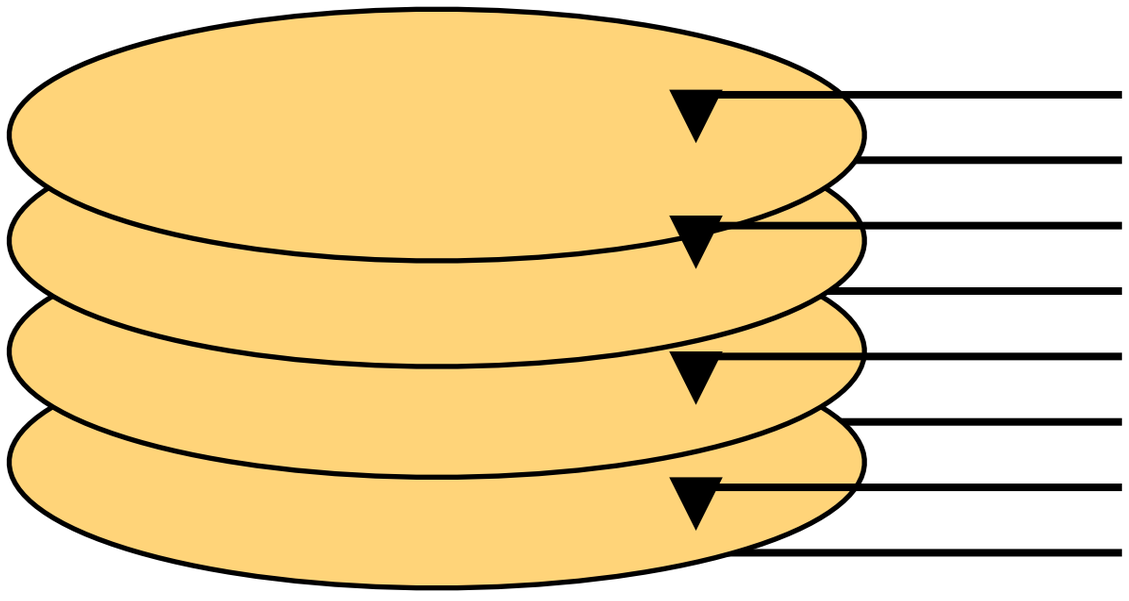
Lecture 11

File Systems

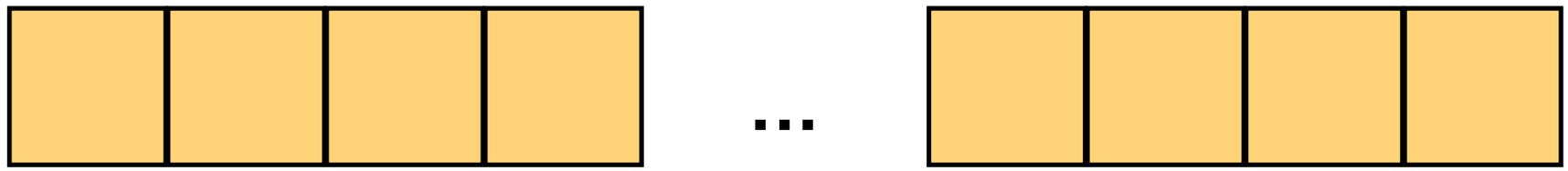
4 November 2011

Magnetic Disks



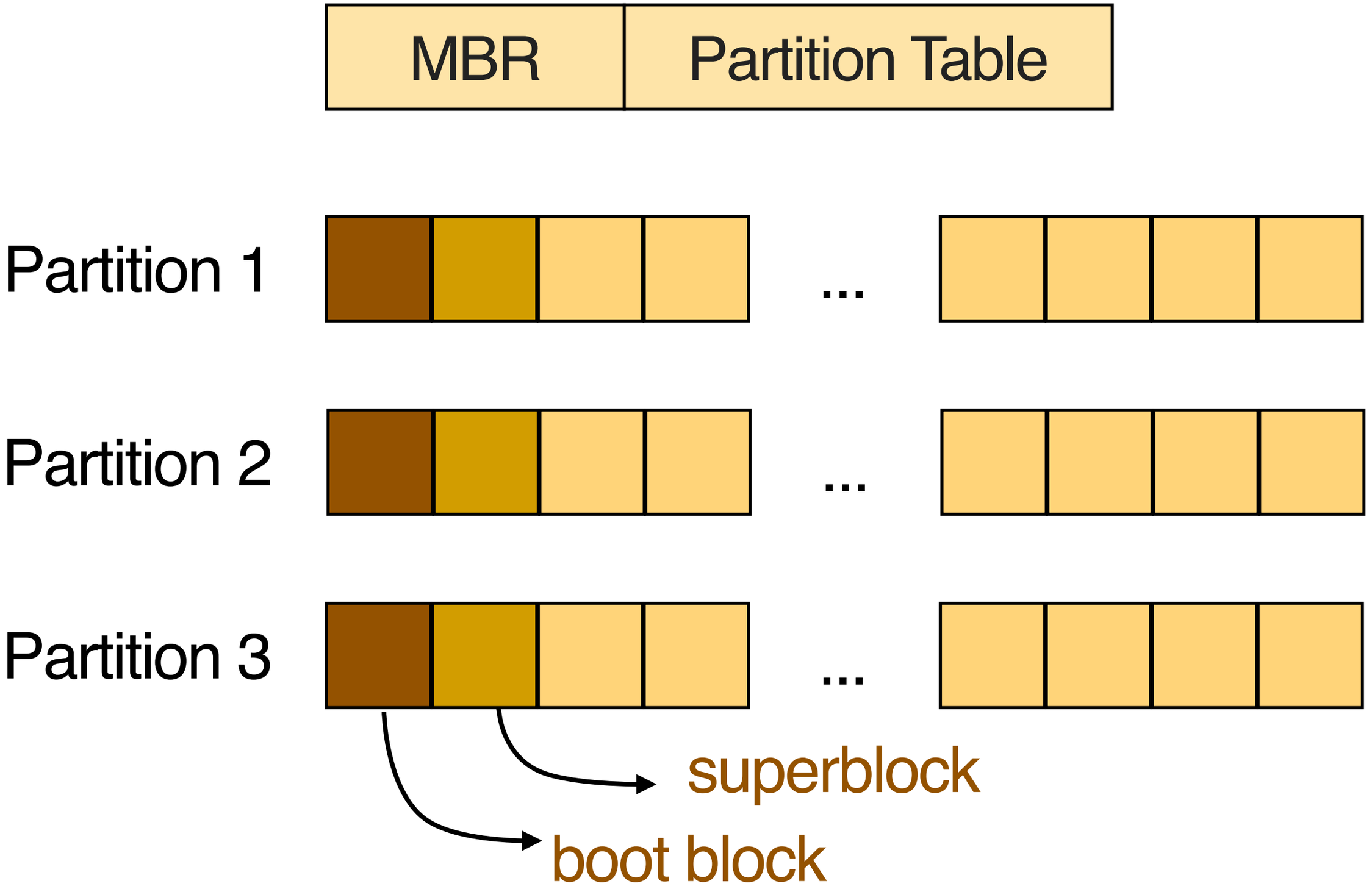
