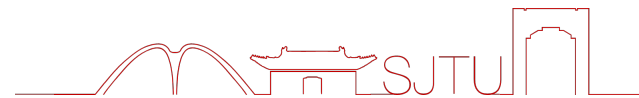




上海交通大学
SHANGHAI JIAO TONG UNIVERSITY



L11. 大容量存储结构

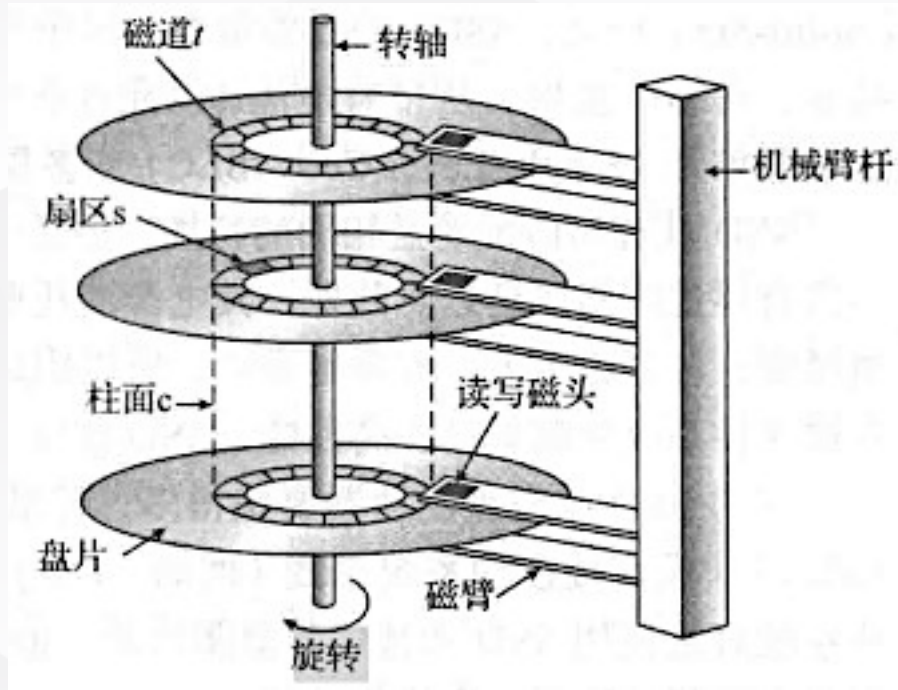
宋卓然

上海交通大学计算机系

songzhuoran@sjtu.edu.cn

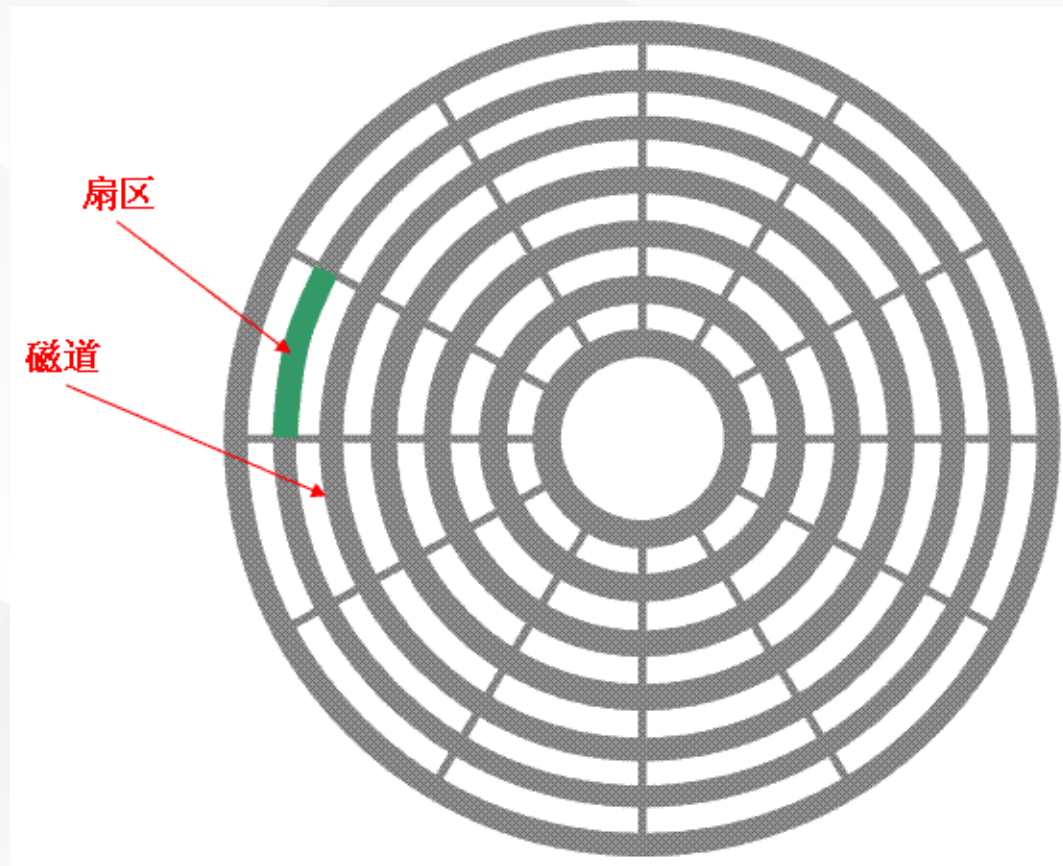
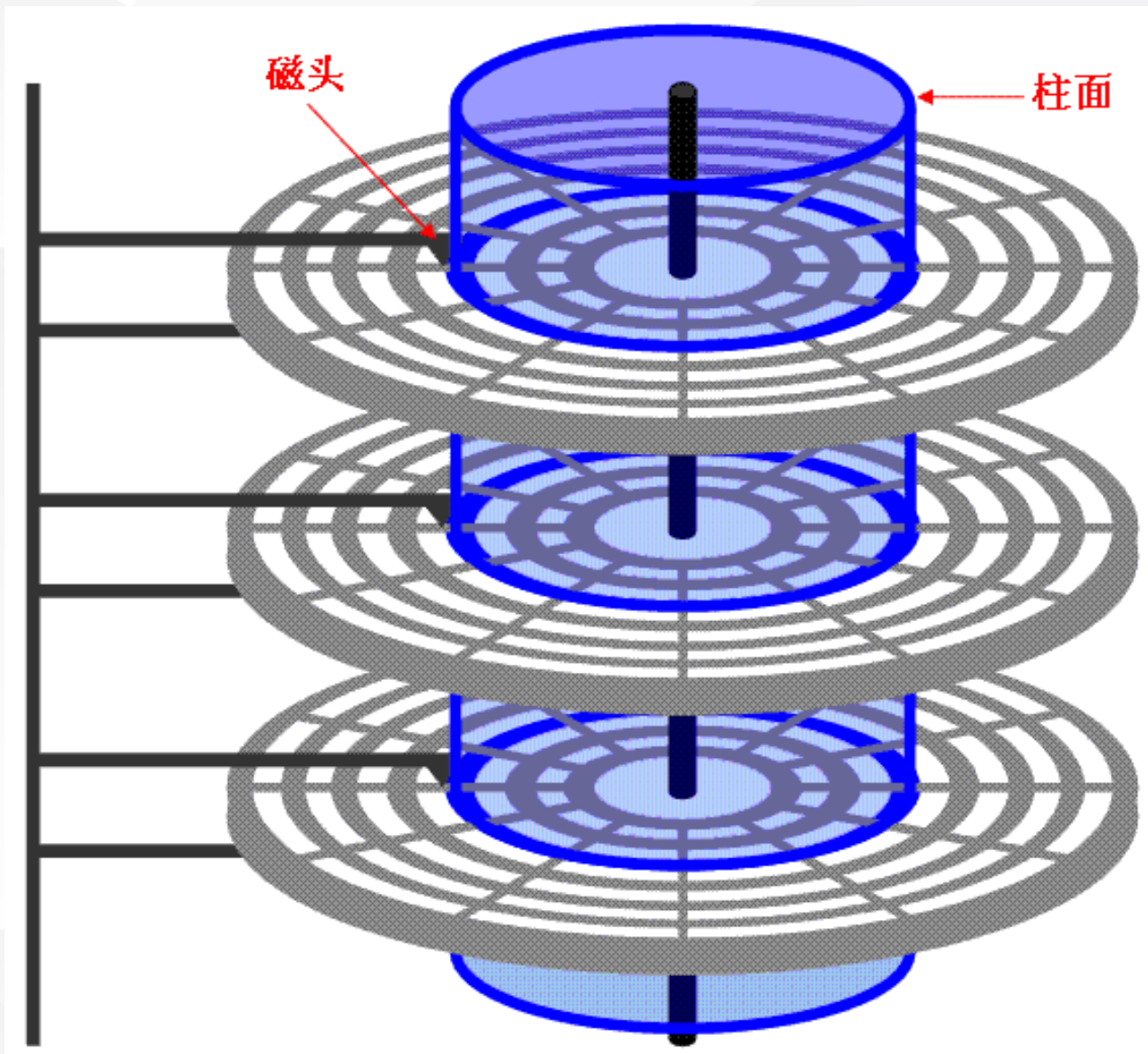
饮水思源 · 爱国荣校

