

# 操作系统作业五（奇数）

2021-06-21

1. 进行以下操作时，RAID-5和RAID-6分别需要访问几个块？ (10分)

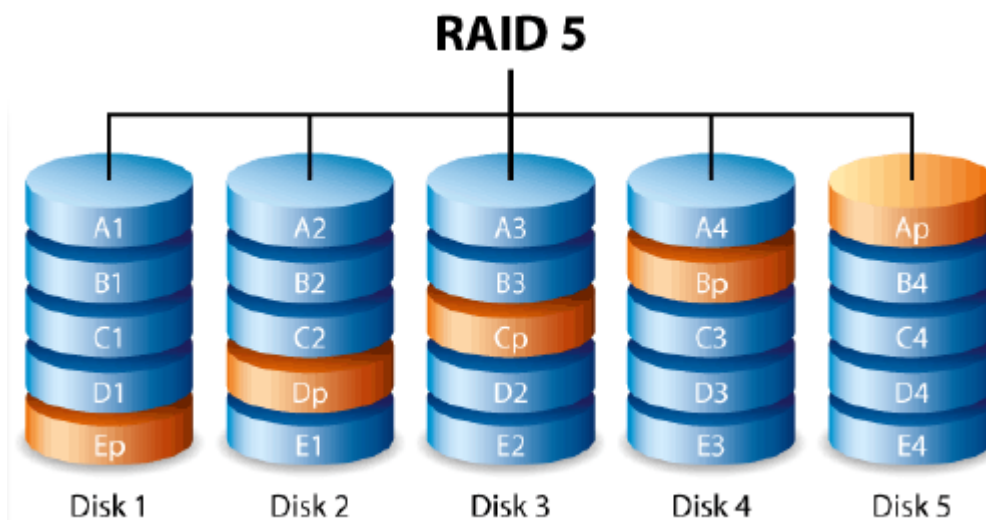
a. 更新一个块 RAID-5

• RMW:

- $A_{p'} = A_p \oplus A_1 \oplus A_1'$
- Read: A1, A<sub>p</sub>
- Write: A1', A<sub>p'</sub>
- 4 accesses to 2 blocks

• RRW:

- $A_{p'} = A_1' \oplus A_2 \oplus A_3 \oplus A_4$
- Read: A2, A3, A4
- Write: A1', A<sub>p'</sub>
- 5 accesses to 5 blocks



1. 进行以下操作时， RAID-5和RAID-6分别需要访问几个块？ (10分)

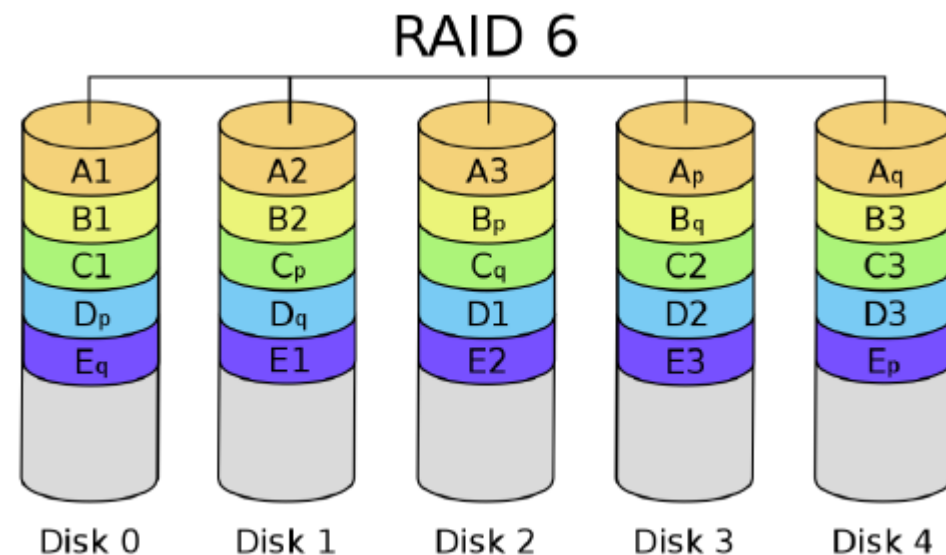
a.更新一个块 RAID-6

• RMW:

- $Ap' = Ap \oplus A1 \oplus A1'$        $Aq' = Aq \oplus c^0 A1 \oplus c^0 A1'$
- Read: A1, Ap, Aq
- Write: A1, Ap, Aq
- 6 accesses to 3 blocks

