

Public signup for this instance is **disabled**. Go to our [Self serve sign up page](#) to request an account.



Hadoop YARN / [YARN-9698 \[Umbrella\] Tools to help migration from Fair Scheduler to Capacity Scheduler](#) / [YARN-9879](#)

# Allow multiple leaf queues with the same name in CapacityScheduler

Details

Type:	Sub-task	Status:	RESOLVED
Priority:	Major	Resolution:	Fixed
Affects Version/s:	None	Fix Version/s:	3.3.0
Component/s:	None		
Labels:	<div>fs2cs</div>		
Target Version/s:	3.3.0		
Hadoop Flags:	Reviewed		

Description

Currently the leaf queue's name must be unique regardless of its position in the queue hierarchy.

Design doc and first proposal is being made, I'll attach it as soon as it's done.

Attachments

<a href="#">CSQueue.getQueueUsage.txt</a>	4 kB	31/Jan/20 13:55
<a href="#">DesignDoc_v1.pdf</a>	68 kB	21/Nov/19 14:47
<a href="#">YARN-9879.014.patch</a>	262 kB	17/Mar/20 13:00
<a href="#">YARN-9879.015.patch</a>	263 kB	24/Mar/20 19:03
<a href="#">YARN-9879.015.patch</a>	263 kB	24/Mar/20 11:38
<a href="#">YARN-9879.POC001.patch</a>	36 kB	21/Jan/20 16:28
<a href="#">YARN-9879.POC002.patch</a>	83 kB	31/Jan/20 13:50
<a href="#">YARN-9879.POC003.patch</a>	141 kB	14/Feb/20 03:13
<a href="#">YARN-9879.POC004.patch</a>	166 kB	20/Feb/20 10:22
<a href="#">YARN-9879.POC005.patch</a>	167 kB	20/Feb/20 16:20
<a href="#">YARN-9879.POC006.patch</a>	174 kB	27/Feb/20 17:17
<a href="#">YARN-9879.POC007.patch</a>	183 kB	28/Feb/20 15:04
<a href="#">YARN-9879.POC008.patch</a>	197 kB	29/Feb/20 21:48
<a href="#">YARN-9879.POC009.patch</a>	207 kB	02/Mar/20 20:39
<a href="#">YARN-9879.POC010.patch</a>	228 kB	05/Mar/20 00:22
<a href="#">YARN-9879.POC011.patch</a>	231 kB	05/Mar/20 22:37
<a href="#">YARN-9879.POC012.patch</a>	258 kB	11/Mar/20 02:06
<a href="#">YARN-9879.POC013.patch</a>	262 kB	13/Mar/20 03:43

Issue Links

breaks		
<a href="#">MAPREDUCE-7269</a> TestNetworkedJob fails		RESOLVED
<a href="#">YARN-10247</a> Application priority queue ACLs are not respected		RESOLVED
Dependent		
<a href="#">YARN-10108</a> FS-CS converter: nestedUserQueue with default rule results in invalid queue mapping		RESOLVED
is duplicated by		
<a href="#">YARN-9772</a> CapacitySchedulerQueueManager has incorrect list of queues		RESOLVED
<a href="#">YARN-9925</a> CapacitySchedulerQueueManager allows unsupported Queue hierarchy		RESOLVED

## Activity



Peter Bacsko added a comment - 12/Dec/19 12:43

Note: after the implementation is complete, remove this part from the converter tool (<https://github.com/apache/hadoop/blob/trunk/hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/fair/converter/FSQueueConverter.java#L83-L94>)

```
private final Set<String> leafQueueNames;
...
public void convertQueueHierarchy(FSQueue queue) {
    List<FSQueue> children = queue.getChildQueues();
    final String queueName = queue.getName();

    if (queue instanceof FSLeafQueue) {
        String shortName = getQueueShortName(queueName);
        if (!leafQueueNames.add(shortName)) {
            throw new ConversionException(
                "Leaf queues must be unique, "
                + shortName + " is defined at least twice");
        }
    }
    ...
}
```

Wilfred Spiegelenburg added a comment - 16/Dec/19 13:32 - edited

I have read through the design document and was wondering if we cannot take a far simpler approach.

If we simply relax the rule that the leaf queue must be unique in the system in favour of the fact that a queue must be unique based on the full queue path. This does not break existing configurations as the unique leaf queue is also unique when you take into account the whole path. That means there is nothing for the current clusters that needs to change. Internally the scheduler does have to change to make sure that all references use the queue path. This will require a lot of changes throughout the scheduler when you look up a queue and the way we store the reference if it is not directly to the leaf queue.

The only other point that we need to correctly handle this now is on the submit side. This must be handled backward compatible. We have two cases to handle: just a queue name and a queue path. I'll discuss updating the configuration is later.

1. When an application is submitted with just a queue name (not a path) we expect that the name is a unique leaf queue name. If that queue does not exist or is not uniquely identifiable we reject the application submission. Resolution of the real leaf queue follows the same steps as it does now. The queue name in the end is converted to the correct leaf queue identified by the a path. For existing configurations nothing has changed. Internally we hide all the changes.
2. When the submit has a queue path (fully qualified or not) we check that the queue exists based on that path. If the leaf queue is not defined using its path the application submission is rejected.

In the case that the scheduler has a non unique leaf queue name submitting to those queues can only be done by using their paths. There is nothing that needs to be configured to switch this behaviour on or off.

The important part is applying a new configuration. If the configuration adds a leaf queue that is not unique the configuration update currently is rejected. With this change we would allow that config to become active. This **could** break existing applications when they try to submit to the leaf queue that is no longer unique.

We should at least log and warn clearly in the response of the update. Maybe even show it in the UI or we could ask for a confirmation. The first update that adds a non unique queue to the configuration should always fail complaining loudly. It should then keep warning the user and rejecting the update unless a confirmation flag is set to force the update through. After the first update that would not be needed anymore.

Reading a config from a file or store which is used to initialise the scheduler should not trigger such behaviour. We still should show a warning in the logs to make sure it is not lost.

What do you think about this approach?

Peter Bacsko added a comment - 07/Jan/20 14:54 - edited

wilfreds based on your suggestion, here's what I came up with:

We can still maintain the HashMap with queueName->CSQueue, however we'd use two levels:

1. Leaf queue -> full path
2. Full path -> CSQueue object

We additionally need an extra map which tells whether a leaf queue is unique.


So after some thinking, this is the semi-pseudocode that could possibly do the job:

```
Map<String, CSQueue> fullPathQueues;
Map<String, String> leafToFullPath;
Map<String, Boolean> leafUnique;

public CSQueue getQueue(String queueName) {
    if (fullPathName(queueName)) {
        return fullPathQueues.get(queueName);
    } else {
        if (leafUnique.get(queueName)) {
```


```
String fullName = leafToFullPath.get(queueName);
return fullPathQueues.get(fullName);
} else {
    throw new YarnException(queueName + " is not unique");
}
}
}
```

Obviously methods like `addQueue()`, `removeQueue()` should be updated too.

▼  [Peter Bacsko](#) added a comment - 07/Jan/20 15:45

Alternatively, we can have `Map<String, CSQueue> fullToCSQueue` and `Map<String, CSQueue> leafToCSQueue`, so we can avoid the double lookup (not that it's really that expensive).

Also it's probably better to have `Map<String, Integer>` to check whether a leaf is unique. When we add/remove a queue, we increase/decrease a counter, so upon removal, we know whether it has become unique or not.

▼  [Wangda Tan](#) added a comment - 07/Jan/20 18:46

[pbacsko](#), thanks for working on the design.

In general, I agree with what [wilfreds](#) mentioned: we should try to avoid change RPC protocols, instead we just change internal logic to make sure multiple queues can be handled.

To me there're two major parts:

1) Whatever logic inside CS to allow multiple queue names. Either solution mentioned in the comment: <https://issues.apache.org/jira/browse/YARN-9879?focusedCommentId=17009845&page=com.atlassian.jira.plugin.system.issuetabpanels:comment-tabpanel#comment-17009845> should be fine. And I expect the lookup of queue name (not queue path) should only be called when submit application.

And once application is submitted to CS, internal to CS, we should make sure we use queue path instead of queue name at all other places. Otherwise we will complicate other logics.

2) When submit app, the scheduler going to accept/reject app based on the uniqueness of queue name or path specified. The core part need to be changed is inside `RMAAppManager`:

```
if (!isRecovery && YarnConfiguration.isAclEnabled(conf)) {
    if (scheduler instanceof CapacityScheduler) {
        String queueName = submissionContext.getQueue();
        String appName = submissionContext.getApplicationName();
        CSQueue csqueue = ((CapacityScheduler) scheduler).getQueue(queueName);
```

Instead of using `scheduler.getQueue`, we may need to consider to add a method like `getAppSubmissionQueue()` to get a queue based on path or name, and after that, we will put normalized `queue_path` back to submission context of application to make sure in the future inside scheduler we all refer to queue path.

For the comment from [wilfreds](#):

The important part is applying a new configuration. If the configuration adds a leaf queue that is not unique the configuration update currently is rejected. With this change we would allow that config to become active. This **could** break existing applications when they try to submit to the leaf queue that is no longer unique.

I personally think it is not a big deal if application reject reasons from RM can clearly guide users to use full qualified queue path when duplicated queue names exists. It is like if a team has only one Peter we can use the first name only otherwise we will add last name to avoid confusion. It isn't counter-intuitive to me.

Also, we need to handle queue mapping for queue-path instead of queue name also, I didn't see it from the design doc or I missed it.

▼  [Wilfred Spiegelenburg](#) added a comment - 08/Jan/20 00:10 - edited

Thank you [leftnoteasy](#) for the comments.

And once application is submitted to CS, internal to CS, we should make sure we use queue path instead of queue name at all other places. Otherwise we will complicate other logics.

I agree that is what I had in mind too. Make it as simple as possible inside the scheduler and that is to use just the full path internally.

For the configuration change: I do not think it is a problem and we can just accept the change. To be fair to the administrator we should show a message when the configuration is loaded or changed and the leaf queues are not unique (any more). However that is probably as far as we need to go.

Instead of using `scheduler.getQueue`, we may need to consider to add a method like `getAppSubmissionQueue()` to get a queue based on path or name, and after that, we will put normalized `queue_path` back to submission context of application to make sure in the future inside scheduler we all refer to queue path.

The FS already does something like this already because it uses a placement rule in all cases. We should leverage a similar mechanism in the CS. We pass the queue from the submission into the queue placement which handles the full path or not. In both cases it just passes back the queue object

which will be using the full path. If the queue is not found or the queue name is not unique it fails as per normal. The returned queue info is updated in the app and submission context.

Far simpler than putting the burden on the core scheduler. It is all hidden in the placement of the app into the queue using the placement engine.

I did not mention queue mapping in my design. Queue mapping itself I thought did not need to change. We already calculate the parent queue in the rules if I am correct so the only change would be the return value. We do all internal handling for queues with the full queue path so it is a logical change. Using the placement rule for the qualified or not qualified mapping does require some changes in that area.

I might have forgotten to mention other bits and pieces like the cli or flow on effects on the UI but that needs to be assessed when we have a design we agree on. There will be more jiras needed to fix separate parts when the change is made to the core.

▼ Peter Bacsko added a comment - 08/Jan/20 08:18

I just mentioned the mapping because based on my (admittedly limited) knowledge, it's the heart of CS when it comes to managing the queues.

It's handled inside <https://github.com/apache/hadoop/blob/trunk/hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/CapacitySchedulerQueueManager.java>.

So I was under the impression that this part needs to be changed, I can be wrong though.

▼ Gergely Pollák added a comment - 09/Jan/20 16:07

Thank you for your feedbacks [wilfreds](#), [pbacsko](#), [leftnoteasy](#)!

The important part is applying a new configuration. If the configuration adds a leaf queue that is not unique the configuration update currently is rejected. With this change we would allow that config to become active. This **could** break existing applications when they try to submit to the leaf queue that is no longer unique.

I think this can be a major issue. In a scenario when the user have let's say a job which runs daily, and it's been working like that for years, can simply break, because an other user, totally unrelated to his team or even department creates a queue with the same name. **The suddenly this application will start failing, and even if we add logs which explicitly tell what is the reason behind the application rejection, user one might not even notice it started failing. (We shouldn't assume everyone has proper monitoring and warning systems). So technically any user who can create a queue, can disable an other user's application if it is started by using single queue name reference. And it concerns me a bit.**

However [adam.antal](#) had an idea to even fix this issue, and then we can move forwards with Wilfred's suggestion:

We should make it possible to flag QUEUES for using full queue name reference, and these flags should mean ALL queues under the flagged queue can ONLY be referenced by full queue name. For regular queues (non-flagged ones), we would still enforce the unique leaf queue policy, while newer or migrating users could stick to the full queue reference. This proposal can also help gradual migration, for older CS users buy slowly flagging queues in which the applications are already started by queue name.

What do you think of this idea?

▼ Wangda Tan added a comment - 09/Jan/20 22:44

Thanks [shuzirra](#), I think adding a flag (suggestion from [adam.antal](#)) will prevent admin to change it accidentally, but it is hard to be understand (thinking about a regular Hadoop user). And we need to maintain it in a long run.

So instead, I would like to allow user to make changes but fail the application submission with a clear message (like you cannot submit the application because there're multiple queue with the name XYZ, you can make change to use the full qualified queue name or remove/rename duplicated queues, etc.). If admin want to regret and make changes back, they can easily do that.

▼ Szilard Nemeth added a comment - 10/Jan/20 20:03 - edited

Thank you for your feedbacks [wilfreds](#), [pbacsko](#), [leftnoteasy](#) and thans [shuzirra](#) for moving this forward!

Chiming in a bit late, but I would like to give my opinion as well. Apologies for being so lengthy, but I wanted to quote and reply some parts of the comments had been given so far.

Please note that some parts of what I'm saying here is taking the FS to CS migration into account.

#### 1. Quoting from Wilfred's comment:

A.

If we simply relax the rule that the leaf queue must be unique in the system in favour of the fact that a queue must be unique based on the full queue path. This does not break existing configurations as the unique leaf queue is also unique when you take into account the whole path. That means there is nothing for the current clusters that needs to change. Internally the scheduler does have to change to make sure that all references use the queue path. This will require a lot of changes throughout the scheduler when you look up a queue and the way we store the reference if it is not directly to the leaf queue.

I do agree with this, existing configurations does not break. However, allowing non-unique leaf queues can make the CS behave weirdly and inconsistently as queues are being added later on, as [shuzirra](#) described this lately in his comment. More on this matter later.

B.

When an application is submitted with just a queue name (not a path) we expect that the name is a unique leaf queue name. If that queue does not exist or is not uniquely identifiable we reject the application submission. Resolution of the real leaf queue follows the same steps as it does now. The queue name in the end is converted to the correct leaf queue identified by the a path. For existing configurations nothing has changed. Internally we hide all the changes.

AND

C.

The important part is applying a new configuration. If the configuration adds a leaf queue that is not unique the configuration update currently is rejected. With this change we would allow that config to become active. This could break existing applications when they try to submit to the leaf queue that is no longer unique.

