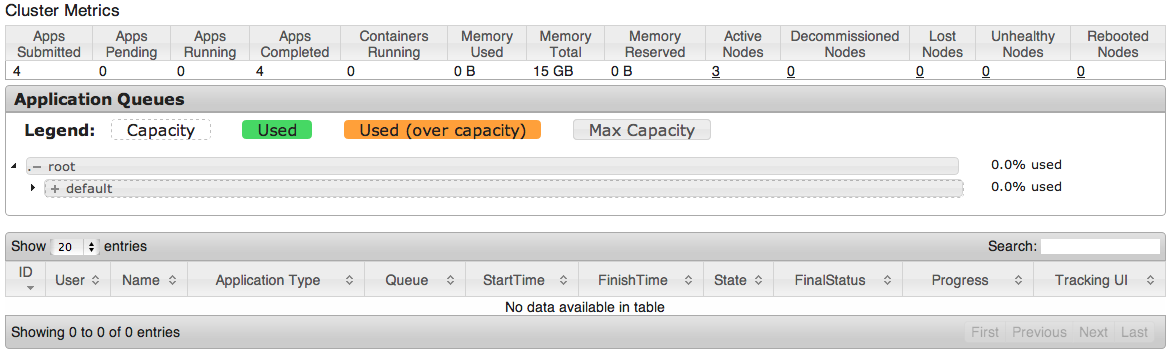
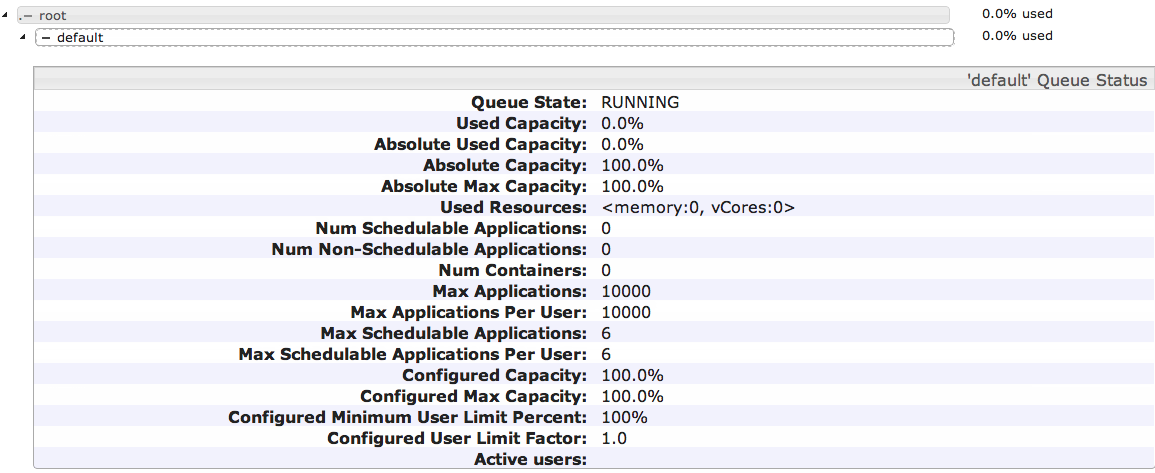
Lab: Configuring the Capacity Scheduler

|  |  |
| --- | --- |
| **Objective:** | Learn how to define and configure queues for the Capacity Scheduler. |
| **Successful Outcome:** | You will have two new queues configured. |
| **Before You Begin:** | Your cluster should be up and running. |

1. View the Status of the Capacity Scheduler
   1. Point your web browser to http://*[server]*:8088 to view the UI of the Resource Manager.
   2. In the menu on the left side, click on the Scheduler link to view the current status of the Capacity Scheduler:

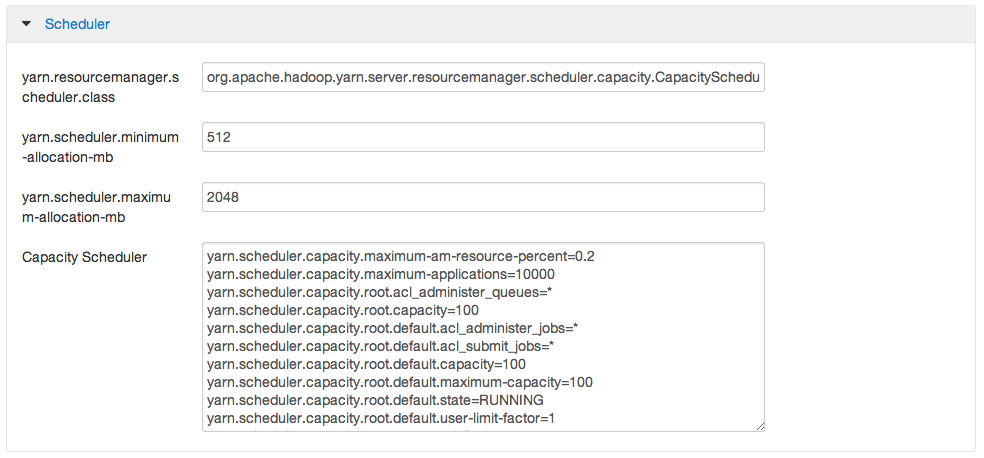


* 1. Notice there is one child queue of root defined. What is the name of the queue? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
  2. Click on the arrow to the left of “default” to expand and view the settings of the default queue:



* 1. Notice since there are no jobs running on your cluster, the status page simply shows 0.0% of default is being used right now.

