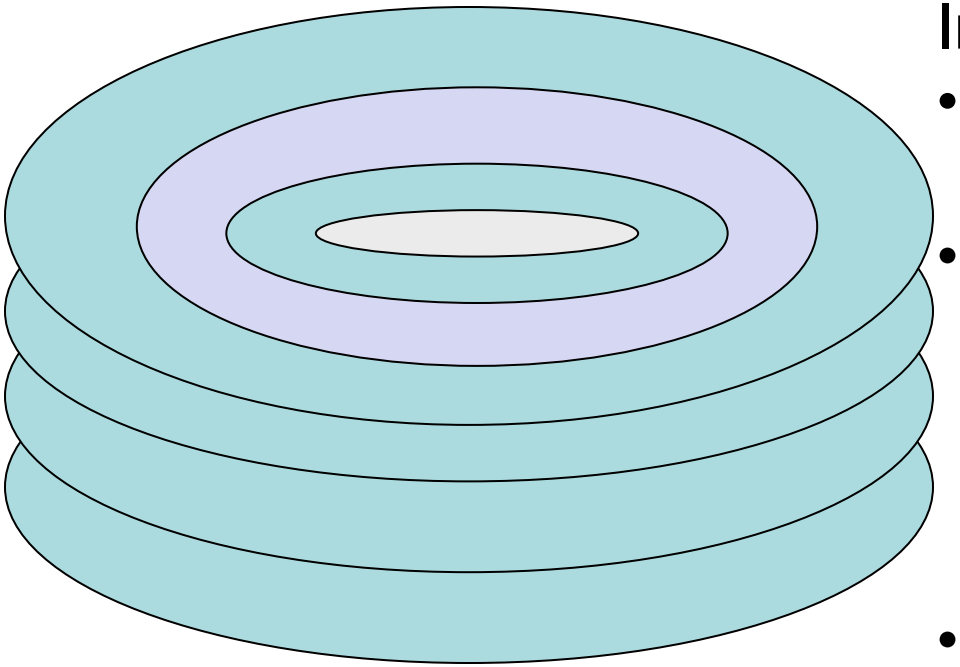
File Systems: Case Study FFS

- Learning Objectives
  - Describe the BSD 4.3 Fast File System
  - Identify strengths and weaknesses of the design
  - Begin to develop a framework in which to think about file system recoverability.
- Topics
  - The Fast File System
  - Problems it solves
  - Problems it does not solve

