

Cluster Utilization Report overview

The Cluster Utilization Report screens in Cloudera Manager display aggregated utilization in Hadoop and Impala jobs.

The reports display CPU utilization, memory utilization, resource allocations made due to the scheduler, and Impala queries. The report displays aggregated utilization for the entire cluster, utilization by tenant, which is either a user or a resource pool. You can configure the report by a range of dates, specific days of the week, and time ranges.

The report displays the current utilization of CPU and memory resources and the resources that are used by the Cloudera Manager resource management features.

Using the information displayed in the Cluster Utilization Report, a Cloudera Runtime cluster administrator can verify that sufficient resources are available for the number and types of jobs running in the cluster. You can use the reports to tune resource allocations so that resources are used efficiently and effectively. Tool tips in the report pages provide suggestions about how to improve performance based on the information displayed in the report. Hover over a label to see these suggestions and other information.

You can tune the following:

- CPU and memory allocations
- Weights for each pool
- Scheduling rules
- Preemption thresholds
- Maximum number of running and queued Impala queries
- Maximum timeout for the queue of Impala queries
- Placement rules
- Number of hosts in a cluster
- Memory capacity of hosts
- Impala Admission Control pool and queue configurations

Enable the Cluster Utilization Report

You must configure several parameters to enable the Cluster Utilization Report.

About this task

Minimum Required Role: Configurator (also provided by Cluster Administrator, Limited Cluster Administrator, and Full Administrator)

By default, the Cluster Utilization Report displays aggregated CPU and memory utilization for Hadoop and for YARN and Impala utilization. You can also view this utilization by tenants, which include Dynamic Resource Pools. To see utilization for a tenant, you must configure the tenant and cluster to collect it.

Procedure

Enable YARN utilization metrics collection:

- In Cloudera Manager, select the YARN service.
- Click the Configuration tab.
- Use the Search function to locate the configuration properties mentioned below.
- In the Container Usage MapReduce Job User property, enter a username for the MapReduce job that collects the metrics. The username you enter must be a Linux user on all the cluster hosts. If you are using an Active Directory KDC, the username must also exist in Active Directory. For secure clusters, the user must not be banned or below the minimum user ID. You can view the list of banned users (banned.users) and the minimum user ID (min.user.id) by under YARN Configuration in Cloudera Manager. The user that is configured with the Container Usage MapReduce Job User property in the YARN service requires permissions to read the subdirectories of the HDFS directory specified with the Cloudera Manager Container Usage Metrics Directory property. The default umask of 022 allows any user to read from that directory. However, if a more strict umask (for example, 027) is used, then those directories are not readable by any user. In that case the user specified with the Container Usage MapReduce Job User property should be added to the same group that owns the subdirectories. For example, if the /tmp/cmYarnContainerMetrics/20161010 subdirectory is owned by user and group yarn:had oop, the user specified in Container Usage MapReduce Job User should be added to the hadoop group. The directories you specify with the Cloudera Manager Container Usage Metrics Directory and Container Usage Output Directory properties should not be located in encryption zones.
- Optionally, enter the resource pool in which the container usage collection MapReduce job runs in the Container Usage MapReduce Job Pool property. Cloudera recommends that you dedicate a resource pool for running this MapReduce job. If you specify a custom resource pool, ensure that the placement rules for the cluster allow for it. The first rule must be for resource pools to be specified at run time with the Create pool if it does not exist option selected. Alternatively, ensure that the pool you specify already exists. If the placement rule is not properly configured or the resource pool does not already exist, the job may run in a different pool.
- You must select Enable Container Usage Metric Collection.
- Click Save Changes.
- Click the Actions button.
- Select Create CM Container Usage Metrics Dir.
- Restart the YARN service.

Enable Impala utilization collection:

- In Cloudera Manager, select the Impala service.
- Click the Configuration tab.
- Search for admission control.
- Find the Enable Impala Admission Control and the Enable Dynamic Resource Pools properties

- Click Save Changes.
- Restart the Impala service.

Configure the Cluster Utilization Report

Configure the Cluster Utilization Report

About this task

To access the Cluster Utilization Report, go to Clusters and then select Utilization Report. Overview tab displays when you first open the report.

