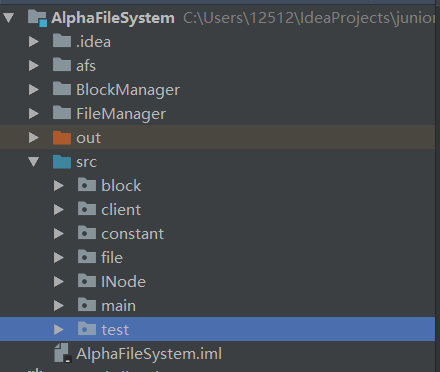
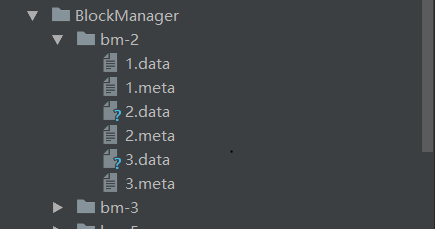
**AlphaFileSystem**

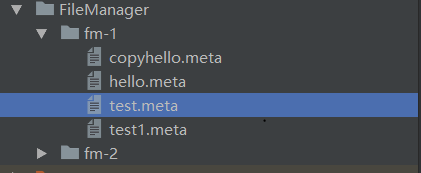
**系统结构**如下:



afs文件夹下存放的是配置文件，系统FileManager，BlockManager的数量（100个）与状态。系统启动时需要读取上一次系统的状态，新系统则进行初始化，我们也可以通过调用system\_init()函数初始化系统，删除所有文件。



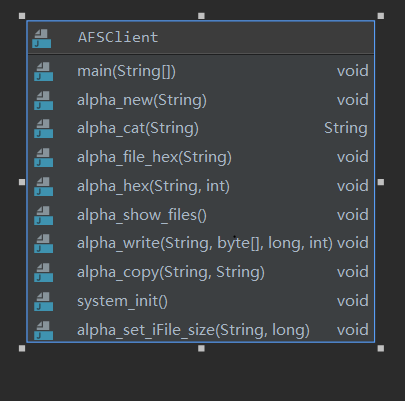
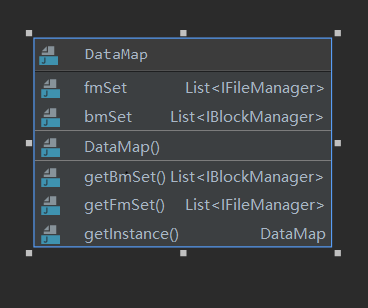
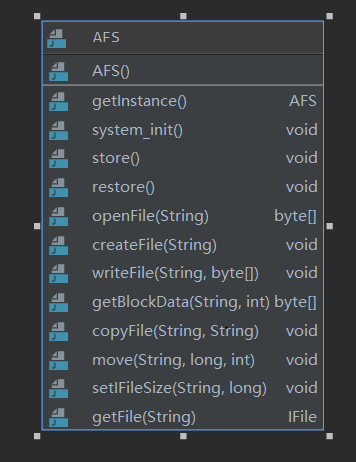
BlockManager文件夹的结构如上图，存放了block的data信息和meta信息。



