

## Managing an Apache Hive workload

**Date of Publish:** 2019-08-26



# Contents

<b>Workload management.....</b>	<b>3</b>
<b>Setting up and using a resource plan.....</b>	<b>5</b>
Create a resource plan.....	5
Enable a resource plan.....	6
Activate a resource plan.....	7
Query sys database for plan data.....	7
Disable a resource plan.....	8
<b>Configure a YARN queue for workload management.....</b>	<b>8</b>
<b>Workload management entity data in sys.....</b>	<b>10</b>

## Workload management

As administrator, you manage workloads by creating resource plans to improve parallel query execution and cluster sharing of queries using low-latency analytical processing (LLAP).

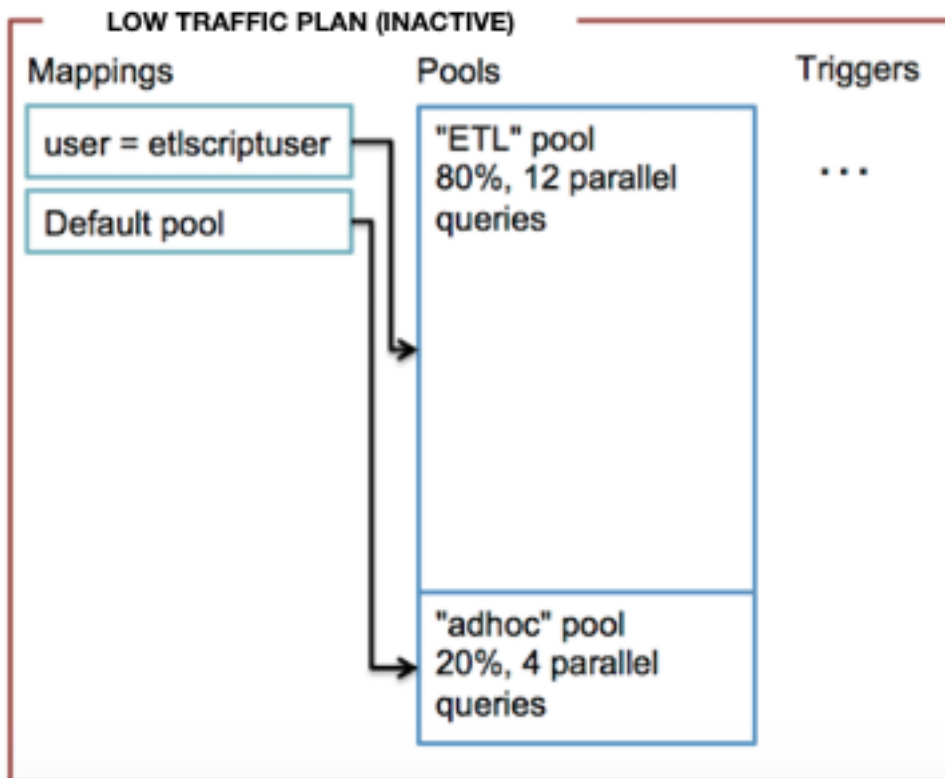
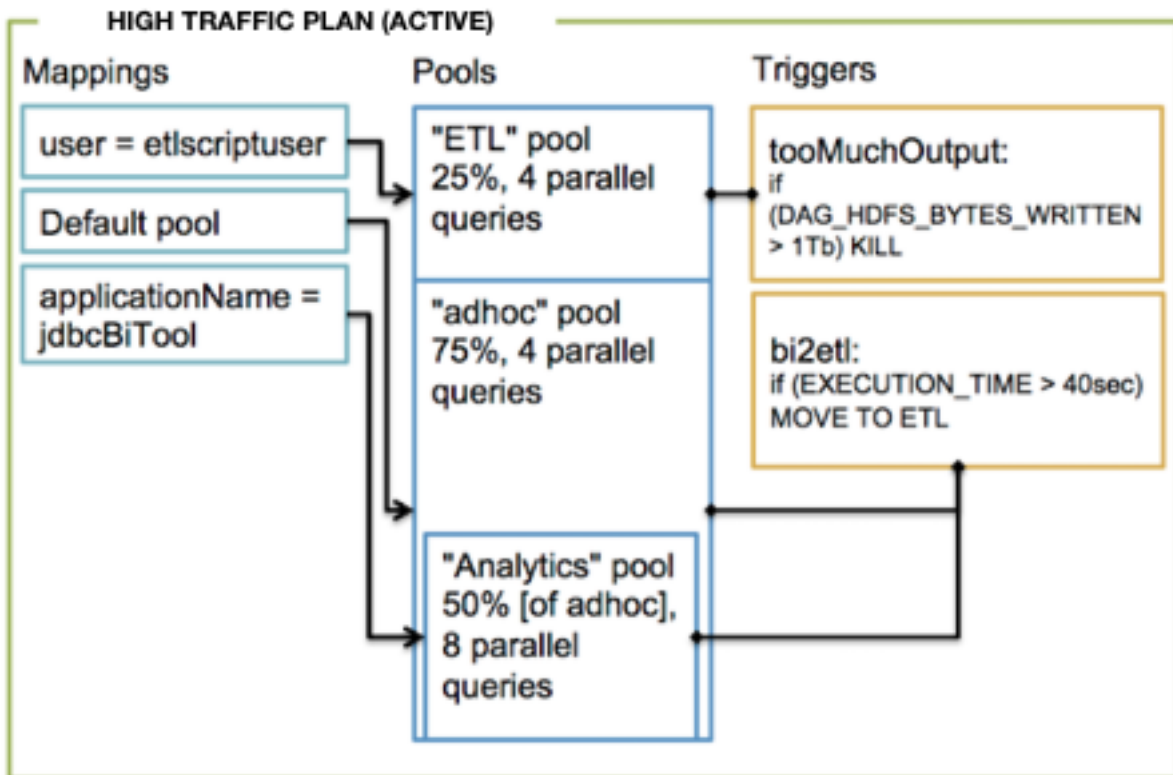
The resource plan is a self-contained, resource-sharing configuration. One resource plan is active on the cluster at a time. Typically, enabling and disabling a resource plan on a live cluster does not affect running queries. You, as administrator, or your script can apply resource plans that configure the cluster for different situations. For example, your script can apply a resource plan that configures the cluster to handle high traffic. When traffic decreases, you can switch the resource plan to support traffic for interactive data visualization, deep ad-hoc analytics, and large-scale BI reporting.