seek, rotate, transfer

Disk Blocks



which data stored where?
which blocks are free?
who owns the block?

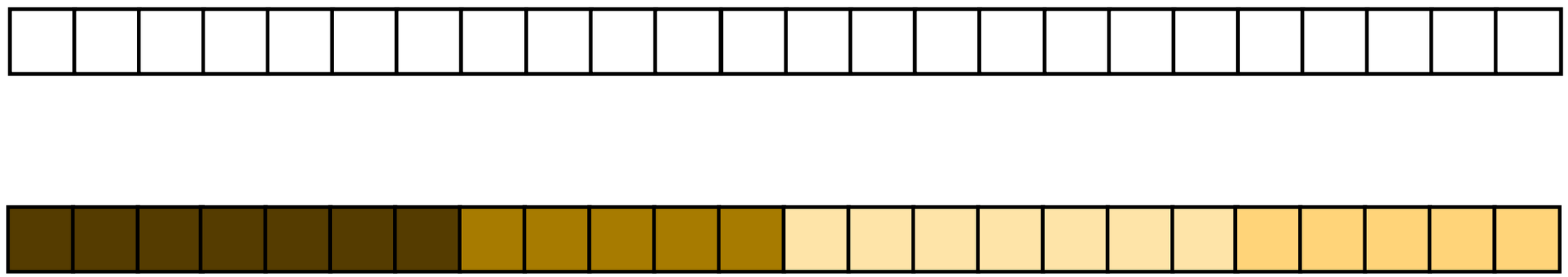
what is a partition?