FileManager文件夹下存放了File的meta信息，每个FileManager只能管理四个文件。

**具体实现：**

①**AFSClient，DataMap和AFS**

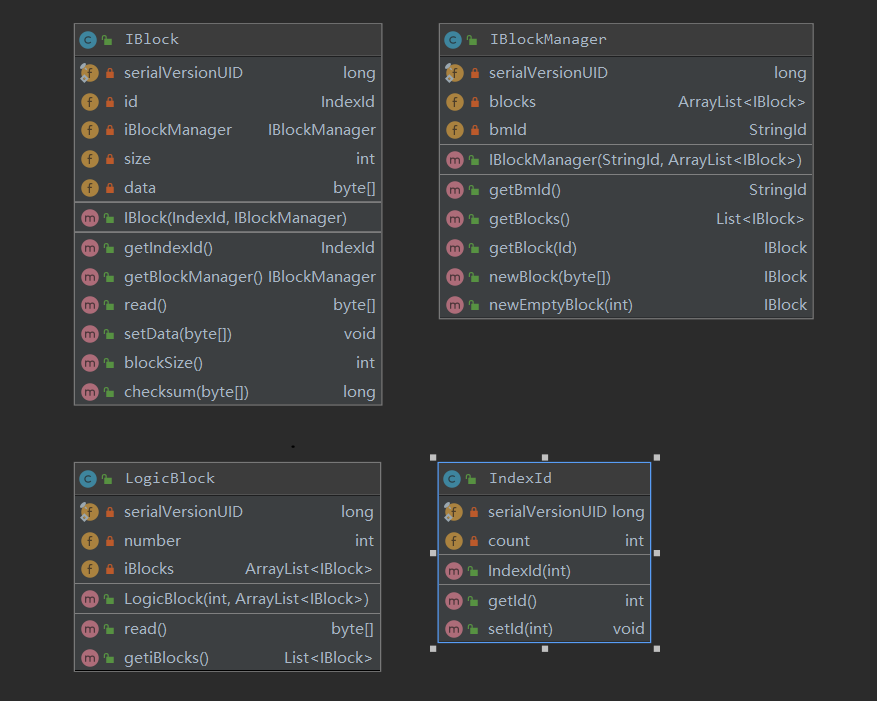
用户通过AFSCient发出操作命令，如cat，hex，copy等，调用AFS对应的方法实现文件的读、写、储存。

DataMap是一个数据储存的类，储存当前所有FileManager和BlockManager的状态，在系统退出时将这些状态写回文件中。

AFS系统启动时，调用restore()指令读取上一次系统的状态，读取到的信息保存在DataMap里，启动后可执行对应的指令。

系统退出时，将DataMap中的储存的信息写回到文件里。

②**IBlock，IBlockManager，LogicBlock，IndexId**



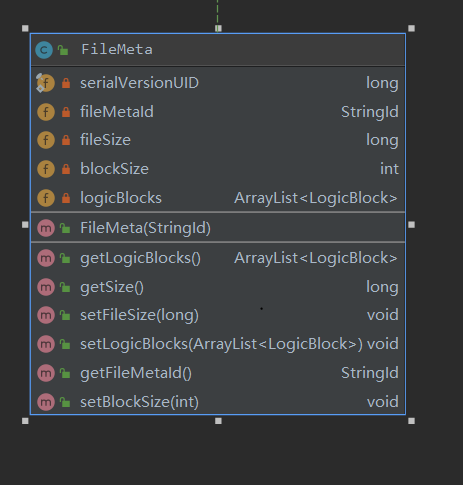
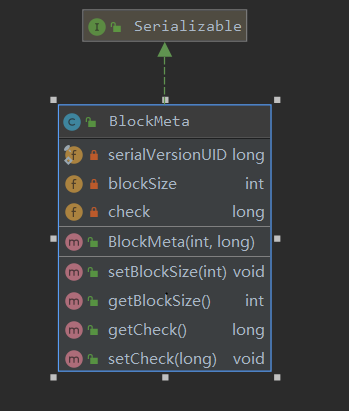
IBlock实现了Block接口，每个Block的大小是512B，data里记录了Block存放的内容。read()方法从data文件中读取数据到data中;

LogicBlock实现了duplication，number是序列号，iBlocks是Block的集合，集合里的Block存放的内容都是相同的；

IndexId实现了Id接口，唯一地标识了BlockManager里的Block;

IBlockManager实现了BlockManager接口，blocks存放了每个IBlockManager管理的Block的集合。newBlock（byte[] b）方法，将b写入data文件中，生成校验码，并在meta文件写入相应信息；getBlock（）方法，返回IBlockManager管理的Block的集合；

