Hive count(distinct)有几个reduce，海量数据会有什么问题

问过的一些公司：字节(2021.07)

参考答案：

count(distinct)只有1个reduce。

为什么只有一个reducer呢，因为使用了distinct和count(full aggreates)，这两个函数产生的mr作业只会产生一个reducer，而且哪怕显式指定set mapred.reduce.tasks=100000也是没用的。

当使用count(distinct)处理海量数据（比如达到一亿以上）时，会使得运行速度变得很慢，熟悉mr原理的就明白这时sql跑的慢的原因，因为出现了很严重的数据倾斜。

案例分析：

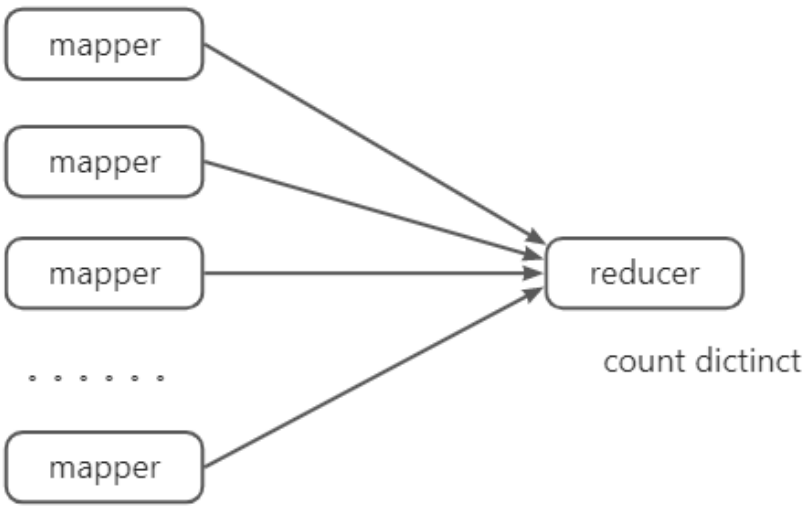
做去重统计时，一般都这么写：

```
select
  count(distinct (bill_no)) as visit_users
from
  i_usoc_user_info_d
where
  p_day = '20210508'
  and bill_no is not null
  and bill_no != ''
```

其实看起来，这没有任务毛病，但我们需要注意的是，此时写的是hql，它的底层引擎是MapReduce，是分布式计算的，所以就会出现数据倾斜这种分布式计算的典型问题，比如上面的使用数仓中一张沉淀了所有用户信息的融合模型来统计所有的手机号码的个数，这种写法肯定是能跑出结果的，但运行时长可能就会有点长。

我们去查下，就会发现记录数至少上亿，去hdfs中查看文件会发现这个分区很大，并且此时，我们通过查看执行计划和日志可以发现只有一个stage。也就是说最后只有一个reduce。

熟悉mr原理的已经明白了这条sql跑的慢的原因，因为出现了很严重的数据倾斜，几百个mapper，1个reducer，所有的数据在mapper处理过后全部只流向了一个reducer，逻辑计划大概如下：

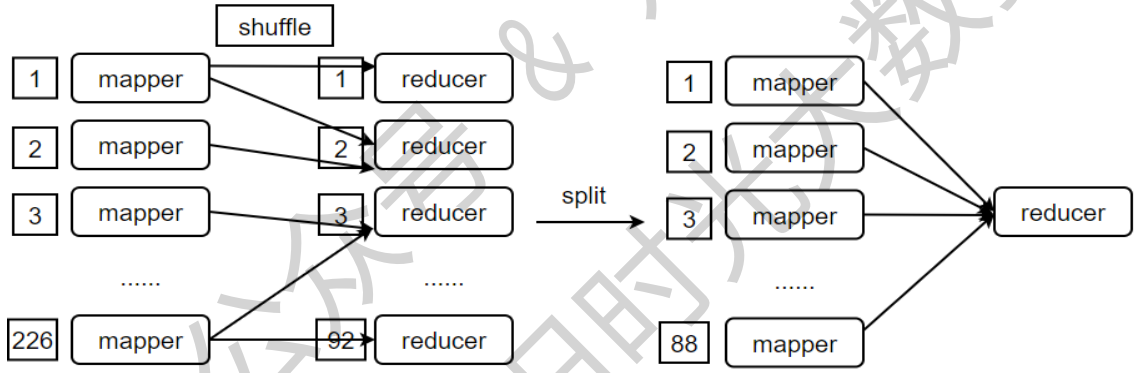


为什么只有一个reducer呢，因为使用了distinct和count(full aggreates)，这两个函数产生的mr作业只会产生一个reducer，而且哪怕显式指定set mapred.reduce.tasks=100000也是没用的。

所以对于这种去重统计，如果在数据量够大，一般是一亿记录数以上(视公司的集群规模，计算能力而定)，建议选择使用count加group by去进行统计：

```
select
  count(a.bill_no)
from
  (
    select
      bill_no
    from
      dwfu_hive_db.i_usoc_user_info_d
    where
      p_day = '20200408'
      and bill_no is not null
      and bill_no != ''
    group by
      bill_no
  ) a
```

这时候再测试，会发现速度会快很多，查看执行计划和日志，会发现启动了多个stage，也就是多个mr作业，这是因为引入了group by将数据分组到了多个reducer上进行处理。逻辑执行图大致如下：



总结：在数据量很大的情况下，使用count+group by替换count(distinct)能使作业执行效率和速度得到很大的提升，一般来说数据量越大提升效果越明显。

注意：开发前最好核查数据量，别什么几万条几十万条几十M数据去重统计就count加groupby就咔咔往上写，最后发现速度根本没有直接count(distinct)快，作业还没起起来人家count(distinct)就完事结果出来了，所以优化还得建立在一个数据量的问题上，这也是跟其他sql的区别。

欢迎加入知识星球，获取《大数据面试题 V4.0》以及更多大数据开发内容