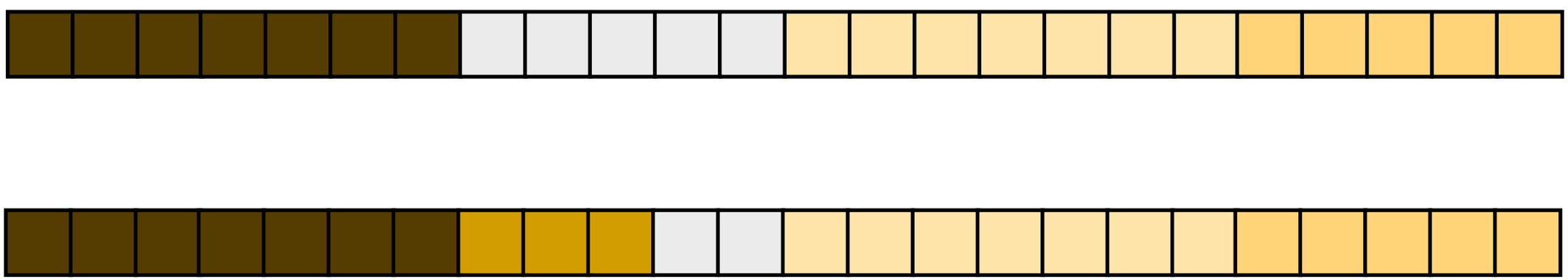

MBR
Partition Table
Partition 1
...
Partition 2
...
Partition 3
...
superblock
boot block

how to map files to blocks?

how to allocate blocks to files?