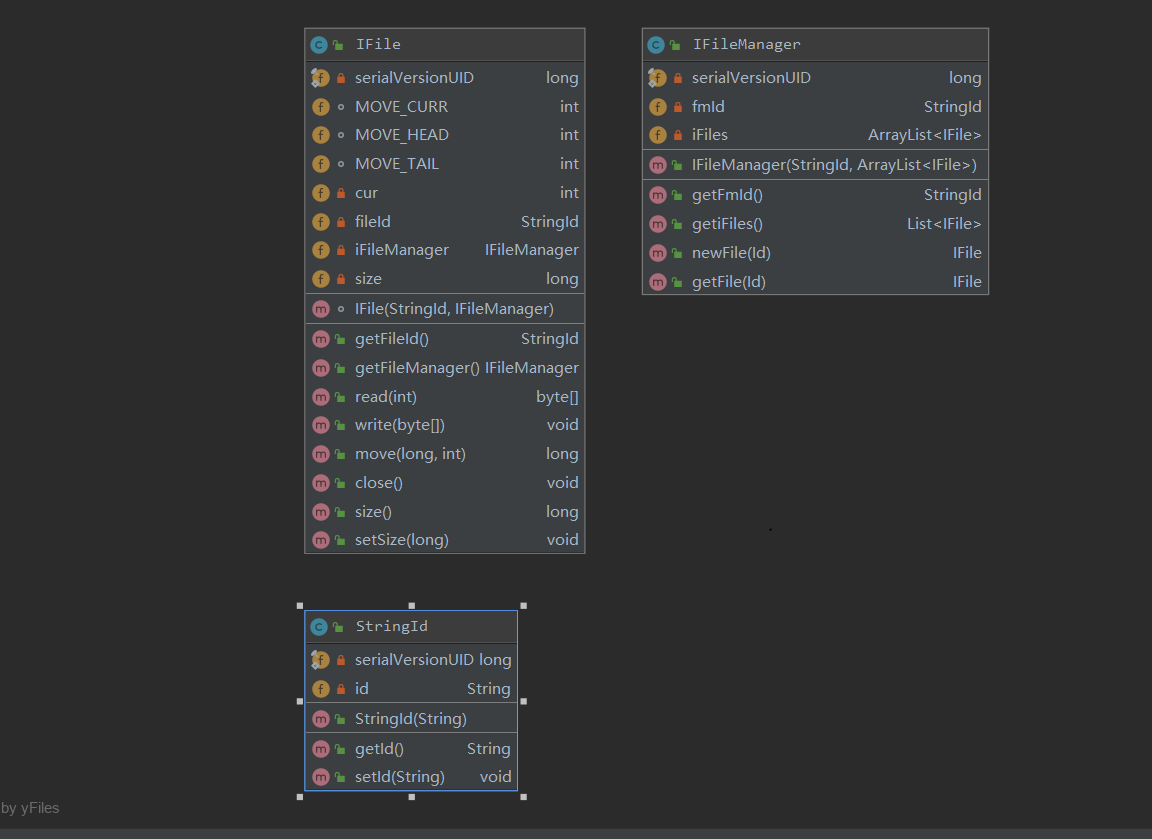
③**FileMeta，BlockMeta**

FileMeta记录File的meta信息，包括File的大小，Block的大小和File存放的LogicBlock的集合。

BlockMeta记录Block的meta信息，包括Block的大小和校验码。

④**IFile，IFileManager，StringId**



StringId标识了File，FileManager，BlockManager，FileMeta和BlockMeta；

IFile实现了File接口，cur存放当前游标的位置，size存放IFile的大小；

1. read(int length)的实现，读取对应的FileMeta的信息，得到File存放的LogicBlock的集合，对于每个LogicBlock，随机地读取其中的一个副本。组合的到对应的length长度的byte即可。
2. write(byte[] b)的实现，从cur位置开始写：