- 磁盘用于存储数据
- 具有盘片，为图中平的圆状
- 普通盘片直径为1.8-3.5英寸
- 盘片两面都涂有磁质材料，通过在盘片上进行磁性记录可以保存信息
- 读写磁头在盘片上滑动，磁头附着在磁臂上，磁臂将所有磁头作为整体一起移动
- 盘片的表面逻辑地分成圆形磁道
- 磁道分为**扇区**，是读写的基本单位
- 同一磁臂位置的磁道集合形成了柱面





磁盘

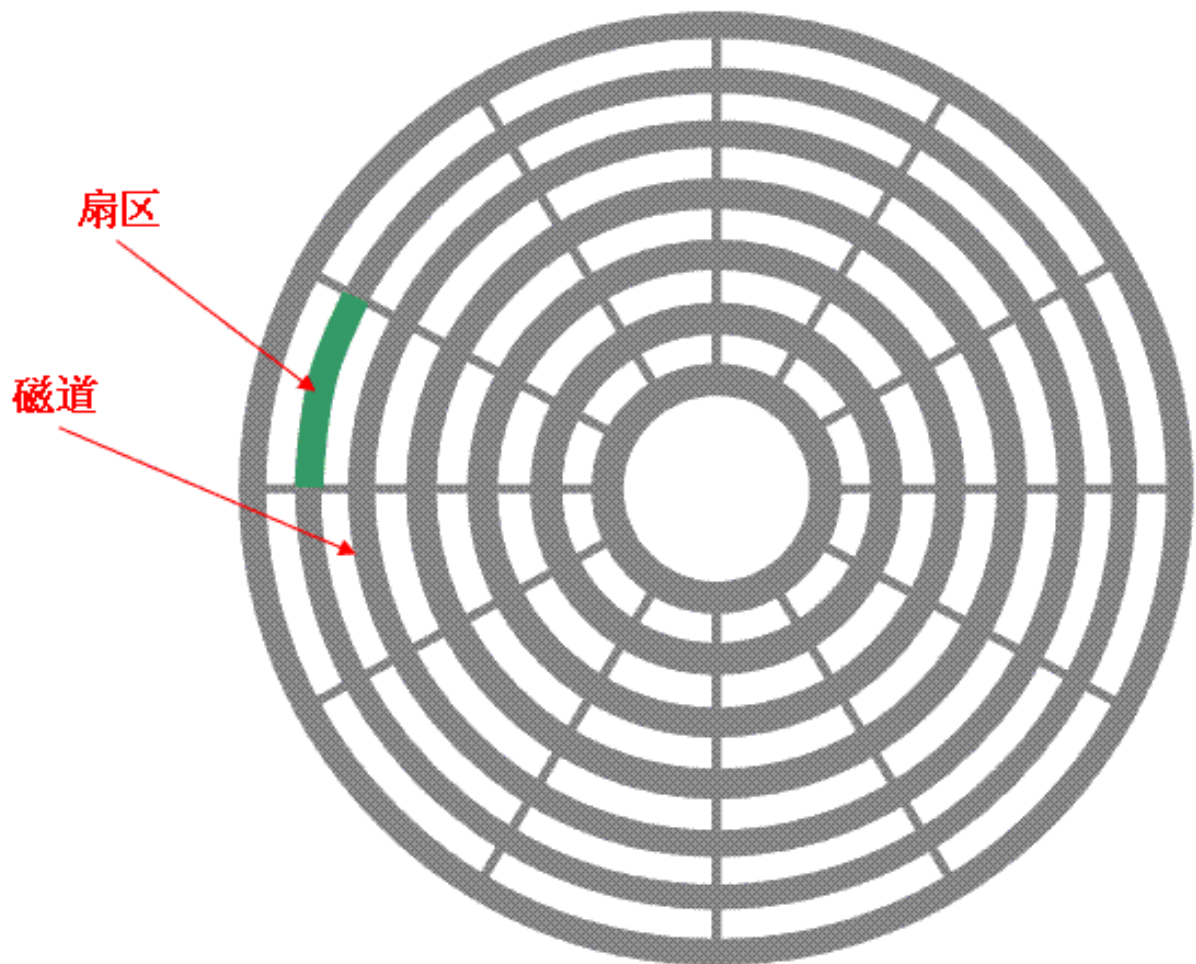




磁盘使用

- 磁盘读取数据的过程
 - 将磁头移动到指定的磁道上
 - 磁道旋转，转到相应的扇区
 - 旋转，磁生电，读取数据
- 磁盘访问时间 =
寻道时间（12ms-8ms，移动磁臂，机械运动）+ 旋转时间（4ms，寻找扇区，机械选择）+ 传输时间（0.3ms）

尽量少寻道！





第一款商用磁盘



1956
IBM RAMDAC computer
included the IBM Model
350 disk storage system

5M (7 bit) characters
50 x 24" platters
Access time = < 1 second



不同大小的磁盘





固态硬盘 (solid-state disk , SSD)