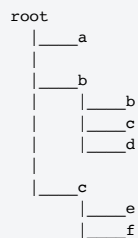
We should at least log and warn clearly in the response of the update. Maybe even show it in the UI or we could ask for a confirmation. The first update that adds a non unique queue to the configuration should always fail complaining loudly. It should then keep warning the user and rejecting the update unless a confirmation flag is set to force the update through. After the first update that would not be needed anymore. Reading a config from a file or store which is used to initialise the scheduler should not trigger such behaviour. We still should show a warning in the logs to make sure it is not lost.

About what you said: "This could break existing applications when they try to submit to the leaf queue that is no longer unique."

Again, this is really problematic and error-prone, as detailed by [shuzirra](#) in his comment above.

We discussed this together with Gergo and I'm trying to give more weight on how tricky this case can get and what I'm doing now is just reiterate on [Gergo's comment](#)

Imagine the following queue setup:



If someone is adding root.c.b, queue 'b' is non-unique anymore without even knowing or taking care of root.b.b is the other queue 'b'.

Even if we log and warn in the response of the update, adding the root.c.b queue can break the app submitted to queue root.b.b.

I can imagine many real-world scenarios where users were submitting apps to a queue (root.b.b) with just the leaf queue name ('b') and their app submission would get suddenly rejected since another queue do exist with the same name. Just consider a user-group mapping and two same usernames exist under different organizations/user groups and we can be in the situation what I depicted. I also agree to Gergo that we can't expect or assume every user and apps submission is guarded by monitoring or warning systems.

I think it's definitely not a good idea to break existing users and app submissions with this change.

## 2. Quoting from Wangda's comment:

I personally think it is not a big deal if application reject reasons from RM can clearly guide users to use full qualified queue path when duplicated queue names exists. It is like if a team has only one Peter we can use the first name only otherwise we will add last name to avoid confusion. It isn't counter-intuitive to me.

How can we know it's not a big deal? As we are not fully familiar how users are submitting apps: They can have scripts (or even many scripts), cron jobs, other unknown setup and whatnot that triggers app submissions. We can't and shouldn't do any non-backward compatible changes. For me, this approach is not an acceptable one as described with my ASCII art + description in detail.

## 3. Quoting from Gergo's comment

A.

I think this can be a major issue. In a scenario when the user have let's say a job which runs daily, and it's been working like that for years, can simply break, because an other user, totally unrelated to his team or even department creates a queue with the same name. The suddenly this application will start failing, and even if we add logs which explicitly tell what is the reason behind the application rejection, user one might not even notice it started failing. (We shouldn't assume everyone has proper monitoring and warning systems). So technically any user who can create a queue, can disable an other user's application if it is started by using single queue name reference. And it concerns me a bit.

Agree with this, the whole point of my comment was based around this fact Gergo shared.

B.

However Adam Antal had an idea to even fix this issue, and then we can move forwards with Wilfred's suggestion:

We should make it possible to flag QUEUES for using full queue name reference, and these flags should mean ALL queues under the flagged queue can ONLY be referenced by full queue name. For regular queues (non-flagged ones), we would still enforce the unique leaf queue



policy, while newer or migrating users could stick to the full queue reference. This proposal can also help gradual migration, for older CS users buy slowly flagging queues in which the applications are already started by queue name.

#### Some things I have in mind, so listing the **ADVANTAGES** and some properties of this approach:

1. First and foremost, duplicate queue names cannot be added under unflagged queues.
  2. Using the department / user analogy here. We don't have the problem anymore that some other department adds a user and this user becomes non-unique as other department has the same username in it, so app submissions can work well for all departments / users. As the new department added a user (root.c.b) has their subqueues flagged, app submissions should only work with full queue paths here and at the same time, the another department had the same user (root.b.b), app submission will still work well as all queues under root.c should be referenced only referenced by full queue path.
  3. Customers can gradually move queues to use full queue path, as much as they want.
  4. FS to CS migration tool should add the flag to the root queue if moved from FS.
4. Quoting from Wangda's comment:

Thanks Gergely Pollak, I think adding a flag (suggestion from Adam Antal) will prevent admin to change it accidentally, but it is hard to be understand (thinking about a regular Hadoop user). And we need to maintain it in a long run.

I'm not sure why this is harder to understand than regular queue and scheduler config parameters, which can be overly complex to setup. With this approach, at least we would give something in our user's hands: They can use the flag to move queues under to the new way to address queues (queue path) gradually. Without the flag-based approach, we don't give users anything but we require them to change their submission code everytime a leaf queue is added that is already in the hierarchy with the same name. If we don't consider this as a hard requirement, apps would fail randomly, from the perspective of the users.

#### As a summary:

**Please note that I did not dive into the CS code to decide what is the development cost of such a flag-based solution, this is up to [shuzirra](#) to give some estimates as the jira is on his name.**

**However, I wanted to state and share my thoughts: Rejecting apps on submission time that worked before and won't work afterwards seems like very far from an acceptable solution.**

Wangda Tan added a comment - 14/Jan/20 21:18 - edited

[snemeth](#), most of the explanation looks reasonable to me. Regarding how to prevent breaking existing CS queue contract. Instead of adding a flag to each queue, I suggest to have a global config in CS about allow duplicated leaf queue name or not.

Why I'm opposed to add the flag to each queue?

To me, use a queue name, or queue path is an intuitive choice of a user (not admin). If the queue name had duplicates, it should fail and give you the right reason.

If everybody think we should not implicitly change the CS behavior to allow duplicate-named leaf queues, a top-level CS config should be sufficient (like `..<cs-prefix>.duplicated-queue-names.allowed`), and clearly document it may cause existing app failures. This won't add any burden for user to understand, and it is also relatively easy for admin to understand. Anything config added to the queue hierarchy seems a bit tricky. (Like admin has to think about how is the queue override looks like, etc.). And for the auto-created queue case it is not obvious to add such configs, etc. My big lesson learned is we should add as less knobs as we could, too many knobs will increase our support areas a lot and make code hard to be maintained.

Wilfred Spiegelenburg added a comment - 15/Jan/20 07:45 - edited

A per queue flag looks very strange. I am also not sure it will help or add anything on top of having a global flag that just prevents the config change. Summarising the proposed solution: add a flag that prevents the admin from adding non unique leaf queue names and thus fail the config change when he/she tries

The behaviour inside the scheduler must all be based on the full queue paths anyway. You cannot have one queue being addressed by the leaf name and the other by the path. The code complexity to do that would be enormous and lead to unsupportable code. That means that after the placement rule(s) are run and the app is placed everything must be based on a full path.

Placement rules throw up a totally different issue here. When we use placement rules we have one of two possible cases:

- the rule generates a queue name and a parent queue name, i.e. a path
- the rule generates just a leaf queue name

Which means that the rule can generate a leaf queue anywhere in the hierarchy without specifying a hierarchy. So no parent is set by the rule but the leaf queue generated could be located below a parent. With that last possibility we have the extra complexity in that the rules are not behaving consistently.

Example:

Two CS definitions to compare both allow queue creation and overwrite of the submitted queue:

1. queues: root.parent.wilfred
2. queues: root

mapping rule defined: `u:%user:%user`

1) user submitting the app is `wilfred` queue given on submission is default


In CS config 1 we submit to the `root.parent.wilfred` queue while in the second CS config we submit to `root.wilfred` queue.

2) user submitting the app is `peter` queue given on submission is default

In both CS configs we submit to the `root.peter` queue.

With different config at the CS level but for the same rule we place the app in a sub queue sometimes but not the other, that is inconsistent.


I think rules even need to start taking this flag into account to preserve this inconsistent behaviour.

▼  [Wangda Tan](#) added a comment - 15/Jan/20 16:14

[wilfreds](#), I agree with,


The behaviour inside the scheduler must all be based on the full queue paths anyway.

I also agree that we need to carefully think about queue mapping and queue path. I would suggest moving queue mapping related changes to a different Jira to avoid putting two big patches together. (If it already considered both scenario we can keep it here).

▼  [Gergely Pollák](#) added a comment - 21/Jan/20 16:34

Uploaded the first POC patch. This version CAN run jobs by their short AND their full name, currently the leaf uniqueness constraint is in place, but the data store is prepared for leaf collisions. Currently working on testing and if nothing seems to be broken, I can remove the uniqueness constraint from leaf queues. Also I can add the feature control flag, as discussed earlier.


The biggest risk I see is the modification in AbstractCSQueue where I've changed the `getQueueName` to return the full name of the queue. Based on my analysis this method is mostly called to get the queue's string identifier, and as it was mentioned earlier we should use full queue names for queue identification. So I need to carefully check every call of this method if it breaks anything. But a quick smoke test was successful, I was able to start a job using only the leaf queue name as well as using the full queue name.

▼  [Wangda Tan](#) added a comment - 21/Jan/20 19:48

[shuzirra](#), I think we should not change semantics of `GetQueueName` of `AbstractCSQueue` to avoid the change of API. (We should keep REST API related to queues unchanged otherwise it will be an incompatible change).

Instead of changing `GetQueueName`, you should check all callers of the `GetQueueName` first. And there's already a `GetQueuePath`, you can leverage that.

I briefly checked `GetQueueName` usages, there're 155 of them in production code. Most of them are just for logging purposes ("org.apache.hadoop.yarn.server.resourcemanager.scheduler.activities.\*" should be considered as logging also). It may take a few hours to identify and change everything, but manually change `GetQueueName` to `GetQueuePath` case-by-case sounds like a safer option to me.

▼  [Wilfred Spiegelenburg](#) added a comment - 22/Jan/20 10:01 - edited

I agree, `getQueueName()` should stay as is. We have a `getQueuePath()` already. Every `CSQueue` can already return both. We should change all non external facing calls getting the name of a queue to the path version. The only calls that can stay are the ones that provide their data in an externally viewable form (REST, UI or IPC) as to not break compatibility.

I also do not see why we would need the ambiguous queue list. The queue is always unique when a path is used. It does not matter if the current leaf queue name uniqueness is enforced or not.

Everything can always be found by its path. If I do not have a path I expect leaf queue uniqueness and can find the queue by just checking the part after the last *dot* in the path.

i.e.

- queue paths defined as: root.parent.child1  
child queue unique flag is set  
find a queue with name: **child1** (no dots, expect leaf queue uniqueness) -> returns the queue correctly
- add a queue defined as: root.otherparent.child1  
child queue unique flag is not set, allowed  
find a queue with name: **child1** (no dots, expect leaf queue uniqueness) -> returns an error

Internally we just store everything using the path, that would remove the whole keeping things in sync and makes the code consistent when combined with using the path everywhere internally

▼  [Gergely Pollák](#) added a comment - 22/Jan/20 17:00 - edited

Thank you for your feedback [leftnoteasy](#) and [wilfreds](#). Originally I tried to keep the `getQueueName`'s behavior, but as I started to investigate it's behavior I've realized we MUST change the way it works.

First let's start with a simple question: What is the purpose of the queue's name? Why does it have one, what do we want to use it for? (Ok these are actually 3 questions)

As I see in the code the queue name's main purpose is to IDENTIFY a queue, and not just some nice display string. This means the name MUST identify uniquely the queue. Queues are looked up by their name, hence it must be unique or all those references can break. So this is the reason I changed it's behavior to return a unique identifier (the queue's path). Obviously I must check if it breaks anything, and fix it, but allowing multiple leaf queues with the same name is inherently a breaking change. I just try to minimize the impact to change the reference internally to full name everywhere (as you both suggested earlier).

About the API breaking. If we have an API which provides us with a queue name, and currently it is a short name, then anyone who uses it to reference to the queue by the provided name will fail in the case of name duplicates. If we return the full name of the queue, then it will still work for them, unless they build on the fact it is just a short name. As long as the queue name is used for queue identification, and not for string operations, it shouldn't cause any problem. Other cases must be identified.

This is why I ended up with this approach. This way we change the queue naming once and for all to use full names, and we adjust services which would fail on this change. But we cannot keep the short queue name as reference and have multiple queues with the same name, it's just impossible. This patch will already introduce some changes which can cause issues in already working systems and it might be better to do all invasive changes at once.

I can use the `getQueuePath` (almost) everywhere where we currently using `getQueueName`, but the result would be the same, with some severe inconsistencies: Using short names would result you being able to get the name of a queue, but you wouldn't be able to get your queue by that very same name from the queue manager. This is just confusing, inconsistent, and not maintainable in my opinion. The `quemanager.get(queue.getQueueName())` call can result in NULL or error! (when the queue name is not unique) This is not good practice in my opinion.

We need the ambiguous queue list, because we provide a remove method, which can result in a previously ambiguous name becoming ambiguous, and it's much faster to get it from a hashmap  $O(1)$ , and then check the size of the Set  $O(1)$ , instead of looking through all queues to see if the collision have been resolved  $O(n)$ .

The short name map has been introduced for the very same reason, when we look up a queue, we just look it up in 2 HashMaps  $2 \times O(1)$ , instead of iterating through all queue names and splicing the last part for short name  $O(n)$ .

So all in all, I've sacrificed some memory space for a drastic speed increase.  $O(n)$  vs  $O(1)$  might not seem a huge improvement in the case of a few queues, but considering the queue parse method will make a get call for each queue to check if it is already present in the store, we have a complexity of  $O(n^2)$ , which IS something to think about.

Please help me to think this through one more time with taking my reasons into consideration, thank you.

Wangda Tan added a comment - 23/Jan/20 01:16

Thanks [shuzirra](#), [wilfreds](#) for sharing your thoughts!

1) Regarding change semantics of `GetQueueName()` to return full qualified queue name v.s. use `GetQueuePath`:

If we decide to go the first route, we need to remove usages of `AbstractCSQueue.GetQueuePath` (which has 128 usages), and add a `GetShortQueueName` in some places. So to me, there are no significant differences to just change internal CS usages to use `GetQueuePath()`.

2) No matter which way we decided to go, I think we should make sure that:

API compatibility, this is critical since I assume there're lots of monitoring framework, JMX metrics, etc. based on this. If we upgrade an existing CS-based cluster, they should expect the same result. Please refer to API compatibility: <https://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/Compatibility.html>

Internal usage of `GetQueuePath` (or `GetShortQueueName` if we choose proposed approach). And externally, we should make sure we can get a queue by short name, or long name. I want to make sure we only check short name / long name in external call (like submit app to specified queue), and in all other places, we use the full queue path to operate. I think introducing a new `CSQueueStore` sounds good, but I recommend to add a separate method to `CSQueueStore` to check both short/long names and make it used by external callers only (And in contrast, internal CS method should check only one `HashMap` instead of two). We can review details of `CSQueueStore` separately.

Peter Bacsko added a comment - 24/Jan/20 06:21 - edited

Hey folks, I can see that a lengthy conversation is already going on, but I'll try to keep my one short.

Regarding `getQueueName()` / `getQueuePath()`, it's up to you to decide, I don't have enough context right now. I'm trying to be constructive from code readability standpoint.

Three things that stand out to me are the following:

#1

```
private final Map<String, Set<String>> ambiguousShortNames = new HashMap<>();
```

My question to [shuzirra](#) is: do we need to keep track of what queues a short name is mapped to? Do we use this information anywhere? Because if we use it as a counter, then it's simply much easier to have a

```
private final Map<String, Integer> leafCount = new HashMap<>();
```

And quite obviously you don't have ambiguity if `leafCount == 1`.

Because of this, the `addShortNameMapping()` is already a bit hard to grasp.

#2 I would synchronize the public method `add()`, not the private method.

