

cp_yang.py思考

一、脚本中几个核心且耗时的操作

1. 递归记录那几个目录下所有文件的文件名和文件路径。
2. 在csv表格中查找当前PRODUCT的所依赖的module的feature。
3. 递归构建所需的yang文件依赖树。
4. 拷贝所需的所有yang文件到指定目录。

思考：

- 如果每次编译的都是5720hi，以上4步在每次编译中都会依次执行一遍。
- 因为每次这4步执行的结果都是相同的，所以很有必要在第一次运行后保存中间结果，以后编译直接读取，从而提升编译效率

二、具体优化方法

前三步的优化：

- 前三步的执行结果都是一个或多个字典变量的形式。我们可以将其序列化成json对象或者pickle对象从而存储起来。在之后编译的时候直接加载即可。

第四步的优化：

1. 对于大量文件的拷贝也是比较耗时的，我们可以预先对某种型号完整执行一遍，编译结束不删除yang文件，然后将其需要的yang文件存储在"5720hi_yang"目录中,当下一次编译的时候，直接将5720hi_yang文件夹重命名为规定的名称，或者直接在5720hi_yang目录下编译生成yin文件，即可节省大量文件拷贝的时间，编译全部结束的时候将yin文件删除即可。
2. 对于每种型号都按照上述操作预先存储yang文件到特定的文件夹下。
3. 如果采取这种优化方法，前三步的优化可以忽略，因为前三步的结果就是找到这些yang文件。

其他：

1. 如果某个型号每次编译生成的yin文件都相同的话也可以保存yin文件，以节省编译生成yin文件的时间。