- SSD相较传统磁盘，速度更快，更昂贵，同时寿命更短
- 写的次数有限





大容量存储结构总结

- 磁盘为现代计算机提供大量辅助存储
- 访问磁盘，速度有两方面
 - 传输速率是数据在驱动器和计算机之间流动的速率
 - 定位时间（随机存取时间）包括：将盘臂移动到所需柱面的时间（寻道时间）和所需扇区在盘头下旋转的时间（旋转延迟）

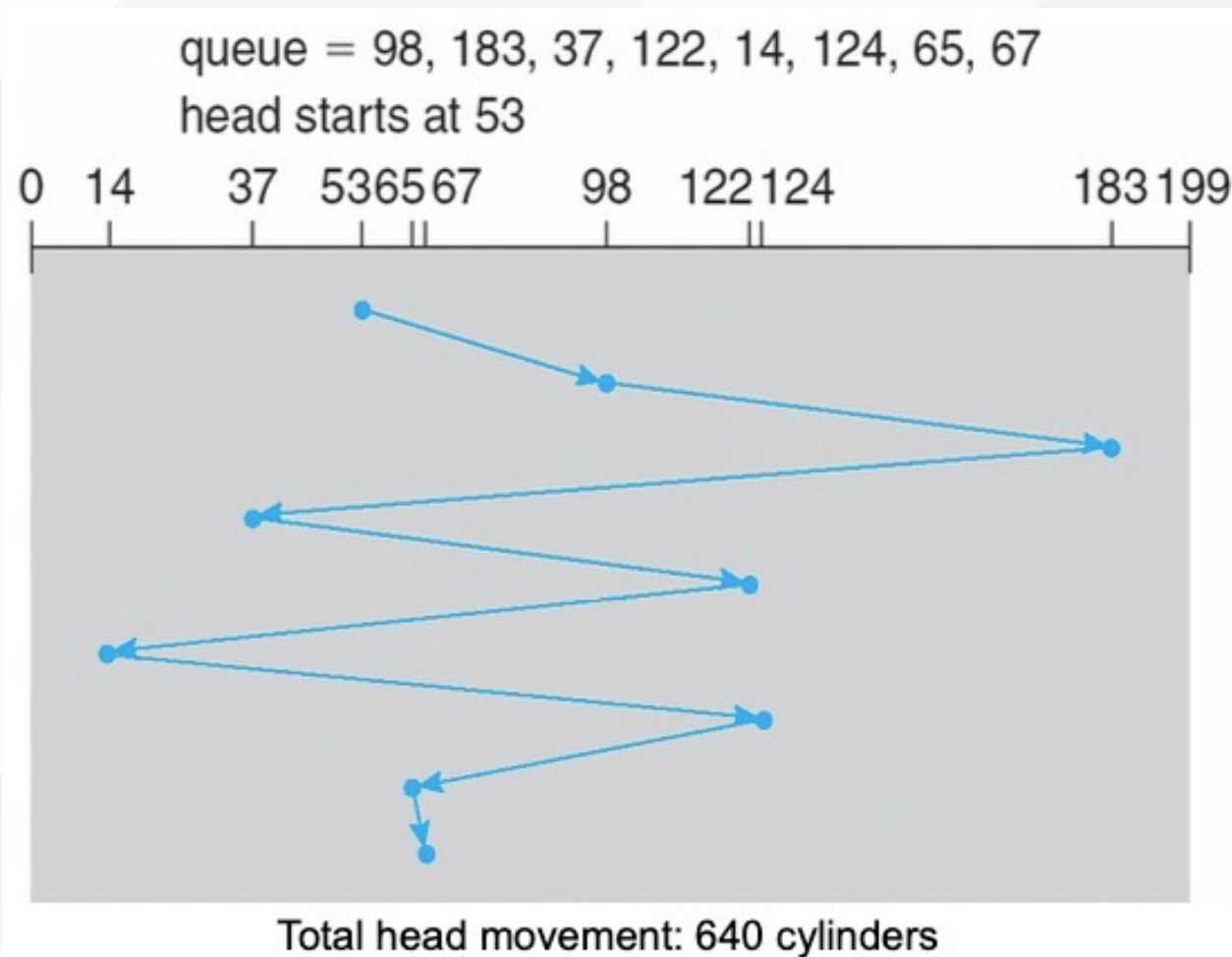


- 磁盘队列中可能有多个待处理的请求，选择哪个？采用磁盘调度算法
 - 先来先服务(First-Come First Served, FCFS)算法
 - 最短寻道时间优先(Shortest-Seek- Time-First, SSTF) 算法
 - 扫描算法(SCAN algorithm)
 - 循环扫描(Circular SCAN,C-SCAN)调度
 - LOOK调度



先来先服务(First-Come First Served, FCFS)算法

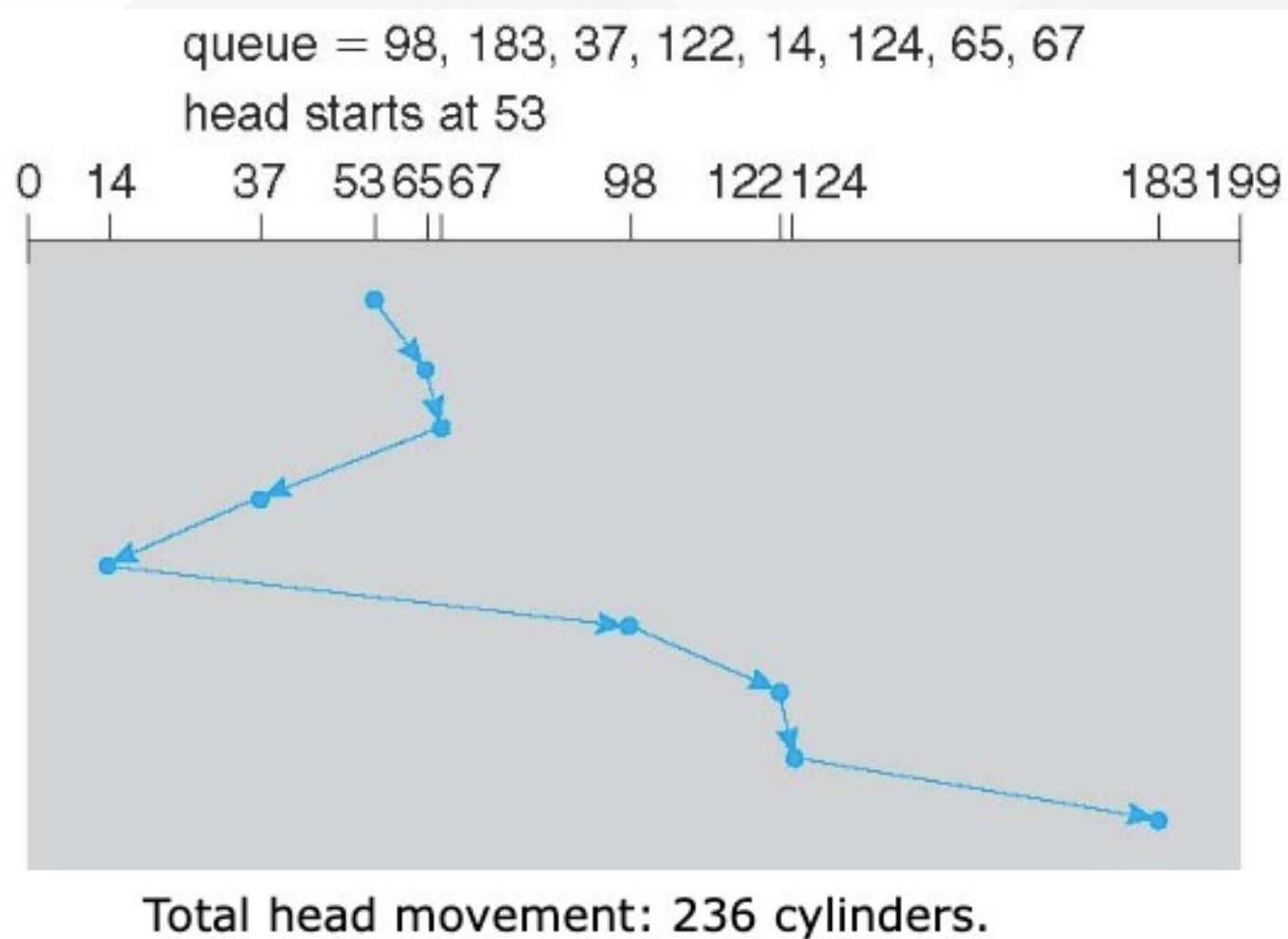
- 考虑请求块的柱面的顺序为：98, 183, 37, 122, 14, 124, 65, 67；起始块为53