• RRW:

- $Ap' = A1' \oplus A2 \oplus A3$
- Read: A2, A3
- Write: A1', Ap', Aq'
- 5 accesses to 5 blocks



1. 进行以下操作时， RAID-5和RAID-6分别需要访问几个块？ (10分)

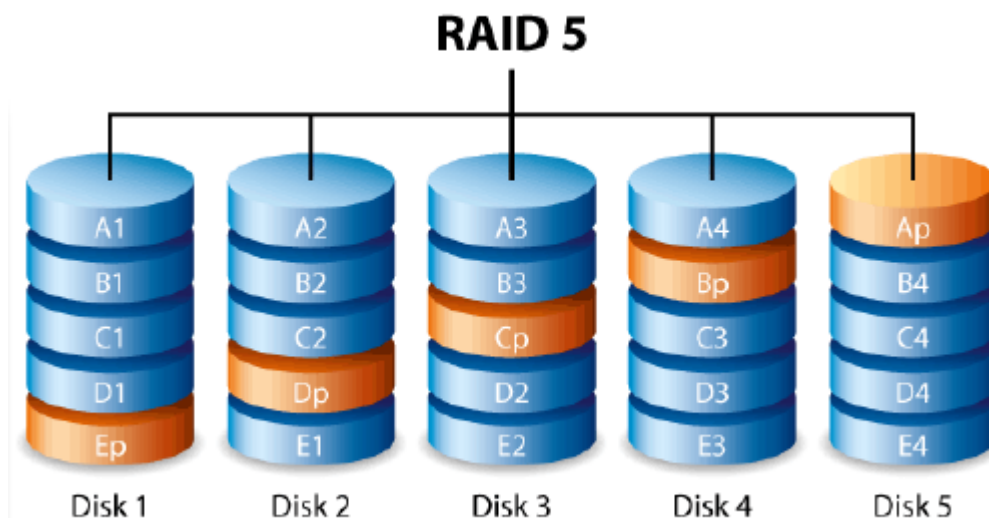
**b.更新连续7个块 RAID-5**

- RMW:

- 2 Read and 2 Write for every blocks
- 28 accesses to 9 blocks
- 14 Read-Write operations

- RRW:

- Write 7 blocks, Read B4, Write Ap' and Bp'
- 10 accesses to 10 blocks



1. 进行以下操作时， RAID-5和RAID-6分别需要访问几个块？ (10分)

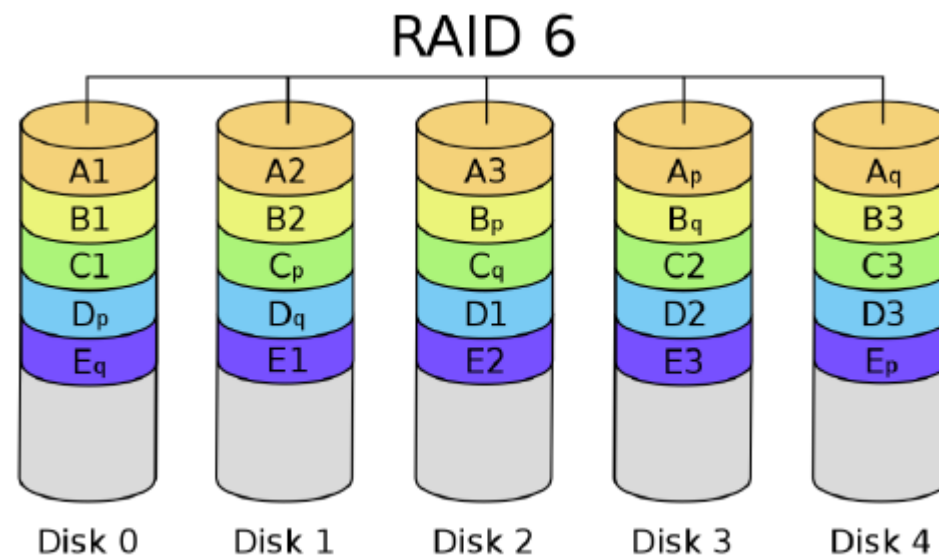
**b.更新连续7个块 RAID-6**

- RMW:

- 3 Read and 3 Write for every blocks
- 42 accesses to 9 blocks
- 21 Read-Write operations

- RRW:

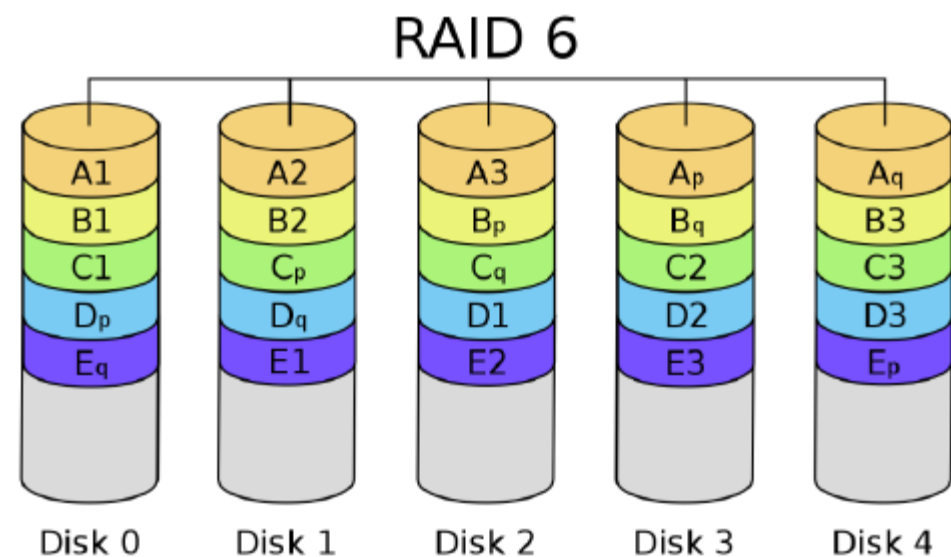
- Write 7 blocks, Read C2, C3, Write  $A_p$ ,  $A_q$ ,  $B_p$ ,  $B_q$ ,  $C_p$ ,  $C_q$
- 15 accesses to 15 blocks



1. 进行以下操作时， RAID-5和RAID-6分别需要访问几个块？ (10分)

**b.更新连续7个块 RAID-6**

- 前6个块使用RRW， C1使用RMW
- 10 accesses for A1~B3
- 6 accesses to 3 blocks for C1
- 16 accesses, 13 blocks
- 会酌情给分。
- 考试题目会尽量避免歧义。
- 答题时尽量写清楚自己的理解。



### 3.解释什么是打开文件表以及我们为什么需要它。(5分)

**打开文件表：**内核对所有打开的文件维护一个系统级别的打开文件表，存储了与一个打开文件相关的全部信息，包括文件偏移量，状态标志信息，i-node对象指针。

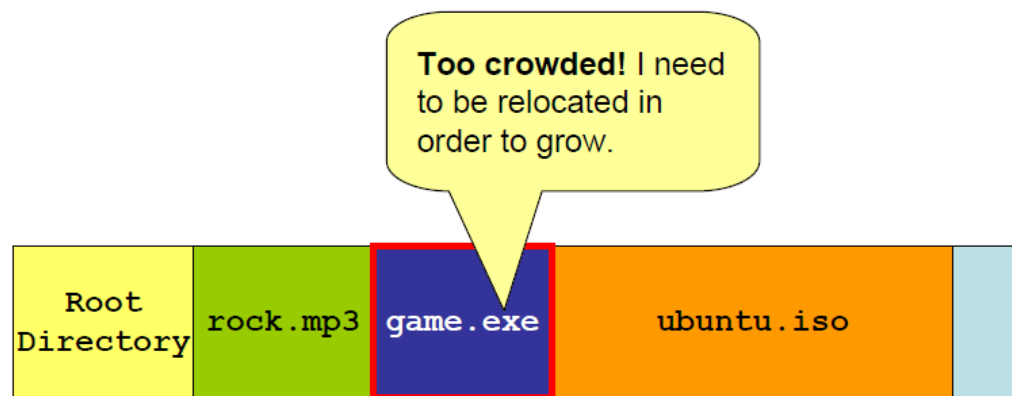
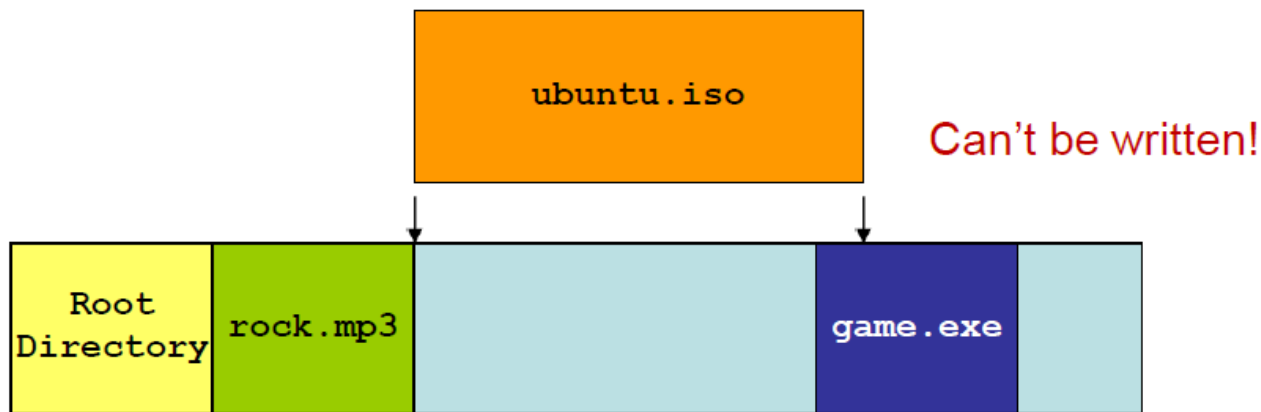
**为什么需要：**通过打开文件表，将已打开的文件信息缓存在内存中，进而通过打开文件表完成文件索引，避免每次访问需要目录遍历，减少IO开销，提高性能。另一方面，全局下的打开文件表可以维护一个文件的打开状态，确认当前是否正在被使用，以正确的进行删除等操作。

