CH8

硬盘结构

物理地址:1. 柱面(相同半径大小的轨道,一个圆柱体)、2. 轨道(盘片被划分成的一个一个圈)、3. 扇区(每个轨道被划分为若干个扇区)

访问

寻道和旋转，寻道是把磁头移动到对应柱面

访问速率:分两种

1. 线速度恒定(CLV)
2. 角速度恒定(CAV)

使用:采用地址映射(address mapping)，(柱面、磁道、扇区)三元组

坏块管理

原因:磁盘可能会发生错误、失效的扇区相当普遍

处理方法:

1. 维护一个坏块表、预留一定的空闲扇区
2. 逻辑上把空闲块映射到坏块，问题:磁盘调度算法失效。应对:每个柱面匀出扇区
3. 重映射到另一扇区(需要数据迁移)

物理格式化硬盘

1. 把硬盘按扇区分开，让控制器可以读写
2. 每个扇区填充一些数据结构(纠错码、扇区号)
3. 在工厂完成

使用硬盘

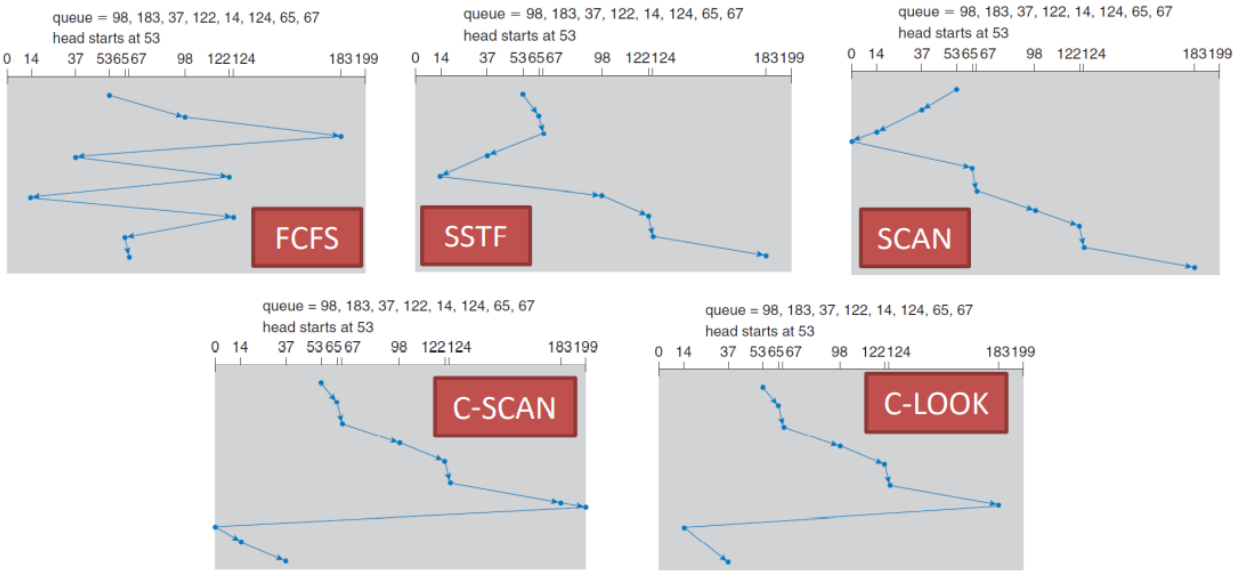
1. 文件系统

    a. 分成数个柱面，即各种盘
    b. 逻辑格式化，存原始的文件系统数据结构
    c. IO优化。磁盘以块读写、文件系统以簇读写。减小随机读写、增加顺序读写

2. 纯硬盘

    a. 把硬盘当成一个很大的数据组，无文件系统
    b. 无视文件系统服务，准确的控制文件的位置

硬盘调度



1. FCFS先到先服务

    优点:公平。缺点:性能差

2. 短时间先服务SSTF

    优点:快。缺点:可能导致饥饿

3. 扫描(从磁盘的一端扫到另一端)