You can create a resource plan to meet a defined data processing benchmark when a workload reaches a high volume of concurrent queries. For example, consider an enterprise that has an ad-hoc analytics application exposed to around 100 analysts. Data set and query patterns dictate that a generated query executes within a few seconds. A resource plan can ensure that when up to 100 users might concurrently use the system, at least 95 percent of queries complete in fewer than 15 seconds.

A resource plan can consist of one or more query pools, mappings, and triggers:

- A query pool shares resources with LLAP cluster processes and queries within the pool, and sets the maximum concurrent queries.
- A mapping routes incoming queries to pools based on specified factors, such as user name, group, or application.
- A trigger initiates an action, such as killing queries in a pool or all queries running in a cluster, based on query metrics represented by Apache Hadoop, Tez, and Hive counters.

The following diagrams depict a simple resource plan. The first diagram shows a resource plan designed for high traffic periods and the second for low traffic periods.



#### Related Information

[Hive Workload Management Command Summary](#)

[Workload trigger counters](#)

[Configure a YARN queue for workload management](#)

## Setting up and using a resource plan

Understanding the prerequisites for using workload management is critical. Prerequisites are running LLAP and setting up a YARN queue for workload management in addition to the LLAP queue. As administrator, you can define a resource plan using DDL statements that typically include event-triggered resource pools for different users or applications. You can then manage workloads using the resource plan. After creating and enabling a resource plan, you must activate it. When you want to alter a resource plan or stop using it, you must disable it.

