



Hadoop YARN / [YARN-2492 \(Clone of YARN-796\) Allow for \(admin\) labels on nodes and resource-requests](#) / [YARN-2496](#)

# Changes for capacity scheduler to support allocate resource respect labels

Details

Type:	Sub-task	Status:	<b>CLOSED</b>
Priority:	Major	Resolution:	Fixed
Affects Version/s:	None	Fix Version/s:	2.6.0
Component/s:	<a href="#">resourcemanager</a>		
Labels:	None		
Target Version/s:	2.6.0		
Hadoop Flags:	Reviewed		

Description

- This JIRA Includes:
- Add/parse labels option to `capacity-scheduler.xml` similar to other options of queue like capacity/maximum-capacity, etc.
  - Include a "default-label-expression" option in queue config, if an app doesn't specify label-expression, "default-label-expression" of queue will be used.
  - Check if labels can be accessed by the queue when submit an app with labels-expression to queue or update ResourceRequest with label-expression
  - Check labels on NM when trying to allocate ResourceRequest on the NM with label-expression
  - Respect labels when calculate headroom/user-limit

Attachments

<a href="#">YARN-2496.patch</a>	164 kB	07/Oct/14 05:20
YARN-2496.patch	163 kB	07/Oct/14 03:05
YARN-2496.patch	149 kB	06/Oct/14 22:25
YARN-2496.patch	148 kB	06/Oct/14 18:22
YARN-2496.patch	102 kB	12/Sep/14 14:30
YARN-2496.patch	100 kB	11/Sep/14 10:36
YARN-2496.patch	100 kB	11/Sep/14 10:05
YARN-2496.patch	81 kB	09/Sep/14 15:53
<a href="#">YARN-2496+YARN-2500.patch</a>	283 kB	15/Oct/14 21:25
YARN-2496+YARN-2500.patch	277 kB	15/Oct/14 17:13
<a href="#">YARN-2496-20141009-1.patch</a>	167 kB	09/Oct/14 22:43
<a href="#">YARN-2496-20141014-1.patch</a>	172 kB	15/Oct/14 00:21
<a href="#">YARN-2496-20141015-1.patch</a>	175 kB	15/Oct/14 17:10
<a href="#">YARN-2496-20141015-2.patch</a>	175 kB	15/Oct/14 21:23

Activity

- [Wangda Tan](#) added a comment - 09/Sep/14 15:53
- Attached patch, this patch is based on trunk, but it cannot be compiled itself.  
It will be hard to separate [YARN-2500](#) with [YARN-2496](#) and make each of them can be compiled. I split them just for easier reviewing.  
This patch is based on [YARN-2493](#) -> [YARN-2494](#) > [YARN-2500](#), you need apply these patches.
- [Hadoop QA](#) added a comment - 09/Sep/14 16:15

-1 overall. Here are the results of testing the latest attachment

<http://issues.apache.org/jira/secure/attachment/12667431/YARN-2496.patch>

against trunk revision 90c8ece.

+1 @author. The patch does not contain any @author tags.

+1 tests included. The patch appears to include 7 new or modified test files.

-1 javac. The patch appears to cause the build to fail.

Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/4855//console>

This message is automatically generated.


▼  Jian He added a comment - 11/Sep/14 04:44

briefly looked at the patch:

- CSQueueUtils.java format change only, we can revert
- why checking `labelManager != null` every where ? we only need to check where it's needed.
- We may not need to change the method signature to add one more parameter, just pass the queues map into `NodeLabelManager#reinitializeQueueLabels`, to avoid a number of test changes.

```
parseQueue(this, conf, null, CapacitySchedulerConfiguration.ROOT,
           queues, queues, noop, queueToLabels);
```

- label initialization code is duplicated between ParentQueue and LeafQueue, how about creating an AbstractCSQueue and put common initialization methods there ?

▼  Wangda Tan added a comment - 11/Sep/14 10:05

Hi jianhe,

Thanks for your comments,

CSQueueUtils.java format change only, we can revert

Reverted

why checking `labelManager != null` every where ? we only need to check where it's needed.

It was used to reduce changes in tests. I think we should remove these checks and improve tests

We may not need to change the method signature to add one more parameter, just pass the queues map into `NodeLabelManager#reinitializeQueueLabels`, to avoid a number of test changes.


Make sense, now reverted changes for related tests and get a queueToLabels after parseQueue

label initialization code is duplicated between ParentQueue and LeafQueue, how about creating an AbstractCSQueue and put common initialization methods there ?

Make sense, there do have lots of common code between PQ and LQ, now I have merged all common parts to abstractCSQueue.

Attached a new patch

Thanks,  
Wangda

▼  Hadoop QA added a comment - 11/Sep/14 10:12

-1 overall. Here are the results of testing the latest attachment


<http://issues.apache.org/jira/secure/attachment/12667992/YARN-2496.patch>

against trunk revision 4be9517.

-1 patch. The patch command could not apply the patch.

Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/4887//console>

This message is automatically generated.

