

MR Streaming

简介

支持用可执行程序或是脚本做Map-Reduce模型的数据处理，是Hadoop Streaming在ODPS上的对应方式。

目的是支持以前的hadoop streaming作业的迁移。

对于新写的任务可以的话还是推荐用Java UDTF或Python UDTF的方式实现，跟SQL和表结合得更好，更灵活。

MR Streaming

简介

预备知识

数据全部都是表

只做纯粹的数据处理

与Hadoop Streaming的不同

Get Started

命令行参数对比

jobconf配置项对比

运行时环境

预备知识

ODPS是多编程模型的离线数据处理平台，已经支持SQL，MR，GRAPH等数据处理方式。

ODPS的MR/SQL的功能和用户接口跟Hadoop/Hive非常类似。

最大的区别在于存储格式 和 运行时安全限制。

数据全部都是表

ODPS的数据都以二维关系表的形式存储和管理，表的schema类似Hive表，但后面的物理存储是结构化的列式存储。

这意味着数据都是一条条的记录 (Record)，每条记录都是一个个的基本类型的字段。

没有文本行，没有列分隔符，没有null indicator。

没有压缩格式。 (默认就是压缩的，按列压缩)

没有textfile, sequencefile等文件格式。

没有文件路径，目录，fs命令。

MR的输入输出是表记录，不是KeyValue。

只做纯粹的数据处理

有严格的运行时安全限制，mr代码有Java沙箱，udf代码有Python和Java沙箱，整个还有进程沙箱。

几乎一切纯计算以外的动作都是被禁止的，不能读写本地文件，不能访问网络。

与Hadoop Streaming的不同

核心的影响来自面向文件和文本行 与 面向表和字段。

streaming的脚本和Map-Reduce框架间是基于文本行和字段分隔符交互的。

对于mapper，会把表的一条记录的所有字段用 `stream.map.input.field.separator` 拼成一行文本，交给mapper脚本处理。把脚本输出按 `stream.map.output.field.separator` 解析，并按 `stream.num.map.output.key.fields` 指定的列数作为shuffle-sort的key。

对于reducer，会把shuffle-sort后的key-value按 `stream.reduce.input.field.separator` 拼成一行文本，交给reducer脚本处理。把脚本输出按 `stream.reduce.output.field.separator` 解析，生成为输出表的一条记录。脚本输出的字段个数需要跟输出表的定义一致。

输出表的字段是有类型的，而streaming生成的字段都是字符串，如果对应的输出表字段不是string类型，会发生**类型转换**。转换规则：

- string字段的结果直接是streaming输出的对应字段。（空字符串仍然是空字符串）
- 非string字段，如果streaming输出是空字符串，表字段将是NULL。
- bigint, double, boolean，会做相应的类型转换，如果转换失败（脏数据），任务将失败。 **FIXME: 异常处理细则**
- 目前不支持其它的类型，比如datetime, decimal。

Get Started

注意：线上目前只在search_offline_dev project启用了该功能。

需要用新版本的console才能提交streaming作业，从[这里下载](#)。

streaming命令格式：

```
jar [generic options] com.aliyun.odps.mapred.bridge.streaming.StreamJob [streaming options]
```

例如：

```
jar com.aliyun.odps.mapred.bridge.streaming.StreamJob
  -input /lyh_meta1/partsrc/pt=pt1
  -output res
  -mapper "python words.py"
  -reducer "python mywc.py"
  -file words.py
  -file mywc.py
```

上面例子用到的脚本：words.py mywc.py

注意：

目前odps的console(odpscmd)是hive风格的，直接odpscmd是进入交互模式的，可以直接输入多行命令，比如jar命令或者sql语句。

这跟hadoop的命令行有不同，hadoop是使用 hadoop jar ... 这样的单行shell命令风格的。odpscmd需要用 odpscmd -e "jar ..." 这样。

已知问题：目前jar命令对参数里的单引号处理有问题， odpscmd -e "jar ..." 里面用到引号时需要用 \"。

命令行参数对比

streaming参数：

参数	hadoop格式	odps格式	说明
-input	hdfs path	[/<project>/]<table>[/pt1=a/pt2=b]	可以有多个-input，指定多个表或分区
input columns	N/A	TODO	这个选项需要跟-input指定的表对应（如果-input有多个表）
-output	hdfs path	[/<project>/]<table>[/pt1=a/pt2=b]	必须指定完整的分区
-mapper	命令	相同	
-reducer	命令	相同	
-jobconf	name=value	相同	
-D	指定jobconf项	指定System Property	语义不同，odps streaming需要用-jobconf
-file	本地路径	相同	目前只支持了指定单个文件
-cacheFile	hdfs://xx/xx#xx	N/A	改用add file 和 -resources
-cacheArchive	hdfs://xx/xx#xx	N/A	改用add archive 和 -resources
-partitioner	org.apache.hadoop.mapred.lib.KeyFieldBasedPartitioner	com.aliyun.odps.mapred.lib.KeyFieldBasedPartitioner	
-combiner	与-reducer相同	N/A	不支持
-comparer	java类	N/A	不支持
-cmdenv		相同	

generic参数：

参数	hadoop格式	odps格式	说明
-resources	N/A	resource_name1,resource_name2	先用add file, add archive 上传，然后直接引用名字（没有文件路径） TODO: 资源别名
-classpath			如果有参数用到了自定义java class，才需要这个

jobconf配置项对比

功能	hadoop 配置项	odps配置项
mapper脚本输入分隔符	stream.map.input.field.separator	相同
mapper脚本输出分隔符	stream.map.output.field.separator	相同
mapper输出key字段个数	stream.num.map.output.key.fields	相同 (对于map-only job则无意义)
partition字段个数	map.output.key.field.separator, num.key.fields.for.partition	相同
reducer脚本输入分隔符	stream.reduce.input.field.separator	相同
reducer脚本输出分隔符	stream.reduce.output.field.separator	相同
reducer输出key字段个数	stream.num.reduce.output.key.fields	无意义
超时时间	mapred.task.timeout	
map input split		odps.stage.mapper.split.size
reduce数	mapred.reduce.tasks	odps.stage.reducer.num
map内存		odps.stage.mapper.mem
reduce内存		odps.stage.reducer.mem
job名字	mapred.job.name	N/A

特别说明：

- 为了支持给不同的input表指定不同的分隔符，`-jobconf stream.map.input.field.separator=<xx>` 和 `-input <tbl partition>` 之间的顺序是有影响的：一个`-jobconf` 指定的输入分隔符会影响它后面的所有`-input`，直到再次用`-jobconf` 改变了分隔符。例如：

```
jar ...
-input tbl_1/ds=20140101
-input tbl_1/ds=20140102
-jobconf stream.map.input.field.separator=\001
-input tbl_2/ds=20140101
-input tbl_2/ds=20140102
-jobconf stream.map.input.field.separator=\002
-input tbl_3
...
```

这里tbl_1的两个分区将使用默认输入分隔符\t, tbl_2的两个分区将使用输入分隔符\001, tbl_3将使用分隔符\002。

- 分隔符可以用\NNN这种八进制转义序列指定，比如\001。也可以直接用字符，比如, :, \t。引号" 的处理有问题，可以用\042代替。也可以用多个字符的字符串做分隔符。

运行时环境

功能	hadoop	odps
环境变量map_input_file		/<project name>/<table name>[/ds=20140101/pt=pt1]
Counter	reporter:counter:g,n,v	相同

站长统计

Last modified on 07/10/2014 12:59:40 PM