Contiguous Allocation





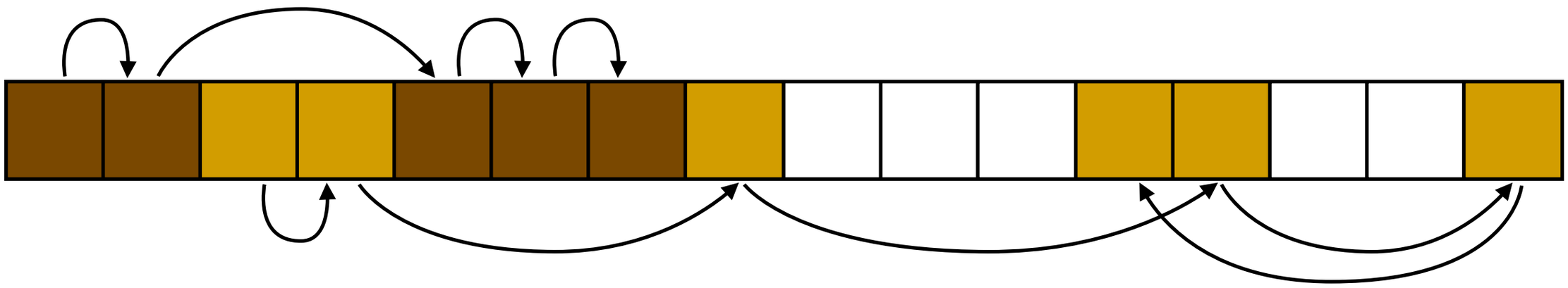
mapping files to blocks is **simple**

reading from disk is very **fast**

but need to know file size fragmentation of space

used in CD-ROMS

Using Linked List



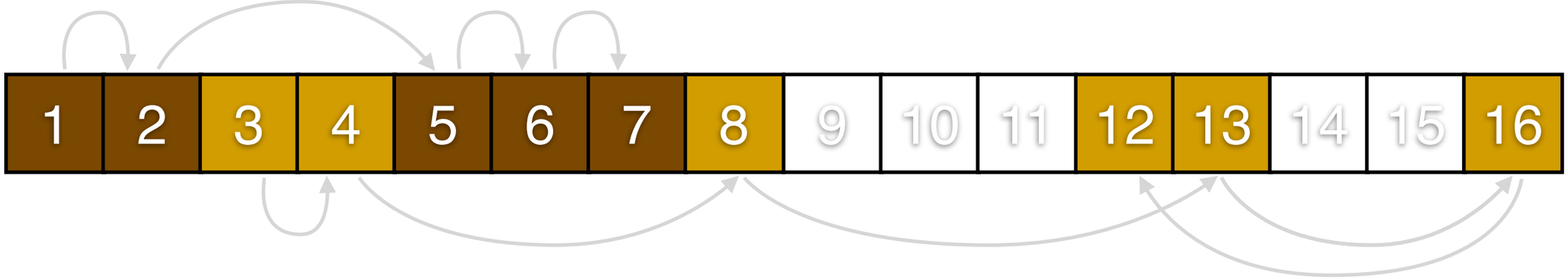
mapping files to blocks is **simple**

reading sequentially is **fast**

but

random access is slow

Using FAT



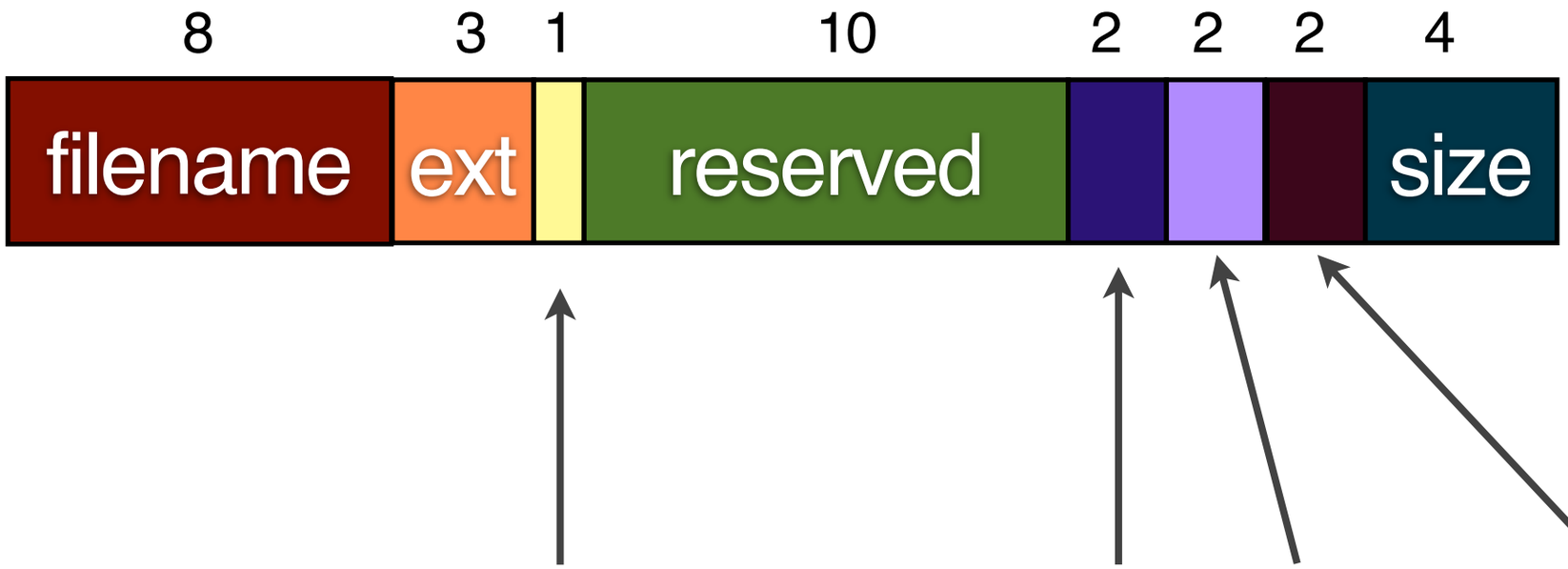
file allocation table (in RAM)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
2	5	4	8	6	7	-1	13				-1	16			12

how big can a FAT get?

assume 256 GB disk, 1 KB block

directory entry in FAT-12 (MSDOS)



how big can a disk partition get?

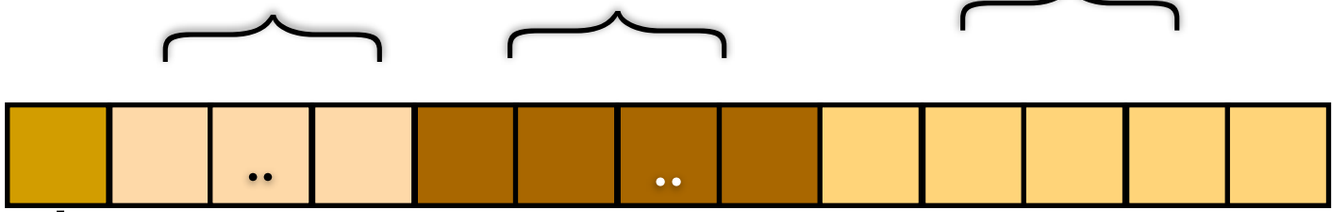
assume 512 bytes per block

Using i-nodes

metadata
(free blocks etc.)

