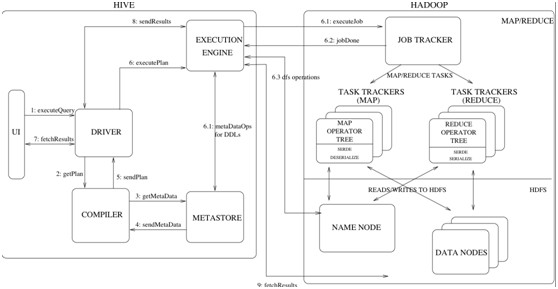
**Hive源码分析(二)-hive执行过程**

2012年09月07日 20:15:08



1.入口  /bin/cli.sh

      调用CliDriver类进行初始化过程

Ø   处理 -e， -f，-h等信息，如果是-h，打印提示信息，并退出

Ø   读取hive的配置文件，设置HiveConf

Ø   创建一个控制台，进入交互模式

2.在交互方式下，读取每一个输命令行，直到’;’为止，然后提交给processLine(cmd)方法处理，该方法将输入的流以;分割成多个命令 ，然后交给processCmd(cmd)方法 。

3.ProcessCmd(cmd) 对输入的命令行进行判断，根据命令的第一个记号(Token),分别进入相应的流程

Ø  quit or exit 系统正常退出

Ø ！开头的命令行,执行操作系统命令

Ø source 开头的，读取外部文件，并执行文件中的命令

Ø list 列出 jar  file archive

其他命令提交给Commandprocess，进行命令的预处理

4.命令的预处理  CommandProcess

   根据输入命令的第一个记号，分别进行处理

Ø set ： 调用SetProcess类，设置hive的环境参数，并保存在该进程的HiveConf中

Ø dfs：   调用DfsProcess类，调用hadoop的shell接口，执行hadoop的相关命令

Ø add：  调用AddResourceProcessor  ，导入外部的资源，只对该进程有效

Ø delete: 与add对应，删除资源

其他  ：提交给Driver类，进行下一步的处理

5.  命令的处理( Driver类的run方法）

(1)  编译 complie（Hive的核心部分）

Ø  通过语言识别工具Antlr，验证语句的合法性

Ø  将sql转换成一个抽象语法树(AST)

Ø  定义树解释器Operator，将AST翻译成逻辑操作树

Ø  调用genMapRed方法，生成物理执行计划

(2)获取读写锁

Ø  对操作的表获取一个读写锁acquireReadWriteLocks

(3)执行 execute

Ø  将生成的Task提交hadoopAPI 处理

Ø  返回任务的执行时间和状态(成功or失败)

6.获取执行的结果

Ø 任务执行失败，抛出异常

Ø  执行成功后，调用Driver的GetReuslt方法，顺序打开每一个输出文件

Ø  获取每一行的输出，并打印到控制台

7.执行清理

Ø  清理hive执行过程中的中间文件和临时文件

Ø  退出该条命令的执行，返回控制台并等待下一条命令的输入

其中Driver类是hive最核心的类。