最短寻道时间优先(SSTF) 算法

- 优先处理靠近当前磁头位置的请求





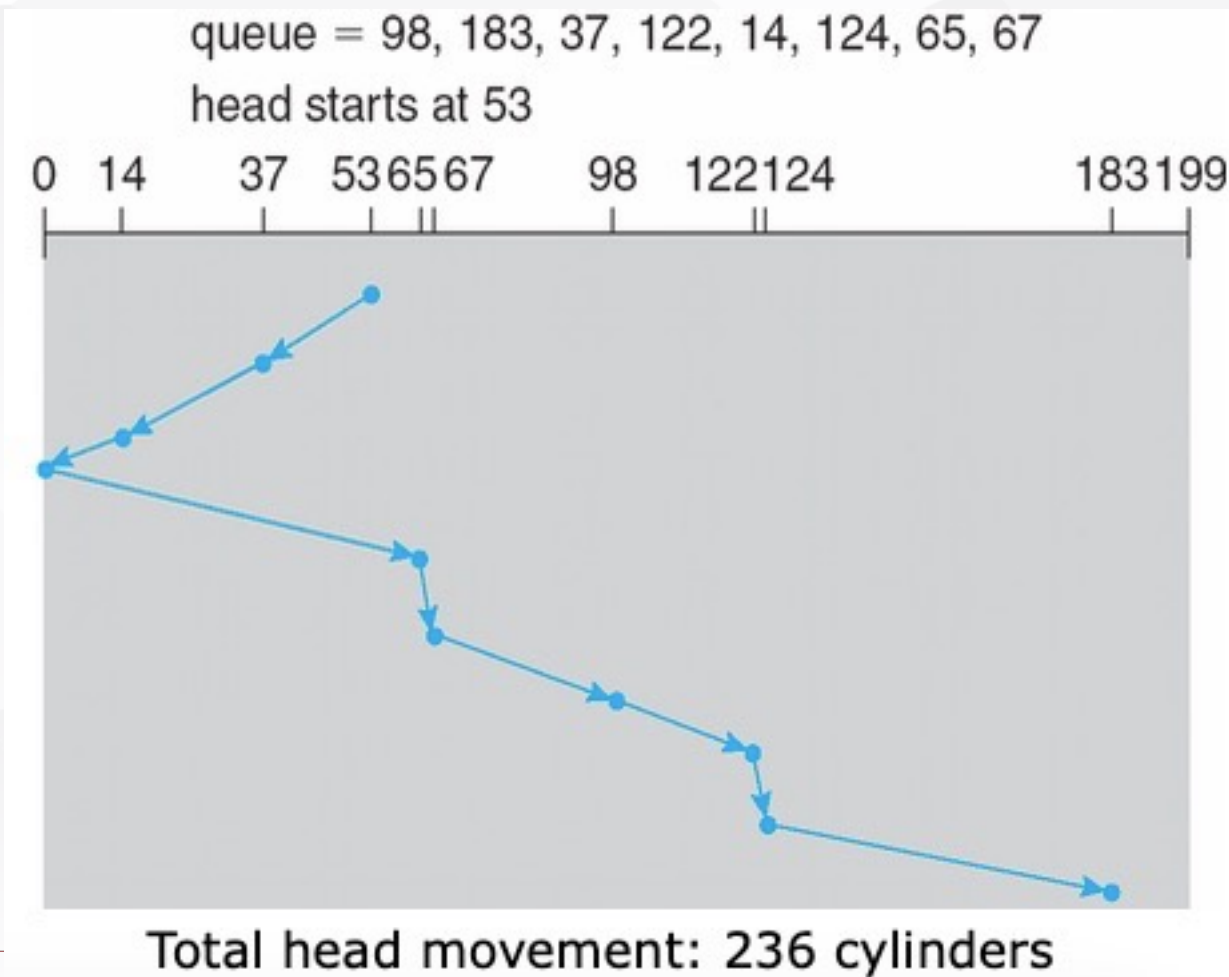
最短寻道时间优先(SSTF) 算法

- 优先处理靠近当前磁头位置的请求
- 依然考虑请求块：98, 183, 37, 122, 14, 124, 65, 67；起始块为53
- 问题：
 - 可能导致一些请求饥饿，考虑假设有两个请求14和186，起始为20，则会先处理请求14，此时若来了一个新请求15，会处理15而非186
 - 性能比FCFS好，但并非最优，对于上述例子，可以移动磁头从53到37（虽然37并不是离53最近的），再到14，再到65、67、98、122、124、183，此时移动到柱面总数为208



扫描算法(SCAN algorithm)

- 磁臂从磁盘的一端开始，向另一端移动；在移过每个柱面时，处理请求。当到达磁盘另一端时，开始反向移动





扫描算法(SCAN algorithm)