    问题:在一端的需要等很久

4. 循环扫描(从磁盘的一端扫到另一端，在返回开始点)

    扫描和循环扫描有效应对高负荷系统

5. C-LOOk(中文不知道)，从一端到另一端，但是观望是否到尽头

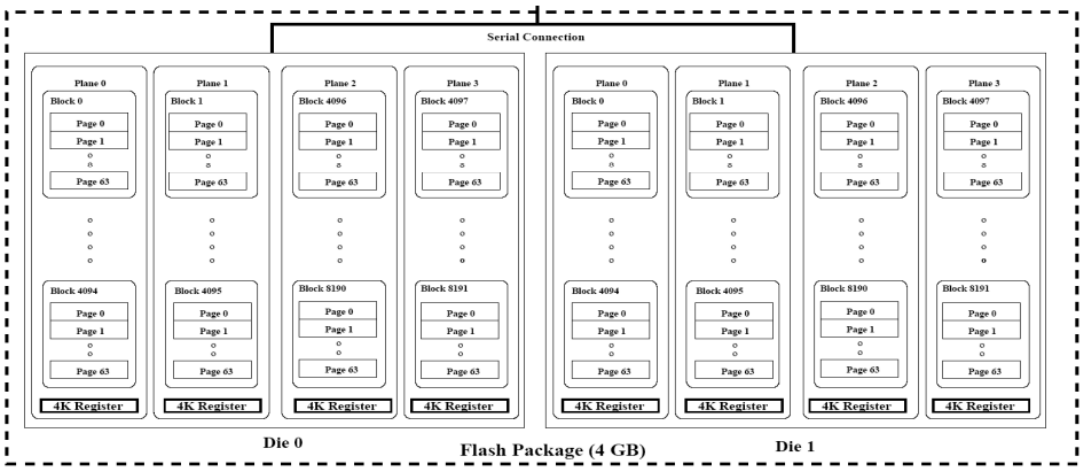
SSD

结构

组成部分:

1. multiple flash package

   Package > die/chip > plane > block > page



   闪存颗粒:

   a. 每个颗粒存1/多个位
   b. 程序只能把硬盘数据从1变成0，擦除从0变成1.
   c. 浮栅晶体管，写次数增多稳定性下降

   NAND闪存:SLC和MLC，SLC是1颗粒1位数据，MLC是1颗粒多数据，SLC更稳定

2. 控制器
3. RAM

SSD工作行为

1. 读

   按页读取

2. 写

   同上

3. 擦除

   按块擦除(64/128页，置1)

   P/E寿命:写入擦除寿命

4. 覆盖和删除

删除:把页标记为无效

覆盖更新:不支持就地覆盖，只能写到未标记为脏的页

RMW和**RRW**

RMW

每次修改时只修改要修改的块和校验块(即使是连续地修改也是如此)

RRW

每次修改时都要读取全部块，修改完后再放回(连续修改时，连续修改完在放回)

FTL

功能:

1. 地址映射
2. 垃圾回收
3. 损耗均衡

地址映射

1. 扇区映射

   闪存中的扇区映射到映射表

   问题:需要的内存很大
2. 块映射

   映射方式:

   $$logicalwriteplace \div blocksize = logicalblock + offset$$

   logicalblock映射可以得到physblock

   读写不精确，需要开销大
3. 混合映射

   先用块映射，再在每一个块用扇区映射

   映射表小、避免大量擦除操作、访问时间更长
4. 日志结构映射

数据块:块映射

日志块:扇区映射

一个日志块写满后再写入数据块

总结:地址映射算法表现与负载密切相关

1. 块映射适合顺序读写
2. 扇区映射适合随机读写
3. 日志映射适合大块顺序和小块随机

垃圾回收

对一个候选块，先把有效页写到其他空闲块，在擦除初始块

开销：1.减小写入删除。2.负载均衡，每个块尽可能平均擦除

RAID和纠错码

RAID

原因

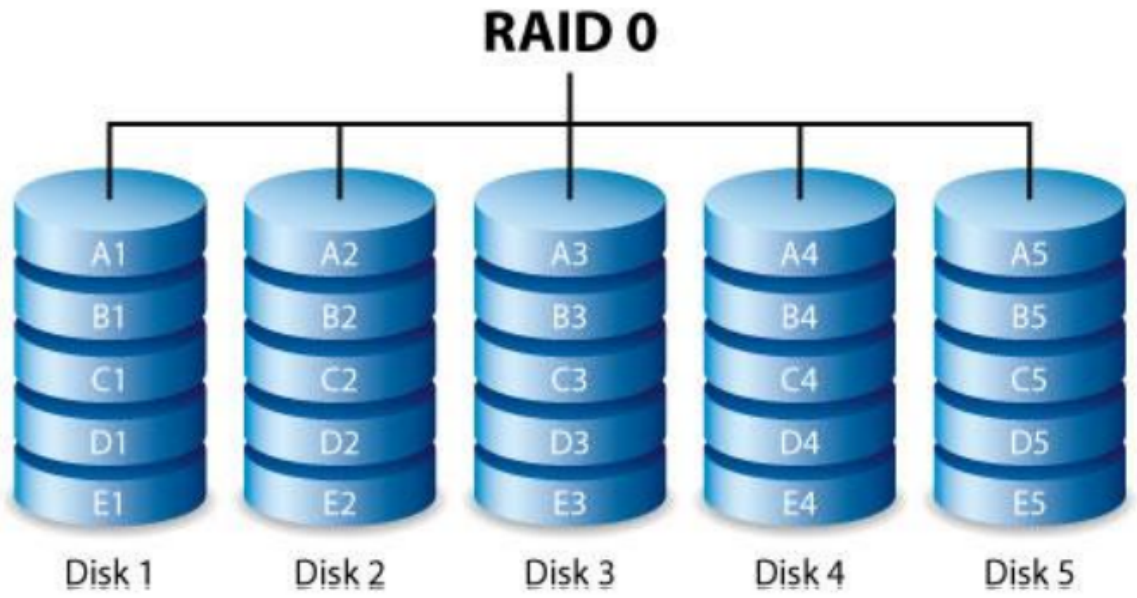
过去：把又便宜又小的硬盘混在一起做为大而昂贵的硬盘的替代

现在：表现更好、可靠性更高、容量更大

介绍

1. RAID 0

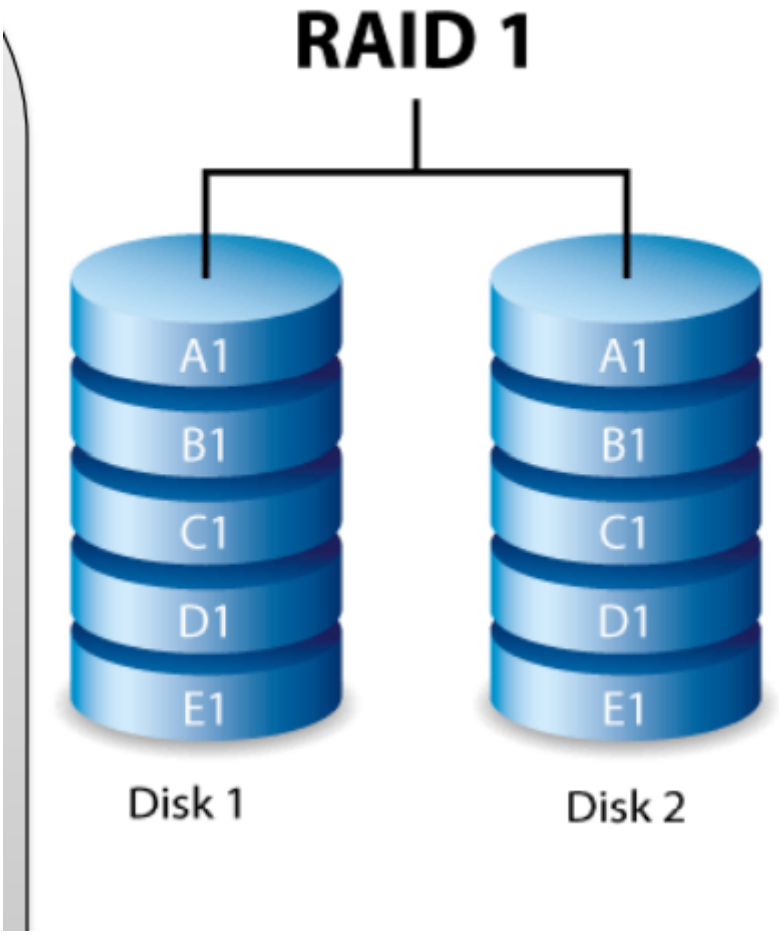
把数据分成5份



优点:速率快

缺点:无冗余、可靠性低

2. RAID 1

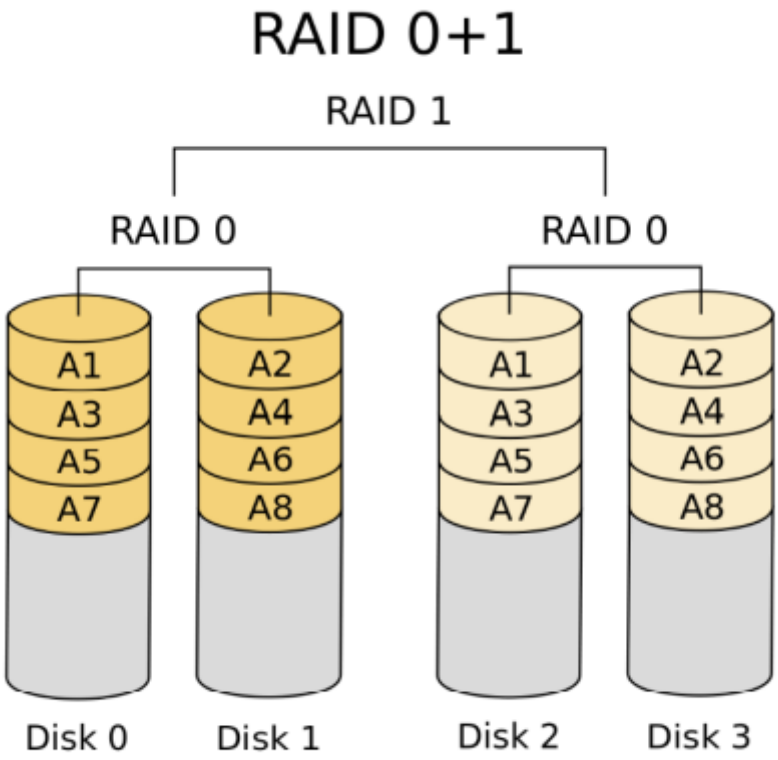


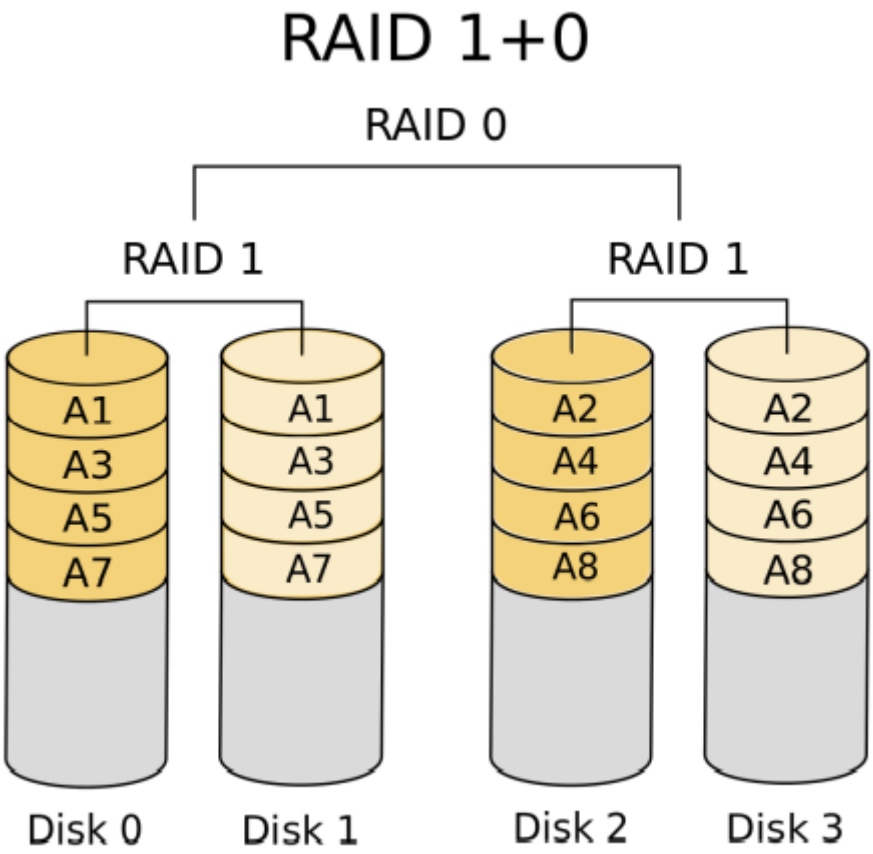