▼  Hadoop QA added a comment - 11/Sep/14 10:55

-1 overall. Here are the results of testing the latest attachment

<http://issues.apache.org/jira/secure/attachment/12668003/YARN-2496.patch>

against trunk revision 4be9517.


+1 @author. The patch does not contain any @author tags.

+1 tests included. The patch appears to include 10 new or modified test files.

-1 javac. The patch appears to cause the build to fail.

Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/4888//console>

This message is automatically generated.

▼  Hadoop QA added a comment - 12/Sep/14 14:45

-1 overall. Here are the results of testing the latest attachment

<http://issues.apache.org/jira/secure/attachment/12668345/YARN-2496.patch>

against trunk revision 78b0483.


+1 @author. The patch does not contain any @author tags.

+1 tests included. The patch appears to include 10 new or modified test files.

-1 javac. The patch appears to cause the build to fail.


Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/4918//console>

This message is automatically generated.

▼  Craig Welch added a comment - 16/Sep/14 17:30

LeafQueue.java

I think there's a bug in the way a user's consumed value is obtained when working with labels (it's unchanged from the existing behavior, but it needs to change, I think) The consumed is going to be for all jobs regardless of label regardless of label expression but it's being subtracted from a value which is limited to the user expression, so - this could be incorrect if user has some jobs w/out labels and some with (or with different labels) in the same queue. Jobs using labels could be starved due to jobs running under the same user with other labels. I believe consumed resources needs to be per label and/or label expression.


▼  Wangda Tan added a comment - 17/Sep/14 07:09

Hi cwelch,

I'm not sure quite understand about this, did you mean we need calculate consumed resource for each label (or label-expression) under each queue? Could you give me an example about how to avoid job starvation with it?

It confuse me that, if we have resource per label/l-expression, should we have resource per host/rack (we can ask for resource only on a host/rack by specifying relax-locality).

Thanks,  
Wangda

▼  Craig Welch added a comment - 19/Sep/14 21:33

Let's say user UserU is running two jobs, JobA and JobB, in queue QueueAB, which has both LabelA and LabelB. JobA has label LabelA, JobB has LabelB. It's a two node cluster, Node1 and Node2. Node1 has LabelA, Node2 has LabelB

Let's say the user has access to 100% of the cluster (just the one queue, etc).

Let's say that JobA is using 4G of Ram and JobB is also using 4G. Let's say each node has 6G.

Headroom Calculation for JobA:


userConsumed = 8G

maxCapacityConsiderLabelA = 6G (Node1 only)

headroom = -2G (assume it will normalize to 0G)

However, the user should still be able to use the remaining 2G for JobA, as they are only using 4 of the 6G available to that label.

The issue I see is userConsumed, as maxCapacityConsiderLabel considers the label, but userConsumed does not, it should only be "userConsumedForLabel", if it is, JobA would see 2G as it should (as the consumed for LabelA is only 4G.) The problem, I think, is in subtracting a cross-label value from a per-label value.

▼  Craig Welch added a comment - 19/Sep/14 21:44


I'm also concerned about this:

Resource maxCapacityConsiderLabel =


labelManager == null ? clusterResource : labelManager.getQueueResource{

queueName, labels, clusterResource);


The "labels" are the labels for the queue, but the resource requests coming from the application can be a subset of that, no? So if application "a" is running on a queue with labels a and b, but it has a label expression of only a, which it is using for resource requests, it's going to get a headroom based on nodes with both labels a and b, but in fact it only has a "real" headroom for nodes with label "a"

▼  [Craig Welch](#) added a comment - 19/Sep/14 21:51


On the parent/leaf refactor to share AbstractCSQueue - a great idea, thought about it myself when seeing the duplication, but I think that doing it while making changes like adding node labels adds confusion and makes it harder to see functional changes, I think it should have been done in isolation at some point (where no other changes were occurring). I don't think you should change course on it now (I'm not suggesting any changes to what you have... I think it would be more risky than not at this point), just a thought for future cases like this.

▼  [Craig Welch](#) added a comment - 19/Sep/14 21:53

CSQueueUtils - just removing a line, should revert

▼  [Craig Welch](#) added a comment - 19/Sep/14 23:05

SchedulerUtils.checkNodeLabelExpression - I think there is an issue here with the \* case, as I read the code a \* case will not properly match against a node with a label - after the first check, there should be a check for ANY in the expression, and if so, return true

▼  [Wangda Tan](#) added a comment - 20/Sep/14 01:38

Hi [c Welch](#),

Thanks for your comments:

1) Regarding,

```
Headroom Calculation for JobA:
userConsumed = 8G
maxCapacityConsiderLabelA = 6G (Node1 only)
headroom = -2G (assume it will normalize to 0G)
```

Currently, we calculate headroom by,

Headroom = min(userLimit, queue-max-cap, max-capacity-consider-label) - consumed

The

{max-capacity-consider-label} is queue-wise not app-wise, so in the queue-wise, the max-capacity-consider-label = node1 + node2.