To show what I was thinking of, here's how I'd code add/remove:

```
// Keep as it as
public synchronized void add(CSQueue queue) {
    String fullName = queue.getQueueName();
    String shortName = queue.getQueueShortName();

    fullNameQueues.put(fullName, queue);
    if (queue instanceof LeafQueue) {
        addShortNameMapping(shortName, fullName);
    }
}

private void addShortNameMapping(String shortName, String fullName) {
    // initialize if necessary
```



```
leafCount.computeIfAbsent(shortName, v -> 0);

if (leafCount.computeIfPresent(shortName, (k,v) -> v + 1) > 1) {
    LOG.warn("Multiple mapping for queue {}", shortName);
} else {
    shortNameToFullName.put(shortName, fullName);
}
}
```

#3 In get ( ) is important to check ambiguous mappings, so an exception must be thrown if leafCount > 1.

▼ Gergely Pollák added a comment - 24/Jan/20 15:28

As per our discussion with wilfreds, snemeth, sunilg, we will be making the behaviour of the getQueueName configurable (by default the getQueueName will return only the leaf name, but it can be changed to return the full path), and will do the name changes as follows:

- External facing APIs will always use the getQueueName
- Internally we will use the full name to store / reference the queues, and we will update the getQueueName method to getQueuePath, to make sure we are working with fully qualified name

This way we minimize the impact for user, but both CS and migrating FS users can use their external monitoring tools as they used to.

▼ Gergely Pollák added a comment - 31/Jan/20 13:51

Uploading next POC, changing to patch available to run tests, there is still a long way to go, in this patch I've replaced the trivial getQueueName->getQueuePath replacements, and collected all the places I still need to check for deeper analysis.

▼ Hadoop QA added a comment - 31/Jan/20 14:38

-1 overall

Vote	Subsystem	Runtime	Comment
0	reexec	0m 0s	Docker mode activated.
-1	patch	0m 8s	YARN-9879 does not apply to trunk. <a href="#">Rebase required?</a> <a href="#">Wrong Branch?</a> See <a href="https://wiki.apache.org/hadoop/HowToContribute">https://wiki.apache.org/hadoop/HowToContribute</a> for help.

Subsystem	Report/Notes
JIRA Issue	<a href="#">YARN-9879</a>
Console output	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25483/console">https://builds.apache.org/job/PreCommit-YARN-Build/25483/console</a>
Powered by	Apache Yetus 0.8.0 <a href="http://yetus.apache.org">http://yetus.apache.org</a>

This message was automatically generated.

▼ Peter Bacsko added a comment - 03/Feb/20 12:00

The latest build picked up CSQueue.getQueueUsage.txt. shuzirra you might want to re-upload the patch again.

▼ Gergely Pollák added a comment - 14/Feb/20 03:24

In this latest patch I've checked all CSQueue.getQueueName calls, and replaced them with getQueuePath, also checked how those values were used, and changed the code if it was necessary. The biggest change was around the placement rules, but I think I've managed to solve the issues with minimal impact and code changes.

The problem with placement rules is they can dynamically create colliding queue names, and at configuration time it is impossible to determine whether the rule will collide with an already existing queue or not. So in this implementation, I changed the config parsing methods to be aware of the the possible name collisions and also added an extra ambiguity check just before creating the placement context, this way the user will be able to see in the logs why the application fails.

The expected behaviour for name collisions is to refuse the application submission if the matching placement rule results in an ambiguous queue name. This is consistent with the app submission behaviour when providing a queue name, but it is a leaf name only.

Next step is to check the application submission side, and make sure the submitted queue names are parsed properly and are compatible with the internal full queue name usage.

Also marked a few parts of the code for cleanup later, we are using 3 classes with very similar behaviour and it was really confusing to understand the difference between them (practically there is none), but since this patch is already growing in size, and I haven't even touched the tests, I'll file separate Jiras about those later.

▼ Hadoop QA added a comment - 14/Feb/20 05:48

-1 overall


Vote	Subsystem	Runtime	Comment
0	reexec	0m 42s	Docker mode activated.
Prechecks			
+1	@author	0m 0s	The patch does not contain any @author tags.
+1	test4tests	0m 0s	The patch appears to include 7 new or modified test files.
trunk Compile Tests			
+1	mvninstall	20m 12s	trunk passed
+1	compile	0m 43s	trunk passed
+1	checkstyle	0m 51s	trunk passed
+1	mvnsite	0m 46s	trunk passed
+1	shadedclient	15m 42s	branch has no errors when building and testing our client artifacts.
+1	findbugs	1m 35s	trunk passed
+1	javadoc	0m 29s	trunk passed
Patch Compile Tests			
+1	mvninstall	0m 43s	the patch passed
+1	compile	0m 38s	the patch passed
+1	javac	0m 38s	the patch passed
-0	checkstyle	0m 46s	hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager: The patch generated 72 new + 1168 unchanged - 5 fixed = 1240 total (was 1173)
+1	mvnsite	0m 40s	the patch passed
+1	whitespace	0m 0s	The patch has no whitespace issues.
+1	shadedclient	14m 13s	patch has no errors when building and testing our client artifacts.
+1	findbugs	1m 37s	the patch passed
+1	javadoc	0m 28s	the patch passed
Other Tests			
-1	unit	87m 9s	hadoop-yarn-server-resourcemanager in the patch passed.
-1	asflicense	0m 28s	The patch generated 2 ASF License warnings.
		147m 24s	

Reason	Tests
Failed junit tests	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestLeafQueue
	hadoop.yarn.server.resourcemanager.reservation.TestCapacitySchedulerPlanFollower
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerNodeLabelUpdate
	hadoop.yarn.server.resourcemanager.webapp.TestRMWebServicesSchedulerActivities
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyIntraQueueUserLimit
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerAutoQueueCreation
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestQueueManagementDynamicEditPolicy
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestQueueStateManager
	hadoop.yarn.server.resourcemanager.placement.TestUserGroupMappingPlacementRule
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyIntraQueueFairOrdering
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicy
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestQueueState
	hadoop.yarn.server.resourcemanager.webapp.TestRMWebServicesForCSWithPartitions
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestQueueParsing
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerAutoCreatedQueuePreemption
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyForReservedContainers
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestParentQueue
	hadoop.yarn.server.resourcemanager.placement.TestAppNameMappingPlacementRule

	hadoop.yarn.server.resourcemanager.TestWorkPreservingRMRestart
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestNodeLabelContainerAllocation
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyIntraQueueWithDRF
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerSchedulingRequestUpdate
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestApplicationPriorityACLs
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerQueueMappingFactory
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerSurgicalPreemption
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestAbsoluteResourceConfiguration
	hadoop.yarn.server.resourcemanager.TestReservationSystemWithRMHA
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyIntraQueue
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestWorkPreservingRMRestartForNodeLabel
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerDynamicBehavior
	hadoop.yarn.server.resourcemanager.webapp.TestRMWebServicesReservation
	hadoop.yarn.server.resourcemanager.webapp.TestRMWebServicesSchedulerActivitiesWithMultiNodesEnabled
	hadoop.yarn.server.resourcemanager.reservation.TestReservationSystem
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacityScheduler
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyPreemptToBalance
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerLazyPreemption
	hadoop.yarn.server.resourcemanager.TestClientRMService

Subsystem	Report/Notes
Docker	Client=19.03.6 Server=19.03.6 Image:yetus/hadoop:c44943d1fc3
JIRA Issue	<a href="#">YARN-9879</a>
JIRA Patch URL	<a href="https://issues.apache.org/jira/secure/attachment/12993447/YARN-9879.POC003.patch">https://issues.apache.org/jira/secure/attachment/12993447/YARN-9879.POC003.patch</a>
Optional Tests	dupname asflicense compile javac javadoc mvninstall mvnsite unit shadedclient findbugs checkstyle
uname	Linux 6542b8eaa2f5 4.15.0-74-generic #84-Ubuntu SMP Thu Dec 19 08:06:28 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
Build tool	maven
Personality	/testptch/patchprocess/precommit/personality/provided.sh
git revision	trunk / 56dee66
maven	version: Apache Maven 3.3.9
Default Java	1.8.0_242
findbugs	v3.1.0-RC1
checkstyle	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25523/artifact/out/diff-checkstyle-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25523/artifact/out/diff-checkstyle-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt</a>
unit	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25523/artifact/out/patch-unit-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25523/artifact/out/patch-unit-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt</a>
Test Results	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25523/testReport/">https://builds.apache.org/job/PreCommit-YARN-Build/25523/testReport/</a>
asflicense	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25523/artifact/out/patch-asflicense-problems.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25523/artifact/out/patch-asflicense-problems.txt</a>
Max. process+thread count	1011 (vs. ulimit of 5500)
modules	C: hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager U: hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager
Console output	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25523/console">https://builds.apache.org/job/PreCommit-YARN-Build/25523/console</a>
Powered by	Apache Yetus 0.8.0 <a href="http://yetus.apache.org">http://yetus.apache.org</a>

This message was automatically generated.

🗲️  **Gergely Pollák** added a comment - 20/Feb/20 10:24

Did some very basic smoke testing, and in this version it is possible to submit applications with leaf queue names and full path as well. However it need some excessive testing, for ACLs, placement rules, preemption just to name a few. But since the initial manual tests worked I've updated the testcases to use getQueuePath instead of getQueueName, probably a lot will fail this way too, but then I'm moving on to fixing those. After the tests are fixed, I start manually testing the more complex features of the CS.

🗲️  **Hadoop QA** added a comment - 20/Feb/20 12:06

-1 overall

Vote	Subsystem	Runtime	Comment
0	reexec	0m 42s	Docker mode activated.
			<b>Prechecks</b>
+1	@author	0m 0s	The patch does not contain any @author tags.
+1	test4tests	0m 0s	The patch appears to include 16 new or modified test files.
			<b>trunk Compile Tests</b>
0	mvndep	1m 17s	Maven dependency ordering for branch
+1	mvninstall	21m 17s	trunk passed
+1	compile	16m 39s	trunk passed
+1	checkstyle	3m 8s	trunk passed
+1	mvnsite	1m 27s	trunk passed
+1	shadedclient	19m 41s	branch has no errors when building and testing our client artifacts.
+1	findbugs	2m 25s	trunk passed
+1	javadoc	1m 5s	trunk passed
			<b>Patch Compile Tests</b>
0	mvndep	0m 20s	Maven dependency ordering for patch
-1	mvninstall	0m 23s	hadoop-yarn-server-resourcemanager in the patch failed.
-1	compile	6m 35s	root in the patch failed.
-1	javac	6m 35s	root in the patch failed.
-0	checkstyle	2m 55s	root: The patch generated 78 new + 2135 unchanged - 5 fixed = 2213 total (was 2140)
-1	mvnsite	0m 24s	hadoop-yarn-server-resourcemanager in the patch failed.
+1	whitespace	0m 0s	The patch has no whitespace issues.
-1	shadedclient	4m 5s	patch has errors when building and testing our client artifacts.
-1	findbugs	0m 24s	hadoop-yarn-server-resourcemanager in the patch failed.
+1	javadoc	0m 45s	the patch passed
			<b>Other Tests</b>
-1	unit	0m 26s	hadoop-yarn-server-resourcemanager in the patch failed.
+1	unit	11m 53s	hadoop-sls in the patch passed.
0	asflicense	0m 44s	ASF License check generated no output?
		96m 7s	

Subsystem	Report/Notes
Docker	Client=19.03.6 Server=19.03.6 Image:yetus/hadoop:c44943d1fc3
JIRA Issue	<a href="#">YARN-9879</a>
JIRA Patch URL	<a href="https://issues.apache.org/jira/secure/attachment/12993985/YARN-9879.POC004.patch">https://issues.apache.org/jira/secure/attachment/12993985/YARN-9879.POC004.patch</a>
Optional Tests	dupname asflicense compile javac javadoc mvninstall mvnsite unit shadedclient findbugs checkstyle
uname	Linux 4435371a8212 4.15.0-74-generic #84-Ubuntu SMP Thu Dec 19 08:06:28 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
Build tool	maven
Personality	/testptch/patchprocess/precommit/personality/provided.sh
git revision	trunk / ec75071
maven	version: Apache Maven 3.3.9
Default Java	1.8.0_242
findbugs	v3.1.0-RC1
mvninstall	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25545/artifact/out/patch-mvninstall-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25545/artifact/out/patch-mvninstall-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt</a>
compile	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25545/artifact/out/patch-compile-root.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25545/artifact/out/patch-compile-root.txt</a>
javac	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25545/artifact/out/patch-compile-root.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25545/artifact/out/patch-compile-root.txt</a>
checkstyle	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25545/artifact/out/diff-checkstyle-root.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25545/artifact/out/diff-checkstyle-root.txt</a>

mvnsite	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25545/artifact/out/patch-mvnsite-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25545/artifact/out/patch-mvnsite-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt</a>
findbugs	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25545/artifact/out/patch-findbugs-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25545/artifact/out/patch-findbugs-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt</a>
unit	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25545/artifact/out/patch-unit-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25545/artifact/out/patch-unit-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt</a>
Test Results	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25545/testReport/">https://builds.apache.org/job/PreCommit-YARN-Build/25545/testReport/</a>
Max. process+thread count	431 (vs. ulimit of 5500)
modules	C: hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-tools/hadoop-sls U: .
Console output	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25545/console">https://builds.apache.org/job/PreCommit-YARN-Build/25545/console</a>
Powered by	Apache Yetus 0.8.0 <a href="http://yetus.apache.org">http://yetus.apache.org</a>

This message was automatically generated.