### Before you begin

- Set up and run Hive low-latency analytical processing (LLAP), which includes a YARN queue configured for LLAP.  
Ambari sets `hive.llap.daemon.queue.name` to the name of this queue.
- Set up a second YARN queue for interactive workload management.
- Add a custom property `hive.server2.tez.interactive.queue` to `hiveserver2-interactive-site` that names the interactive workload management queue.

### Related Information

[Hive Workload Management Command Summary](#)

[Workload trigger counters](#)

[Configure a YARN queue for workload management](#)

[Set up LLAP](#)

[Configure a YARN queue for workload management](#)

## Create a resource plan

As administrator, you can create resource plans that configure your cluster for different conditions, enabling you to improve parallel query execution and share queries on nodes running LLAP. You can allocate resources to a pool of users, groups, or applications and trigger actions in a plan.

### Before you begin

You started Hive from the Beeline shell.

### About this task

As administrator, you can either create a new resource plan or clone another plan and then modify it.

### Procedure

#### 1. Create a resource plan.

- Create a new resource plan (for example, one named `high_concurr_rp`):

```
CREATE RESOURCE PLAN high_concurr_rp;
```

- Clone an existing resource plan (for example, `ha_plan`) and rename it `high_concurr_rp`:

```
CREATE RESOURCE PLAN high_concurr_rp LIKE ha_plan;
```

- Create a new resource plan (for example, named `high_concurr_rp`) and specify a query concurrency limit:

```
CREATE RESOURCE PLAN high_concurr_rp WITH QUERY_PARALLELISM=15;
```

2. Optionally, add one or more query pools to the plan.

```
CREATE POOL high_concurr_rp.bi_pool
WITH ALLOC_FRACTION=75,QUERY_PARALLELISM=5;

CREATE POOL high_concurr_rp.etl_pool
WITH ALLOC_FRACTION=25,QUERY_PARALLELISM=10;
```

This example configuration allocates 75 percent of the LLAP cluster resources to one pool and 25 percent to the other, and allows 5 concurrent queries by users in one pool and 10 concurrent queries by users in the other.

3. Optionally, map an application to a pool.

```
CREATE APPLICATION MAPPING tableau_to_bi IN bi_pool;
```

This example mapping of the Tableau application to the `bi_pool` requires Tableau to connect to HiveServer using a JDBC connection string that includes `applicationName="Tableau"`. In a mapping of a group to a pool, group authorization depends on the HDFS group configuration on the cluster. You can configure Lightweight Directory Protocol (LDAP) and other mechanisms.

4. Optionally, create a trigger. For example, create a trigger that moves a query to the `etl_pool` for execution when the execution time exceeds one minute.

```
CREATE TRIGGER high_concurr_rp.slow_query
WHEN execution_time_ms > 60000
DO MOVE TO etl_pool;
```

5. If you are running the LLAP service, add a trigger to the pool or resource plan as shown in the following example.  
`ALTER TRIGGER high_concurr_rp.slow_query ADD TO POOL bi_pool;`

You must add a trigger to a pool or resource plan when running under LLAP; otherwise, the trigger is considered global and runs only in Tez.

### Related Information

[CREATE MAPPING](#)

[CREATE POOL](#)

[CREATE TRIGGER](#)

[Hive Workload Management Command Summary](#)

[Workload trigger counters](#)

## Enable a resource plan

You can enable, and optionally activate, a resource plan to manage a workload by running a query.

### Before you begin

- You connected to HiveServer over ODBC or JDBC on the cluster you want to manage.
- You started Hive from the Beeline shell.

### About this task

Part of the procedure for enabling the resource plan is to validate the plan. Enabling the plan does not apply the plan to a query workload. You cannot modify the plan while enabled. You can enable multiple plans, but you can activate only one at a time on a cluster. Enabling and activating the plan also validates the plan.

### Procedure

1. Validate the resource plan called `myplan`.  
`ALTER RESOURCE PLAN myplan VALIDATE;`

2. Enable the resource plan using one of the following methods, depending on whether or not you want to activate the plan now or later and if you already have another plan activated.