You can think that, the {max-capacity-consider-label}

can guarantee it's always larger or equals than total resource of the queue will use.

2) Regarding

The "labels" are the labels for the queue, but the resource requests coming from the application can be a subset of that, no? So if application "a" is running on a queue with labels a and b, but it has a label expression of only a, which it is using for resource requests, it's going to get a headroom based on nodes with both labels a and b, but in fact it only has a "real" headroom for nodes with label "a"

Yes/No, because even if app-a has "a" label in app-level, it's ResourceRequest(s) can also overwrite it and use b. Label in app-level is just a default label expression when its ResourceRequest doesn't set. So, app-a can still use all labels of queue.

3) Regarding,

On the parent/leaf refactor to share AbstractCSQueue - a great idea, thought about it myself when seeing the duplication,

I agree with that, I think it may not too risky but it will hide functional changes we made. Let's get more ideas about this, because reverting it need some efforts.

4) Regarding,

CSQueueUtils - just removing a line, should revert

Will do


5) Regarding,

SchedulerUtils.checkNodeLabelExpression - I think there is an issue here with the \* case

`checkNodeLabelExpression` is used for check if a `ResourceRequest` can be allocated on a node, we don't support specifying `*` in any label-expression (including `ResourceRequest`, `ASC`, `queue-default-label-expression`), that will cause many problem. Instead, we support `*` in queue's labels (not default-label-expression), which means queue **can** access any labels. The checking methods are `checkQueueAccessToNode` and `checkQueueLabelExpression`.

Thanks,  
Wangda


---

▼  [Craig Welch](#) added a comment - 20/Sep/14 03:55

Ok, so it sounds like 1, 3 and 4 are ok.

I think that 2 is still a problem though - headroom is an app level value, and even though an app may be able to use either label in a resource request it will, in some cases, not be able to. A typical case will be where a subset of nodes in


---

▼  [Craig Welch](#) added a comment - 20/Sep/14 04:02

ach, not finished - anyway, re 2 - a particular job may well only be able to use nodes with one label in a queue, and so if the headroom includes nodes without that label, we'll end up with another deadlock case where it spins up reducers too early and then can't complete its maps. It is definitely a valid usecase to have a queue with two labels (a and b, as in this example) and an app which is consistently requesting only one of those two labels (from submission to completion...) - perhaps only "a" nodes have the "special resource" it needs (special hardware capability, etc). For this reason, headroom should reflect the labels in the last resource request from the application, not the queue's labels.

(-re 5, I though `*` could be in requests, if no, then should not be an issue.)

---

▼  [Wangda Tan](#) added a comment - 20/Sep/14 14:28

Craig, still about #2, I think what you commented is make sense to me, AM can get a more precise headroom to plan its following resource usage, but I think:

1) It may not enough as what you said:

For this reason, headroom should reflect the labels in the last resource request from the application, not the queue's labels.

It is possible an AM sent resource requests with different label expression, so what we will response headroom back to AM? I think maybe we need a new field in `AllocateRequest` to request different headrooms under different label expression.

2) Even with 1), I cannot think of a good way to fast computing random label expression in an acceptable time complexity, it is possible thousands of different label expression existed in a big cluster at the same time. Our current implementation can make sure resource of labels of a queue will up-to-date whenever resource change happened.

With 1) and 2). I suggest to make it as a pending task, and we can deal it in the future.


About

-re 5, I though `*` could be in requests, if no, then should not be an issue.

Yes, we doesn't support specify `*` in requests, because it may cause some possible resource wastage. AM should clearly know what resource it needed.

Thanks,  
Wangda

---

▼  [Craig Welch](#) added a comment - 22/Sep/14 23:04

So, re the headroom issue (2) - the short version - I don't think we can put off addressing this, because I think it is going to be a typical case and will be problematic. I think the most realistic solution is to support only a short list of pre-configured label expressions per queue. Another option is to limit nodes to supporting only 1 label per node (which, realistically, might be sufficient). A third option is to limit the number of labels which a queue can access to a very small value + the "all" value (1-2). Basically, one of the factors pushing the large set of possible values which must be considered to properly calculate headroom needs to be made finite/dramatically reduced.

longer version...


I don't think we should move forward without addressing it. I say this because I think it is likely to be a typical situation to have a queue which has more than one label associated with it- most likely, the simple case of a queue which can address all nodes some of which have a label and some of which do not. Jobs entering these queues using a restrictive label expression will hit this headroom issue - it's especially true in cases where there are lower resources, which is what one would expect from a "small set of special machines" (e.g. typical node label case). It's important to make sure headroom is correctly handled as we add node labels, and as things stand, we know it is not.