相当于是一个镜像

优点:可靠性高

缺点：储存成本太高、效率低

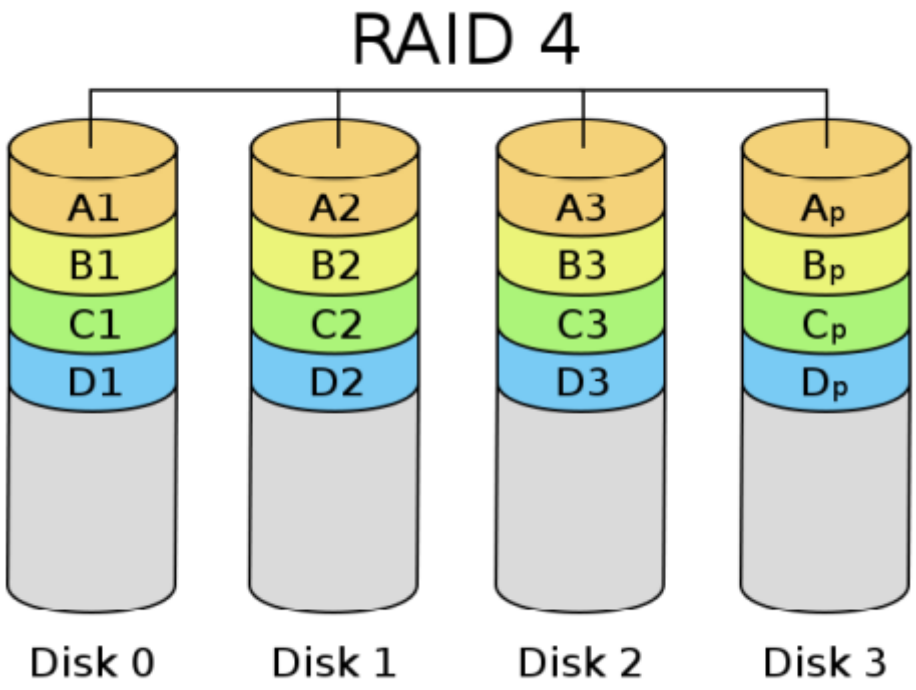
3. 混合





在前面的是后做的

4. RAID 4

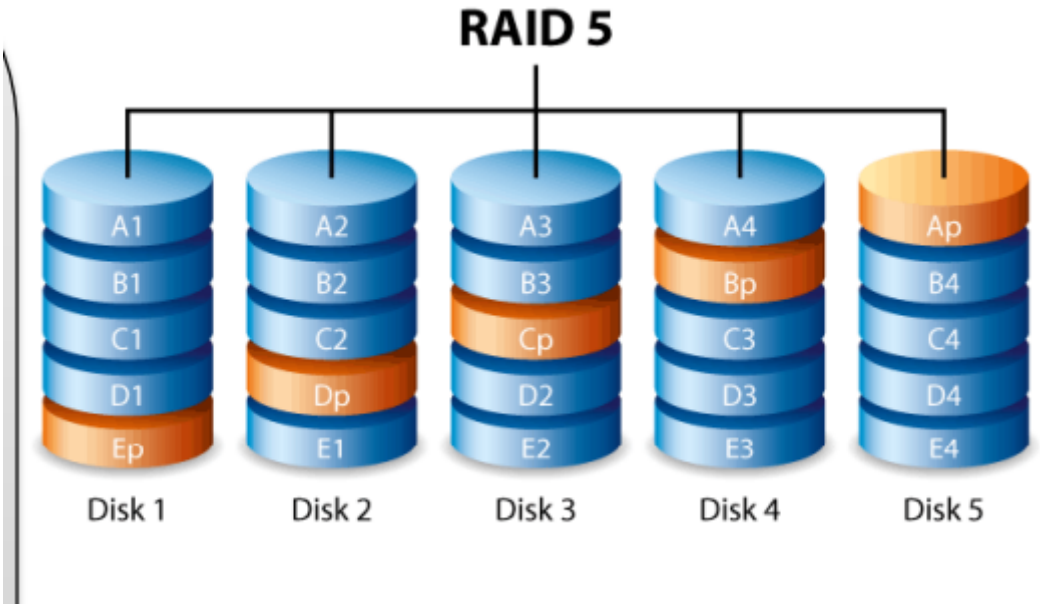


数据分三份，还有一个校验盘。

优点:兼顾了纠错和速度

缺点:校验码块被反复读写，寿命下降

5. RAID 5

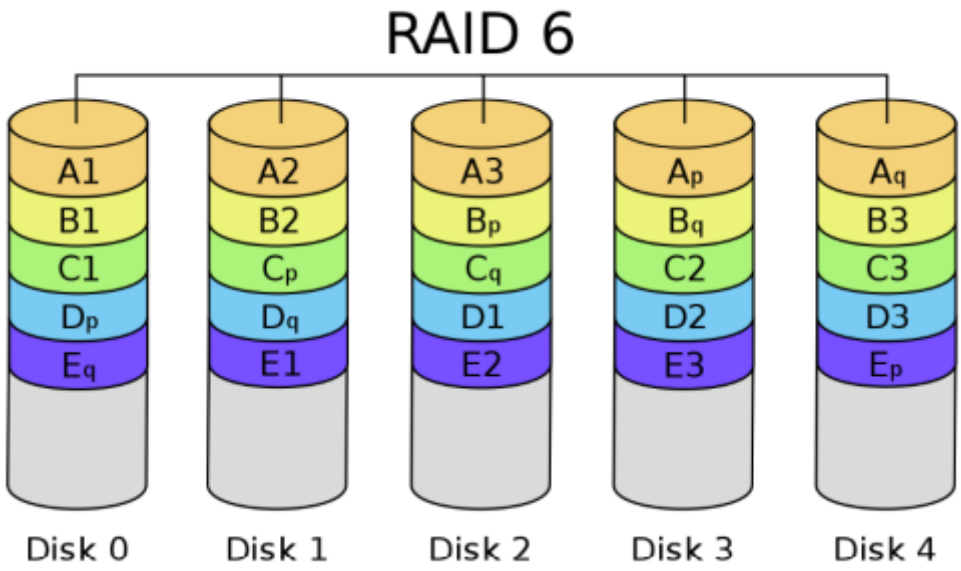


四个数据盘，还有一个校验盘

优点:兼顾了速率和纠错、还实现了负载均衡

- good performance
