract Long-Term Storage



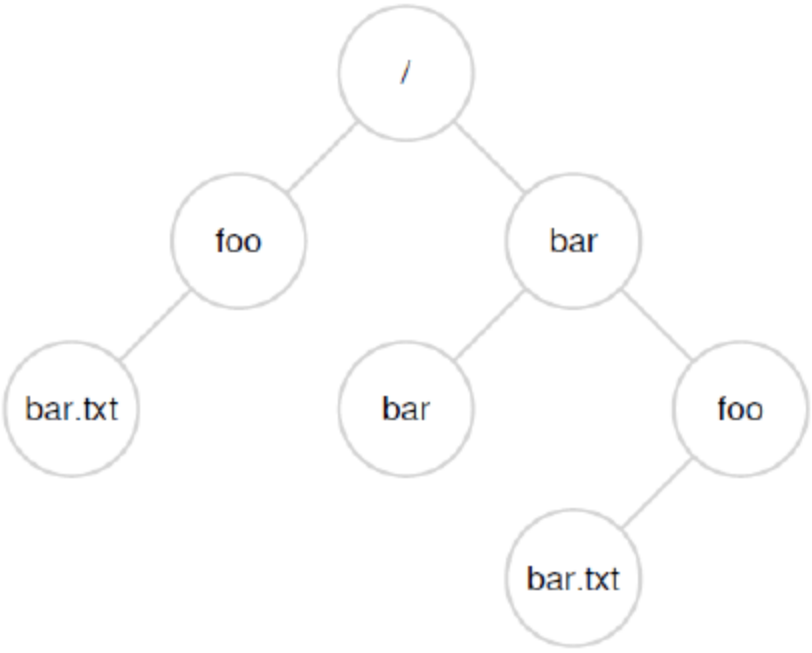
- Today's class is all from user POV

Long Term Storage

- Cannot keep everythign in RAM
- Requirements are...
  - Store large quantities of info
  - Persistance of information
  - Multiple processes able to process / open the file

▼ Two Key Abstractions

1. Files — Linear array of bytes
2. Directories



- Directories and files are just a bunch of strings to users and processes

▼ Three issues w/ long-term storage

- How to find info?
- How to protect files from other users?
- How to know which **blocks** are free?

> For now, and till we discuss disks in I/O lectures, let's assume a disk consists of a **linear sequence of fixed-size blocks** supporting two operations: read block x and write block x.

Files

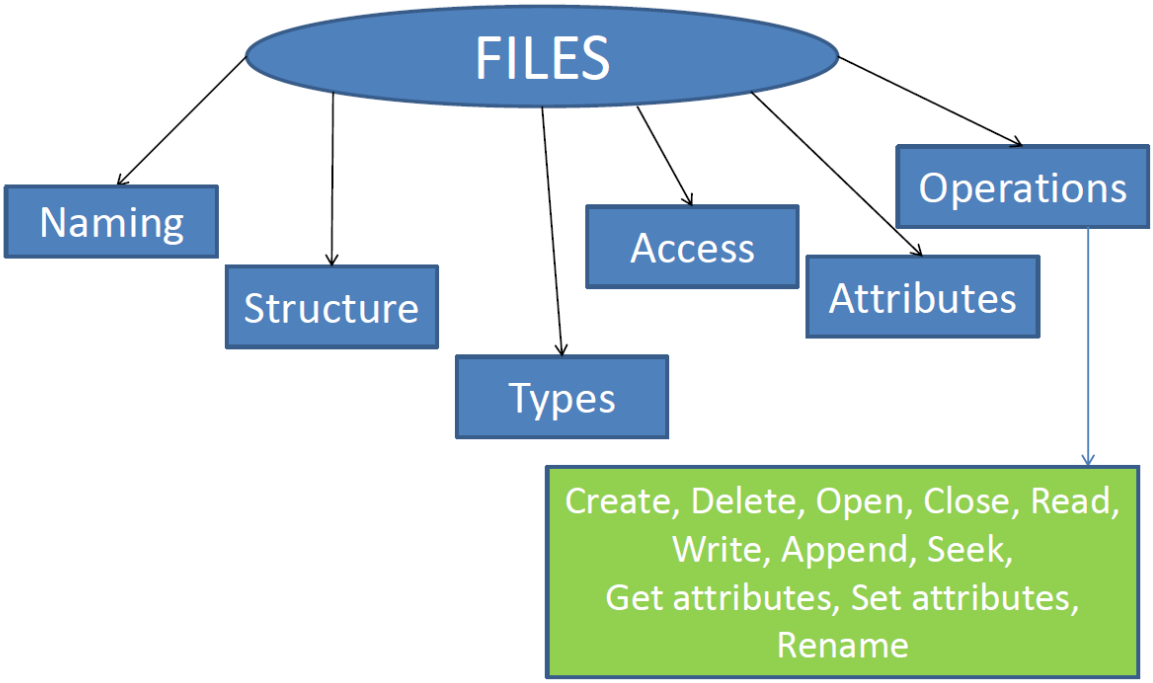
- A file is a data collection created by users or logical units of information created by processes
- Desirable properties include..

    - Long-term existence — Files are stored on disk or other secondary storage and do not disappear when a user logs off
    - Shareable between processes — Files have names and can have associated access permissions that permit controlled sharing
    - Structure — Files can be organized into hierarchical or more complex structures to reflect the relationships among files
- Files model disks, not RAM, and are managed by OS — i.e. moved to/retrieved from disk, security of files, etc.
    - All done via OS **File System**

File Systems

- Provide means to store data and organize things as a colleciton of functions that can be performed on files
- Maintains attributes associated with file
- Files aren't opened outside of processors → User never directly interacts with OS (again, GUIs)

▼ Files Web Diagram

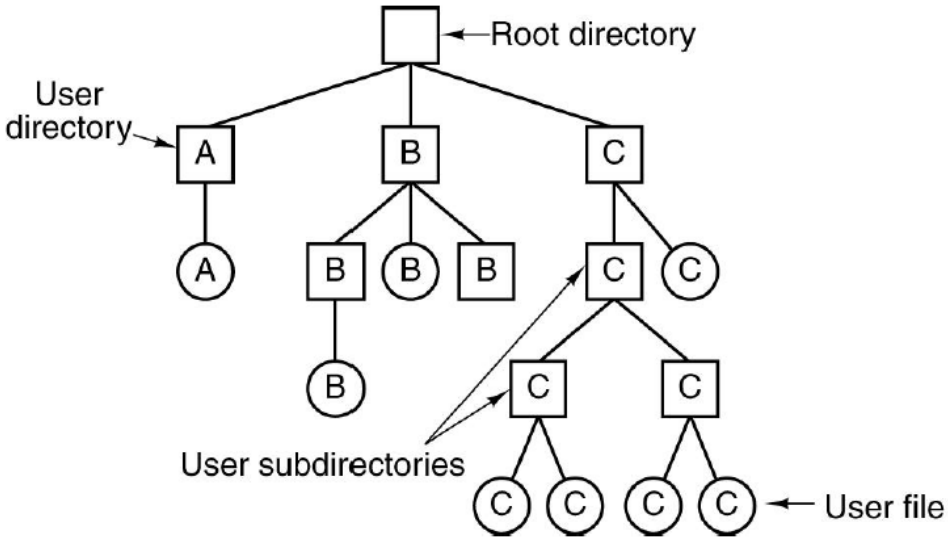


Directories

- Early on, single level directory systems simple, but obviously no longer adequate

▼ Hierachical Directory Systems

- Group related files together
- Tree of directories



- Need path names, either *absolute* or *relative* (to working directory)