Hadoop QA added a comment - 20/Feb/20 19:46

-1 overall

Vote	Subsystem	Runtime	Comment
0	reexec	0m 44s	Docker mode activated.
			Prechecks
+1	@author	0m 0s	The patch does not contain any @author tags.
+1	test4tests	0m 0s	The patch appears to include 16 new or modified test files.
			trunk Compile Tests
0	mvndep	0m 23s	Maven dependency ordering for branch
+1	mvninstall	20m 19s	trunk passed
+1	compile	20m 35s	trunk passed
+1	checkstyle	3m 7s	trunk passed
+1	mvnsite	1m 27s	trunk passed
+1	shadedclient	19m 53s	branch has no errors when building and testing our client artifacts.
+1	findbugs	2m 27s	trunk passed
+1	javadoc	1m 5s	trunk passed
			Patch Compile Tests
0	mvndep	0m 20s	Maven dependency ordering for patch
+1	mvninstall	1m 5s	the patch passed
+1	compile	15m 47s	the patch passed
+1	javac	15m 47s	the patch passed
-0	checkstyle	3m 8s	root: The patch generated 78 new + 2135 unchanged - 5 fixed = 2213 total (was 2140)
+1	mvnsite	1m 25s	the patch passed
+1	whitespace	0m 0s	The patch has no whitespace issues.
+1	shadedclient	14m 46s	patch has no errors when building and testing our client artifacts.
+1	findbugs	2m 36s	the patch passed
+1	javadoc	1m 4s	the patch passed
			Other Tests
-1	unit	88m 9s	hadoop-yarn-server-resourcemanager in the patch passed.
-1	unit	8m 19s	hadoop-sls in the patch passed.
-1	asflicense	0m 41s	The patch generated 2 ASF License warnings.
		206m 2s	

Reason	Tests
Failed junit tests	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestLeafQueue



	hadoop.yarn.server.resourcemanager.reservation.TestCapacitySchedulerPlanFollower
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerNodeLabelUpdate
	hadoop.yarn.server.resourcemanager.webapp.TestRMWebServicesSchedulerActivities
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyIntraQueueUserLimit
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerAutoQueueCreation
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestQueueManagementDynamicEditPolicy
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyMockFramework
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestQueueStateManager
	hadoop.yarn.server.resourcemanager.placement.TestUserGroupMappingPlacementRule
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyIntraQueueFairOrdering
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicy
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestQueueState
	hadoop.yarn.server.resourcemanager.webapp.TestRMWebServicesForCSWithPartitions
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestQueueParsing
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerAutoCreatedQueuePreemption
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyForReservedContainers
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestParentQueue
	hadoop.yarn.server.resourcemanager.placement.TestAppNameMappingPlacementRule
	hadoop.yarn.server.resourcemanager.TestWorkPreservingRMRestart
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestNodeLabelContainerAllocation
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyIntraQueueWithDRF
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerSchedulingRequestUpdate
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestApplicationPriorityACLs
	hadoop.yarn.server.resourcemanager.scheduler.TestAbstractYarnScheduler
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerQueueMappingFactory
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerSurgicalPreemption
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestAbsoluteResourceConfiguration
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyInterQueueWithDRF
	hadoop.yarn.server.resourcemanager.TestReservationSystemWithRMHA
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyIntraQueue
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestWorkPreservingRMRestartForNodeLabel
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyForNodePartitions
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerDynamicBehavior
	hadoop.yarn.server.resourcemanager.webapp.TestRMWebServicesReservation
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestPreemptionForQueueWithPriorities
	hadoop.yarn.server.resourcemanager.webapp.TestRMWebServicesSchedulerActivitiesWithMultiNodesEnabled
	hadoop.yarn.server.resourcemanager.reservation.TestReservationSystem
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacityScheduler
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyPreemptToBalance
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerLazyPreemption
	hadoop.yarn.server.resourcemanager.TestClientRMService
	hadoop.yarn.sls.TestSLSRunner
	hadoop.yarn.sls.TestSLStreamAMSynth
	hadoop.yarn.sls.TestSLSGenericSynth
	hadoop.yarn.sls.TestReservationSystemInvariants
	hadoop.yarn.sls.appmaster.TestAMSimulator

Subsystem	Report/Notes
Docker	Client=19.03.6 Server=19.03.6 Image:yetus/hadoop:c44943d1fc3
JIRA Issue	<a href="#">YARN-9879</a>
JIRA Patch URL	<a href="https://issues.apache.org/jira/secure/attachment/12994014/YARN-9879.POC005.patch">https://issues.apache.org/jira/secure/attachment/12994014/YARN-9879.POC005.patch</a>

Optional Tests	dupname asflicense compile javac javadoc mvninstall mvnsite unit shadedclient findbugs checkstyle
uname	Linux 5d9ab07081c2 4.15.0-74-generic #84-Ubuntu SMP Thu Dec 19 08:06:28 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
Build tool	maven
Personality	/testptch/patchprocess/precommit/personality/provided.sh
git revision	trunk / 181e6d0
maven	version: Apache Maven 3.3.9
Default Java	1.8.0_242
findbugs	v3.1.0-RC1
checkstyle	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25553/artifact/out/diff-checkstyle-root.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25553/artifact/out/diff-checkstyle-root.txt</a>
unit	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25553/artifact/out/patch-unit-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25553/artifact/out/patch-unit-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt</a>
unit	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25553/artifact/out/patch-unit-hadoop-tools_hadoop-sls.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25553/artifact/out/patch-unit-hadoop-tools_hadoop-sls.txt</a>
Test Results	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25553/testReport/">https://builds.apache.org/job/PreCommit-YARN-Build/25553/testReport/</a>
asflicense	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25553/artifact/out/patch-asflicense-problems.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25553/artifact/out/patch-asflicense-problems.txt</a>
Max. process+thread count	1002 (vs. ulimit of 5500)
modules	C: hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-tools/hadoop-sls U: .
Console output	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25553/console">https://builds.apache.org/job/PreCommit-YARN-Build/25553/console</a>
Powered by	Apache Yetus 0.8.0 <a href="http://yetus.apache.org">http://yetus.apache.org</a>

This message was automatically generated.

Hadoop QA added a comment - 27/Feb/20 20:37

-1 overall

Vote	Subsystem	Runtime	Comment
0	reexec	0m 28s	Docker mode activated.
			Prechecks
+1	@author	0m 0s	The patch does not contain any @author tags.
+1	test4tests	0m 0s	The patch appears to include 16 new or modified test files.
			trunk Compile Tests
0	mvndep	1m 8s	Maven dependency ordering for branch
+1	mvninstall	17m 47s	trunk passed
+1	compile	14m 53s	trunk passed
+1	checkstyle	3m 18s	trunk passed
+1	mvnsite	1m 42s	trunk passed
+1	shadedclient	19m 1s	branch has no errors when building and testing our client artifacts.
+1	findbugs	2m 31s	trunk passed
+1	javadoc	1m 20s	trunk passed
			Patch Compile Tests
0	mvndep	0m 24s	Maven dependency ordering for patch
+1	mvninstall	1m 9s	the patch passed
+1	compile	14m 16s	the patch passed
+1	javac	14m 16s	the patch passed
-0	checkstyle	3m 15s	root: The patch generated 83 new + 2144 unchanged - 5 fixed = 2227 total (was 2149)
+1	mvnsite	1m 41s	the patch passed
+1	whitespace	0m 0s	The patch has no whitespace issues.
+1	shadedclient	13m 15s	patch has no errors when building and testing our client artifacts.
+1	findbugs	2m 44s	the patch passed
+1	javadoc	1m 22s	the patch passed
			Other Tests
-1	unit	84m 40s	hadoop-yarn-server-resourcemanager in the patch passed.

+1	unit	12m 7s	hadoop-slfs in the patch passed.
-1	asflicense	0m 51s	The patch generated 2 ASF License warnings.
		196m 42s	

Reason	Tests
Failed junit tests	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestParentQueue
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerSurgicalPreemption
	hadoop.yarn.server.resourcemanager.TestClientRMService
	hadoop.yarn.server.resourcemanager.webapp.TestRMWebServicesSchedulerActivitiesWithMultiNodesEnabled
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyPreemptToBalance
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestAbsoluteResourceConfiguration
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestQueueParsing
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerNodeLabelUpdate
	hadoop.yarn.server.resourcemanager.placement.TestAppNameMappingPlacementRule
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestQueueState
	hadoop.yarn.server.resourcemanager.webapp.TestRMWebServicesForCSWithPartitions
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyForReservedContainers
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerQueueMappingFactory
	hadoop.yarn.server.resourcemanager.reservation.TestReservationSystem
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerAutoQueueCreation
	hadoop.yarn.server.resourcemanager.TestReservationSystemWithRMHA
	hadoop.yarn.server.resourcemanager.webapp.TestRMWebServicesReservation
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestApplicationPriorityACLs
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerAutoCreatedQueuePreemption
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerLazyPreemption
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicy
	hadoop.yarn.server.resourcemanager.scheduler.TestAbstractYarnScheduler
	hadoop.yarn.server.resourcemanager.webapp.TestRMWebServicesSchedulerActivities
	hadoop.yarn.server.resourcemanager.placement.TestUserGroupMappingPlacementRule
	hadoop.yarn.server.resourcemanager.TestWorkPreservingRMRestart

Subsystem	Report/Notes
Docker	Client=19.03.6 Server=19.03.6 Image:yetus/hadoop:c44943d1fc3
JIRA Issue	<a href="#">YARN-9879</a>
JIRA Patch URL	<a href="https://issues.apache.org/jira/secure/attachment/12994805/YARN-9879.POC006.patch">https://issues.apache.org/jira/secure/attachment/12994805/YARN-9879.POC006.patch</a>
Optional Tests	dupname asflicense compile javac javadoc mvninstall mvnsite unit shadedclient findbugs checkstyle
uname	Linux 971d5fbedce3 4.15.0-58-generic #64-Ubuntu SMP Tue Aug 6 11:12:41 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
Build tool	maven
Personality	/testptch/patchprocess/precommit/personality/provided.sh
git revision	trunk / cd2c6b1
maven	version: Apache Maven 3.3.9
Default Java	1.8.0_242
findbugs	v3.1.0-RC1
checkstyle	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25598/artifact/out/diff-checkstyle-root.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25598/artifact/out/diff-checkstyle-root.txt</a>
unit	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25598/artifact/out/patch-unit-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25598/artifact/out/patch-unit-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt</a>
Test Results	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25598/testReport/">https://builds.apache.org/job/PreCommit-YARN-Build/25598/testReport/</a>
asflicense	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25598/artifact/out/patch-asflicense-problems.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25598/artifact/out/patch-asflicense-problems.txt</a>
Max. process+thread count	872 (vs. ulimit of 5500)

modules	C: hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-tools/hadoop-sls U:
Console output	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25598/console">https://builds.apache.org/job/PreCommit-YARN-Build/25598/console</a>
Powered by	Apache Yetus 0.8.0 <a href="http://yetus.apache.org">http://yetus.apache.org</a>

This message was automatically generated.

🗲️ Gergely Pollák added a comment - 28/Feb/20 15:12

Latest 2 patches are fixes to get rid of the enormous amount of unit test failures. Some cleanup for the CSQueueStore are required, since I had to reintroduce a feature which caused a bug in the current queue management solution, since too many tests were relying on it.

The tests often times get non leaf queues by their short name, the issue with this is it is possible to overwrite a leaf queue with a non-leaf queue. Eg if we have the two following queues:

root.a.leaf

root.b.c.leaf.e

Then the get('leaf') will return the root.b.c.leaf instead of root.a.leaf. Also in the current implementation it is quite hectic what happens if there is a non-leaf queue with the same name as a leaf queue. So I thought I would only allow referencing by short names only to leaf queues, but it won't work.

So here is my proposal:

Leaf queues will have priority, so if there is leaf queue and a non-leaf queue with the short name, the name will always refer to the leaf queue, if there are two leaf queues with the same name, none will be accessible. This is not the best solution but it will fix the bug in the current implementation, and we keep the possibility to reference non-leaf queues by their short name, and our tests will be grateful for this.

🗲️ Hadoop QA added a comment - 28/Feb/20 18:39

-1 overall

Vote	Subsystem	Runtime	Comment
0	reexec	0m 47s	Docker mode activated.
			Prechecks
+1	@author	0m 0s	The patch does not contain any @author tags.
+1	test4tests	0m 0s	The patch appears to include 19 new or modified test files.
			trunk Compile Tests
0	mvndep	1m 17s	Maven dependency ordering for branch
+1	mvninstall	20m 27s	trunk passed
+1	compile	15m 44s	trunk passed
+1	checkstyle	3m 4s	trunk passed
+1	mvnsite	1m 26s	trunk passed
-1	shadedclient	20m 46s	branch has errors when building and testing our client artifacts.
-1	findbugs	0m 59s	hadoop-yarn-server-resourcemanager in trunk failed.
+1	javadoc	1m 10s	trunk passed
			Patch Compile Tests
0	mvndep	0m 30s	Maven dependency ordering for patch
+1	mvninstall	1m 14s	the patch passed
+1	compile	15m 49s	the patch passed
+1	javac	15m 49s	the patch passed
-0	checkstyle	3m 7s	root: The patch generated 90 new + 2161 unchanged - 6 fixed = 2251 total (was 2167)
+1	mvnsite	1m 24s	the patch passed
+1	whitespace	0m 0s	The patch has no whitespace issues.
+1	shadedclient	14m 17s	patch has no errors when building and testing our client artifacts.
+1	findbugs	2m 38s	the patch passed
+1	javadoc	1m 5s	the patch passed
			Other Tests
-1	unit	90m 18s	hadoop-yarn-server-resourcemanager in the patch passed.
+1	unit	11m 59s	hadoop-sls in the patch passed.
-1	asflicense	0m 43s	The patch generated 2 ASF License warnings.

	208m 45s	
--	----------	--


Reason	Tests
Failed junit tests	hadoop.yarn.server.resourcemanager.reservation.TestCapacitySchedulerPlanFollower
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerNodeLabelUpdate
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerAutoQueueCreation
	hadoop.yarn.server.resourcemanager.placement.TestUserGroupMappingPlacementRule
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicy
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestQueueState
	hadoop.yarn.server.resourcemanager.webapp.TestRMWebServicesForCSWithPartitions
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestQueueParsing
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerAutoCreatedQueuePreemption
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyForReservedContainers
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestParentQueue
	hadoop.yarn.server.resourcemanager.placement.TestAppNameMappingPlacementRule
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerQueueACLs
	hadoop.yarn.server.resourcemanager.TestWorkPreservingRMRestart
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestApplicationPriorityACLs
	hadoop.yarn.server.resourcemanager.scheduler.TestAbstractYarnScheduler
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerQueueMappingFactory
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerSurgicalPreemption
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestAbsoluteResourceConfiguration
	hadoop.yarn.server.resourcemanager.TestReservationSystemWithRMHA
	hadoop.yarn.server.resourcemanager.reservation.TestReservationSystem
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyPreemptToBalance
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerLazyPreemption

Subsystem	Report/Notes
Docker	Client=19.03.6 Server=19.03.6 Image:yetus/hadoop:c44943d1fc3
JIRA Issue	<a href="#">YARN-9879</a>
JIRA Patch URL	<a href="https://issues.apache.org/jira/secure/attachment/12994906/YARN-9879.POC007.patch">https://issues.apache.org/jira/secure/attachment/12994906/YARN-9879.POC007.patch</a>
Optional Tests	dupname asflicense compile javac javadoc mvninstall mvnsite unit shadedclient findbugs checkstyle
uname	Linux a68487065fa2 4.15.0-74-generic #84-Ubuntu SMP Thu Dec 19 08:06:28 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
Build tool	maven
Personality	/testptch/patchprocess/precommit/personality/provided.sh
git revision	trunk / fccfb02
maven	version: Apache Maven 3.3.9
Default Java	1.8.0_242
findbugs	v3.1.0-RC1
findbugs	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25604/artifact/out/branch-findbugs-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25604/artifact/out/branch-findbugs-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt</a>
checkstyle	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25604/artifact/out/diff-checkstyle-root.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25604/artifact/out/diff-checkstyle-root.txt</a>
unit	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25604/artifact/out/patch-unit-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25604/artifact/out/patch-unit-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt</a>
Test Results	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25604/testReport/">https://builds.apache.org/job/PreCommit-YARN-Build/25604/testReport/</a>
asflicense	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25604/artifact/out/patch-asflicense-problems.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25604/artifact/out/patch-asflicense-problems.txt</a>
Max. process+thread count	838 (vs. ulimit of 5500)
modules	C: hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-tools/hadoop-sls U: .
Console output	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25604/console">https://builds.apache.org/job/PreCommit-YARN-Build/25604/console</a>



Powered by	Apache Yetus 0.8.0 <a href="http://yetus.apache.org">http://yetus.apache.org</a>
------------	--

This message was automatically generated.