- good fault tolerance
- high capacity
- storage efficiency

6. RAID 6



与RAID 5相似，但多了一个校验块

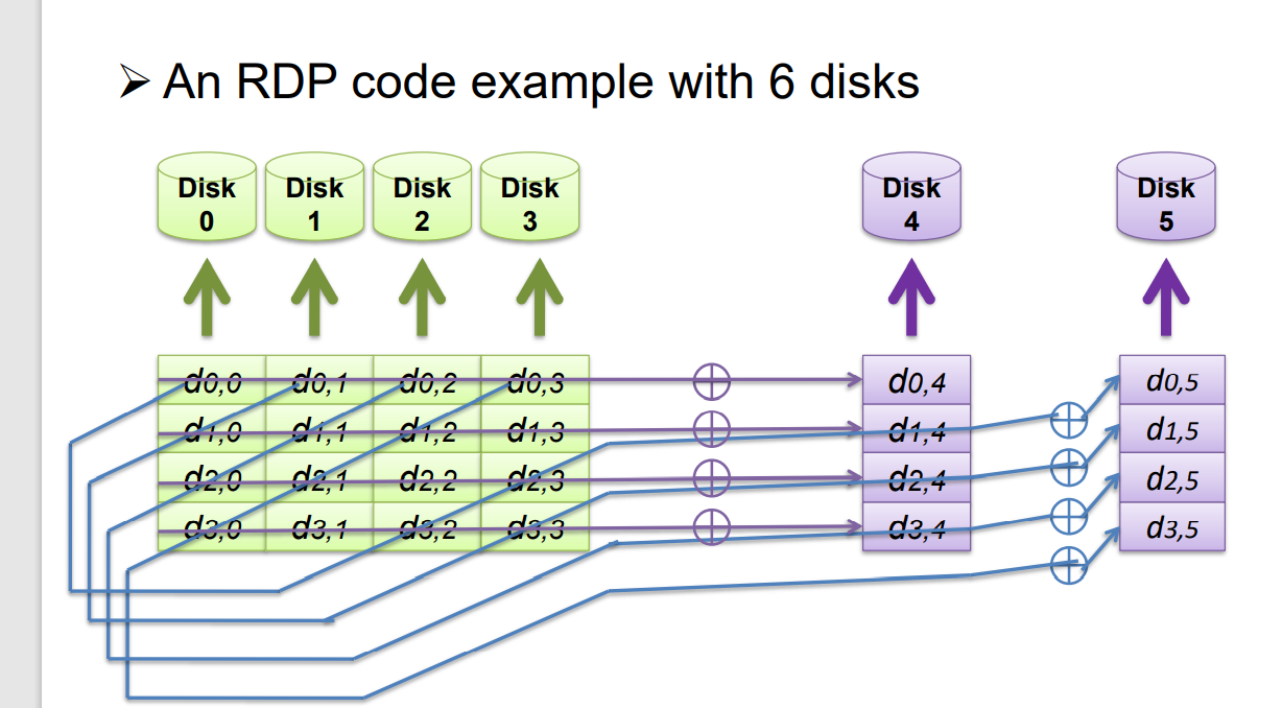
$$A_p = A_1 \bigoplus A_2 \bigoplus A_3$$

$$A_q = c^0 A_1 \bigoplus c^1 A2 \bigoplus c^3 A3$$

最多允许两块硬盘出问题

更新覆盖的代价更高

RDP code



了解即可?

纠删码EC

介绍

1. 容忍2错误:RDP，EXENODD，X-code
2. 容忍3错误:STAR
3. 【转】Reed Solomon纠删码 - fukan - 博客园 (cnblogs.com)

RAID和EC

optimizing parity updates(优化校验码更新)

首先RAID提供设备级的容错(每个ssd均有数据和校验码)

限制:校验码的更新，即更新校验码一定会带来额外的IO和垃圾回收。而对SSD来说，会影响性能和寿命

不足之处:pre-read:带来额外的读。Per-stripe basis:带来了额外的日志块、还有局部的并行

解决:EPLOG(没搞懂，貌似很新)

优点:

1. 对RAID通用
2. 寿命增加、校验块修改少
3. 性能更好

数据恢复