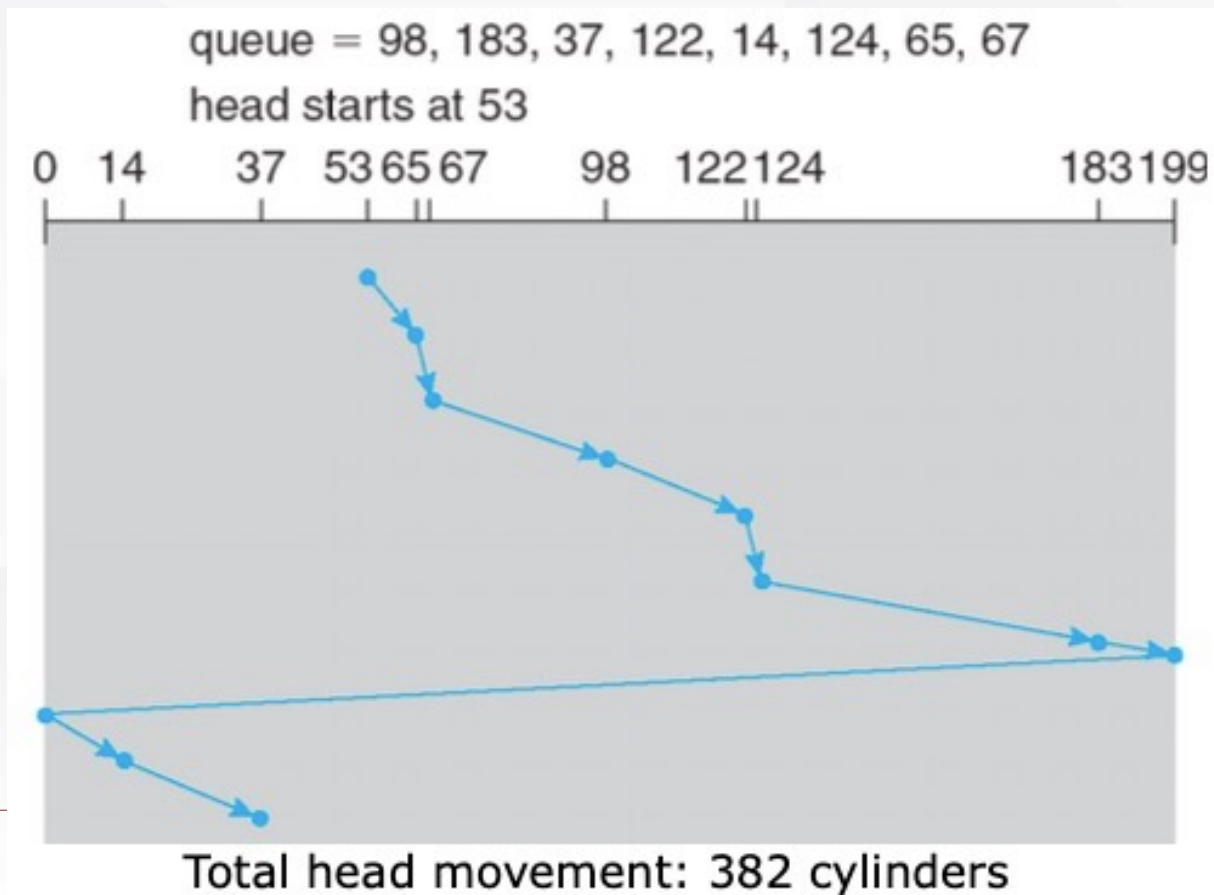


- 磁臂从磁盘的一端开始，向另一端移动；在移过每个柱面时，处理请求。当到达磁盘另一端时，开始反向移动
- 适合请求柱面较均匀的情况
- 若紧靠磁头前方的请求相对较少，因为最近处理过这些柱面。磁盘另一端的请求密度却是最多，那么为什么不先处理请求密度更高的一端？循环扫描调度算法的由来



循环扫描(C-SCAN)调度

- 类似SCAN调度，C-SCAN移动磁头从磁盘的一端到另一端，并处理行程中的请求
- 但当磁头到达另一端时，它立即返回到磁盘的开头，而不处理任何回程上的请求

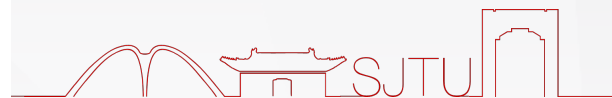
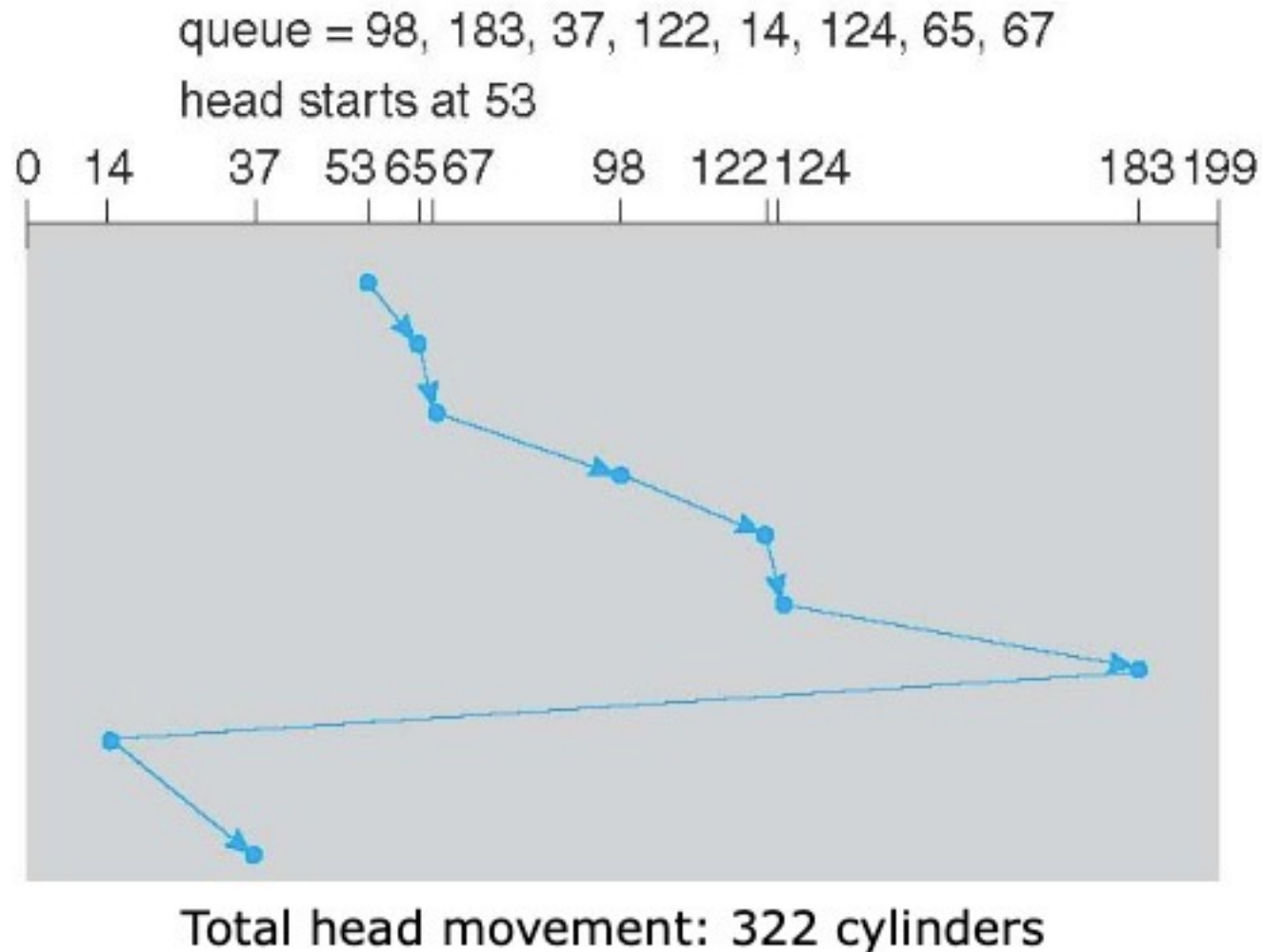