▼  Hadoop QA added a comment - 01/Mar/20 01:32

 -1 overall

Vote	Subsystem	Runtime	Comment
0	reexec	0m 44s	Docker mode activated.
			Prechecks
+1	@author	0m 0s	The patch does not contain any @author tags.
+1	test4tests	0m 0s	The patch appears to include 23 new or modified test files.
			trunk Compile Tests
0	mvndep	1m 13s	Maven dependency ordering for branch
+1	mvninstall	19m 57s	trunk passed
+1	compile	16m 4s	trunk passed
+1	checkstyle	3m 9s	trunk passed
+1	mvnsite	1m 26s	trunk passed
+1	shadedclient	19m 40s	branch has no errors when building and testing our client artifacts.
+1	findbugs	2m 23s	trunk passed
+1	javadoc	1m 5s	trunk passed
			Patch Compile Tests
0	mvndep	0m 21s	Maven dependency ordering for patch
+1	mvninstall	1m 6s	the patch passed
+1	compile	15m 20s	the patch passed
+1	javac	15m 20s	the patch passed
-0	checkstyle	3m 10s	root: The patch generated 99 new + 2262 unchanged - 6 fixed = 2361 total (was 2268)
+1	mvnsite	1m 26s	the patch passed
+1	whitespace	0m 0s	The patch has no whitespace issues.
+1	shadedclient	14m 22s	patch has no errors when building and testing our client artifacts.
+1	findbugs	2m 38s	the patch passed
+1	javadoc	1m 6s	the patch passed
			Other Tests
-1	unit	104m 37s	hadoop-yarn-server-resourcemanager in the patch passed.
+1	unit	12m 22s	hadoop-slfs in the patch passed.
-1	asflicense	0m 51s	The patch generated 2 ASF License warnings.
		221m 46s	

Reason	Tests
Failed junit tests	hadoop.yarn.server.resourcemanager.reservation.TestCapacitySchedulerPlanFollower
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerNodeLabelUpdate
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestSchedulingRequestContainerAllocationAsync
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerAutoQueueCreation
	hadoop.yarn.server.resourcemanager.placement.TestUserGroupMappingPlacementRule
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicy
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestQueueState
	hadoop.yarn.server.resourcemanager.webapp.TestRMWebServicesForCSWithPartitions
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestQueueParsing
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyForReservedContainers
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestParentQueue
	hadoop.yarn.server.resourcemanager.placement.TestAppNameMappingPlacementRule

	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestCapacitySchedulerQueueACLs
	hadoop.yarn.server.resourcemanager.TestWorkPreservingRMRestart
	hadoop.yarn.server.resourcemanager.scheduler.capacity.TestApplicationPriorityACLs
	hadoop.yarn.server.resourcemanager.scheduler.TestAbstractYarnScheduler
	hadoop.yarn.server.resourcemanager.TestReservationSystemWithRMHA
	hadoop.yarn.server.resourcemanager.reservation.TestReservationSystem
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyPreemptToBalance

Subsystem	Report/Notes
Docker	Client=19.03.6 Server=19.03.6 Image:yetus/hadoop:c44943d1fc3
JIRA Issue	<a href="#">YARN-9879</a>
JIRA Patch URL	<a href="https://issues.apache.org/jira/secure/attachment/12995042/YARN-9879.POC008.patch">https://issues.apache.org/jira/secure/attachment/12995042/YARN-9879.POC008.patch</a>
Optional Tests	dupname asflicense compile javac javadoc mvninstall mvnsite unit shadedclient findbugs checkstyle
uname	Linux 656d0428ce20 4.15.0-74-generic #84-Ubuntu SMP Thu Dec 19 08:06:28 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
Build tool	maven
Personality	/testptch/patchprocess/precommit/personality/provided.sh
git revision	trunk / 1a636da
maven	version: Apache Maven 3.3.9
Default Java	1.8.0_242
findbugs	v3.1.0-RC1
checkstyle	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25607/artifact/out/diff-checkstyle-root.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25607/artifact/out/diff-checkstyle-root.txt</a>
unit	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25607/artifact/out/patch-unit-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25607/artifact/out/patch-unit-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt</a>
Test Results	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25607/testReport/">https://builds.apache.org/job/PreCommit-YARN-Build/25607/testReport/</a>
asflicense	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25607/artifact/out/patch-asflicense-problems.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25607/artifact/out/patch-asflicense-problems.txt</a>
Max. process+thread count	829 (vs. ulimit of 5500)
modules	C: hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-tools/hadoop-sls U: .
Console output	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25607/console">https://builds.apache.org/job/PreCommit-YARN-Build/25607/console</a>
Powered by	Apache Yetus 0.8.0 <a href="http://yetus.apache.org">http://yetus.apache.org</a>

This message was automatically generated.

[Gergely Pollák](#) added a comment - 02/Mar/20 20:42

Latest patch (no. 9) is still about test failure fixes, if I there won't be many test failures, then I'll get to the code cleanup of the new classes, also adding own testcases for new code. And when all tests pass, I'll remove the constraint which prevents multiple leafqueues to be created with the same name. It shouldn't break too many tests but I think its better to cleanup the current test failures first, to see the change's impact.

[Hadoop QA](#) added a comment - 03/Mar/20 00:06

-1 overall

Vote	Subsystem	Runtime	Comment
0	<a href="#">reexec</a>	0m 42s	<a href="#">Docker mode activated.</a>
			<b>Prechecks</b>
+1	<a href="#">@author</a>	0m 0s	The patch does not contain any <a href="#">@author</a> tags.
+1	<a href="#">test4tests</a>	0m 0s	The patch appears to include 26 new or modified test files.
			<b>trunk Compile Tests</b>
0	<a href="#">mvndep</a>	0m 28s	<a href="#">Maven dependency ordering for branch</a>
+1	<a href="#">mvninstall</a>	21m 42s	trunk passed
+1	<a href="#">compile</a>	15m 56s	trunk passed
+1	<a href="#">checkstyle</a>	3m 8s	trunk passed

+1	mvnsite	1m 27s	trunk passed
+1	shadedclient	19m 36s	branch has no errors when building and testing our client artifacts.
+1	findbugs	2m 22s	trunk passed
+1	javadoc	1m 5s	trunk passed
			<b>Patch Compile Tests</b>
0	mvndep	0m 20s	Maven dependency ordering for patch
+1	mvninstall	1m 6s	the patch passed
+1	compile	15m 23s	the patch passed
+1	javac	15m 23s	the patch passed
-0	checkstyle	3m 12s	root: The patch generated 112 new + 2290 unchanged - 9 fixed = 2402 total (was 2299)
+1	mvnsite	1m 26s	the patch passed
+1	whitespace	0m 0s	The patch has no whitespace issues.
+1	shadedclient	14m 13s	patch has no errors when building and testing our client artifacts.
+1	findbugs	2m 39s	the patch passed
+1	javadoc	1m 6s	the patch passed
			<b>Other Tests</b>
-1	unit	88m 19s	hadoop-yarn-server-resourcemanager in the patch passed.
+1	unit	11m 54s	hadoop-slfs in the patch passed.
0	asflicense	1m 16s	ASF License check generated no output?
		206m 2s	

Reason	Tests
Failed junit tests	hadoop.yarn.server.resourcemanager.placement.TestUserGroupMappingPlacementRule
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicy
	hadoop.yarn.server.resourcemanager.webapp.TestRMWebServicesForCSWithPartitions
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyForReservedContainers
	hadoop.yarn.server.resourcemanager.placement.TestAppNameMappingPlacementRule
	hadoop.yarn.server.resourcemanager.reservation.TestCapacityOverTimePolicy
	hadoop.yarn.server.resourcemanager.scheduler.TestAbstractYarnScheduler
	hadoop.yarn.server.resourcemanager.scheduler.fair.TestFairSchedulerPreemption
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyPreemptToBalance

Subsystem	Report/Notes
Docker	Client=19.03.6 Server=19.03.6 Image:yetus/hadoop:c44943d1fc3
JIRA Issue	<a href="#">YARN-9879</a>
JIRA Patch URL	<a href="https://issues.apache.org/jira/secure/attachment/12995362/YARN-9879.POC009.patch">https://issues.apache.org/jira/secure/attachment/12995362/YARN-9879.POC009.patch</a>
Optional Tests	dupname asflicense compile javac javadoc mvninstall mvnsite unit shadedclient findbugs checkstyle
uname	Linux 26ca44f07f43 4.15.0-74-generic #84-Ubuntu SMP Thu Dec 19 08:06:28 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
Build tool	maven
Personality	/testptch/patchprocess/precommit/personality/provided.sh
git revision	trunk / edc2e9d
maven	version: Apache Maven 3.3.9
Default Java	1.8.0_242
findbugs	v3.1.0-RC1
checkstyle	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25616/artifact/out/diff-checkstyle-root.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25616/artifact/out/diff-checkstyle-root.txt</a>
unit	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25616/artifact/out/patch-unit-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25616/artifact/out/patch-unit-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt</a>
Test Results	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25616/testReport/">https://builds.apache.org/job/PreCommit-YARN-Build/25616/testReport/</a>
Max. process+thread count	830 (vs. ulimit of 5500)

modules	C: hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-tools/hadoop-sls U:
Console output	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25616/console">https://builds.apache.org/job/PreCommit-YARN-Build/25616/console</a>
Powered by	Apache Yetus 0.8.0 <a href="http://yetus.apache.org">http://yetus.apache.org</a>

This message was automatically generated.

👤 Szilard Nemeth added a comment - 04/Mar/20 18:05

Hi shuzirra ,

Phew... This was pretty hard to review, spent at least 2 hours with it.

Thanks for working on this patch, good job, this change is incredible. 😊

**In general, +1 for your approach by introducing CSQueueStore.**

#### MY MAIN COMMENTS:

1. I can see many TODOs in the patch. Do you want to address them in the next patch?

Make sure you remove all the TODOs you have added to the code as they could not be part of the commit.

2. Make sure to adhere to the 80 chars line limit: I saw some very long comments.

3. I think I spot something suspicious with CapacitySchedulerPreemptionContext:

The reader methods (getQueuePartition, getQueuePartitions) are using the full queue path, like you are using it in

FifoIntraQueuePreemptionPlugin#skipContainerBasedOnIntraQueuePolicy

However, the implementation of this interface is passing a simple queue name in

ProportionalCapacityPreemptionPolicy#addPartitionToUnderServedQueues. Can you please double check your changes around this code?

4. In QueuePath#QueuePath(java.lang.String): The value of leafQueue is assigned to two fields: leafQueue, fullPath. Could you please add a comment here? I don't get what's going on by just reading the code.

5. Please add javadoc to methods: CapacityScheduler#normalizeQueueName and CapacityScheduler#isAmbiguous

6. TestCSQueueStore: I guess you will add something more in here 😊

#### COMMENTS FOR CSQueueStore:

1. Can you please add a javadoc to the class CSQueueStore, to its fields and to its main methods (at least the publicly accessible ones + addShortNameMapping)?

2. Method getFullNameQueues can be package-private.

3. Method getShortNameQueues can be package-private.

4. There's an unnecessary comment in this class:

```
// shortNameToFullName.entrySet().stream().forEach(e -> System.out.println("<>" + e));
// return null;
```

5. Comment could be a javadoc instead:

```
//we must synchronize here because we need to maintain multiple maps to be
//in sync, and concurrent hashMap does not help with that
```

6. Unnecessary commented code in method add

7. In method remove, you have an unnecessary containsKey check, IntelliJ reports this as well:

```
if (shortNameToFullName.containsKey(shortName)) {
    shortNameToFullName.remove(shortName);
}
```

Remove will remove the mapping if it does exist, otherwise it won't do anything.

I would remove the containsKey check, unless you explicitly want to highlight it.

8. Can you please add curly braces to the if in remove(java.lang.String)?

9. Method getQueueCount is unused

10. Method getByFullName can be package-private

11. Method getByShortName can be package-private

12. Method isAmbiguous can be package-private

13. In method getByShortName, can you add curly braces to the if?

RENAMINGS:

1. I found many occurrences of:

```
queueName = queue.getQueuePath
```

and

```
String leafQueueName = leafQueue.getQueuePath();
```

throughout your patch.  
Please rename ALL the variables to queuePath (or maybe fullQueueName) as it's pretty confusing like this.

2. Please rename the first parameter of GuaranteedOrZeroCapacityOverTimePolicy.LeafQueueState#addLeafQueueStateIfNotExists to queuePath as this method now receives a queue path instead of a name of the leaf queue.

3. You have a call in FifoIntraQueuePreemptionPlugin#skipContainerBasedOnIntraQueuePolicy:

```
TempQueuePerPartition tq = context.getQueueByPartition(queueName, partition);
```

I think it'd be a good idea to rename the parameter of CapacitySchedulerPreemptionContext#getQueueByPartition to fullQueueName or add a short javadoc to this method.

4. Please rename the 'queueName' parameter to 'fullQueueName' in CapacityScheduler#checkAndGetApplicationPriority

5. Please rename the 'leafQueueName' parameter to 'fullQueueName' in QueuePlacementRuleUtils#validateQueueMappingUnderParentQueue.


6. Please rename the parameter "queueName" to "fullQueueName" in methods checkAbsoluteCapacity / checkMaxCapacity  
CSQueueUtils#checkMaxCapacity


7. Please rename local variable called "leafQueueName" to "fullQueueName" in CapacityScheduler#markContainerForKillable.

8. Please rename local variable called "leafQueueName" to "fullQueueName" in CapacityScheduler#markContainerForNonKillable.

NITS:

- 1. There's an unused import in QueuePlacementRuleUtils
- 2. I can see some whitespace only changes in CapacitySchedulerConfigValidator#validateQueueHierarchy. Please remove them from the patch if they are not necessary.
- 3. CapacitySchedulerQueueManager#normalizeQueueName(String name) could be private.
- 4. CapacitySchedulerQueueManager#getQueueByShortName is unused.
- 5. Parameters of CapacitySchedulerConfigValidator#validateQueueHierarchy look very weird.

▼  [Gergely Pollák](#) added a comment - 05/Mar/20 00:23  
[snemeth](#) thank you for the feedback, I'll start implementing the suggested changes as soon all tests pass.

