Unix Filesystems

Overview Unix filesystem is a collection of files and directories that has the following properties:

- It has a root directory (`/`) that contains other files and directories.
- Each file or directory is uniquely identified by its name, the directory in which it resides, and a unique identifier, typically called an **inode**.
- By convention, the root directory has an inode number of 2. Inode numbers 0 and 1 are not used (file inode numbers can be seen using "`ls -i`").
- It is self-contained. It does not depend on other filesystems.

Disks, slices, partitions, and volumes Each hard disk is typically split into a number of separate, different sized units called **partitions** or **slices**. Each disk contains some for of partition table, called a **VTOC (Volume Table of Contents)**, which describes where the slices start and what their size is. Each slide may then be used to store bootstrap information, a filesystem, swap space, or left as a *raw parition* for database access or other use

Mounting and unmounting filesystems The root filesystem is mounted by the kernel during system startup. Each filesystem can be mounted on any directory in the root filesystem except `/`. A **mount point** is simple a directory. `mount` program displays all mounted filesystems.

Querying filesystem statistics We can use `stat` program to display file or filesystem status. For example, "`stat -f /`" shows the statistics of root filesystem.

On-disk layout of Unix filesystem Typically a disk is viewed as an array of blocks by the Unix system.

- Block 0 (**bootblock**): unused
- Block 1 (**superblock**): holds info about the filesystem as a whole such as the number of blocks in the filesystem, the number of inodes (files), and the number of free inodes and data blocks (layout defined by `struct filesys`
- Block $2 \sim k$ (**inodes**): each inode is defined by `struct inode`
- Block $k+1 \sim$ (**data blocks**)

Inode