Yarn7 任务推测执行

任务的推测执行

1.作业完成时间取决于最慢的任务完成时间

> 一个作业由若干个Map任务和Reduce任务构成。因硬件老化、软件Bug等，某些任务可能运行非常慢。

典型案例：系统中有99%的Map任务都完成了，只有少数几个Map老是进度很慢，完不成，怎么办？

2.推测执行机制

> 发现拖后腿的任务，比如某个任务运行速度远慢于任务平均速度。为拖后腿任务启动一个备份任务，同时运行。谁先运行完，则采用谁的结果

3.执行推测任务的前提条件

```
（1）每个Task只能有一个备份任务；
（2）当前Job已完成的Task必须不小于0.05（5%）
（3）开启推测执行参数设置。Hadoop2.7.2 mapred-site.xml文件中默认是打开的。
```

```
# mapred-site.xml
<property>
    <name>mapreduce.map.speculative</name>
    <value>true</value>
    <description>If true, then multiple instances of some map tasks may be
executed in parallel.
    </description>
</property>
<property>
    <name>mapreduce.reduce.speculative</name>
    <value>true</value>
    <description>If true, then multiple instances of some reduce tasks may be
executed in parallel.
    </description>
</property>
```

不能启用推测执行机制情况

```
（1）任务间存在严重的负载倾斜；
（2）特殊任务，比如任务向数据库中写数据
```

5. 推测执行算法的原理

某一时刻，任务T的执行进度为progress，则可通过一定的算法推出该任务的最终完成时刻estimateEndTime。另一个方面，如果此时为该任务启动一个备份任务，则可以推断它可能完成时刻estimateEndTime'，于是可以得到以下几个公式：

$$estimateEndTime = estmateRunTime + taskStartTime \quad (1)$$

$$推测执行完成时刻60 = 推测运行时间（60s） + 任务启动时刻（0） \quad (2)$$

$$estimateRunTime = (currentTimeStamp + taskStartTime)/progress \quad (3)$$

$$\text{推测运行时间}60s = (\text{当前时刻}6 - \text{任务启动时刻}0)/\text{任务运行比例（}10 \quad (4)$$

$$estimateEndTime = (currentTimeStamp + averageRunTime) \quad (5)$$

$$\text{备份任务推测完成时刻}16 = (\text{当前时刻}6 + \text{运行完成任务的平均时间}) \quad (6)$$

总结：

1. MR总是选择(estimateEndTime - estimateEndTime`)差值最大的任务，并为之启动备份任务
2. 为了防止大量任务同时启动备份造成的资源浪费，MR为每个作业设置了同时启动的备份任务数目上限
3. 推测执行机制实际上采用了经典的优化算法：以空间换时间，它同时启动多个相同任务处理相同的数据，并让这些任务竞争以缩短数据处理时间，显然，这种方法需要占用更多的资源，在集群资源紧缺的情况下，应合理使用该机制，争取在多用少量资源的情况下，减少作业的计算时间。