▼  [Hadoop QA](#) added a comment - 05/Mar/20 04:23

 -1 overall

Vote	Subsystem	Runtime	Comment
0	reexec	1m 39s	Docker mode activated.
			Prechecks
+1	@author	0m 0s	The patch does not contain any @author tags.
+1	test4tests	0m 0s	The patch appears to include 31 new or modified test files.
			trunk Compile Tests
0	mvndep	1m 33s	Maven dependency ordering for branch
+1	mvninstall	30m 14s	trunk passed



+1	compile	25m 51s	trunk passed
+1	checkstyle	4m 38s	trunk passed
+1	mvnsite	2m 6s	trunk passed
+1	shadedclient	28m 49s	branch has no errors when building and testing our client artifacts.
+1	findbugs	2m 35s	trunk passed
+1	javadoc	1m 48s	trunk passed
<b>Patch Compile Tests</b>			
0	mvndep	0m 27s	Maven dependency ordering for patch
+1	mvninstall	1m 41s	the patch passed
-1	compile	9m 46s	root in the patch failed.
-1	javac	9m 46s	root in the patch failed.
-0	checkstyle	3m 43s	root: The patch generated 123 new + 2328 unchanged - 9 fixed = 2451 total (was 2337)
+1	mvnsite	1m 10s	the patch passed
+1	whitespace	0m 0s	The patch has no whitespace issues.
+1	shadedclient	15m 39s	patch has no errors when building and testing our client artifacts.
+1	findbugs	2m 20s	the patch passed
+1	javadoc	0m 45s	the patch passed
<b>Other Tests</b>			
-1	unit	88m 31s	hadoop-yarn-server-resourcemanager in the patch passed.
-1	unit	11m 52s	hadoop-sls in the patch passed.
-1	asflicense	0m 29s	The patch generated 2 ASF License warnings.
		232m 29s	

Reason	Tests
Failed junit tests	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyMockFramework
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyIntraQueueUserLimit
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicy
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyPreemptToBalance
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyIntraQueueFairOrdering
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyIntraQueue
	hadoop.yarn.server.resourcemanager.monitor.capacity.TestProportionalCapacityPreemptionPolicyIntraQueueWithDRF
	hadoop.yarn.sls.appmaster.TestAMSimulator

Subsystem	Report/Notes
Docker	Client=19.03.7 Server=19.03.7 Image:yetus/hadoop:c44943d1fc3
JIRA Issue	<a href="#">YARN-9879</a>
JIRA Patch URL	<a href="https://issues.apache.org/jira/secure/attachment/12995675/YARN-9879.POC010.patch">https://issues.apache.org/jira/secure/attachment/12995675/YARN-9879.POC010.patch</a>
Optional Tests	dupname asflicense compile javac javadoc mvninstall mvnsite unit shadedclient findbugs checkstyle
uname	Linux 95f34b0c6e34 4.15.0-74-generic #84-Ubuntu SMP Thu Dec 19 08:06:28 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
Build tool	maven
Personality	/testptch/patchprocess/precommit/personality/provided.sh
git revision	trunk / 2649f8b
maven	version: Apache Maven 3.3.9
Default Java	1.8.0_242
findbugs	v3.1.0-RC1
compile	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25633/artifact/out/patch-compile-root.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25633/artifact/out/patch-compile-root.txt</a>
javac	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25633/artifact/out/patch-compile-root.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25633/artifact/out/patch-compile-root.txt</a>
checkstyle	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25633/artifact/out/diff-checkstyle-root.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25633/artifact/out/diff-checkstyle-root.txt</a>
unit	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25633/artifact/out/patch-unit-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25633/artifact/out/patch-unit-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt</a>

unit	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25633/artifact/out/patch-unit-hadoop-tools_hadoop-sls.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25633/artifact/out/patch-unit-hadoop-tools_hadoop-sls.txt</a>
Test Results	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25633/testReport/">https://builds.apache.org/job/PreCommit-YARN-Build/25633/testReport/</a>
asflicense	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25633/artifact/out/patch-asflicense-problems.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25633/artifact/out/patch-asflicense-problems.txt</a>
Max. process+thread count	815 (vs. ulimit of 5500)
modules	C: hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-tools/hadoop-sls U: .
Console output	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25633/console">https://builds.apache.org/job/PreCommit-YARN-Build/25633/console</a>
Powered by	Apache Yetus 0.8.0 <a href="http://yetus.apache.org">http://yetus.apache.org</a>

This message was automatically generated.

Wangda Tan added a comment - 06/Mar/20 00:29

Thanks shuzirra for the monster patch!

Took a quick look at the patch, overall looks good. (I skipped the hardest queue mapping module to leave to other folks to review).

1) Make sure commented code is not part of the final patch.

2) CSQueueStore:

- Only fullNameQueues is ConcurrentHashMap, is it intentional?
- getByShortName can be converted to private method, and the CapacitySchedulerQueueManager#getQueueByShortName is not used, can be removed.
- Instead of Synchronized lock, I suggest to use ReadWriteLock, the method like get is not safe since it access multiple fields. There's very infrequent write to queue map comparing to read.

3) CapacityScheduler.java:

```
1144 Queue queue = attempt.getQueue();
1145 CSQueue csQueue = queue instanceof CSQueue
```

This check is uncessceary. When CS is enabled, all queues in the RM is CSQueue.

4) CapacitySchedulerConfigValidator.java:  
validateQueueHierarchy: Have mixed usage of queueName and queuePath, suggest to move to queuePath for less ambiguous.

5) There're 18 TODOs in the patch, I suggest to mark "must-to-fix" TODOs to FIXME, in most cases TODO means we will never do it 😊. In Hadoop there're 731 TODOs.

Hadoop QA added a comment - 06/Mar/20 02:14

-1 overall


Vote	Subsystem	Runtime	Comment
0	reexec	0m 44s	Docker mode activated.
			Prechecks
+1	@author	0m 0s	The patch does not contain any @author tags.
+1	test4tests	0m 0s	The patch appears to include 32 new or modified test files.
			trunk Compile Tests
0	mvndep	0m 24s	Maven dependency ordering for branch
+1	mvninstall	23m 26s	trunk passed
+1	compile	17m 16s	trunk passed
+1	checkstyle	3m 11s	trunk passed
+1	mvnsite	1m 28s	trunk passed
+1	shadedclient	19m 44s	branch has no errors when building and testing our client artifacts.
+1	findbugs	2m 21s	trunk passed
+1	javadoc	1m 6s	trunk passed
			Patch Compile Tests
0	mvndep	0m 20s	Maven dependency ordering for patch
+1	mvninstall	1m 6s	the patch passed
+1	compile	15m 41s	the patch passed
+1	javac	15m 41s	the patch passed

-0	checkstyle	4m 36s	root: The patch generated 126 new + 2329 unchanged - 10 fixed = 2455 total (was 2339)
+1	mvnsite	1m 50s	the patch passed
+1	whitespace	0m 0s	The patch has no whitespace issues.
+1	shadedclient	15m 10s	patch has no errors when building and testing our client artifacts.
+1	findbugs	2m 40s	the patch passed
+1	javadoc	1m 20s	the patch passed
Other Tests			
-1	unit	90m 9s	hadoop-yarn-server-resourcemanager in the patch passed.
+1	unit	11m 58s	hadoop-sls in the patch passed.
-1	asflicense	0m 42s	The patch generated 2 ASF License warnings.
		214m 10s	

Reason	Tests
Failed junit tests	hadoop.yarn.server.resourcemanager.reservation.TestCapacityOverTimePolicy

Subsystem	Report/Notes
Docker	Client=19.03.7 Server=19.03.7 Image:yetus/hadoop:c44943d1fc3
JIRA Issue	<a href="#">YARN-9879</a>
JIRA Patch URL	<a href="https://issues.apache.org/jira/secure/attachment/12995789/YARN-9879.POC011.patch">https://issues.apache.org/jira/secure/attachment/12995789/YARN-9879.POC011.patch</a>
Optional Tests	dupname asflicense compile javac javadoc mvninstall mvnsite unit shadedclient findbugs checkstyle
uname	Linux 29b7f201cf10 4.15.0-74-generic #84-Ubuntu SMP Thu Dec 19 08:06:28 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
Build tool	maven
Personality	/testptch/patchprocess/precommit/personality/provided.sh
git revision	trunk / 004e955
maven	version: Apache Maven 3.3.9
Default Java	1.8.0_242
findbugs	v3.1.0-RC1
checkstyle	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25645/artifact/out/diff-checkstyle-root.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25645/artifact/out/diff-checkstyle-root.txt</a>
unit	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25645/artifact/out/patch-unit-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25645/artifact/out/patch-unit-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt</a>
Test Results	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25645/testReport/">https://builds.apache.org/job/PreCommit-YARN-Build/25645/testReport/</a>
asflicense	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25645/artifact/out/patch-asflicense-problems.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25645/artifact/out/patch-asflicense-problems.txt</a>
Max. process+thread count	840 (vs. ulimit of 5500)
modules	C: hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-tools/hadoop-sls U: .
Console output	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25645/console">https://builds.apache.org/job/PreCommit-YARN-Build/25645/console</a>
Powered by	Apache Yetus 0.8.0 <a href="http://yetus.apache.org">http://yetus.apache.org</a>

This message was automatically generated.

▾  **Prabhu Joseph** added a comment - 06/Mar/20 11:22

Thanks [shuzirra](#) for the patch. Have tested below scenarios with the patch and it works fine except two issues.

1. Job Submission with leaf queue name and full queue path.
2. Queue Placement
3. Auto Creation of Leaf Queue.
4. RM UI
5. RMWebService Scheduler response.
6. RMAdminService RefreshQueues
7. Scheduler Configuration Mutation API - add / remove / update queue.
8. Recovery
9. RM JMX Metrics - [YARN-9772](#)

**Issue 1: RM fails to start when a dynamic parent queue "batch" (auto-create-child-queue.enabled=true) and another leaf queue "batch" exists.**

CS Config:

root.batch -> (auto-create-child-queue.enabled=true)

root.default

root.A.batch

yarn.scheduler.capacity.queue-mappings = u:%user:batch.%user\*

```
2020-03-06 00:54:59,239 ERROR org.apache.hadoop.yarn.server.resourcemanager.ResourceManager: Error starting ResourceManager
org.apache.hadoop.service.ServiceStateException: org.apache.hadoop.yarn.exceptions.YarnException: Failed to initialize queues
    at org.apache.hadoop.service.ServiceStateException.convert(ServiceStateException.java:105)
    at org.apache.hadoop.service.AbstractService.init(AbstractService.java:173)
    at org.apache.hadoop.service.CompositeService.serviceInit(CompositeService.java:109)
    at org.apache.hadoop.yarn.server.resourcemanager.ResourceManager$RMActiveServices.serviceInit(ResourceManager.java:876)
    at org.apache.hadoop.service.AbstractService.init(AbstractService.java:164)
    at org.apache.hadoop.yarn.server.resourcemanager.ResourceManager.createAndInitActiveServices(ResourceManager.java:1288)
    at org.apache.hadoop.yarn.server.resourcemanager.ResourceManager.serviceInit(ResourceManager.java:339)
    at org.apache.hadoop.service.AbstractService.init(AbstractService.java:164)
    at org.apache.hadoop.yarn.server.resourcemanager.ResourceManager.main(ResourceManager.java:1576)
Caused by: org.apache.hadoop.yarn.exceptions.YarnException: Failed to initialize queues
    at org.apache.hadoop.yarn.server.resourcemanager.scheduler.capacity.CapacityScheduler.initializeQueues(CapacityScheduler.java:757)
    at org.apache.hadoop.yarn.server.resourcemanager.scheduler.capacity.CapacityScheduler.initScheduler(CapacityScheduler.java:342)
    at org.apache.hadoop.yarn.server.resourcemanager.scheduler.capacity.CapacityScheduler.serviceInit(CapacityScheduler.java:418)
    at org.apache.hadoop.service.AbstractService.init(AbstractService.java:164)
    ... 7 more
Caused by: java.io.IOException: mapping contains invalid or non-leaf queue [%user] and invalid parent queue [batch]
    at
    org.apache.hadoop.yarn.server.resourcemanager.placement.QueuePlacementRuleUtils.validateQueueMappingUnderParentQueue(QueuePlacementRuleUt
    at
    org.apache.hadoop.yarn.server.resourcemanager.placement.UserGroupMappingPlacementRule.validateAndGetAutoCreatedQueueMapping(UserGroupMapp
    at
```

Complete CS Config to repro above issue:

```
<configuration xmlns:xi="http://www.w3.org/2001/XInclude">

<property><name>yarn.scheduler.capacity.root.batch.leaf-queue-template.capacity</name>
<value>40</value></property>

<property><name>yarn.scheduler.capacity.queue-mappings</name>
<value>u:%user:batch.%user</value></property>

<property><name>yarn.scheduler.capacity.root.batch.auto-create-child-queue.enabled</name>
<value>true</value></property>

<property>
<name>yarn.scheduler.capacity.root.queues</name>
<value>default,batch,A</value>
</property>

<property>
<name>yarn.scheduler.capacity.queue-mappings-override.enable</name>
<value>false</value>
</property>

<property>
<name>yarn.scheduler.capacity.root.capacity</name>
```

Issue 2:

RM Starts fine with below queue config but when submitting job with queue name "A" it fails. The job submission works fine when specifying the full queue name root.B.A. There is only one leaf queue with queue name "A" and the placement has to find that right?

root.A.B

root.B.A

```
yarn jar /HADOOP/hadoop-3.3.0-SNAPSHOT/share/hadoop/mapreduce/hadoop-mapreduce-client-jobclient-3.3.0-SNAPSHOT-tests.jar sleep -
Dmapreduce.job.queueName=A -m 1 -mt 1

Caused by: org.apache.hadoop.yarn.exceptions.YarnException: Failed to submit application_1583486216805_0002 to YARN : Application
application_1583486216805_0002 submitted by user hive to unknown queue: A
    at org.apache.hadoop.yarn.client.api.impl.YarnClientImpl.submitApplication(YarnClientImpl.java:336)
    at org.apache.hadoop.mapred.ResourceMgrDelegate.submitApplication(ResourceMgrDelegate.java:304)
    at org.apache.hadoop.mapred.YARNRunner.submitJob(YARNRunner.java:331)
    ... 25 more
```

  Peter Bacsko added a comment - 06/Mar/20 11:39

prabhujoseph could you try it with this mapping rule?

u:%user:root.batch.%user

That is, you give the full path, not just the leaf queue name. Although I believe `QueueManager.get()` should be able to retrieve both.

Prabhu Joseph added a comment - 06/Mar/20 12:07

pbacsko Have tested with full queue name in mapping and which failed with different error.

```
<property><name>yarn.scheduler.capacity.queue-mappings</name>
<value>u:%user:root.batch.%user</value></property>

Caused by: java.io.IOException: mapping contains invalid or non-leaf queue [%user] and invalid parent queue which does not match existing
leaf queue's parent : [root.batch] does not match [ batch]
    at
    org.apache.hadoop.yarn.server.resourcemanager.placement.QueuePlacementRuleUtils.validateQueueMappingUnderParentQueue(QueuePlacementRuleUt
    at
    org.apache.hadoop.yarn.server.resourcemanager.placement.UserGroupMappingPlacementRule.validateAndGetAutoCreatedQueueMapping(UserGroupMapp
    at
    org.apache.hadoop.yarn.server.resourcemanager.placement.UserGroupMappingPlacementRule.initialize(UserGroupMappingPlacementRule.java:298)
    at
    org.apache.hadoop.yarn.server.resourcemanager.scheduler.capacity.CapacityScheduler.getUserGroupMappingPlacementRule(CapacityScheduler.jav
    at
    org.apache.hadoop.yarn.server.resourcemanager.scheduler.capacity.CapacityScheduler.updatePlacementRules(CapacityScheduler.java:709)
    at
    org.apache.hadoop.yarn.server.resourcemanager.scheduler.capacity.CapacityScheduler.initializeQueues(CapacityScheduler.java:750)
```

The existing way it worked is by just setting parent queue name like below.

```
<property><name>yarn.scheduler.capacity.queue-mappings</name>
<value>u:%user:batch.%user</value></property>
```

Reference: <https://hadoop.apache.org/docs/r3.2.0/hadoop-yarn/hadoop-yarn-site/CapacityScheduler.html>

```
user-group queue mapping(s) listed in yarn.scheduler.capacity.queue-mappings need to specify an additional parent queue parameter to
identify which parent queue the auto-created leaf queues need to be created under. Refer above Queue Mapping based on User or Group
section for more details. Please note that such parent queues also need to enable auto-creation of child queues as mentioned in Parent
queue configuration for dynamic leaf queue creation and management section below

Example:

<property>
  <name>yarn.scheduler.capacity.queue-mappings</name>
  <value>u:user1:queue1,g:group1:queue2,u:user2:%primary_group,u:%user:parent1.%user</value>
  <description>
    Here, u:%user:parent1.%user mapping allows any <user> other than user1,
    user2 to be mapped to its own user specific leaf queue which
    will be auto-created under <parent1>.
  </description>
</property>
```

Peter Bacsko added a comment - 06/Mar/20 12:16

prabhujoseph thanks, I think it's likely that this piece of code is missing that I mentioned here: <https://issues.apache.org/jira/browse/YARN-10108?focusedCommentId=17025143&page=com.atlassian.jira.plugin.system.issuetabpanels:comment-tabpanel#comment-17025143>

Gergely Pollák added a comment - 11/Mar/20 02:10

wangda, prabhujoseph and pbacsko thank you for your feedback. Latest patch contains a lot of fixes from Szilard's review, and some serious changes to CSQueueStore to fix the queue overwrite problem. Also added some tests for the aforementioned class. I removed the condition which prevented creating multiple leaf queues with the same name, so I expect some regression, due to the changes, but I don't expect anything serious. I'll get to implementing changes suggested in the rest of the reviews, and checking the issues reported by Prabhu. Hopefully next iteration of the patch will be something very close to committable material.