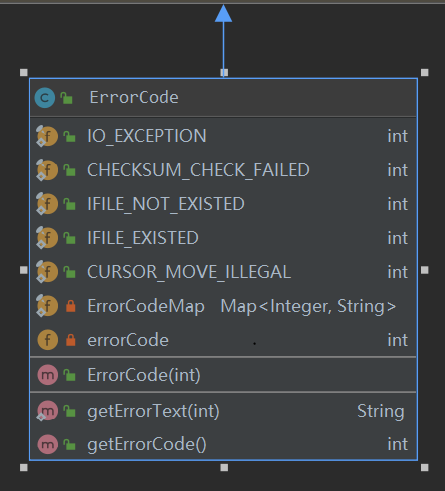
Ⅰ. cur/blocksize取到当前LogicBlock，对于当前块之前的块保留，移除当前块之后的块；读取当前块cur前面的文件的内容，和要写的内容拼接到一起，为其分配Block；

Ⅱ. 将对应的信息写入该文件的meta文件中。

1. move(long,int)的实现，根据参数改变cur的位置即可，cur小于0或者超出File的size都是违规操作。
2. setSize(long)的实现，如果传入的新size大于原文件的大小，在后面补0，小于原文件的大小，改掉size即可，对于大于文件大小部分的内容是不可访问的。

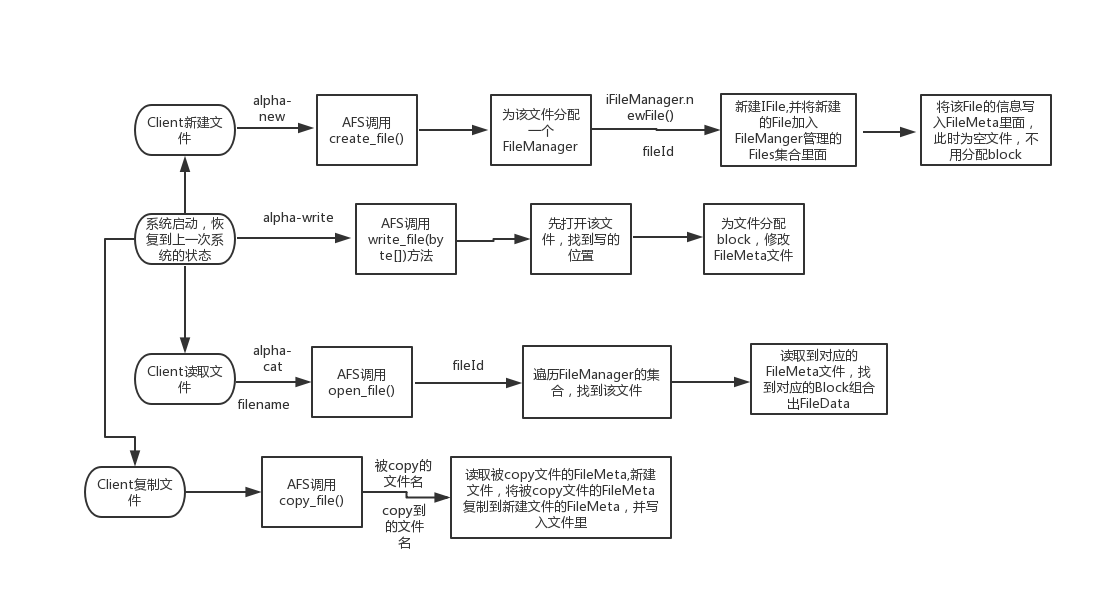
IFileManager实现了FileManager接口，iFiles是IFileManager管理的File的集合；getFile()方法，遍历IFiles找到对应的IFile即可；newFile()创建一个空文件，不用分配block，将对应信息写入FileMeta里，并将创建的文件添加到IFileManager管理的文件集合里。

