June 10th, 2013

1. FileIO: 如果目前没找到而且还有nextfile，那么就继续找因为初级目录和次级目录的名字都是.和..。那么前两个目录先比较，肯定不是。如果是directory，不对；如果是file，比较只是文件名一样不，如果一样，将ASearchPath抄写进入AFoundFile，然后将路径和名字合并，并且返回true。然后将路径和名字合并，并且返回true。
2. JumpTable{

Int listOffset

Int listSize

}

LECJobFileHeader{

Unsigned short magicNumber

Unsigned short numberObjects

Int tableOffset}

LECObjectHeader{

Char type[200]

Unsigned short numProperties

Int tableOffset}

LECPropertyHeader{

Int size

Char structName[50]

Char propertyName[50]

Int dataOffset

Unsigned short arrayRank}

1. InteropEnums.cs是所有error code以及返回值的举例。
2. JobFunctions：
3. NewJob: first check if the job is already existed. If it is, delete the existed one and add into the new one. And add the new one to the active job list and send the reference number back to the caller.
4. LoadJobFromFile: 和NewJob一样只是多了获取路径这个以及要找的仅是PC Job传达了这个参数。
5. AbortJob在LECJobManager.cpp中，先AbortMotion，但是是在哪里abort的呢？
6. Snapshot是用来通过存储一串连续的job（在当下这个job的附近的那些按时间顺序发生的序列存在一个stack中，每个job里面也都有这样的一个序列），实现redo以及undo。其中redo就是把当前的指针转移到当前的job的下一个JOBRECORD的数据类型中，而undo就是把当前的指针转移到当前的job。的上一个JOBRECORD 的数据类型中。但是为什么在snapshot函数中需要先erase那些？是不是再加入新的snapshot（也就是action时当前snapshot之后的就都disappear、erase了？）
7. dqSnapshotList: 是一个stack of jobRecord就是用来进行上面的操怍的。
8. GetLECPath用来干嘛？
9. JOBRECORD{

Unsigned char\* bytes

Int size

UndoRedoState action

}

Bytes里存放所有的内容，size代表内容的大小，代表因为在check how many的时候，这个加的是总的memories的大小。

1. GertBytes： 根据obt\_Snapshot的不同将这个current job变成下一种形式。
2. Undo只是把指针后退一个