- Enable the resource plan:

```
ALTER RESOURCE PLAN myplan ENABLE;
```

- Enable and activate the resource plan:

```
ALTER RESOURCE PLAN myplan ENABLE ACTIVATE;
```

- Archive the active resource plan and replace it with the named resource plan:

```
ALTER RESOURCE PLAN myplan ENABLE ACTIVATE WITH REPLACE
```

### Related Information

[Start a Hive shell locally](#)

[Hive Workload Management Command Summary](#)

[Workload trigger counters](#)

## Activate a resource plan

You can activate an enabled resource plan to apply the plan to a query workload to start managing the workload.

### Before you begin

- You connected to HiveServer over ODBC or JDBC on the cluster you want to manage, and started Hive from the Beeline shell.
- You enabled the resource plan.

### About this task

You can make only one resource plan active at a time.

### Procedure

1. Activate a resource plan.  
`ALTER RESOURCE PLAN plan1 ACTIVATE;`  
This example activates plan1.
2. Deactivate one resource plan by activating another.  
`ALTER RESOURCE PLAN plan2 ACTIVATE;`  
This example activates plan2 and deactivates plan1, but plan1 is still enabled.

### Related Information

[Start a Hive shell locally](#)

[Hive Workload Management Command Summary](#)

[Workload trigger counters](#)

## Query sys database for plan data

You query the sys database in the Hive Metastore to retrieve information about workload management entities, such as a resource plan.

### Before you begin

- You established a connection to HiveServer on the cluster you want to manage, and you started Hive from the Beeline shell, for example.

- A resource plan for managing a workload on your cluster was created.

### About this task

Assuming you created the `high_concurr_rp` resource plan as described earlier, query the `sys` database for information about the plan.

### Procedure

1. Get views of the query pools for the resource plan, for example.

```
SELECT * FROM SYS.WM_POOLS WHERE RP_NAME = 'high_concurr_rp';
```

2. Get information about a trigger named `slow_query` in the pool named `etl_pool` pool of the plan.

```
SELECT TRIGGER_NAME.slow_query FROM SYS.WM_POOLS_TO_TRIGGERS WHERE RP_NAME = 'high_concurr_rp';
```

## Disable a resource plan

You can disable a resource plan to modify it or to stop managing a workload by running a query.

### Before you begin

- You connected to HiveServer over ODBC or JDBC on the cluster you want to manage.
- You started Hive from the Beeline shell.

### Procedure

Disable the resource plan.

```
ALTER RESOURCE PLAN myplan DISABLE;
```

### Related Information

[Start a Hive shell locally](#)

[Hive Workload Management Command Summary](#)

[Workload trigger counters](#)

## Configure a YARN queue for workload management

You must add a custom YARN queue for workload management if you enable workload management; otherwise, it is likely that workload management will not start. This queue is in addition to the LLAP queue, named `llap` by default that is also required for interactive querying and workload management.

### About this task

The `llap` queue capacity is reserved for launching LLAP daemons and pre-provisioned YARN Application Masters (AMs), which correspond to Maximum Total Concurrent Queries that you configure during LLAP setup. Typically, the entire `llap` queue is used when LLAP daemons and AMs are running. There is insufficient capacity for workload management to use the same `llap` queue because workload management requires more AMs/sessions. Configuring an interactive YARN queue for workload management solves this problem.

Configuration of the workload management queue includes specifying capacity. You need to consider the total cluster capacity, the capacity you have allocated to the `llap` queue, and the workloads you want to accommodate. For example:

- Total cluster capacity = 100 GB

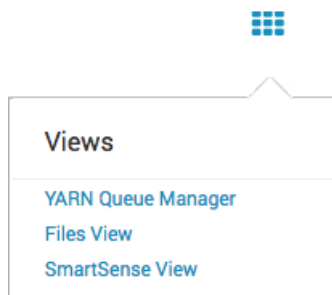


- llap queue capacity = 50 GB
- Workload management queue capacity = 20 GB

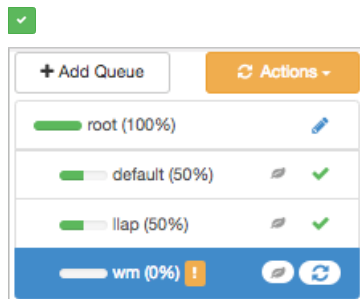
Workload management can launch only 10 AMs/sessions of 2GB container size each. A resource plan that defines query\_parallelism greater than 10 does not work (20 \* 2GB container = 40GB and only 20GB capacity is available). Plan carefully to create resource plans in accordance with the workload management queue capacity.