⑤**ErrorCode**



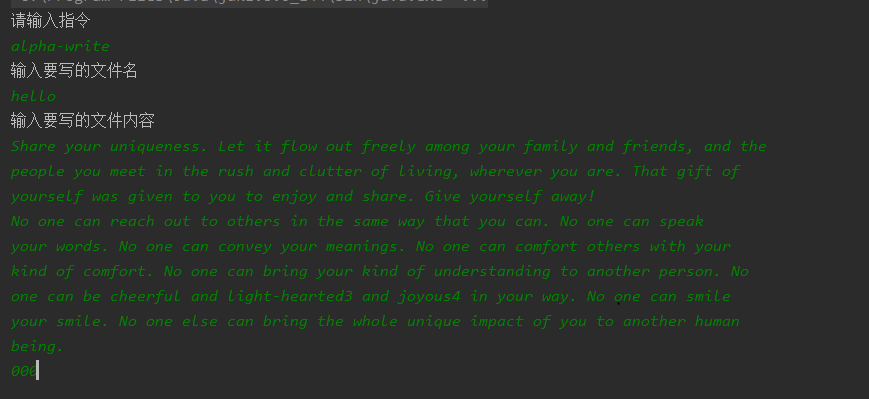
ErrorCode主要处理IO的异常，读取时文件校验码失败的异常，读取时文件不存在的异常，创建文件时文件已存在的异常，移动光标非法的异常，在遇到相应问题是抛出异常即可。

**流程图**

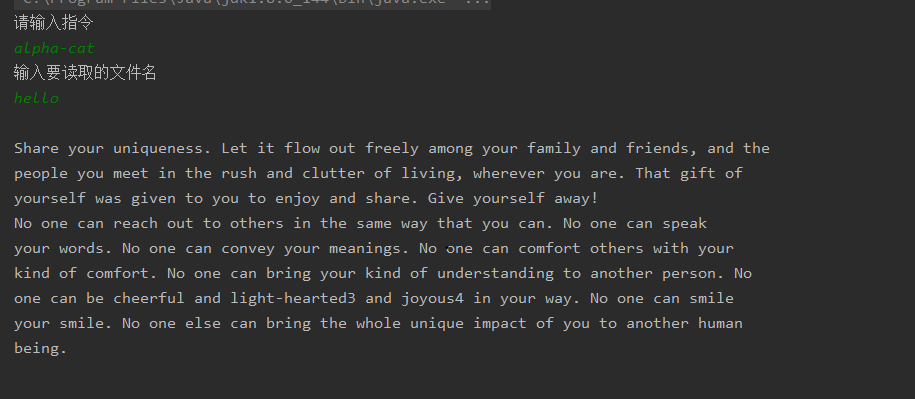


**运行示例：**

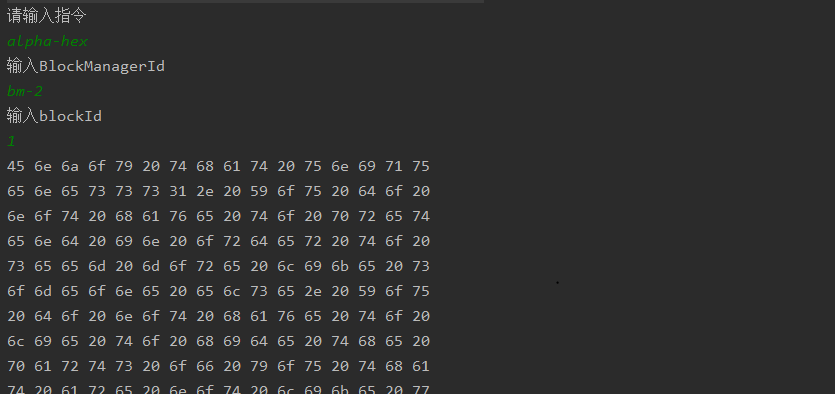
alpha-write : 新建文件hello，并写入一串字符，输入000结束输入，alpha-cat查询发现成功写入。



alpha-cat：读取已有的文件hello。



alpha-hex:打开Id为bm-2的BlockManager下的第一个Block (即1.data)，一个Block有512/16=32行。



alpha-copy : 将文件hello里的内容copy到copyhello文件里。并查询copyhello里的文件内容。

