hive数据倾斜:

数据倾斜是进行大数据计算时最经常遇到的问题之一。当我们在执行HiveQL或者运行MapReduce作业时候，如果遇到一直卡在map100%,reduce99%一般就是遇到了数据倾斜的问题。数据倾斜其实是进行分布式计算的时候，某些节点的计算能力比较强或者需要计算的数据比较少，早早执行完了，某些节点计算的能力较差或者由于此节点需要计算的数据比较多，导致出现其他节点的reduce阶段任务执行完成，但是这种节点的数据处理任务还没有执行完成。

在hive中产生数据倾斜的原因和解决方法：

1）group by,我使用Hive对数据做一些类型统计的时候遇到过某种类型的数据量特别多，而其他类型数据的数据量特别少。当按照类型进行group by的时候，会将相同的group by字段的reduce任务需要的数据拉取到同一个节点进行聚合，而当其中每一组的数据量过大时，会出现其他组的计算已经完成而这里还没计算完成，其他节点的一直等待这个节点的任务执行完成，所以会看到一直map 100%  reduce 99%的情况。

解决方法：set hive.map.aggr=true

　　　set hive.groupby.skewindata=true

原理：hive.map.aggr=true 这个配置项代表是否在map端进行聚合

　hive.groupby.skwindata=true

当选项设定为 true，生成的查询计划会有两个 MR Job。第一个 MR Job 中，Map 的输出结果集合会随机分布到 Reduce 中，每个 Reduce 做部分聚合操作，并输出结果，这样处理的结果是相同的 Group By Key 有可能被分发到不同的 Reduce 中，从而达到负载均衡的目的；第二个 MR Job 再根据预处理的数据结果按照 Group By Key 分布到 Reduce 中（这个过程可以保证相同的 Group By Key 被分布到同一个 Reduce 中），最后完成最终的聚合操作。

2）map和reduce优化。

1.当出现小文件过多，需要合并小文件。可以通过set hive.merge.mapfiles=true来解决。

2.单个文件大小稍稍大于配置的block块的大写，此时需要适当增加map的个数。解决方法：

set mapred.map.tasks个数

3.文件大小适中，但map端计算量非常大，如select id,count(\*),sum(case when...),sum(case when...)...需要增加map个数。解决方法：set mapred.map.tasks个数，set mapred.reduce.tasks个数

4.当HiveQL中包含count（distinct）时，如果数据量非常大，执行如select a,count(distinct b) from t group by a;类型的SQL时，会出现数据倾斜的问题。

解决方法：使用sum...group by代替。如select a,sum(1) from (select a, b from t group by a,b) group by a;

5.当遇到一个大表和一个小表进行join操作时。

解决方法：使用mapjoin 将小表加载到内存中。

如：select /\*+ MAPJOIN(a) \*/ a.c1, b.c1 ,b.c2 from a join b where a.c1 = b.c1;

6.遇到需要进行join的但是关联字段有数据为空，如表一的id需要和表二的id进行关联

解决方法1：id为空的不参与关联

比如：select \* from log a join users b　on a.id is not null and a.id = b.id　union all select \* from log a where a.id is null;

解决方法2：给空值分配随机的key值

如：select \* from log a left outer join users b on case when a.user\_id is null then concat(‘hive’,rand() ) else a.user\_id end = b.user\_id;