## 5.说明使用连续分配作为文件系统布局导致的问题，并给出解决方案。（5分）

### 问题：

- (1) 连续分配会产生外部碎片，导致创建大文件时没有足够的连续空间。
- (2) 当文件需要追加写入时，不能正常扩展文件大小。

**解决方法：** 将文件分块存储。





7.考虑一个类似于 UNIX 使用的索引分配的文件系统，并假设每个文件只使用一个块。以下两种情况中，读取一个小文件/a/b/c分别需要多少次IO？给出详细流程。（5分）

a. 没有缓存任何inode或者磁盘块

7次IO: (1) root directory (2) inode for /a (3) disk block for /a  
(4) inode for /a/b (5) disk block for /a/b (6) inode for /a/b/c  
(7) disk block for /a/b/c

### b. 仅缓存inode

4次IO, (2)、(4)、(6)被Cache

9. FAT32文件系统的8+3命名规则是什么，  
如何管理长文件名？ (5分)

**8+3命名规则：**

8 个字符表示文件名 +3个字符表示文件拓展名

**长文件名：**

对于长文件名，添加更多项来表示文件名，每个LFN条目用Unicode表示13个字符，即每个字符2个字节。

# 11.硬链接和符号链接的区别。 (5分)

- **硬链接：**

- (1)创建一个目录条目指向一个存在的文件
- (2)硬链接条目和原条目共同指向一个inode，并不创建新的文件内容。
- (3)对应link count+1,实际上是一个文件有两个路径

- **符号链接：**

- (1)创造了一个新的目录条目和inode，它的目录条目指向的是这个新的inode
- (2)这个inode里存储的是指向的文件路径。
- (3)对应link count没有改变，实际上可以看作是一个文件的快捷方式

# 13.数据日志和元数据日志有什么区别？ 解释两种日志方式的操作顺序 (10分)

区别： (5分)

- 流程不同：

- data journaling: 先写日志，再写文件系统
- metadata journaling: 先将元数据写日志，数据写文件系统，日志提交后再写元数据到文件系统

- 写入数据量不同：

- data journaling: 数据写了两次
- metadata journaling: 数据只写了一次

- 操作顺序：见右图 (5分)

data journaling:

TxB	Journal Contents		TxE	File System	
	(metadata)	(data)		Metadata	Data
issue	issue	issue			
complete	complete				
		complete			
			issue		
			complete		
				issue	issue
				complete	complete

metadata journaling:

TxB	Journal Contents		TxE	File System	
	(metadata)	(data)		Metadata	Data
issue	issue				issue
complete					complete
	complete				
			issue		
			complete		
				issue	
				complete	

15.列出至少三种 I/O 设备并解释如何提供标准和统一的应用程序 I/O 接口? (5分)

- I/O设备
  - 块设备(磁盘)
  - 字符设备(键盘, 鼠标, 串口)
  - 网络设备
  - .....
- 通过抽象(Abstraction), 封装(encapsulation), 分层(layering)的方法