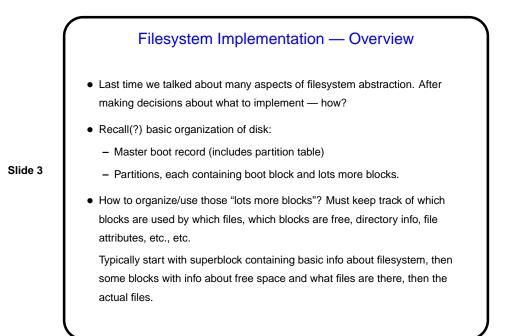
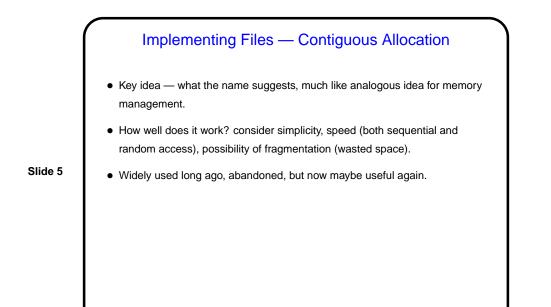


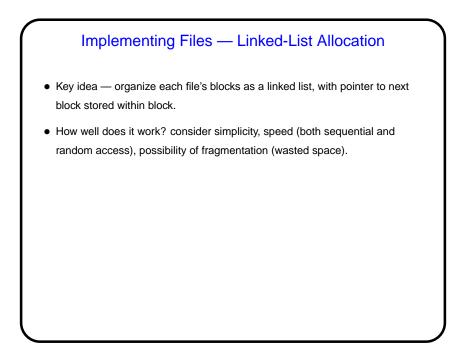
Minute Essay From Last Lecture
File-naming conventions can be different, which could lead to problems. (Example — filenames that are valid on source system but not on target, or filenames that are distinct on source but not on target.)
Structure and organization of files can be different. (Well, I say a copy program can probably fix some things. Maybe not all, though.)
Difference in metadata can be a problem too. (Windows doesn't have permissions? well, no, I think NTFS does, though the old DOS filesystems don't.)

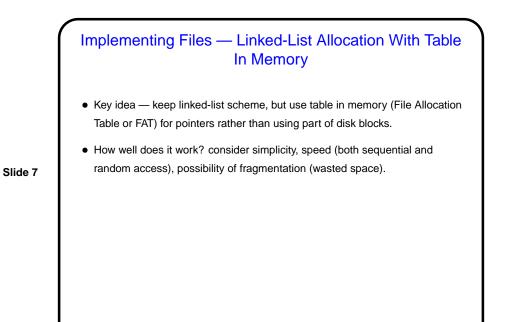


## **Implementing Files**

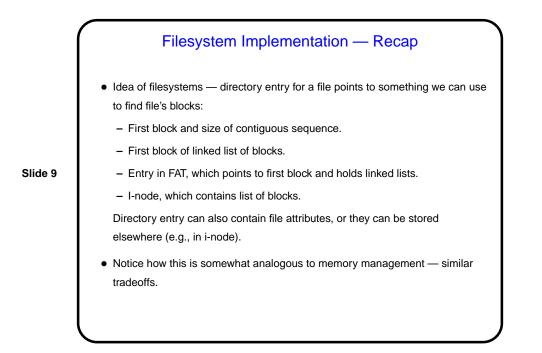
- One problem is keeping track of which disk blocks belong to which files.
- No surprise there are several approaches. (All assume some outside "directory"-type structure with some information about each file a starting block, e.g.)

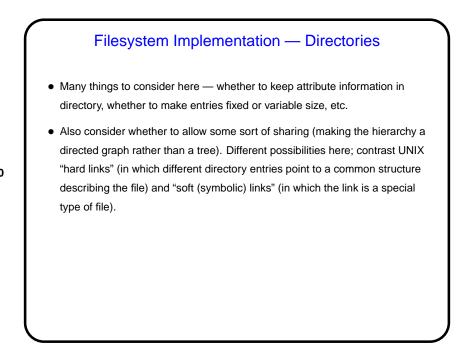


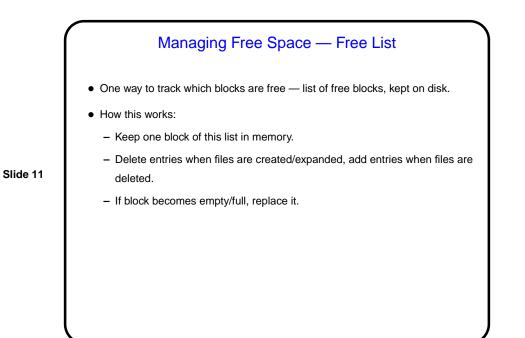




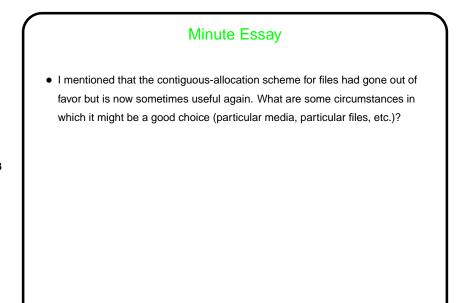
Implementing Files — I-Nodes • Key idea — associate with each file a data structure ("index node" or i-node) containing file attributes and disk block numbers, keep in memory. • How well does it work? consider simplicity, speed (both sequential and random access), possibility of fragmentation (wasted space).











## Minute Essay Answer It could be a good choice for a write-once medium, assuming every files is written all at once (rather than, say, writing some blocks of one file, then some blocks of another, and so forth). It might even be a good choice in situations in which faster access outweighs other considerations (such as difficulties in making files bigger, or the potential for fragmentation).