**操作系统 作业14**

**陈彦帆 2018K8009918002**

1. **设读请求与写请求的命中率相同。**

（1）**设write through结合No-write allocate。**每次读不命中需要1次IO，每次写命中或写不命中需要1次IO。读命中不需要IO。

共需 10000\*0.4+3000 = 7000 次IO。

（2）**设write back结合write allocate。设不命中时，每次替换的块都是脏块。**每次读不命中或写不命中需要2次IO。读命中或写命中不需要IO。

共需(10000+3000)\*0.4\*2 = 10400 次IO。

2. **设inode array 只有一个块。**

**设题目中用户打开文件时为open操作，只需返回inode，无须读取文件块的内容。**

（1）读根目录的i-node和它的第一块。需要读2个块。此时inode array进入文件缓存。

读home目录的i-node和它的第一块。由于inode array已经在缓存中，只需要读1个块。

读OS18目录的i-node和它的第一块，需要读1个块。

从OS18目录块读fs01.pdf的i-node，不需要访问磁盘。

共需读4个块。

（2）**设所有访问过的磁盘块都进入文件缓存，且没有发生替换。**

则不需要从磁盘读取块。

（3）在执行了（1）（2）操作的基础上：

需要访问的块已经在文件缓存中。无须访问磁盘。

（4）在执行了（1）（2）（3）操作的基础上：

inode块，根目录、home目录、OS18目录块均已在文件缓存中。

从上层目录块找到OS18目录块，从OS18目录块中找到fs03.pdf的ino，从inode块中找到fs03.pdf的inode。不需要访问磁盘。

3. （1）每个循环向文件里写4字节，循环1000次，共4000字节。

（2）写1000次磁盘，每次4字节（注：实际写的大小与写粒度有关，每次写的大小不能小于写粒度）。

此外，若tmpfile原来不存在，则创建文件还需要写inode array, inode bitmap, data block map, 目录块。

（3）设采用写回策略。总共写1次磁盘，4000字节（注：实际写的大小与写粒度有关）。

此外，若tmpfile原来不存在，则创建文件还需要写inode array, inode bitmap, data block map, 目录块。

4. 设采用先写数据，再写元数据的方法：

设fs03.pdf总共为1个块。

（1）路径解析，分配数据块，写数据块。

（若此时宕机，数据块丢失，无不一致）

（2）写inode。

（若此时宕机，inode和数据块一致。但无法通过目录找到inode）

（3）写inode bitmap和数据块bitmap

（若此时宕机，无不一致。但产生垃圾块）

（4）更新目录块

5. （1）操作为：写日志（TxB，inode日志，bitmap日志，数据块日志），提交日志（TxE），checkpoint，清除日志。

若宕机发生在提交日志之前，则恢复后文件A的内容为修改前的内容。若宕机发生在提交日志之后，则恢复后文件A的内容为修改后的内容。

（2）操作为：写数据块，写日志（TxB，inode日志，bitmap日志），提交日志（TxE），checkpoint，清除日志。

若宕机发生在提交日志之前，则恢复后文件A的内容为修改前的内容。若宕机发生在提交日志之后，则恢复后文件A的内容为修改后的内容。

6. 设1KB=1024B

（1） （块）

（2） （块）

每个CR需要5个块。若磁盘有两个CR，共需10个块。

（3）CR块号：

imap块号：

inode号：

查找过程：查找第0个CR块，找第0个CR块中的第638个imap块地址，在找到的imap块中找第1009个inode地址，即为ino=654321的磁盘地址。

7.

（1）20MB需要2级间址。

从CR可找到imap的地址，从imap可找到foo的inode地址，foo的inode下有直接索引指针，一级间址指针，二级间址指针，三级间址指针。通过直接索引、一级间址和二级间址指针可以访问foo的各个数据块。

（2）假设已知foo的ino。打开foo文件需要读CR，读imap，读inode。

第2560块需要2级间址。

需要通过2级间址指针，读一级间址块，读二级间址块。

在日志末尾写新数据块，写修改后的二级间址块，写修改后的一级间址块，写修改后的inode。写修改后的imap。（更新CR是固定周期进行，不算在内）。

以上操作有5次读操作（具体读磁盘操作数与foo在磁盘上的分散程度有关）。一共写入3个块和1个inode和对应的imap，不超过4MB（即1个segment），故只需要进行1次写。

（3）假设已知foo的ino。打开foo文件需要读inode array，得到inode。

需要通过2级间址指针，读一级间址块，读二级间址块，找到第2560块的地址。

修改原2560块对应位置的数据。

修改inode（如更新时间戳等）。

以上操作有3次读操作，2次写操作。

（4）假设已知foo的ino。打开foo文件需要读inode array，得到inode。

需要通过2级间址指针，读一级间址块，读二级间址块，找到第2560块的地址。

修改原2560块对应位置的数据。

写日志（TxB，inode日志）和提交日志（TxE），checkpoint（更新inode），清除日志。

以上操作有3次读操作，4次写操作。