i-nodes

data



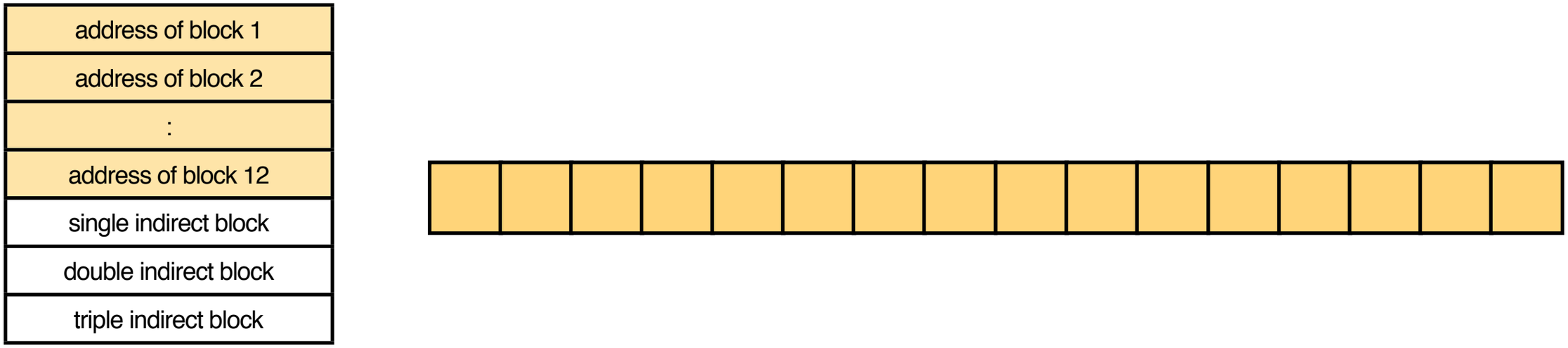
superblock

one **i-node** per file, containing info about files on disk

(owner, file type, size, address, last access time, last modified time etc.)

i-node addresses

address of block 1
address of block 2
.
:
address of block 12
single indirect block
double indirect block
triple indirect block


address of block 1
address of block 2
:
address of block 12
single indirect block
double indirect block
triple indirect block

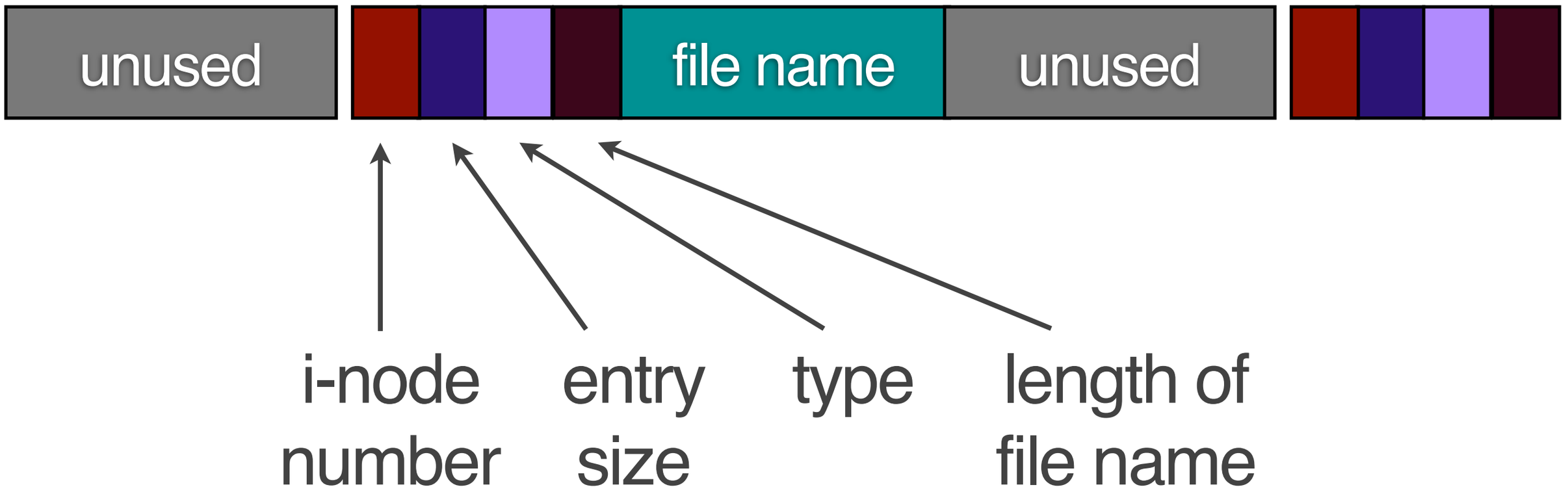
only stores i-nodes of opened files in memory (in i-node table)

access to small files is fast

still support large files

directories in UNIX are just files

directory entry on Linux (content of a directory “file”)



example: opening /home/user1/lab.c

read dir entries (data blocks) of /
look for i-node number for home
read i-node for /home
read data blocks of /home
look for i-node number for user1

:

:

:

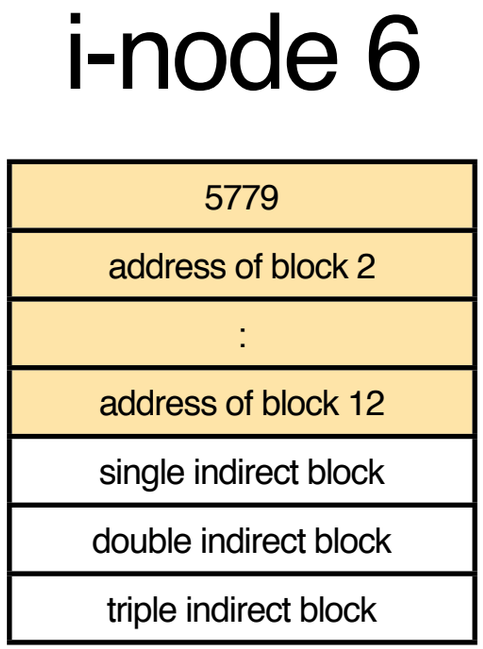
caching of directory entries in memory improves access time

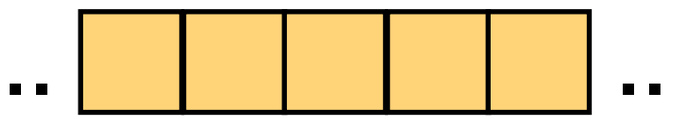
Sharing files

hard link

soft link

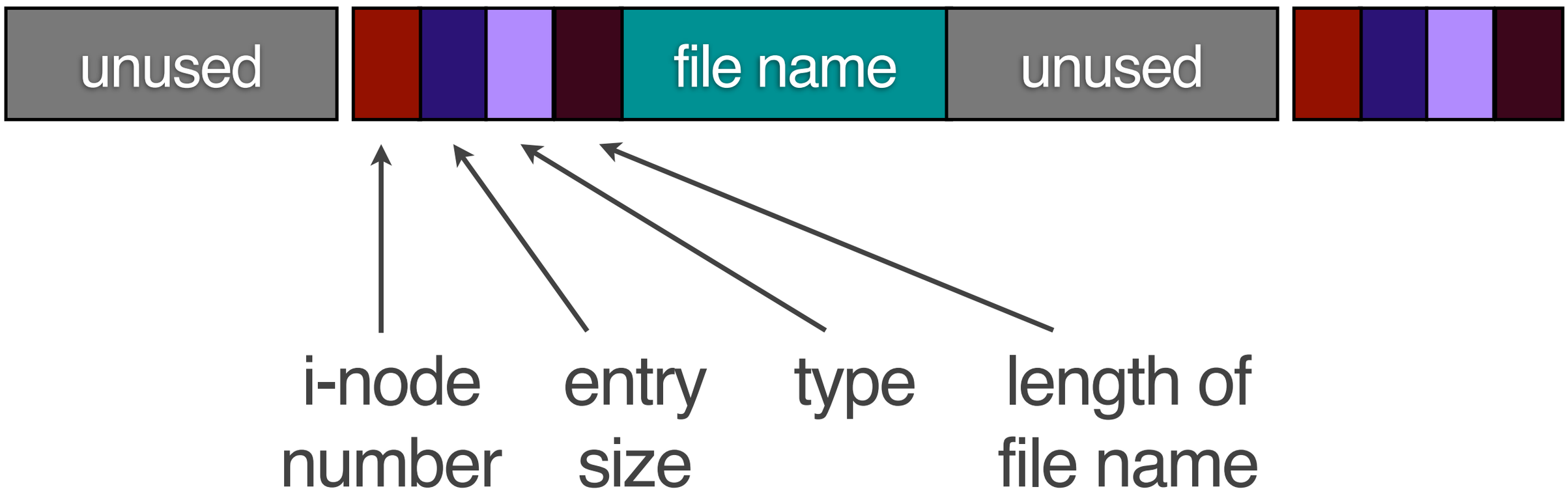
i-node number





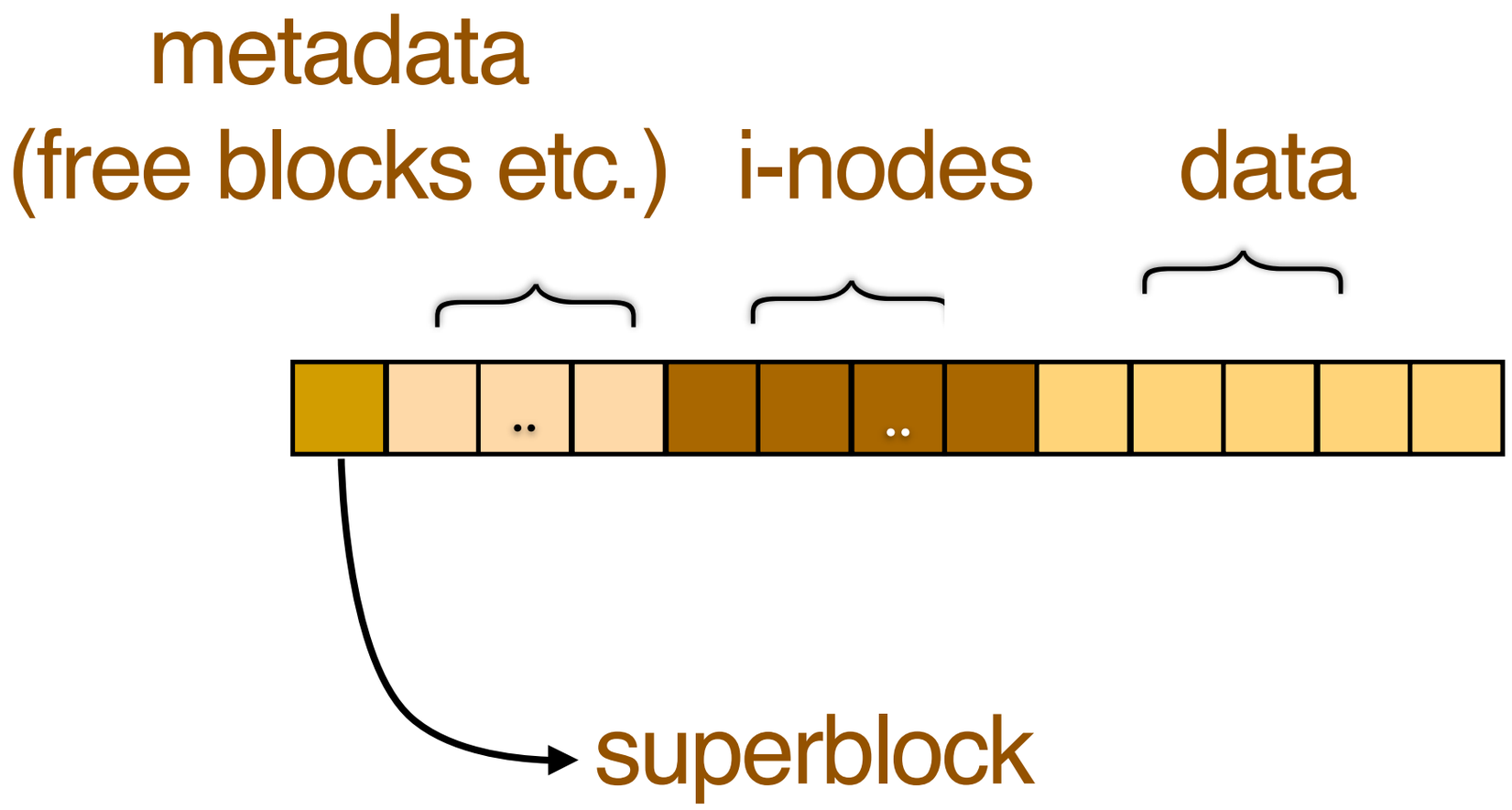
Removing a file

removes directory entry



marks blocks as free

marks i-node as free



what if the system crashes?

remove dir entry

release blocks

release i-node

journaling file system

write a log to disk

remove file

remove log from disk

journaling is used in

NTFS (Windows NT)

ext3 (Linux 2.4)

HFS+ (Mac OSX)

Improving File Systems Performance

1. Buffer Cache

caches disk blocks in memory

critical dirty blocks are written immediately to disk

data blocks are written periodically via `sync()`


The disk was not ejected properly. If possible, always eject a disk before unplugging it or turning it off.
To eject a disk, select it in the Finder and choose File > Eject. The next time you connect the disk, Mac OS X will attempt to repair any damage to the information on the disk.
OK


Safe To Remove Hardware
The 'USB Mass Storage Device' device can now be safely removed from the system.