LOOK调度

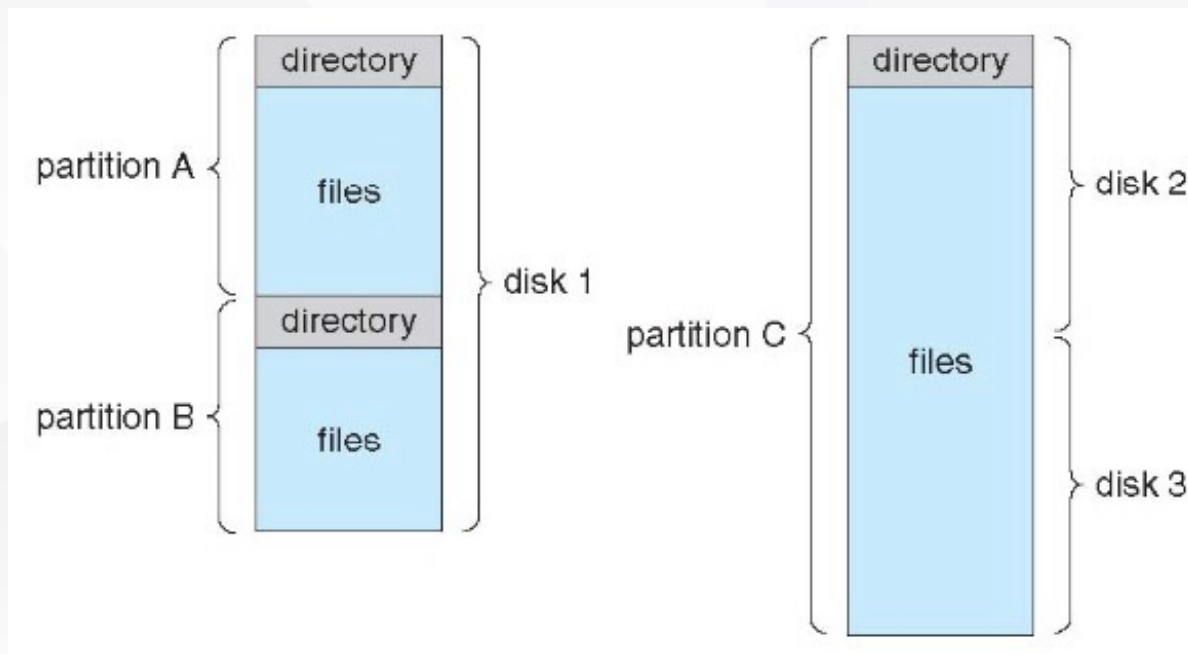


- SCAN和C-SCAN在磁盘的整个宽度内移动磁臂，浪费
- LOOK和C-LOOK调度则只需要将磁臂移动到最远请求为止





- 分区：硬件磁盘的一种适合操作系统指定格式的划分
- 卷：两个磁盘一起服务于一个文件系统
 - 多个磁盘是否可以提高文件访问的效率？
 - 让两个磁盘存放文件系统同样的内容，是否可以提高文件系统的可靠性？



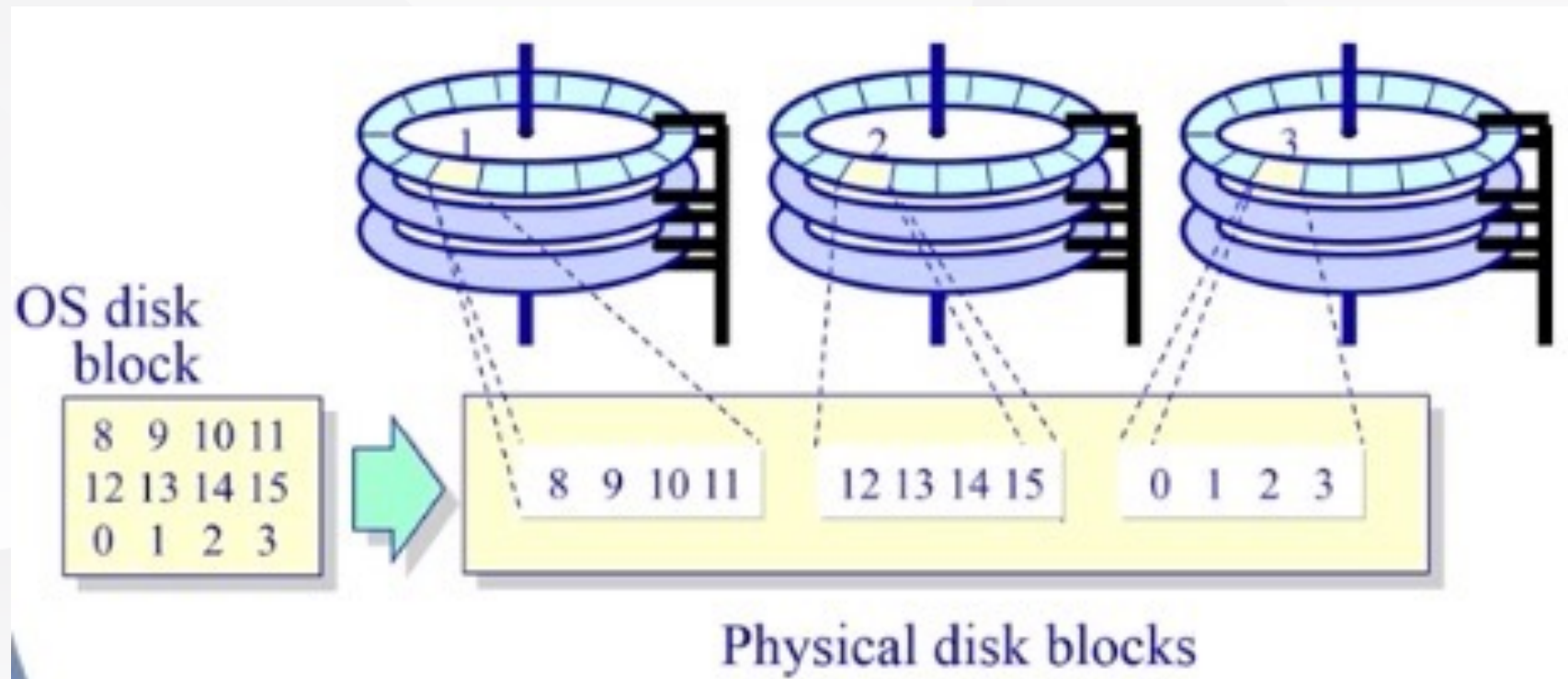


- 使用多个并行磁盘
 - 吞吐量（通过并行）
 - 可靠性和可用性（通过冗余）
- 出现了 RAID 冗余磁盘阵列
 - RAID有多级：RAID-0、RAID-1、RAID-5