I'm afraid it is something of a design issue, allowing arbitrary node label expressions with multiple labels on queues, etc, is leading to something of a combinatorial explosion. It may be that the right solution is to narrow the feature set a bit for this iteration. We could choose to only support a restricted set of expressions on a given queue. This could even mean only supporting the default label expression - I'm

concerned that this may be too restrictive - and so that we would need to support a set of expressions. This could then be a finite list which is pre-calculated. I think, in practical terms, this will probably meet people's needs. A second option is to restrict the number of labels supported on a queue, a small enough set could be pre-calculated for all possibilities. I'm suspicious of this latter option, though, it would have to be a very small number of labels to be manageable and I think it reduces, realistically, to the restricted set of expressions.


I also don't see any performant way to support arbitrary node-label expressions on every request with unlimited labels per queue and node - things as they are. It appears to me you would need to keep track of all resource values for intersection of all label combinations. If we limited the number of possible labels on a node to one then we could calculate based on expressions at runtime (possibly for a very small number > 1, but again, growth is exponential? I believe... and functionally complex)

---

▼  **Hadoop QA** added a comment - 06/Oct/14 18:22

Attached updated patch

---

▼  **Hadoop QA** added a comment - 06/Oct/14 18:46

-1 **overall**. Here are the results of testing the latest attachment  
<http://issues.apache.org/jira/secure/attachment/12673145/YARN-2496.patch>  
against trunk revision ea26cc0.

+1 **@author**. The patch does not contain any @author tags.


+1 **tests included**. The patch appears to include 12 new or modified test files.

-1 **javac**. The patch appears to cause the build to fail.

Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/5277//console>

This message is automatically generated.

---

▼  **Hadoop QA** added a comment - 06/Oct/14 22:46

-1 **overall**. Here are the results of testing the latest attachment  
<http://issues.apache.org/jira/secure/attachment/12673224/YARN-2496.patch>  
against trunk revision 8dc6abf.

+1 **@author**. The patch does not contain any @author tags.


+1 **tests included**. The patch appears to include 12 new or modified test files.

-1 **javac**. The patch appears to cause the build to fail.

Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/5287//console>

This message is automatically generated.

---

▼  **Hadoop QA** added a comment - 07/Oct/14 03:11


-1 **overall**. Here are the results of testing the latest attachment  
<http://issues.apache.org/jira/secure/attachment/12673283/YARN-2496.patch>  
against trunk revision 519e5a7.

-1 **patch**. The patch command could not apply the patch.

Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/5297//console>


This message is automatically generated.

---

▼  **Wangda Tan** added a comment - 07/Oct/14 05:20

Attached new patch against latest trunk

---

▼  **Hadoop QA** added a comment - 07/Oct/14 06:41

-1 **overall**. Here are the results of testing the latest attachment  
<http://issues.apache.org/jira/secure/attachment/12673291/YARN-2496.patch>  
against trunk revision 0fb2735.

+1 **@author**. The patch does not contain any @author tags.

+1 **tests included**. The patch appears to include 5 new or modified test files.

+1 **javac**. The applied patch does not increase the total number of javac compiler warnings.

+1 **javadoc**. There were no new javadoc warning messages.

+1 **eclipse:eclipse**. The patch built with eclipse:eclipse.

+1 **findbugs**. The patch does not introduce any new Findbugs (version 2.0.3) warnings.

+1 **release audit**. The applied patch does not increase the total number of release audit warnings.

-1 **core tests**. The patch failed these unit tests in hadoop-yarn-project/hadoop-yarn/hadoop-yarn-api hadoop-yarn-project/hadoop-yarn/hadoop-yarn-client hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager:


org.apache.hadoop.yarn.client.TestApplicationClientProtocolOnHA  
org.apache.hadoop.yarn.server.resourcemanager.security.TestAMRMTokens  
org.apache.hadoop.yarn.server.resourcemanager.TestResourceManager  
org.apache.hadoop.yarn.server.resourcemanager.TestRMRestart

+1 **contrib tests**. The patch passed contrib unit tests.

Test results: <https://builds.apache.org/job/PreCommit-YARN-Build/5300/testReport/>

Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/5300/console>

This message is automatically generated.

▼  **Hadoop QA** added a comment - 09/Oct/14 23:06

-1 **overall**. Here are the results of testing the latest attachment

<http://issues.apache.org/jira/secure/attachment/12674034/YARN-2496-20141009-1.patch>  
against trunk revision 596702a.


+1 **@author**. The patch does not contain any @author tags.

+1 **tests included**. The patch appears to include 12 new or modified test files.

-1 **javac**. The patch appears to cause the build to fail.

Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/5348/console>

This message is automatically generated.

▼  **Vinod Kumar Vavilapalli** added a comment - 14/Oct/14 01:47

Started looking at the patch. Some early comments

CapacitySchedulerConfiguration

```
LOG.debug("CSConf - getCapacityOfLabel: prefix="
    + getNodeLabelPrefix(queue, label) + ", capacity=" + capacity);
```

Wrap in LOG.isDebugEnabled()

Rename continueLooking variable in setupQueueConfigs to refer to reservation

LeafQueue.getRequestLabelSetByExpression() can be a static method.

