Schedule mode in Spark

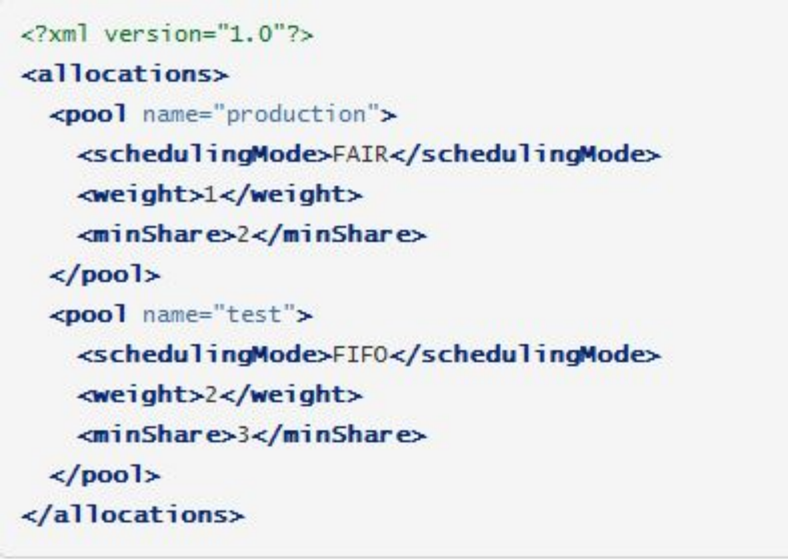
王涛(wangtaothetonic@163.com)

Introduction

- In Spark an "action" tirggers a job running which contains some tasks, which is presented as a TaskSet in Spark.
- If there are many jobs, which are not related, submitted by different threads(like in Spark Thrift Server), they share one cluster resources.
- By default, they are scheduled in FIFO manner, which means the job first submitted can run first.
- If the fore job takes all the resources, then the rear one should wait.
- There is another fasion of scheduling, which is called FAIR mode.

FAIR mode(introduced in Spark 0.8)

- You can set *spark.scheduler.mode* to *FAIR* in SparkConf to turn on FAIR schedule mode.
- You can specify a pool for a job in FAIR schedule mode by using *sc.setLocalProperty("spark.scheduler.pool", "pool1").* Once the pool get resources, jobs in it can be launched.
- You SHOULD specify a xml file for pools definition. Like this one:



```xml
<?xml version="1.0"?>
<allocations>
  <pool name="production">
    <schedulingMode>FAIR</schedulingMode>
    <weight>1</weight>
    <minShare>2</minShare>
  </pool>
  <pool name="test">
    <schedulingMode>FIFO</schedulingMode>
    <weight>2</weight>
    <minShare>3</minShare>
  </pool>
</allocations>
```

Buiding the pool

- While TaskScheduler being initialized, the pool is built. Only in FAIR mode, it will read xml file specified by *spark.scheduler.allocation.file*, and use it to build pool.
- Building the pool is just add child nodes according to the scheduler file for root pool.
- The code location is in *FairSchedulableBuilder.buildFairSchedulerPool()*

Filling in the pool

- Once DAGScheduler submit tasks(be wrapped into a TaskSet), it will call *TaskScheduler.submitTasks* in which it will create a TaskSetManager for this TaskSet and add the TaskSetManager into schedulableBuilder(This schedulableBuilder is either FIFO or FAIR depending on setting).
- In FAIR mode, it will look up pool name for this TaskSet, then add this TaskSetManager into the returned pool. If the pool name is not found, will create a default one for it.
- See *FairSchedulableBuilder.addTaskSetManager*

Getting tasks from the pool

- After every submit of TaskSet, *SchedulerBackend* will make offers for all of them.
- After getting all resources offered by workers, *SchedulerBackend* will get a sorted task set by calling *rootPool.getSortedTaskSetQueue.*
- There are two level of scheduling, one between pools and another one within a pool. They all are done in one function but we will look into them in two pages.
- They will fetch all *Schedulable* objects and sorted them using *taskSetSchedulingAlgorithm.comparator*. See *Pool.getSortedTaskSetQueue.*

Scheduling in FIFO manner

- In FIFO manner, there is only one root pool containing many TaskSetManager.
- The comparator first compares priority of TaskSet, which is actually their jobId. The TaskSet who has **minor jobId runs first**. If the jobId are same, it depends on who **having minor stageId**.

Scheduling in FAIR manner(between pools)

- In FAIR mode, the comparator works like below:

1. Who is **needy** runs first. Jump to step 2 if both are needy.
2. Whose **minShareRatio lower** runs first. If equals jump to step 3.
3. Whose **taskToWeightRatio lower** runs first. If equals jump to step 4.
4. Whose **name** ranks forward runs first.