2. Read Ahead

read more blocks than requested

3. Reducing Disk Arm Motion

(a) put relevant blocks together on the same cylinder

3. Reducing Disk Arm Motion

(b) schedule disk arm motion carefully

Disk Addressing

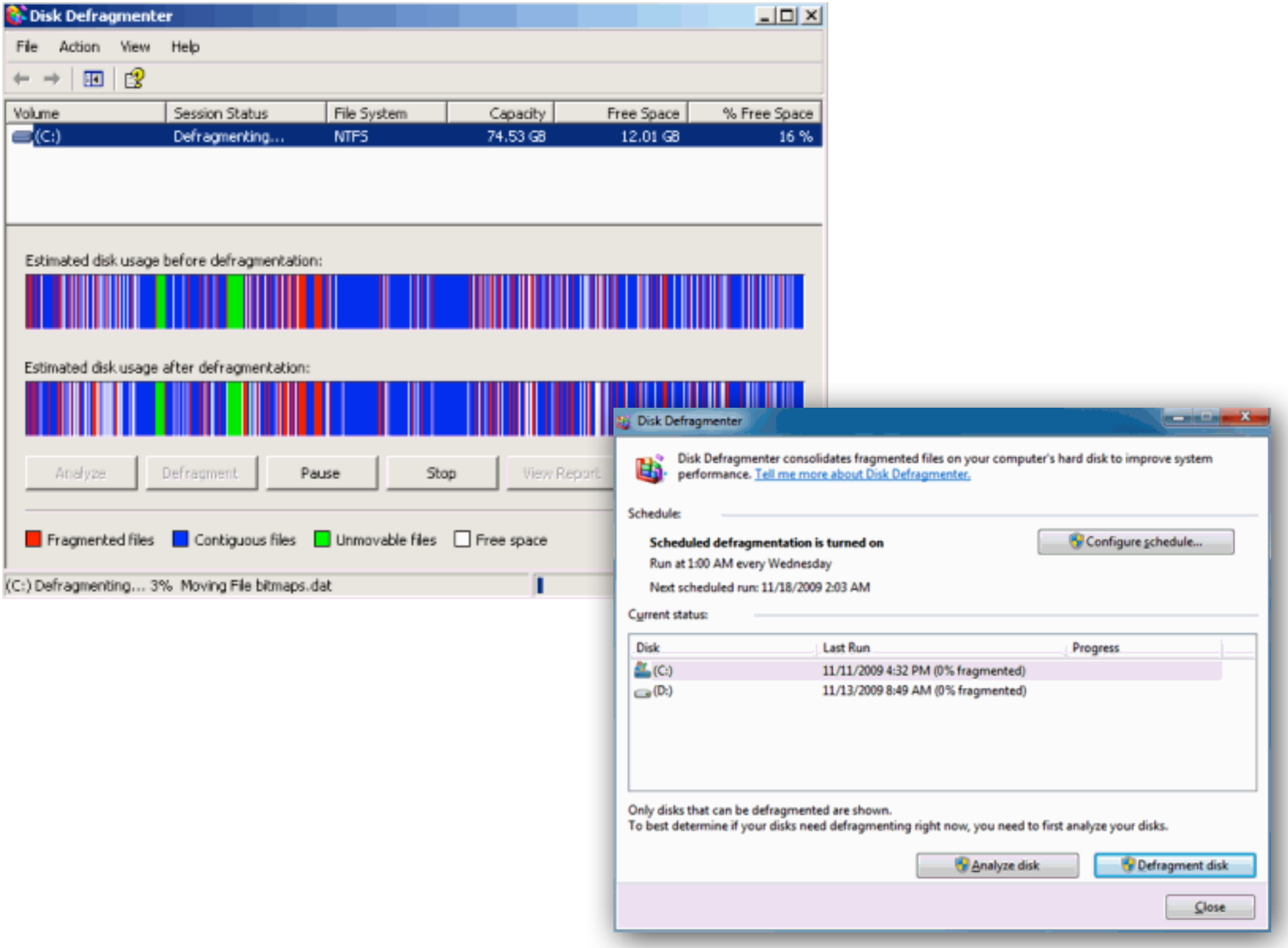
1. (cylinder, head, sector)
2. logical block address

FCFS
SSF
Elevator

Cylinder requests: 10, 9, 1, 11, 18, 19

4. Defragmentation

reorganize files on disk
to keep them contiguous


Disk Defragmenter
File Action View Help
Volume Session Status File System Capacity Free Space % Free Space
(C:) Defragmenting... NTFS 74.53 GB 12.01 GB 16 %
Estimated disk usage before defragmentation:
Estimated disk usage after defragmentation:
Analyze Defragment Pause Stop View Report
Fragmented files Contiguous files Unmovable files Free space
(C:) Defragmenting... 3% Moving File bitmaps.dat
Disk Defragmenter
Disk Defragmenter consolidates fragmented files on your computer's hard disk to improve system performance. Tell me more about Disk Defragmenter.
Schedule:
Scheduled defragmentation is turned on
Run at 1:00 AM every Wednesday
Next scheduled run: 11/18/2009 2:03 AM
Configure schedule...
Current status:
Disk Last Run Progress
(C:) 11/11/2009 4:32 PM (0% fragmented)
(D:) 11/13/2009 8:49 AM (0% fragmented)
Only disks that can be defragmented are shown.
To best determine if your disks need defragmenting right now, you need to first analyze your disks.
Analyze disk
Defragment disk
Close