网络本身的异常处理

*NO\_AVAILABLE\_BLOCK：在服务器相关的block manager未启动的情况下抛出的异常。*

*NOT\_FOUND\_FILEMANAGER：在服务器相关的file manager未启动的情况下抛出的异常。*

*REMOTE\_CONNECT\_ERROR：远程服务器连接失败，可能是由于服务器未启动。*

容错机制

①客户端

T1：对用户输入格式进行校验，例如游标位置为数值，长度为数值，否则提醒用户重新输入或者重试。

T2：客户端提示相关BM或者FM未启动，提醒稍后重试，而不抛出错误。

T3：拷贝操作要求客户首先创建被拷贝文件，否则会提醒用户创建重试。新创建的文件尚无内容，不能被读取，此时会报用户输入错误，提醒用户重试。

T4：对弈一个BM下不存在的block，提醒用户其不存在或者无内容，请检查后重试。

②服务端

T1：可以随时关闭或者启动一个FM或者BM。

T2：如果启动或者关闭指令存在错误，会提醒服务器检查并重试，而不会因此宕机。

T3：服务器端检查备份，检查到如果备份中一方下线，则使用另一方。如果两者均存在问题，提醒客户端稍后再试。

T4：写入FileMeta存盘失败，会根据缓存回滚，不会造成不一致。

异常处理汇总

我将NotBoundExcepption和RemoteException等一系列新引入的异常与Lab1中异常进行依据情况分类，汇总如下：

***ErrorCodeMap***.put(***IO\_EXCEPTION***, **"IO exception"**);  
***ErrorCodeMap***.put(***CHECKSUM\_CHECK\_FAILED***, **"block checksum check failed"**);  
***ErrorCodeMap***.put(***BUILD\_DIRECTORY\_OF\_BM\_FAILED***,**"build directory of block manager failed"**);  
***ErrorCodeMap***.put(***BUILD\_FILE\_OF\_BLOCK\_FAILED***,**"build file for block failed"**);  
***ErrorCodeMap***.put(***BUILD\_FILE\_OF\_FILEMETA\_FAILED***,**"build file for file meta failed"**);  
***ErrorCodeMap***.put(***GET\_CONTENTS\_FAILED***,**"get contents failed"**);  
***ErrorCodeMap***.put(***WRITE\_CONTENTS\_FAILED***,**"write contents failed"**);  
***ErrorCodeMap***.put(***WRITE\_FILEMETA\_FAILED***,**"write file meta failed"**);  
***ErrorCodeMap***.put(***ALPHA\_CAT\_READ\_ERROR***,**"alpha-cat read error"**);  
***ErrorCodeMap***.put(***ALPHA\_COPY\_READ\_ERROR***,**"alpha-copy read error"**);  
***ErrorCodeMap***.put(***INVALID\_SIZE***,**"error size"**);  
***ErrorCodeMap***.put(***NOT\_FOUND\_FILEMANAGER***,**"can not found file manager"**);  
***ErrorCodeMap***.put(***NO\_AVAILABLE\_BLOCK***,**"可能未启动相应block manager，请重试。"**);  
***ErrorCodeMap***.put(***FILE\_BROKEN***,**"文件被违规损坏，请避免对该文件操作。"**);  
***ErrorCodeMap***.put(***INPUT\_ERROR***,**"输入存在错误，请检查。"**);  
***ErrorCodeMap***.put(***REMOTE\_CONNECT\_ERROR***,**"远程连接错误，请检查。"**);  
***ErrorCodeMap***.put(***UNKNOWN***, **"unknown"**);