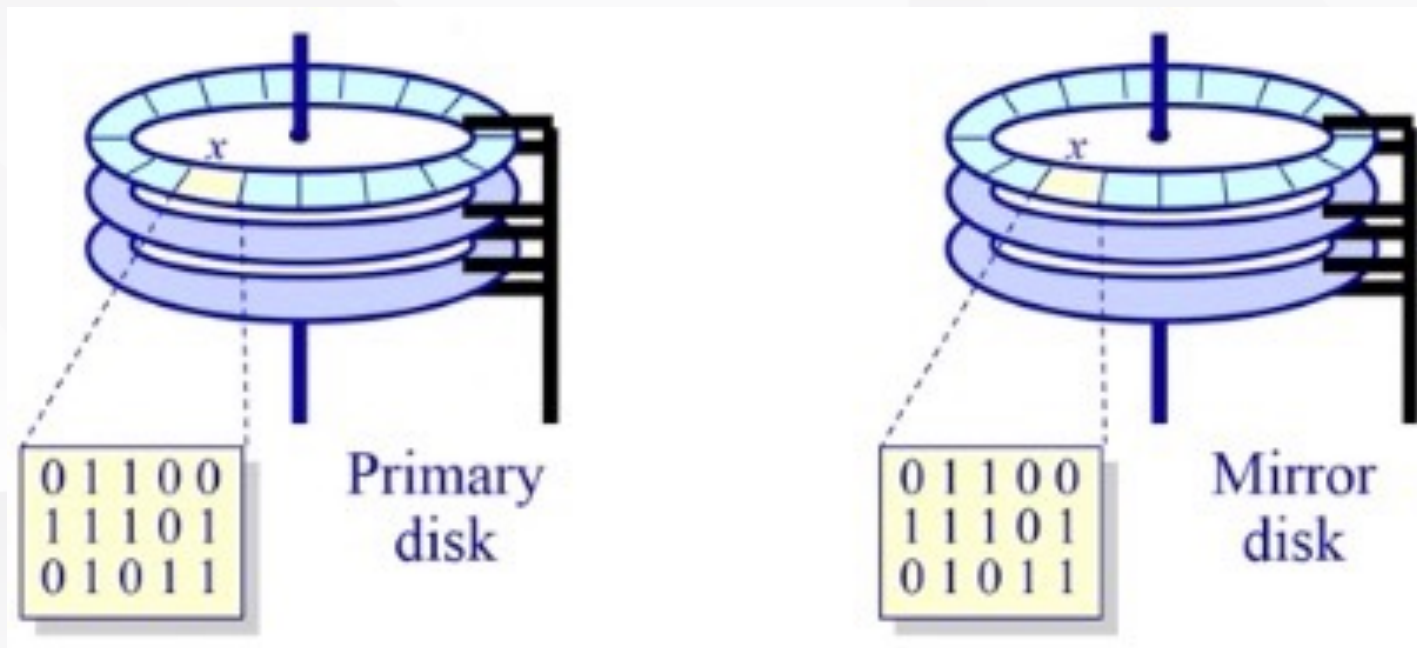
多磁盘管理 RAID-0

- 数据块分成多个子块，存储在独立的磁盘中
- 通过更大的有效块大小来提供更大的磁盘带宽



多磁盘管理 RAID-1

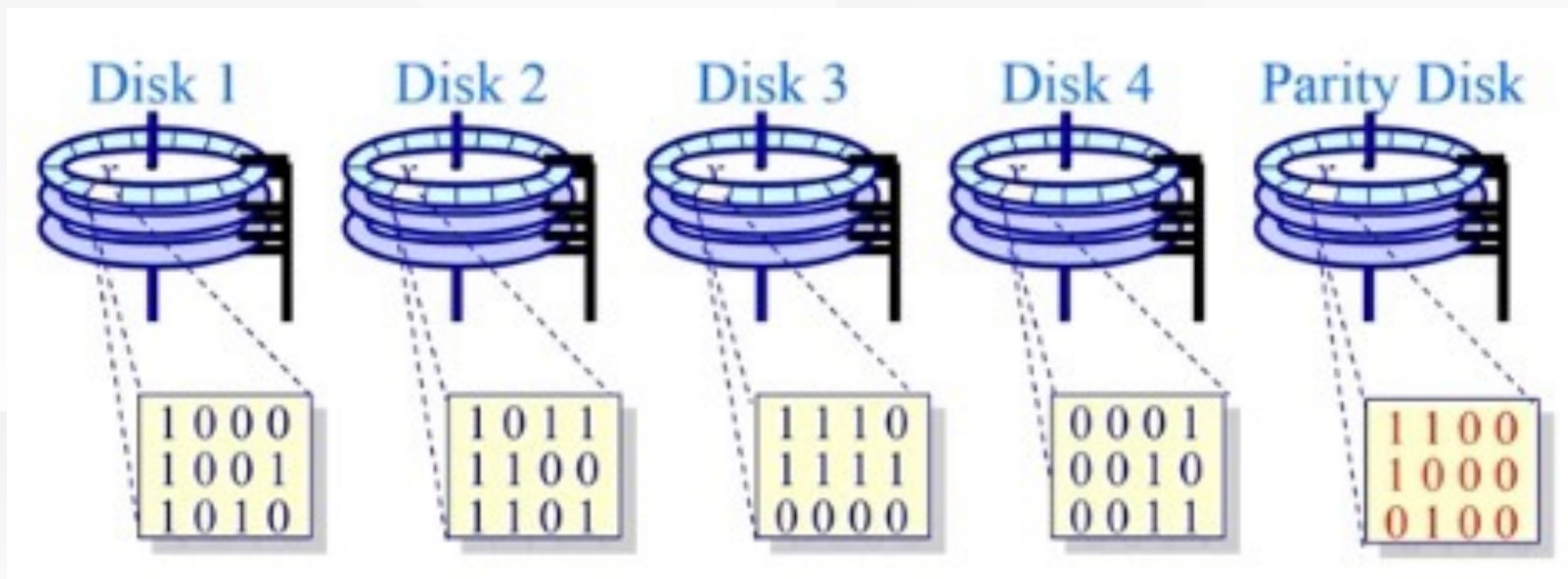
- 可靠性成倍增长
 - 两个磁盘存一样的数据
- 读取性能线性增长
 - 从两个磁盘写入，从任何一个读取





