

Lecture 36 - File Systems Implementation

CprE 308

April 7, 2014

Intro

Today's topics: File System Implementation

- Data Structures for files
- Unix files, Inodes

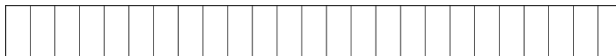
Blocks

Unix File System - Inodes

- data structure on disk
- one inode per file

Owner snt
Group cpre308
Type regular file
Perms rwxr-xr-x
Accessed oct 23 9.34pm
Modified
Inode modified ...
Size 74003 bytes
Disk addresses

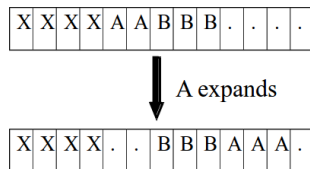
Problem



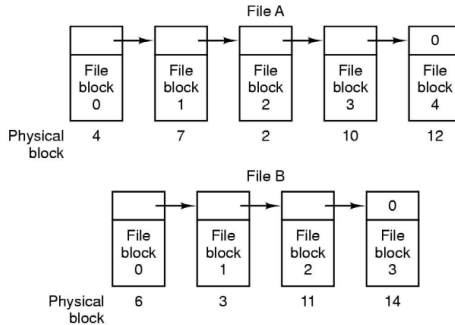
- Disk = (long) sequence of block
- Keep track of the blocks associated with a file

Contiguous Allocation

- All disk blocks of a file allocated sequentially
- Advantages
 - (very) Fast read
 - Useful for read-only file systems (CD-ROM)
 - Keeping track of blocks of a file is easy
- Problems
 - Fragmentation with deletes
 - File growth might be expensive



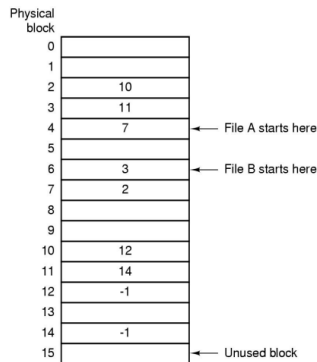
Linked List of Blocks



Sequential access is fast, random access is slow

File Allocation Tables (FAT)

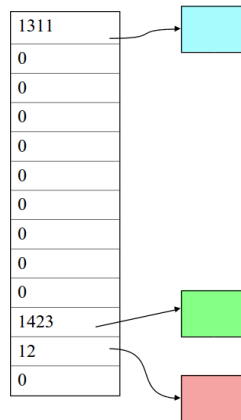
- One entry per physical disk block;
- FAT can be in main memory



The diagram illustrates a hierarchical tree structure. On the left, there is a vertical column of 13 input nodes, numbered 0 through 12. These nodes are grouped into four color-coded categories: nodes 0-6 are light purple, nodes 7-9 are light green, nodes 10-11 are light red, and node 12 is light yellow. Arrows indicate the flow from these input nodes to a series of intermediate nodes, which are represented by colored squares (light blue, light green, light red, and light yellow). The structure branches out from left to right, with each level of nodes connected to the next level by arrows, forming a complex, multi-level hierarchy. The final output is a large, multi-colored square on the right, which is the result of the entire branching process.

Optimization for Sparse Files

- Suppose a file was large, but mostly zeros
- Could be produced using lseek and write



Additional Enhancements

- Performance depends on: How many disk access are needed to read a file?
- Store some data in the inode itself
 - Perhaps the whole file will fit in!
 - Need only 1 disk access for a small file
- Increase block size

Directories

File Systems

- Data Structures for Directories
- Shared Files

Unix Directory (V7)

Directories are files whose data is a list of filenames & inodes

filename (14 bytes)

inode number (2 bytes)

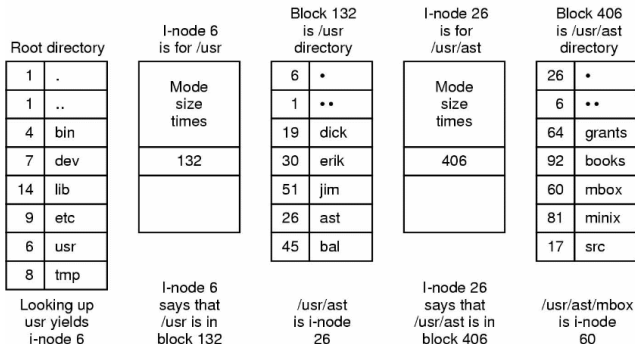
.	12
..	14
etc	134
mail	346
crash	5
init	175
mount	586

Example inode

Owner smt
Group cpre308
Type regular file
Perms rwxr-xr-x
Accessed oct 23 9:34pm
Modified
Inode modified ...
Size 74003 bytes
Disk addresses

Max filename size = 14 chars

The UNIX V7 File System



The steps in looking up `/usr/ast/mbox`