Sunil G added a comment - 11/Mar/20 02:14

Thanks shuzirra. Appreciate the same.

Hadoop QA added a comment - 11/Mar/20 06:19

-1 overall

Vote	Subsystem	Runtime	Comment
0	reexec	0m 42s	Docker mode activated.
			Prechecks
+1	@author	0m 0s	The patch does not contain any @author tags.




+1	test4tests	0m 0s	The patch appears to include 32 new or modified test files.
			<b>trunk Compile Tests</b>
0	mvndep	1m 9s	Maven dependency ordering for branch
+1	mvninstall	24m 8s	trunk passed
+1	compile	21m 59s	trunk passed
+1	checkstyle	4m 10s	trunk passed
+1	mvnsite	2m 3s	trunk passed
+1	shadedclient	24m 17s	branch has no errors when building and testing our client artifacts.
+1	findbugs	3m 3s	trunk passed
+1	javadoc	1m 27s	trunk passed
			<b>Patch Compile Tests</b>
0	mvndep	0m 24s	Maven dependency ordering for patch
+1	mvninstall	1m 18s	the patch passed
+1	compile	21m 14s	the patch passed
+1	javac	21m 14s	the patch passed
-0	checkstyle	4m 5s	root: The patch generated 129 new + 2374 unchanged - 11 fixed = 2503 total (was 2385)
+1	mvnsite	1m 55s	the patch passed
+1	whitespace	0m 0s	The patch has no whitespace issues.
+1	shadedclient	16m 39s	patch has no errors when building and testing our client artifacts.
+1	findbugs	3m 30s	the patch passed
+1	javadoc	1m 13s	the patch passed
			<b>Other Tests</b>
+1	unit	105m 24s	hadoop-yarn-server-resourcemanager in the patch passed.
-1	unit	11m 56s	hadoop-sls in the patch passed.
+1	asflicense	0m 44s	The patch does not generate ASF License warnings.
		249m 2s	

Reason	Tests
Failed junit tests	hadoop.yarn.sls.appmaster.TestAMSimulator

Subsystem	Report/Notes
Docker	Client=19.03.7 Server=19.03.7 Image:yetus/hadoop:c44943d1fc3
JIRA Issue	<a href="#">YARN-9879</a>
JIRA Patch URL	<a href="https://issues.apache.org/jira/secure/attachment/12996363/YARN-9879.POC012.patch">https://issues.apache.org/jira/secure/attachment/12996363/YARN-9879.POC012.patch</a>
Optional Tests	dupname asflicense compile javac javadoc mvninstall mvnsite unit shadedclient findbugs checkstyle
uname	Linux e3a483ad8cea 4.15.0-74-generic #84-Ubuntu SMP Thu Dec 19 08:06:28 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
Build tool	maven
Personality	/testptch/patchprocess/precommit/personality/provided.sh
git revision	trunk / cf9cf83
maven	version: Apache Maven 3.3.9
Default Java	1.8.0_242
findbugs	v3.1.0-RC1
checkstyle	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25673/artifact/out/diff-checkstyle-root.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25673/artifact/out/diff-checkstyle-root.txt</a>
unit	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25673/artifact/out/patch-unit-hadoop-tools_hadoop-sls.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25673/artifact/out/patch-unit-hadoop-tools_hadoop-sls.txt</a>
Test Results	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25673/testReport/">https://builds.apache.org/job/PreCommit-YARN-Build/25673/testReport/</a>
Max. process+thread count	820 (vs. ulimit of 5500)
modules	C: hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-tools/hadoop-sls U: .
Console output	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25673/console">https://builds.apache.org/job/PreCommit-YARN-Build/25673/console</a>

Powered by	Apache Yetus 0.8.0 <a href="http://yetus.apache.org">http://yetus.apache.org</a>
------------	--

This message was automatically generated.

▼  [Wangda Tan](#) added a comment - 11/Mar/20 17:54 - edited

Thanks [shuzirra](#) for uploading another monster patch!

I didn't check other places, I only checked the CSQueueStore related items:

\*Nits: \*

- CapacitySchedulerQueueManager#getShortNameQueues please mark @VisibleByTests
- Similarly, mark CSQueueStore#getShortNameQueues @VisibleByTests

\*Primary: Locking of the class still have many issues: \*

For all methods will be accessed by external class. Make sure that:

- 1) Avoid using synchronized lock when read/write lock present.
- 2) ALL external read-only methods protected by readlock.
- 3) ALL external writable methods protected by writelock.
- 4) Use

```
try {
    lock.(read/or write).lock()


    .. your logic ..
} catch (exception) {
    // if there's any
} finally {
    lock.(read/or write).unlock()
}
```

To make sure lock is always released: Example: CapacityScheduler#serviceStop

5) After the above changes, you can remove all usage of ConcurrentHashMap, it is bad for performance with locks. Hashmap will be way faster under the protection of lock.

▼  [Sunil G](#) added a comment - 13/Mar/20 03:53

[shuzirra](#) while uploading next patch, please remove POC string from patch name.

▼  [Gergely Pollák](#) added a comment - 13/Mar/20 03:56

[wangda](#) Thank you for the feedback, my original thought process was I will keep the getMap and fullQueue list as a concurrent hash map, since those are the most frequently read maps, and probably ConcurrentHashMap can handle the locking internally more efficiently than me externally. The other internal maps were protected by the locks you suggested. About synchronized, to be honest I simply missed removing those, I did not plan to use both, thank you for bringing it to my attention.

However the try / catch wrapping for locks was a huge deal, thanks for pointing it out.

Since the current partial locking solution was confusing, I've just simply removed the ConcurrentHashmaps, and added locking to all externally accessible methods, which access any of the internal data structures, just as you've suggested.

[prabhujoseph](#) I've managed to test the issues you've brought to my attention, and thank you for that again. Issue

1) Batch is ambiguous, so the rule will fail, if you use root.batch, as [pbacsko](#) suggested, it will work from now on, it had not before this fix, thanks for finding the issue.


2) It is a bit more tricky, I had a discussion with [wangda](#) and [wilfreds](#) and the conclusion was, if there are any queue name collisions (even between parent and leaf), we simply mark those ambiguous, and we won't allow access any of those queues by their short name. With the exception of root, root is always accessible, and always means root, root is protected, root is king. This goes against what I've said in one of my previous comments, but currently if we have these queues:

root.a.b

root.b.a

Neither a or b can be accessed by their name. It is possible to give leaf queue a priority, however in that case the ambiguity check becomes a bit harder, and the CSQueue store will become a lot more complex. But I can do it, however we need a consensus about it.

Also fixed a bunch of checkstyle issues, if there is no regression due to the changes I've made to fix Prabhu's findings, I only have to remove the TODOs and create Jiras about them, and then we are getting ready.

▼  [Hadoop QA](#) added a comment - 13/Mar/20 07:17



Vote	Subsystem	Runtime	Comment
0	reexec	0m 41s	Docker mode activated.
			<b>Prechecks</b>
+1	@author	0m 0s	The patch does not contain any @author tags.
+1	test4tests	0m 0s	The patch appears to include 32 new or modified test files.
			<b>trunk Compile Tests</b>
0	mvndep	0m 21s	Maven dependency ordering for branch
+1	mvninstall	20m 56s	trunk passed
+1	compile	17m 28s	trunk passed
+1	checkstyle	3m 11s	trunk passed
+1	mvnsite	1m 26s	trunk passed
+1	shadedclient	19m 43s	branch has no errors when building and testing our client artifacts.
+1	findbugs	2m 22s	trunk passed
+1	javadoc	1m 6s	trunk passed
			<b>Patch Compile Tests</b>
0	mvndep	0m 20s	Maven dependency ordering for patch
+1	mvninstall	1m 7s	the patch passed
+1	compile	15m 17s	the patch passed
+1	javac	15m 17s	the patch passed
-0	checkstyle	3m 9s	root: The patch generated 12 new + 2369 unchanged - 16 fixed = 2381 total (was 2385)
+1	mvnsite	1m 25s	the patch passed
+1	whitespace	0m 0s	The patch has no whitespace issues.
+1	shadedclient	14m 24s	patch has no errors when building and testing our client artifacts.
+1	findbugs	2m 36s	the patch passed
+1	javadoc	1m 4s	the patch passed
			<b>Other Tests</b>
+1	unit	88m 17s	hadoop-yarn-server-resourcemanager in the patch passed.
+1	unit	12m 5s	hadoop-sls in the patch passed.
+1	asflicense	0m 46s	The patch does not generate ASF License warnings.
		206m 27s	

Subsystem	Report/Notes
Docker	Client=19.03.8 Server=19.03.8 Image:yetus/hadoop:c44943d1fc3
JIRA Issue	<a href="#">YARN-9879</a>
JIRA Patch URL	<a href="https://issues.apache.org/jira/secure/attachment/12996614/YARN-9879.POC013.patch">https://issues.apache.org/jira/secure/attachment/12996614/YARN-9879.POC013.patch</a>
Optional Tests	dupname asflicense compile javac javadoc mvninstall mvnsite unit shadedclient findbugs checkstyle
uname	Linux 07aeff3c839e 4.15.0-74-generic #84-Ubuntu SMP Thu Dec 19 08:06:28 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
Build tool	maven
Personality	/testptch/patchprocess/precommit/personality/provided.sh
git revision	trunk / 20903f7
maven	version: Apache Maven 3.3.9
Default Java	1.8.0_242
findbugs	v3.1.0-RC1
checkstyle	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25686/artifact/out/diff-checkstyle-root.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25686/artifact/out/diff-checkstyle-root.txt</a>
Test Results	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25686/testReport/">https://builds.apache.org/job/PreCommit-YARN-Build/25686/testReport/</a>
Max. process+thread count	818 (vs. ulimit of 5500)
modules	C: hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-tools/hadoop-sls U: .

Console output	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25686/console">https://builds.apache.org/job/PreCommit-YARN-Build/25686/console</a>
Powered by	Apache Yetus 0.8.0 <a href="http://yetus.apache.org">http://yetus.apache.org</a>

This message was automatically generated.

Wangda Tan added a comment - 15/Mar/20 01:14

Thanks shuzirra for the update.  
I only checked the updates of CSQueueStore, now the class looks good to me. I will let others to check the rest of the patch. 😊

Gergely Pollák added a comment - 17/Mar/20 13:02 - edited

In the latest patch, there are mostly cosmetic changes. I've removed most of the TODOs as wangda and snemeth suggested, will create follow up Jira based on those. Fixed additional checkstyle issues, and added a proper error message when a queue is ambiguously referenced during submission, so the user will now get feedback, that the queue exists, but actually multiple queues exist with the same name.  
Also dropped the POC prefix, since it's getting near completion, so the latest patch is  
[YARN-9879.014.patch](#)

Hadoop QA added a comment - 17/Mar/20 17:10



 -1 overall

Vote	Subsystem	Runtime	Comment
0	reexec	32m 57s	Docker mode activated.
			Prechecks
+1	@author	0m 0s	The patch does not contain any @author tags.
+1	test4tests	0m 0s	The patch appears to include 32 new or modified test files.
			trunk Compile Tests
0	mvndep	1m 30s	Maven dependency ordering for branch
+1	mvninstall	20m 20s	trunk passed
+1	compile	16m 28s	trunk passed
+1	checkstyle	3m 9s	trunk passed
+1	mvnsite	1m 28s	trunk passed
+1	shadedclient	19m 57s	branch has no errors when building and testing our client artifacts.
+1	findbugs	2m 21s	trunk passed
+1	javadoc	1m 8s	trunk passed
			Patch Compile Tests
0	mvndep	0m 20s	Maven dependency ordering for patch
+1	mvninstall	1m 5s	the patch passed
+1	compile	15m 44s	the patch passed
+1	javac	15m 44s	the patch passed
-0	checkstyle	3m 10s	root: The patch generated 2 new + 2367 unchanged - 16 fixed = 2369 total (was 2383)
+1	mvnsite	1m 26s	the patch passed
+1	whitespace	0m 0s	The patch has no whitespace issues.
+1	shadedclient	14m 20s	patch has no errors when building and testing our client artifacts.
+1	findbugs	2m 40s	the patch passed
+1	javadoc	1m 6s	the patch passed
			Other Tests
-1	unit	88m 49s	hadoop-yarn-server-resourcemanager in the patch passed.
+1	unit	12m 2s	hadoop-sls in the patch passed.
+1	asflicense	0m 56s	The patch does not generate ASF License warnings.
		239m 52s	

Reason	Tests
Failed junit tests	hadoop.yarn.server.resourcemanager.security.TestDelegationTokenRenewer

Subsystem	Report/Notes
Docker	Client=19.03.8 Server=19.03.8 Image:yetus/hadoop:c44943d1fc3
JIRA Issue	<a href="#">YARN-9879</a>
JIRA Patch URL	<a href="https://issues.apache.org/jira/secure/attachment/12996923/YARN-9879.014.patch">https://issues.apache.org/jira/secure/attachment/12996923/YARN-9879.014.patch</a>
Optional Tests	dupname asflicense compile javac javadoc mvninstall mvnsite unit shadedclient findbugs checkstyle
uname	Linux bfe2905efa8e 4.15.0-74-generic #84-Ubuntu SMP Thu Dec 19 08:06:28 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
Build tool	maven
Personality	/testptch/patchprocess/precommit/personality/provided.sh
git revision	trunk / 8d63734
maven	version: Apache Maven 3.3.9
Default Java	1.8.0_242
findbugs	v3.1.0-RC1
checkstyle	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25701/artifact/out/diff-checkstyle-root.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25701/artifact/out/diff-checkstyle-root.txt</a>
unit	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25701/artifact/out/patch-unit-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25701/artifact/out/patch-unit-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt</a>
Test Results	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25701/testReport/">https://builds.apache.org/job/PreCommit-YARN-Build/25701/testReport/</a>
Max. process+thread count	820 (vs. ulimit of 5500)
modules	C: hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-tools/hadoop-sls U: .
Console output	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25701/console">https://builds.apache.org/job/PreCommit-YARN-Build/25701/console</a>
Powered by	Apache Yetus 0.8.0 <a href="http://yetus.apache.org">http://yetus.apache.org</a>

This message was automatically generated.

