

Homework 14

钟赞 2016K8009915009

Question 1

1) 数据日志:

如果写数据日志时发生宕机, A 相比于原来没有任何变化。

如果写数据日志后发生宕机, 操作系统可以按照数据日志进行恢复, 文件 A 的块 0 和块 1 被写入了新数据。

2) 元数据日志:

采用元数据日志时, 操作系统最开始会写数据块。

如果写数据块之前发生宕机, 宕机恢复之后 A 相比于原来没有任何变化。

如果写数据块 0 之后发生宕机, 则文件 A 的块 0 被写入了新数据。数据和元数据不一致。

如果写数据块 0, 1 之后发生宕机, 则文件 A 的块 0 和块 1 被写入了新数据。数据和元数据不一致。

如果写日志之后发生宕机, 操作系统可以按照数据日志进行恢复, 文件 A 的块 0 和块 1 被写入了新数据。

Question 2

1) 一个磁盘地址占 4B, 所以一个磁盘块可以存储 $4K/4B = 1024$ 个磁盘地址。

imap 块来存储 i-node 和磁盘地址, 所以需要: $5000000/1024 = 4883$ 个 imap 块。

2) 一个磁盘地址占 4B, 所以一个磁盘块可以存储 $4K/4B = 1024$ 个磁盘地址。

CR 记录每个 imap 块的磁盘地址, 所以需要: $4883/1024 = 5$ 个。

3) $654321/1024 = 639$ 个, $639/1024 = 1$ 个, 先找到第一个 CR 块, 得到第 639 个 imap 块的地址。

Question 3

1) foo 需要 $\frac{20MB}{4KB} = 5K$ 个 LFS 的块。foo 的索引结构为 10 个直接索引, 1 个一级间址索引, 1 个二级间址索引。

2) 他们写在磁盘上的顺序是: CR 块, 数据块, inode 块, imap 块。一次写一个 Segment, 因此操作系统会把写的东西存到缓存, 直到到达 4MB 的 Segment 时, 一次性向磁盘写入 1K 个块, 即共计 1K 次 I/O。

3) 需要写文件的数据块和 i-node 的块, 共计 2 次 I/O。

4) 首先写数据块, 然后需要写日志 (TxB、i-node 日志、bitmap 日志), 接下来提交日志 commit: 写 TxE, 最后清除日志。总共需要 4 次 I/O。

Question 4

这里没有给出 SSD 每一块大小, 我们不妨设 SSD 的一块大小为 128KB。现在以每秒发 200,000

个写请求的速率写, 平均下来每块的请求是: $\frac{20000 \times 128KB}{400GB} = 0.064$ 个/秒。

则 SSD 总共可使用的时间为: $\frac{100000}{0.064} = 1562500s$

Question 5

1) Switch Merge 需要直接把日志块转成数据块，然后把原来的数据块回收擦除。故总共需要 2ms。

2) Partial Merge 需要把数据块中有效页拷贝到日志块，然后把日志块转成数据块，最后把原来的数据块回收擦除。

因为只有 50% 的有效页，故只需要读 32 页，写 32 页，共 $(200 + 25) \times 32\mu s = 7.2ms$ 。故共需 9.2ms。

3) Full merge 首先要分配一个新的日志块，从数据块和日志块分别拷贝有效页到新日志块，然后把新日志块转成数据块，最后把原来的数据块和日志块都回收擦除。

需要读 64 页，写 64 页，即 $(200 + 25) \times 64\mu s = 14.4ms$ ，然后擦除原先日志块和数据块的时间 4ms，所以一共需要 18.4ms。