### Procedure

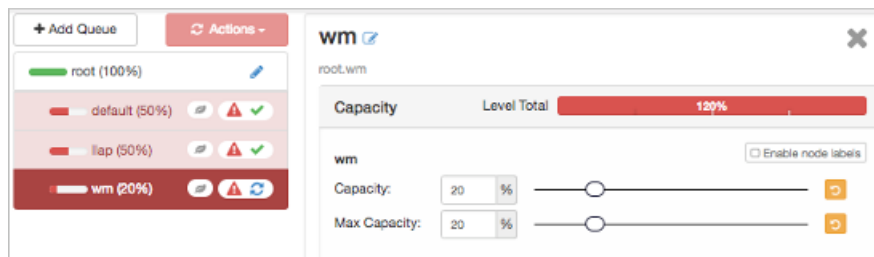
1. In Ambari, select Services > YARN > Configs, and from the hamburger menu Views (top right), select YARN Queue Manager.



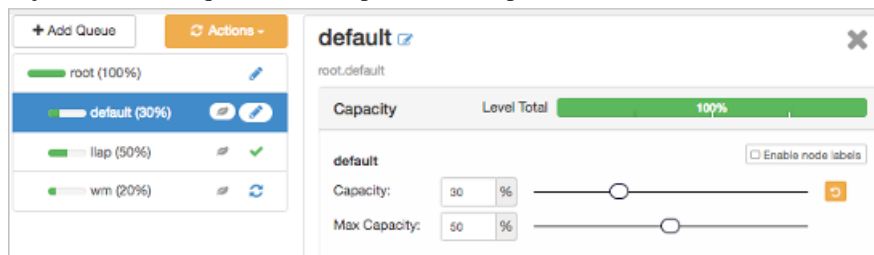
2. Click Add Queue, and enter a queue path (name) wm, for example. Click



3. Configure the capacity of the wm queue to greater than zero.



4. Make adjustments in the capacity and max capacity of other queues as necessary to resolve the error. For example, adjust the default queue from 50 percent to 30 percent:



5. Select the wm queue under Add Queue, and in Resources that appears on the right, set User Limit Factor to 1, and set Priority to greater than 0 (1 for example).
6. Select **Actions** > **Save** and Refresh Queues.
7. In **Services** > **YARN** > **Summary** restart any YARN services as prompted.

8. In Ambari, navigate to Services > Hive > Configs > Advanced > Custom hiveserver2-interactive-site.
9. Add the following property and set its value to the name of the workload management queue, wm, for example:  
Key: hive.server2.tez.interactive.queue Value: wm
10. Save the configuration.

## Workload management entity data in sys

From the Hive sys database, you can get information about workload management and other Hive entities.

### Workload Management Views

Hive has a special database named sys, which is similar to a system catalog or information catalog. The sys database is part of the Hive Metastore. In the sys database you can query for views of all Hive entities, including workload management entities. The following workload management views are available:

- SYS.WM\_RESOURCEPLANS (NAME string, STATUS string, QUERY\_PARALLELISM int, DEFAULT\_POOL\_PATH string)
- SYS.WM\_POOLS (RP\_NAME string, PATH string, ALLOC\_FRACTION double, QUERY\_PARALLELISM int, SCHEDULING\_POLICY string)
- SYS.WM\_MAPPINGS (RP\_NAME string, ENTITY\_TYPE string, ENTITY\_NAME string, POOL\_PATH string, ORDERING int)
- SYS.WM\_TRIGGERS (RP\_NAME string, NAME string, TRIGGER\_EXPRESSION string, ACTION\_EXPRESSION string)
- SYS.WM\_POOLS\_TO\_TRIGGERS (RP\_NAME string, POOL\_PATH string, TRIGGER\_NAME string)