AbstractQueue

- absoluteNodeLabelCapacities -> absoluteCapacitiesByNodeLabels
- nodeLabelCapacities -> capacitiesByNodeLabels
- absoluteMaximumNodeLabelCapacities -> absoluteMaxCapacityByNodeLabels
- maximumNodeLabelCapacities -> maxCapacitiesByNodeLabels
- Similarly the CSQueueUtils.compute\* methods
- labels -> accessibleLabels
- releaseResource(): nodeLabels.isEmpty() should also do a null check in case admin removed accessible labels for this queue?
- getCapacityByNodeLabel(): If parent is null, simply return 1; Similarly no need to check !nodeLabelCapacities.containsKey(label)
- Same as above in getAbsoluteCapacityByNodeLabel(). Also, for root, it is set to 1 incorrectly.

SchedulerUtils.checkAndThrowIfLabelNotIncluded -> checkIfLabelInTheCollection()

Seems like removing accessible labels from a queue doesn't work given the changes in AbstractCSQueue.

▼  **Vinod Kumar Vavilapalli** added a comment - 14/Oct/14 20:18

More comments

ParentQueue

- assignToQueue(): We are only checking that at least one label is within maximum capacity. Bug?
- assignToQueue() -> canAssignToThisQueue
- Not related to your patch, but removeApplication() can be private. Similarly assignContainersToChildQueues, printChildQueues.
- Can avoid multiple calls to labelManager.getLabelsOnNode(node.getNodeID()) inside assignContainers.



- getACLs() should be pushed up to AbstractQueue.
- I think reservations are still not handled per accessible node-labels in the patch. We can fix it separately though.
- Sorting queues doesn't take node-labels into account. Again, we can fix it separately.
- Explicitly mark calls to allocateResource() and releaseResource with super for better readability.
- We should change printChildQueues() and getChildQueuesToPrint() to print node-label associations too.
- The following check in LeafQueue needs to be present in ParentQueue too?

```
// if our queue cannot access this node, just return
if (!SchedulerUtils.checkQueueAccessToNode(accessibleLabels,
    labelManager.getLabelsOnNode(node.getNodeID()))) {
    return NULL_ASSIGNMENT;
}
```

#### AbstractQueue

- queueComparator should be pushed down to ParentQueue.
- releaseResource() should be protected

More to come.

Wangda Tan added a comment - 15/Oct/14 00:21

vinodkv, thanks for your comments. They all make sense to me and addressed in the new patch, except a few,

SchedulerUtils.checkAndThrowIfLabelNotIncluded -> checkIfLabelInTheCollection()

Changed to checkIfLabelInClusterNodeLabels

Seems like removing accessible labels from a queue doesn't work given the changes in AbstractCSQueue.

Removing accessible labels from a queue should be done by first modifying capacity scheduler configuration, then call refreshQueue by RMAdminCLI

I think reservations are still not handled per accessible node-labels in the patch. We can fix it separately though.

Reservation is partially works. Resource will be reserved only when the node satisfies queue accessible-label and the node satisfies node-label-expression of the resource request.

The reservation will not work with the changes of [YARN-1769](#) (continuous reservation looking). Which is required to count reserved resource by node label, we haven't counted reserved resource by node label.

Hadoop QA added a comment - 15/Oct/14 00:46

-1 overall. Here are the results of testing the latest attachment <http://issues.apache.org/jira/secure/attachment/12674883/YARN-2496-20141014-1.patch> against trunk revision cdce883.

+1 @author. The patch does not contain any @author tags.

+1 tests included. The patch appears to include 12 new or modified test files.

-1 javac. The patch appears to cause the build to fail.

Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/5393//console>

This message is automatically generated.

Vinod Kumar Vavilapalli added a comment - 15/Oct/14 02:19

Last set of code comments. Still to review tests:

- numApplications can be pushed down to ParentQueue
- LeafQueue
  - assignToQueue -> canAssignToThisQueue
  - Type in potencialNewCapacity
- We are only respecting label against ANY ResourceRequest. Let's document it for now. It is very difficult to fix the API, let's atleast validate in AMRMClient and ApplicationMasterService to accept labels only against ANY. Let's file a ticket for testing this.
- Multiple node-labels are not supported in the ResourceRequest for now. Similar to the above, document it and validate in AMRMClient and ApplicationMasterService. Let's file a ticket for testing this.

Wangda Tan added a comment - 15/Oct/14 17:10



vinodkv, Thanks for your comments!


Updated patch addressed all your comments above, will separated JIRAs as what you suggested to solve other problems right now.

---

▼  Wangda Tan added a comment - 15/Oct/14 17:13

Since [YARN-2496](#) and [YARN-2500](#) need apply together to make it compilable. So updated a patch contains [YARN-2496](#) and [YARN-2500](#) to get Jenkins report.

---


▼  Wangda Tan added a comment - 15/Oct/14 18:07

Filed [YARN-2694](#) for disable multiple node labels in resource request, and check node label expression only specified in ANY level.

Filed [YARN-2695](#) for respecting continuous container reservation with node label

Filed [YARN-2696](#) for considering node labels when sorting queues.