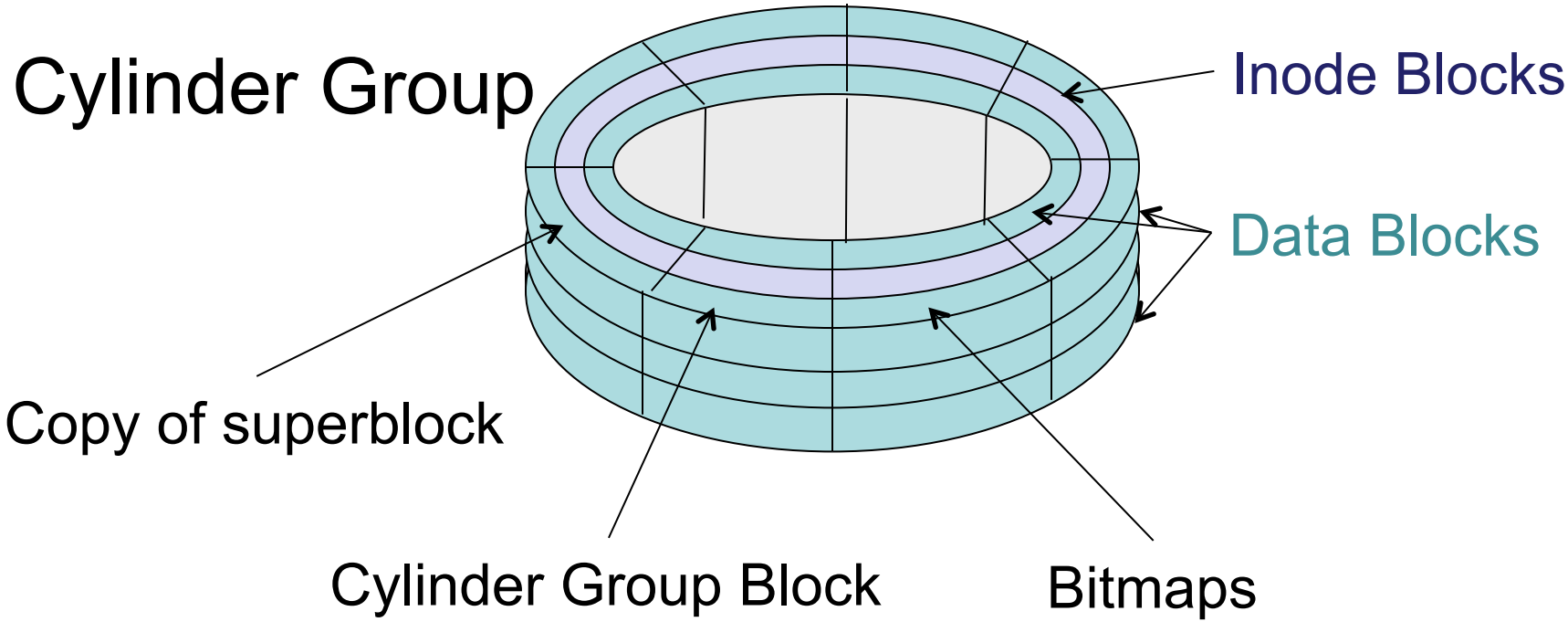
FFS Disk Organization



Introduce Cylinder Groups

- View disk as a set of cylinder groups
- A cylinder group is a collection of tens or a small number of hundreds of tracks aligned on each platter.
- Provides a management unit.

FFS Cylinder Group



FFS Details (1)

- Copies of the superblock spiral around the disk, so you get superblocks on every platter at every rotational position.
  - Cylinder group meta-data is at different cylinder offset in every cylinder.
- Inodes contain hybrid index
  - 12 direct blocks
  - 1 singly indirect block
  - 1 doubly indirect block
  - 1 triply indirect block

FFS Details (2)

- Allocation policies: placement
  - Files placed in same cylinder group as their parent directory.
  - Subdirectories allocated to different cylinder groups.
  - Large files splits across multiple groups (at indirect block boundary).
- Allocation policies: blocks management
  - Two blocks sizes: blocks and fragments
  - Block is either 4 or 8 fragments.
  - Only the last block of a file can be a fragment.
  - No fragments for “big” files (big = “has indirect block”).
  - Allocations at “rotationally optimal” positions.
- Allocation mechanisms:
  - Bitmaps (not freelist) maintained in fragment granularity
  - Must find a contiguous set of free bits to allocate a block.
  - Keep allocation tables indicating where partial blocks of different sizes are available.

FFS Details (3)

- Disk divided into 8 rotational regions
  - Allocate “next” file block in a nearby block at the appropriate rotational position.
  - With 10% reserve, can almost always allocate a good block.
- Addressed functional limitations:
  - File names of “nearly arbitrary length” (somewhat less than 512 bytes)
  - Symbolic links
  - Added advisory locking
  - Atomic rename instead of three-call sequence (can rename directories)
  - Added quota support

Exercise 1

- In what ways is FFS similar to the V6 file system?

- In what ways is FFS different from the V6 file system?

Design Rationale

- Speculate why each of the following design decisions was made, or identify a problem that you believe the design is addressing.
- The superblock is replicated many times in the file system (new idea); replicated on different platters and different cylinders.
  - Improves robustness: can still recover the file system, even if you lose a block, even a very important one.
  - Can recover if you have a head crash or if you lose a platter.
- Disk partition is divided into chunks called *cylinder groups*.
  - A cylinder group is a collection of adjacent cylinders.
  - A cylinder group has allocation information, inodes, data blocks, a copy of the superblock.
- Larger and variable-sized blocks; two sizes of allocation – blocks and fragments; on the order of 4-8 fragments per block; blocks on the order of 512 bytes to 64 KB.
  - Improve transfer bandwidth of blocks.
  - Avoid external fragmentation for small files.
- Bitmap-based allocation
  - Facilitates smart block allocation
  - Blocks in a file can be placed close together and positioned for good rotational behavior.

Exercise 2

- Assume that you can cache file system data and metadata in memory.
- What kinds of errors might you see after a crash?
- What techniques might you employ to avoid such problems?