多磁盘管理 RAID-4

- 结合RAID-0和RAID-1的优势
- 数据块级磁盘配有专用奇偶校验磁盘
 - 允许从任意一个故障磁盘中恢复





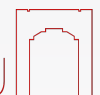
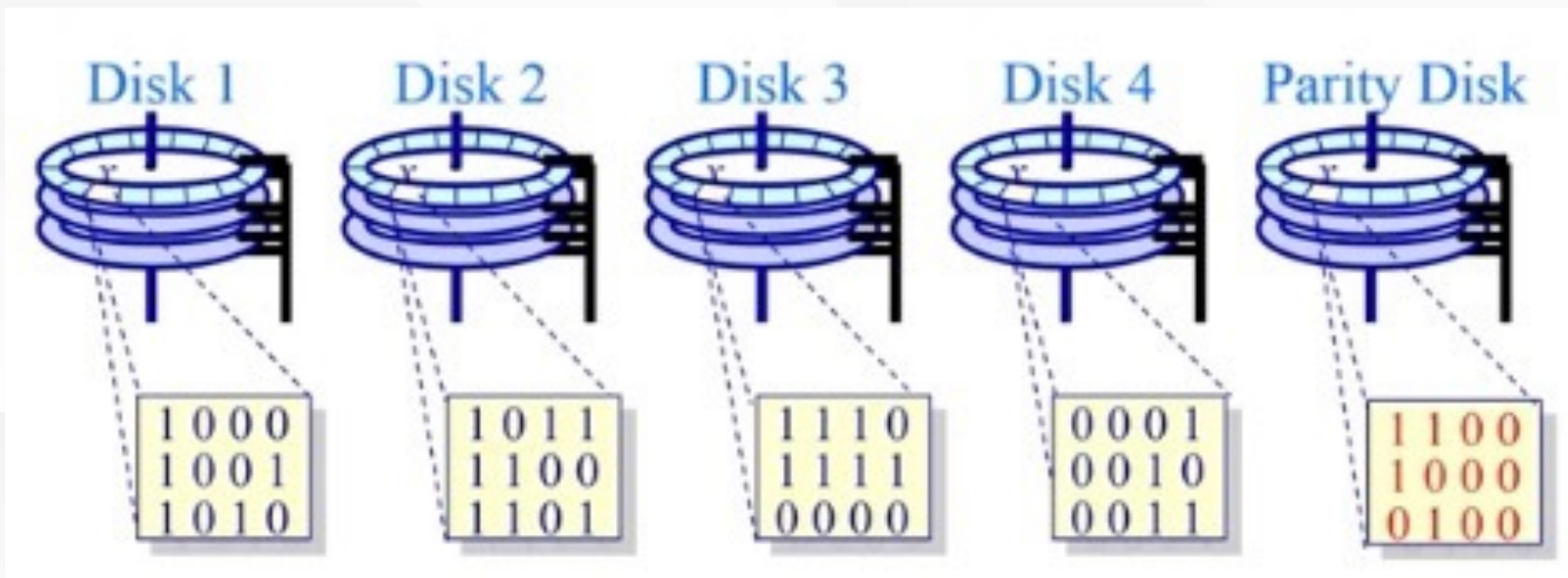
- 奇偶校验
 - 奇偶校验是在通信过程中确保节点之间准确数据传输的过程。奇偶校验位附加到原始数据位以创建偶数或奇数位
 - 假设存储了8位数，并增加一位错误检测位。数据为1、1、1、0、0、1、0、1，那么把每个位相加（ $1+1+1+0+0+1+0+1=5$ ），结果是奇数
 - 奇校验：8位加上检测位对应数字的和为奇数，因此设置错误检测位为0
 - 偶校验：8位加上检测位对应数字的和为偶数，因此设置错误检测位为1
- 奇偶校验只能在一定程度上判断是否出错，无法知道**错误位置**



多磁盘管理 RAID-4

- 结合RAID-0和RAID-1的优势
- 数据块级磁盘配有专用奇偶校验磁盘
 - 允许从任意一个故障磁盘中恢复
 - 向数据磁盘写入数据时，也需要更新奇偶校验磁盘

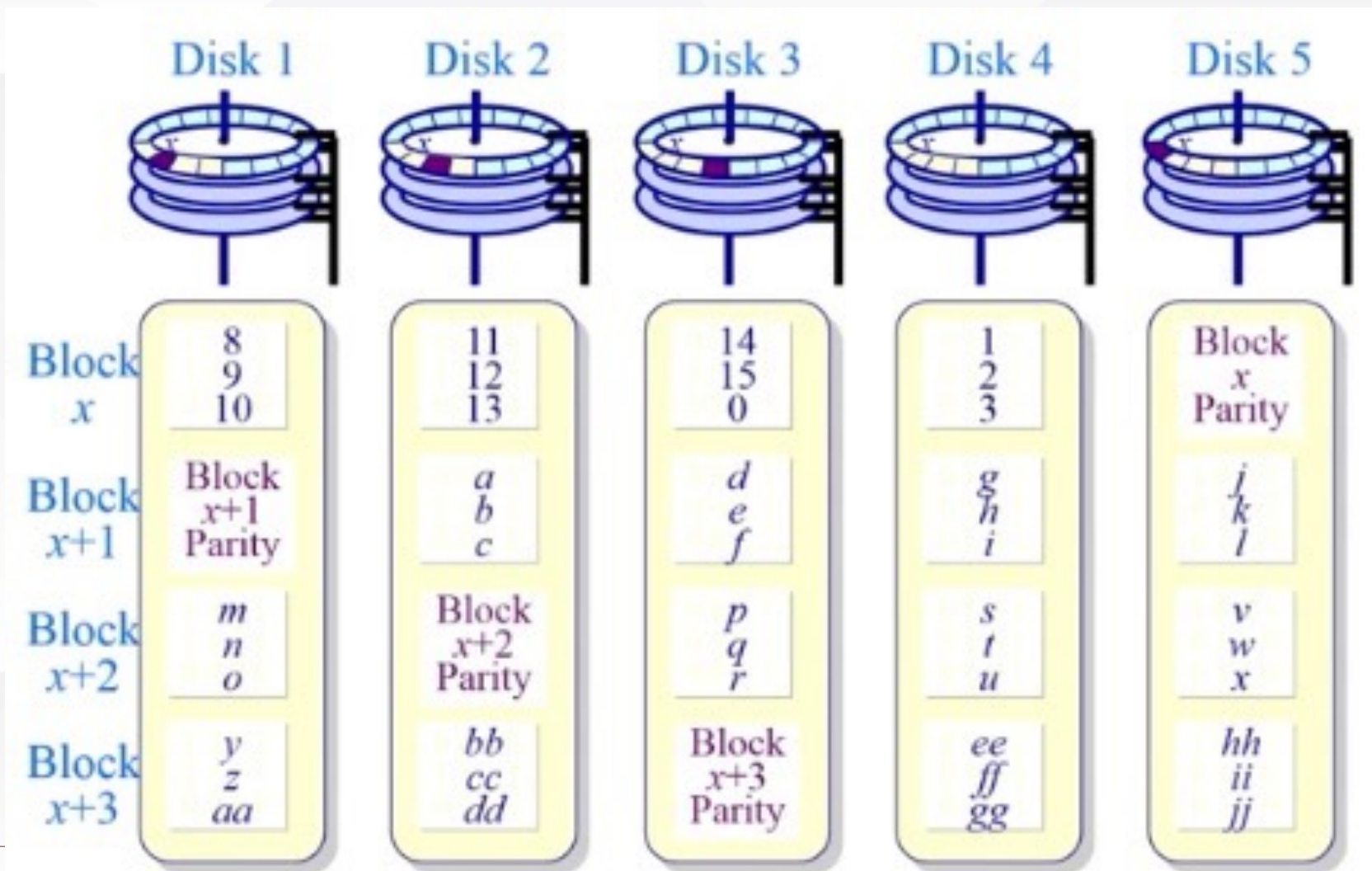
奇偶校验磁盘
写入过于频繁！





多磁盘管理 RAID-5

- 将奇偶校验块均匀地分布在数据块里





- 分层次的组织方式
 - 高效
 - 容错