1. View the Settings of the Capacity Scheduler
   1. Go to the **Ambari** Dashboard page (port 8080).
   2. Click on the YARN link in the list of services, then click on the Configs tab and scroll down to the Scheduler section:



* 1. Notice this is where you configure the settings for the scheduler of the Resource Manager. Which type of scheduler is currently being used? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Define Custom Queues
   1. From the Configs tab of the YARN Services page, add two new queues named A and B to the Capacity Scheduler (and leave the default queue defined also):

yarn.scheduler.capacity.root.queues=default,A,B

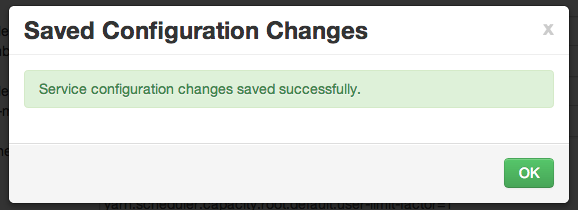
* 1. Assign the following capacities to the three queues:

default capacity = 20%  
 A capacity = 50%  
 B capacity = 30%

* 1. Assign the following maximum capacities to the three queues:

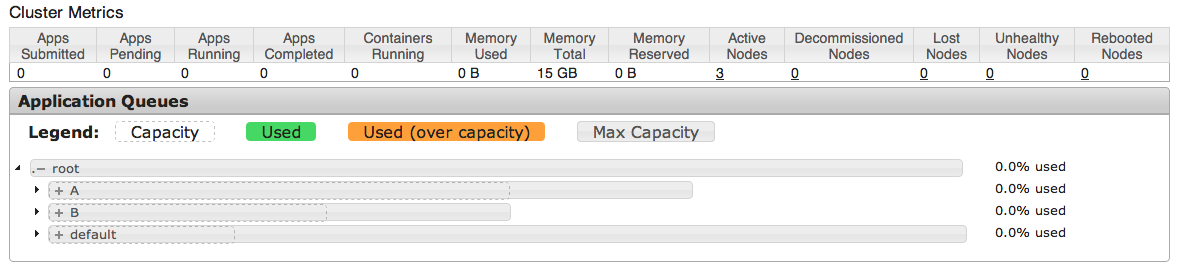
default maximum capacity = 100%  
 A maximum capacity = 70%  
 B maximum capacity = 50%

* 1. Save your changes by clicking the Save button at the bottom of the page. You should see the following confirmation dialog:



* 1. Click OK to close the dialog window.

1. ReStart YARN
   1. Go back to the Summary page of YARN and click the Start button to start the YARN service again.
2. Verify the Changes
   1. Go back to the Scheduler web page at http://*[localhost]*:8088/cluster/scheduler.
   2. You should see the A and B queues now:



* 1. Expand the A queue and verify its capacity is 50% and its maximum capacity is 70%:

Screen Shot 2013-11-08 at 10

* 1. Similarly, verify B and default are configured correctly.
  2. Leave this web page open - you are going to view it in the next step.

1. Submit a Job to a Specific Queue
   1. Let’s submit a MapReduce job to each new queue. SSH into **hadoop-master**, and change directories to the /root/labs folder:

[root@node1 ~]# cd ~/labs

* 1. Make sure environment variable JAVA\_HOME is defined, so enter the following command:

# export JAVA\_HOME=/usr/jdk64/jdk1.6.0\_31/

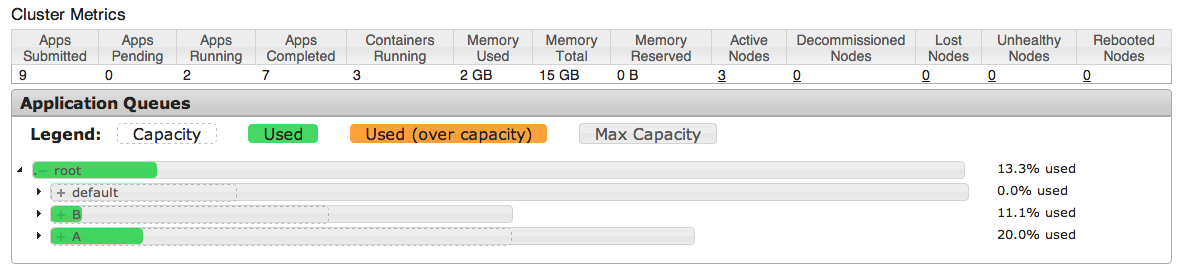
* 1. Make sure you have a file in /user/root in HDFS named hbase.jar. If not, put the HBase jar from the labs folder into HDFS, giving it the name hbase.jar.
  2. In the first window, submit the test1.pig script to queue A by running the following command (all on a single line):

# pig -Dmapreduce.job.queuename=A test1.pig 1>pig1.out &>pig1.err &

* 1. While test1.pig is running, submit the test2.pig script to queue B in the other terminal window:

# pig -Dmapreduce.job.queuename=B test2.pig 1>pig2.out &>pig2.err &

* 1. While both jobs are running, refresh the Scheduler status page. (It may take a minute for both jobs to run long enough to show up in the queues, so refresh the page often until they do):



* 1. You should see resources being used in both the A and B queues.

**RESULT**: You just defined two queues for the Capacity Scheduler, configured specific capacities for each queue, and submitted a job to each queue.

**ANSWERS**:

1.3: default

2.3: The Capacity Scheduler