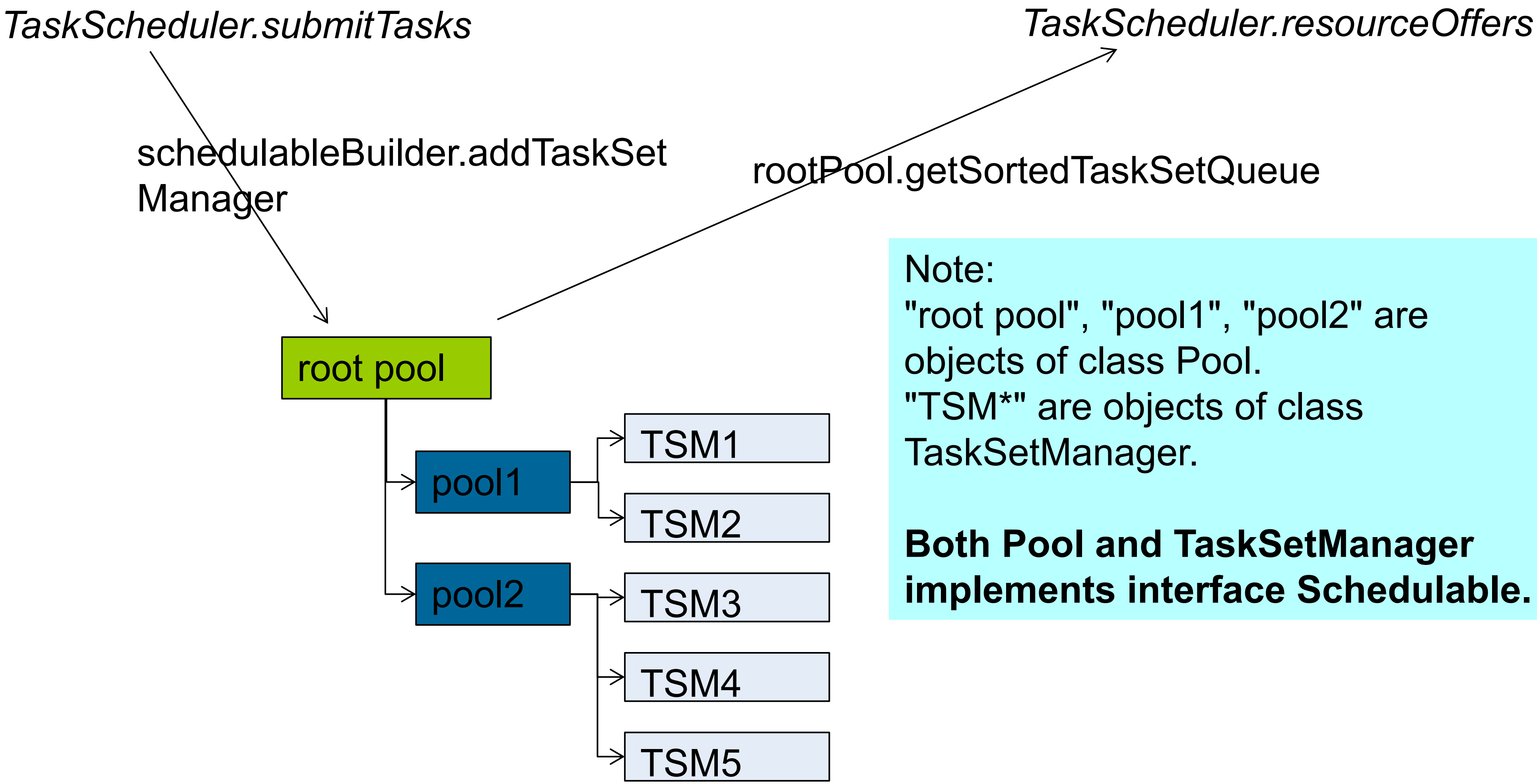
Note:

needy means #runningTasks < minShare
minShareRatio = #runningTasks / max(minShare, 1.0)
taskToWeightRatio = #runningTasks / weight
name ranks in dictionary manner

Scheduling in FAIR manner(within pools)

- The pool in FAIR schedle mode has two levels, the first one is pool like the front page shows and the second one is TaskSetManager in them.
- Actually the TaskSetManager will be sorted just like the pool, the only difference is that the minShare in all TaskSetManager is 0, and the weight is all 1.
- So the TaskSetManager will be sorted **first with #running tasks, then name.**

A Sketch Map



TaskScheduler.submitTasks
TaskScheduler.resourceOffers
schedulableBuilder.addTaskSet Manager
rootPool.getSortedTaskSetQueue
root pool
pool1
pool2
TSM1
TSM2
TSM3
TSM4
TSM5

Note:
"root pool", "pool1", "pool2" are objects of class Pool.
"TSM*" are objects of class TaskSetManager.

Both Pool and TaskSetManager implements interface Schedulable.

THANKS