  Hadoop QA added a comment - 18/Mar/20 13:06

 -1 overall

Vote	Subsystem	Runtime	Comment
0	reexec	0m 38s	Docker mode activated.
			<b>Prechecks</b>
+1	@author	0m 0s	The patch does not contain any @author tags.
+1	test4tests	0m 0s	The patch appears to include 32 new or modified test files.
			<b>trunk Compile Tests</b>
0	mvndep	0m 57s	Maven dependency ordering for branch
+1	mvninstall	19m 51s	trunk passed
+1	compile	15m 48s	trunk passed
+1	checkstyle	3m 10s	trunk passed
+1	mvnsite	1m 26s	trunk passed
+1	shadedclient	20m 27s	branch has no errors when building and testing our client artifacts.
+1	findbugs	2m 23s	trunk passed
+1	javadoc	1m 5s	trunk passed
			<b>Patch Compile Tests</b>
0	mvndep	0m 22s	Maven dependency ordering for patch
+1	mvninstall	1m 7s	the patch passed
+1	compile	15m 32s	the patch passed
+1	javac	15m 32s	the patch passed
-0	checkstyle	4m 4s	root: The patch generated 2 new + 2369 unchanged - 16 fixed = 2371 total (was 2385)
+1	mvnsite	1m 31s	the patch passed
+1	whitespace	0m 0s	The patch has no whitespace issues.
+1	shadedclient	14m 53s	patch has no errors when building and testing our client artifacts.

+1	findbugs	2m 39s	the patch passed
+1	javadoc	1m 5s	the patch passed
			Other Tests
-1	unit	94m 49s	hadoop-yarn-server-resourcemanager in the patch passed.
-1	unit	11m 48s	hadoop-sls in the patch passed.
+1	asflicense	0m 44s	The patch does not generate ASF License warnings.
		213m 8s	

Reason	Tests
Failed junit tests	hadoop.yarn.server.resourcemanager.scheduler.fair.TestFairSchedulerPreemption
	hadoop.yarn.sls.TestSLSRunner

Subsystem	Report/Notes
Docker	Client=19.03.8 Server=19.03.8 Image:yetus/hadoop:c44943d1fc3
JIRA Issue	<a href="#">YARN-9879</a>
JIRA Patch URL	<a href="https://issues.apache.org/jira/secure/attachment/12996923/YARN-9879.014.patch">https://issues.apache.org/jira/secure/attachment/12996923/YARN-9879.014.patch</a>
Optional Tests	dupname asflicense compile javac javadoc mvninstall mvnsite unit shadedclient findbugs checkstyle
uname	Linux b8c6f5cfa774 4.15.0-74-generic #84-Ubuntu SMP Thu Dec 19 08:06:28 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
Build tool	maven
Personality	/testptch/patchprocess/precommit/personality/provided.sh
git revision	trunk / 8d63734
maven	version: Apache Maven 3.3.9
Default Java	1.8.0_242
findbugs	v3.1.0-RC1
checkstyle	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25706/artifact/out/diff-checkstyle-root.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25706/artifact/out/diff-checkstyle-root.txt</a>
unit	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25706/artifact/out/patch-unit-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25706/artifact/out/patch-unit-hadoop-yarn-project_hadoop-yarn_hadoop-yarn-server_hadoop-yarn-server-resourcemanager.txt</a>
unit	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25706/artifact/out/patch-unit-hadoop-tools_hadoop-sls.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25706/artifact/out/patch-unit-hadoop-tools_hadoop-sls.txt</a>
Test Results	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25706/testReport/">https://builds.apache.org/job/PreCommit-YARN-Build/25706/testReport/</a>
Max. process+thread count	816 (vs. ulimit of 5500)
modules	C: hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-tools/hadoop-sls U: .
Console output	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25706/console">https://builds.apache.org/job/PreCommit-YARN-Build/25706/console</a>
Powered by	Apache Yetus 0.8.0 <a href="http://yetus.apache.org">http://yetus.apache.org</a>

This message was automatically generated.

- Sunil G added a comment - 19/Mar/20 06:33

shuzirra cud pls check latest errors.
- Gergely Pollák added a comment - 19/Mar/20 13:40

sunilg

The latest errors are flaky tests, we keep retriggering the jenkins job to get a green, but if you check the last 2 runs, completely different tests fail, for the same patch set. Also last patch only contains cosmetic and log message changes, so it shouldn't cause any issue, but trying to get a jenkins +1
- Sunil G added a comment - 21/Mar/20 14:00

Somehow jenkins results are not showing here.
- Gergely Pollák added a comment - 24/Mar/20 11:39

Somehow <https://issues.apache.org/jira/browse/YARN-10198> was committed, so I had to rebase and resolve, uploading new patch for testing.
- Gergely Pollák added a comment - 24/Mar/20 19:05



▼

Hadoop QA added a comment - 24/Mar/20 23:07

✖

-1 overall

Vote	Subsystem	Runtime	Comment
0	reexec	31m 0s	Docker mode activated.
			Prechecks
+1	@author	0m 0s	The patch does not contain any @author tags.
+1	test4tests	0m 0s	The patch appears to include 32 new or modified test files.
			trunk Compile Tests
0	mvndep	1m 12s	Maven dependency ordering for branch
+1	mvninstall	20m 9s	trunk passed
+1	compile	15m 46s	trunk passed
+1	checkstyle	3m 12s	trunk passed
+1	mvnsite	1m 24s	trunk passed
+1	shadedclient	19m 37s	branch has no errors when building and testing our client artifacts.
+1	findbugs	2m 24s	trunk passed
+1	javadoc	1m 5s	trunk passed
			Patch Compile Tests
0	mvndep	0m 19s	Maven dependency ordering for patch
+1	mvninstall	1m 5s	the patch passed
+1	compile	15m 15s	the patch passed
+1	javac	15m 15s	the patch passed
-0	checkstyle	3m 9s	root: The patch generated 4 new + 2366 unchanged - 16 fixed = 2370 total (was 2382)
+1	mvnsite	1m 25s	the patch passed
+1	whitespace	0m 0s	The patch has no whitespace issues.
+1	shadedclient	14m 33s	patch has no errors when building and testing our client artifacts.
+1	findbugs	2m 39s	the patch passed
+1	javadoc	1m 6s	the patch passed
			Other Tests
+1	unit	88m 46s	hadoop-yarn-server-resourcemanager in the patch passed.
-1	unit	10m 36s	hadoop-slfs in the patch passed.
+1	asflicense	0m 47s	The patch does not generate ASF License warnings.
		234m 31s	

Reason	Tests
Failed junit tests	hadoop.yarn.sls.TestReservationSystemInvariants


Subsystem	Report/Notes
Docker	Client=19.03.8 Server=19.03.8 Image:yetus/hadoop:4454c6d14b7
JIRA Issue	<a href="#">YARN-9879</a>
JIRA Patch URL	<a href="https://issues.apache.org/jira/secure/attachment/12997595/YARN-9879.015.patch">https://issues.apache.org/jira/secure/attachment/12997595/YARN-9879.015.patch</a>
Optional Tests	dupname asflicense compile javac javadoc mvninstall mvnsite unit shadedclient findbugs checkstyle
uname	Linux 431dcc001ba6 4.15.0-74-generic #84-Ubuntu SMP Thu Dec 19 08:06:28 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
Build tool	maven
Personality	/testptch/patchprocess/precommit/personality/provided.sh

git revision	trunk / d9c4f11
maven	version: Apache Maven 3.3.9
Default Java	1.8.0_242
findbugs	v3.1.0-RC1
checkstyle	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25746/artifact/out/diff-checkstyle-root.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25746/artifact/out/diff-checkstyle-root.txt</a>
unit	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25746/artifact/out/patch-unit-hadoop-tools_hadoop-sls.txt">https://builds.apache.org/job/PreCommit-YARN-Build/25746/artifact/out/patch-unit-hadoop-tools_hadoop-sls.txt</a>
Test Results	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25746/testReport/">https://builds.apache.org/job/PreCommit-YARN-Build/25746/testReport/</a>
Max. process+thread count	830 (vs. ulimit of 5500)
modules	C: hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager hadoop-tools/hadoop-sls U: .
Console output	<a href="https://builds.apache.org/job/PreCommit-YARN-Build/25746/console">https://builds.apache.org/job/PreCommit-YARN-Build/25746/console</a>
Powered by	Apache Yetus 0.8.0 <a href="http://yetus.apache.org">http://yetus.apache.org</a>

This message was automatically generated.

▼  [Sunil G](#) added a comment - 25/Mar/20 02:52

[shuzirra](#) test case failures seems not related, could you please confirm the same?

▼  [Gergely Pollák](#) added a comment - 25/Mar/20 09:48

[sunilg](#) yes, that is correct, it is unrelated, and SLS tests are failing quite often with no real reason. But to be sure I've executed the test case a few times manually, and it was passing properly, so this one seems to be a flaky one.

▼  [Sunil G](#) added a comment - 25/Mar/20 11:02

Thanks [shuzirra](#)

Lets get this in now. +1 to the latest patch.

▼  [Sunil G](#) added a comment - 25/Mar/20 11:32

Thanks [shuzirra](#) for your contributions. Appreciate the same.

Thanks [wangda](#) [snemeth](#) [pbacsko](#) [prabhujoseph](#) for your valuable comments

pushed to trunk

▼  [Hudson](#) added a comment - 25/Mar/20 11:46

SUCCESS: Integrated in Jenkins build Hadoop-trunk-Commit #18085 (See <https://builds.apache.org/job/Hadoop-trunk-Commit/18085/>)  
[YARN-9879](#). Allow multiple leaf queues with the same name in (sunilg: rev cdb2107066a2d8557270888c0a9a75f29a6853bf)

- (edit) hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/webapp/TestRMWebServicesSchedulerActivities.java
- (edit) hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/webapp/TestRMWebServicesForCSWithPartitions.java
- (add) hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/CSQueueStore.java
- (edit) hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestCapacitySchedulerQueueMappingFactory.jav
- (edit) 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
- (edit) hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/placement/QueueMappingEntity.java
- (edit) hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestApplicationLimitsByPartition.java
- (edit) hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestAbsoluteResourceConfiguration.java
- (edit) hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestCapacitySchedulerPerf.java
- (edit) 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
- (add) hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestCSQueueStore.java
- (edit) hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/AbstractYarnScheduler.java

- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/monitor/capacity/FifoIntraQueuePreemptionPlugin.java`
- (edit) `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`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/activities/ActivitiesManager.java`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/placement/QueuePlacementRuleUtils.java`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/reservation/TestReservationSystem.java`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/monitor/capacity/ProportionalCapacityPreemptionPolicy.java`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/SchedulerApplicationAttempt.java`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/WorkflowPriorityMappingsManager.java`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestCapacitySchedulerNodeLabelUpdate.java`
- (edit) `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`
- (edit) `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`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/monitor/capacity/TestProportionalCapacityPreemptionPolicyForReset.java`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/reservation/ReservationSystemTestUtil.java`
- (edit) `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`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/UsersManager.java`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestQueueState.java`
- (edit) `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`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/CapacitySchedulerQueueManager.java`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/security/QueueACLsManager.java`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestCapacitySchedulerAutoCreatedQueueBase.java`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestCapacitySchedulerSurgicalPreemption.java`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/monitor/capacity/CapacitySchedulerPreemptionUtils.java`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/placement/UserGroupMappingPlacementRule.java`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/preemption/PreemptionManager.java`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/AbstractAutoCreatedLeafQueue.java`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/AbstractManagedParentQueue.java`
- (edit) `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`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestCapacitySchedulerAutoQueueCreation.java`
- (edit) `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`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/placement/AppNameMappingPlacementRule.java`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/placement/TestUserGroupMappingPlacementRule.java`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/AutoCreatedLeafQueue.java`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/ManagedParentQueue.java`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/common/fica/FiCaSchedulerApp.java`
- (edit) `hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/CapacitySchedulerConfigValidator.java`

- (edit) [hadoop-tools/hadoop-sls/src/main/java/org/apache/hadoop/yarn/sls/scheduler/SLSCapacityScheduler.java](#)
- (edit) [hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/monitor/capacity/TestProportionalCapacityPreemptionPolicy.java](#)
- (edit) [hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/queuemanagement/GuaranteedOrZeroCapacity](#)
- (edit) [hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/monitor/capacity/TestProportionalCapacityPreemptionPolicyPreempt](#)
- (edit) [hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/placement/QueuePath.java](#)
- (edit) [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](#)
- (edit) [hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/YarnScheduler.java](#)
- (edit) [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](#)
- (edit) [hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/policy/TestPriorityUtilizationQueueOrderingPolicy](#)
- (edit) [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](#)
- (edit) [hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/placement/QueueMapping.java](#)
- (edit) [hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/QueueManagementChange.java](#)
- (edit) [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](#)
- (edit) [hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/TestReservationSystemWithRMHA.java](#)
- (edit) [hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/reservation/CapacityReservationSystem.java](#)
- (edit) [hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/TestAbstractYarnScheduler.java](#)
- (edit) [hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/placement/TestAppNameMappingPlacementRule.java](#)
- (edit) [hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/monitor/capacity/TestProportionalCapacityPreemptionPolicyIntraQue](#)
- (edit) [hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/TestCapacitySchedulerLazyPreemption.java](#)
- (edit) [hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/test/java/org/apache/hadoop/yarn/server/resourcemanager/monitor/capacity/ProportionalCapacityPreemptionPolicyMockFramew](#)
- (edit) [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](#)
- (edit) [hadoop-yarn-project/hadoop-yarn/hadoop-yarn-server/hadoop-yarn-server-resourcemanager/src/main/java/org/apache/hadoop/yarn/server/resourcemanager/scheduler/capacity/QueueManagementDynamicEditPolicy.java](#)

▼  Akira Aijisaka added a comment - 06/Apr/20 02:05

Hi [shuzirra](#) and [sunilq](#). Probably this broke [MAPREDUCE-7269](#). Would you check the PR?

Now I'm thinking we need to mark this issue as "incompatible change" because this issue changed the output of `o.a.h.mapred.JobClient#getQueueAclsForCurrentUser`, which is `@public` and `@stable`. What do you think?

▼  Nie Gus added a comment - 30/Dec/20 14:07

Hi shuzirra ,

really appreciate this patch, we are using this patch in our branch, it worked quite well, but still we found there are lots of place still using "getQueueName" , and we also see "fullPathQueueNamingPolicy" could change the output of getQueueName to queuePath directly, but the code is set it to final false. Is that for future work ? or something still block the directly replacement between queueName and queuePath, so currently we set it false, should we consider to change it into conf ?

[illegible]

```
private final boolean fullPathQueueNamingPolicy = false;
```

```
@Override
public String getQueueName() {
    if (fullPathQueueNamingPolicy)
    { return queuePath; }

    return queueName;
}
```

▼ People

---

Assignee:

 Gergely Pollák

Reporter:

 Gergely Pollák

Votes:

 0 Vote for this issue

Watchers:

 12 Start watching this issue

▼ Dates

---

Created:

08/Oct/19 15:22

Updated:

30/Dec/20 14:07

Resolved:

25/Mar/20 11:32