---

▼  Hadoop QA added a comment - 15/Oct/14 18:36

-1 overall. Here are the results of testing the latest attachment

<http://issues.apache.org/jira/secure/attachment/12675049/YARN-2496%2BYARN-2500.patch>  
against trunk revision b9edad6.

+1 @author. The patch does not contain any @author tags.

+1 tests included. The patch appears to include 27 new or modified test files.

+1 javac. The applied patch does not increase the total number of javac compiler warnings.

+1 javadoc. There were no new javadoc warning messages.

+1 eclipse:eclipse. The patch built with eclipse:eclipse.

-1 findbugs. The patch appears to introduce 1 new Findbugs (version 2.0.3) warnings.

+1 release audit. The applied patch does not increase the total number of release audit warnings.

-1 core tests. The patch failed these unit tests in hadoop-tools/hadoop-sls hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager:

org.apache.hadoop.yarn.server.resourcemanager.webapp.TestRMWebApp

org.apache.hadoop.yarn.server.resourcemanager.scheduler.fair.TestFairScheduler

org.apache.hadoop.yarn.server.resourcemanager.webapp.TestNodesPage

The following test timeouts occurred in hadoop-tools/hadoop-sls hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager:

org.apache.hadoop.yarn.server.resourcemanager.TestRMHA

+1 contrib tests. The patch passed contrib unit tests.


Test results: <https://builds.apache.org/job/PreCommit-YARN-Build/5410/testReport/>

Findbugs warnings: <https://builds.apache.org/job/PreCommit-YARN-Build/5410/artifact/patchprocess/newPatchFindbugsWarningshadoop-yarn-server-resourcemanager.html>

Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/5410/console>


This message is automatically generated.

---

▼  Wangda Tan added a comment - 15/Oct/14 21:25

Attached a new patch contains [YARN-2496](#) and [YARN-2500](#) to kick jenkins to get result.

---

▼  Hadoop QA added a comment - 15/Oct/14 22:45

+1 overall. Here are the results of testing the latest attachment

<http://issues.apache.org/jira/secure/attachment/12675117/YARN-2496%2BYARN-2500.patch>  
against trunk revision f19771a.

+1 @author. The patch does not contain any @author tags.

+1 tests included. The patch appears to include 29 new or modified test files.

+1 javac. The applied patch does not increase the total number of javac compiler warnings.

+1 javadoc. There were no new javadoc warning messages.

+1 eclipse:eclipse. The patch built with eclipse:eclipse.

+1 findbugs. The patch does not introduce any new Findbugs (version 2.0.3) warnings.

+1 **release audit**. The applied patch does not increase the total number of release audit warnings.


+1 **core tests**. The patch passed unit tests in hadoop-tools/hadoop-sls hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager.

+1 **contrib tests**. The patch passed contrib unit tests.


Test results: <https://builds.apache.org/job/PreCommit-YARN-Build/5415/testReport/>

Console output: <https://builds.apache.org/job/PreCommit-YARN-Build/5415/console>

This message is automatically generated.

▼  Vinod Kumar Vavilapalli added a comment - 16/Oct/14 01:26

+1, looks good! Checking this in, let's make sure we follow up on the remaining items.

▼  Vinod Kumar Vavilapalli added a comment - 16/Oct/14 01:36

Committed this to trunk, branch-2 and branch-2.6. Thanks Wangda!

▼  Hudson added a comment - 16/Oct/14 01:42

FAILURE: Integrated in Hadoop-trunk-Commit #6269 (See <https://builds.apache.org/job/Hadoop-trunk-Commit/6269/>)

~~YARN-2496~~. Enhanced Capacity Scheduler to have basic support for allocating resources based on node-labels. Contributed by Wangda Tan. (vinodkv: rev f2ea555ac6c06a3f2f6559731f48711fff05d3f1)

- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/rmapp/TestRMAppTransitions.java
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestLeafQueue.java
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/fifo/FifoScheduler.java
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/CapacityScheduler.java
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/RMContext.java
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/TestAppManager.java
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/TestClientRMService.java
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestApplicationLimits.java
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/rmapp/attempt/RMAppAttemptImpl.java
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestQueueMappings.java
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/SchedulerUtils.java
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/webapp/TestRMWebApp.java
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/rmnode/RMNodeImpl.java
- hadoop-yarn-project/hadoop-yarn/dev-support/findbugs-exclude.xml
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/reservation/CapacitySchedulerPlanFollower.java
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/reservation/ReservationSystemTestUtil.java
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/Application.java
- hadoop-yarn-project/CHANGES.txt
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/ResourceManager.java
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/RMHAstTestBase.java
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestContainerAllocation.java
- hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/ParentQueue.java

- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/MockAM.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestParentQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/Queue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/ApplicationMasterService.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/AbstractCSQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestUtils.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestQueueParsing.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/rmapp/attempt/TestRMAppAttemptTransitions.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/TestWorkPreservingRMRestart.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestCapacityScheduler.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/fair/FairSchedulerTestBase.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/MockRM.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/TestApplicationACLs.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/ReservationQueue.java`
- `hadoop-tools/hadoop-sls/src/main/java/org/apache/hadoop/yarn/sls/scheduler/RMNodeWrapper.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/CapacitySchedulerConfiguration.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestCSQueueUtils.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/fair/TestFairScheduler.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/PlanQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestChildQueueOrder.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/MockNodes.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/RMServerUtils.java`
- `hadoop-tools/hadoop-sls/src/main/java/org/apache/hadoop/yarn/sls/nodemanager/NodeInfo.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/RMContextImpl.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/CSQueueUtils.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestReservationQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/TestSchedulerUtils.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/rmapp/RMAppImpl.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestReservations.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/fair/FSQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/CSQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/TestRMHA.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/LeafQueue.java`

- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/RMAppManager.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/rmnode/RMNode.java`

 **Hudson** added a comment - 16/Oct/14 11:39

FAILURE: Integrated in Hadoop-Yarn-trunk #713 (See <https://builds.apache.org/job/Hadoop-Yarn-trunk/713/>)

~~YARN-2496~~. Enhanced Capacity Scheduler to have basic support for allocating resources based on node-labels. Contributed by Wangda Tan. (vinodkv: rev f2ea555ac6c06a3f2f6559731f48711fff05d3f1)

- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/LeafQueue.java`
- `hadoop-yarn-project/CHANGES.txt`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/TestSchedulerUtils.java`
- `hadoop-tools/hadoop-sls/src/main/java/org/apache/hadoop/yarn/sls/scheduler/RMNodeWrapper.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/PlanQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/rmapp/attempt/TestRMAppAttemptTransitions.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestParentQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/Application.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestChildQueueOrder.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/RMContext.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/rmapp/TestRMAppTransitions.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/rmnode/RMNodeImpl.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/ReservationQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/reservation/ReservationSystemTestUtil.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestContainerAllocation.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/CapacityScheduler.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/TestApplicationACLs.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/ApplicationMasterService.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/rmapp/attempt/RMAppAttemptImpl.java`
- `hadoop-yarn-project/hadoop-yarn/dev-support/findbugs-exclude.xml`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/TestRMHA.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/rmnode/RMNode.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/MockRM.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/TestClientRMService.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestReservationQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/Queue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/reservation/CapacitySchedulerPlanFollower.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/AbstractCSQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestReservations.java`

- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/webapp/TestRMWebApp.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/TestWorkPreservingRMRestart.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestQueueMappings.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/MockAM.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestApplicationLimits.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/CSQueue.java`
- `hadoop-tools/hadoop-sls/src/main/java/org/apache/hadoop/yarn/sls/nodemanager/NodeInfo.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/fifo/FifoScheduler.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/rmapp/RMAppImpl.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/ParentQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestCSQueueUtils.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/CSQueueUtils.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestCapacityScheduler.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/RMHAstTestBase.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/TestAppManager.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/RMContextImpl.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestLeafQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/fair/FSQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestUtils.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestQueueParsing.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/fair/FairSchedulerTestBase.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/fair/TestFairScheduler.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/SchedulerUtils.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/CapacitySchedulerConfiguration.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/RMServerUtils.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/ResourceManager.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/RMAppManager.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/MockNodes.java`

▼  Hudson added a comment - 16/Oct/14 13:58

FAILURE: Integrated in Hadoop-Hdfs-trunk #1903 (See <https://builds.apache.org/job/Hadoop-Hdfs-trunk/1903/>)

[YARN-2496](#). Enhanced Capacity Scheduler to have basic support for allocating resources based on node-labels. Contributed by Wangda Tan. (vinodkv: rev f2ea555ac6c06a3f2f6559731f48711fff05d3f1)

- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/Application.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/reservation/ReservationSystemTestUtil.java`



- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestContainerAllocation.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/fair/FSQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/TestRMHA.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestQueueMappings.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/RMContextImpl.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/SchedulerUtils.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/MockAM.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestCapacityScheduler.java`
- `hadoop-yarn-project/CHANGES.txt`
- `hadoop-tools/hadoop-sls/src/main/java/org/apache/hadoop/yarn/sls/scheduler/RMNodeWrapper.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestApplicationLimits.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/fair/FairSchedulerTestBase.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/fair/TestFairScheduler.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/TestApplicationACLs.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/MockRM.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/rmnode/RMNode.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/CSQueueUtils.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestReservations.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/RMServerUtils.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/ParentQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestCSQueueUtils.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestUtils.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestReservationQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/TestClientRMService.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/TestAppManager.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/Queue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/webapp/TestRMWebApp.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/PlanQueue.java`
- `hadoop-tools/hadoop-sls/src/main/java/org/apache/hadoop/yarn/sls/nodemanager/NodeInfo.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/MockNodes.java`
- `hadoop-yarn-project/hadoop-yarn/dev-support/findbugs-exclude.xml`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/RMAppManager.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/fifo/FifoScheduler.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/AbstractCSQueue.java`

- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/CapacityScheduler.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/rmapp/TestRMAppTransitions.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/rmapp/attempt/TestRMAppAttemptTransitions.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/rmapp/attempt/RMAppAttemptImpl.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestLeafQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/CSQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/TestWorkPreservingRMRestart.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestChildQueueOrder.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/TestSchedulerUtils.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/ReservationQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/rmapp/RMAppImpl.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/rmnode/RMNodeImpl.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/ApplicationMasterService.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/RMContext.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/RMHATestBase.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/ResourceManager.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestQueueParsing.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/LeafQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/reservation/CapacitySchedulerPlanFollower.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestParentQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/CapacitySchedulerConfiguration.java`

▼  Hudson added a comment - 16/Oct/14 13:59

FAILURE: Integrated in Hadoop-Mapreduce-trunk #1928 (See <https://builds.apache.org/job/Hadoop-Mapreduce-trunk/1928/>) [YARN-2496](#). Enhanced Capacity Scheduler to have basic support for allocating resources based on node-labels. Contributed by Wangda Tan. (vinodkv: rev f2ea555ac6c06a3f2f6559731f48711fff05d3f1)

- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/fifo/FifoScheduler.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestQueueMappings.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestApplicationLimits.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/AbstractCSQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/RMContext.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/MockAM.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/TestApplicationACLs.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/rmapp/RMAppImpl.java`



- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/ApplicationMasterService.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/CSQueue.java`
- `hadoop-yarn-project/hadoop-yarn/dev-support/findbugs-exclude.xml`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/MockNodes.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestReservationQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/rmapp/attempt/TestRMAppAttemptTransitions.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/rmnode/RMNode.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/ResourceManager.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/RMContextImpl.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/PlanQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/MockRM.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/SchedulerUtils.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/rmnode/RMNodeImpl.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestCSQueueUtils.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/RMAppManager.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestContainerAllocation.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/reservation/ReservationSystemTestUtil.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/CSQueueUtils.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/ReservationQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/fair/FSQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestReservations.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestLeafQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/RMServerUtils.java`
- `hadoop-tools/hadoop-sls/src/main/java/org/apache/hadoop/yarn/sls/scheduler/RMNodeWrapper.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/webapp/TestRMWebApp.java`
- `hadoop-yarn-project/CHANGES.txt`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/ParentQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestParentQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/fair/FairSchedulerTestBase.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/Application.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/TestRMHA.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/fair/TestFairScheduler.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestUtils.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/rmapp/TestRMAppTransitions.java`

- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcecemanager/src/test/java/org/apache/hadoop/yarn/server/resourcecemanager/TestClientRMService.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcecemanager/src/main/java/org/apache/hadoop/yarn/server/resourcecemanager/scheduler/Queue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcecemanager/src/test/java/org/apache/hadoop/yarn/server/resourcecemanager/TestAppManager.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcecemanager/src/test/java/org/apache/hadoop/yarn/server/resourcecemanager/scheduler/TestSchedulerUtils.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcecemanager/src/main/java/org/apache/hadoop/yarn/server/resourcecemanager/rmapp/attempt/RMAppAttemptImpl.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcecemanager/src/main/java/org/apache/hadoop/yarn/server/resourcecemanager/reservation/CapacitySchedulerPlanFollower.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcecemanager/src/test/java/org/apache/hadoop/yarn/server/resourcecemanager/scheduler/capacity/TestQueueParsing.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcecemanager/src/test/java/org/apache/hadoop/yarn/server/resourcecemanager/TestWorkPreservingRMRestart.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcecemanager/src/test/java/org/apache/hadoop/yarn/server/resourcecemanager/scheduler/capacity/TestChildQueueOrder.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcecemanager/src/test/java/org/apache/hadoop/yarn/server/resourcecemanager/scheduler/capacity/TestCapacityScheduler.java`
- `hadoop-tools/hadoop-sls/src/main/java/org/apache/hadoop/yarn/sls/nodemanager/NodeInfo.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcecemanager/src/main/java/org/apache/hadoop/yarn/server/resourcecemanager/scheduler/capacity/LeafQueue.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcecemanager/src/main/java/org/apache/hadoop/yarn/server/resourcecemanager/scheduler/capacity/CapacitySchedulerConfiguration.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcecemanager/src/main/java/org/apache/hadoop/yarn/server/resourcecemanager/scheduler/capacity/CapacityScheduler.java`
- `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcecemanager/src/test/java/org/apache/hadoop/yarn/server/resourcecemanager/RMHATestBase.java`

---

**People**

Assignee:



Wangda Tan

Reporter:



Wangda Tan

Votes:



0 Vote for this issue

Watchers:



10 Start watching this issue

---

**Dates**

Created:

03/Sep/14 08:55

Updated:

01/Dec/14 03:08

Resolved:

16/Oct/14 01:36