The upper-right part of the page has two controls that you use to configure the Cluster Utilization Report.

Procedure

- Click the Configuration drop-down menu.
- Select one of the configured options, or create a new configuration. If you want to create a new configuration, do the following:
 - Click Create New Configuration.
 - Enter a Configuration Name.
 - Select the Tenant Type: Pool or User.
 - Select the days of the week for which you want to report utilization.
 - Select All Day, or use the drop-down menus to specify a utilization time range for the report.
- Click Create. The configuration you created is now available from the Configuration drop-down menu. Select a date range for the report:
 - Click the date range button.
 - Select one of the range options (Today, Yesterday, Last 7 Days, Last 30 Days, or This Month) or click Custom Range and select the beginning and ending dates for the date range.

Use the Cluster Utilization Report to manage resources

The Cluster Utilization Report provides information about CPU and memory utilization in three tabs: YARN and Impala tabs.

To access the Cluster Utilization Report, go to Clusters and then select Utilization Report. Overview tab of the report displays.

Note: The report updates utilization information every hour. The utilization information for YARN queries does not display in the Cluster Utilization Report until captured by the hourly report.

Figure 1: Cluster Utilization Report Overview Tab

The Cluster Utilization Report is divided into the following tabs:

- Overview Tab
- YARN Tab
- Impala Tab

Overview Tab

The overview tab of the cluster utilization report provides a summary of CPU and memory utilization. The Overview tab provides a summary of CPU and memory utilization for the entire cluster and applications and Impala queries. Two sections, CPU Utilization and Memory Utilization, display information:

CPU Utilization Memory Utilization
Overall Cluster Utilization

- Total CPU Cores – Average number of CPU cores available during the reporting window.
- Average Utilization – Average CPU utilization for the entire cluster, including resources consumed by user applications and Cloudera Runtime services.
- Maximum Utilization – Maximum CPU utilization for the entire cluster during the reporting window, including resources consumed by user applications and Cloudera Runtime services. If this value is high, consider adding more hosts to the cluster. Click the drop-down menu next to the date and select one of the following to view details about jobs running when maximum utilization occurred:
 - View YARN Applications Running at the Time
 - View Impala Queries Running at the Time
- Average Daily Peak – Average daily peak CPU consumption for the entire cluster during the reporting window. This includes resources consumed by user applications and Cloudera Runtime services. The number is computed by averaging the maximum resource consumption for each day of the reporting period. Click View Time Series Chart to view a chart of peak utilization.

Overall Cluster Utilization

- Total Physical Memory – Average physical memory available in the cluster during the reporting window.
- Average Utilization – Average memory consumption for the entire cluster, including resources consumed by user applications and Cloudera Runtime services.
- Maximum Utilization – Maximum memory consumption for the entire cluster during the reporting window, including resources consumed by user applications and Cloudera Runtime services. If this value is high, consider adding more hosts to the cluster. Click the drop-down menu next to the date and select one of the following to view details about jobs running when maximum utilization occurred:
 - View YARN Applications Running at the Time
 - View Impala Queries Running at the Time
- Average Daily Peak – Average daily peak memory consumption for the entire cluster during the reporting window, including resources consumed by user applications and Cloudera Runtime services. The number is computed by averaging the maximum memory utilization for each day of the reporting period. Click View Time Series Chart to view a chart of peak

utilization. YARN + Impala Utilization

- Average Utilization – Average resource consumption by YARN applications and Impala queries that ran on the cluster.
- Maximum Utilization – Maximum resource consumption by YARN applications and Impala queries that ran on the cluster. Click the drop-down menu next to the date and select one of the following to view details about jobs running when maximum utilization occurred:
 - View YARN Applications Running at the Time
 - View Impala Queries Running at the Time
- Average Daily Peak – Average daily peak resource consumption by YARN applications and Impala queries during the reporting window. The number is computed by finding the maximum resource consumption per day and calculating the mean. Click View Time Series Chart to view a chart of peak utilization.

YARN + Impala Utilization

- Average Utilization – Average memory consumption by YARN applications and Impala queries that ran on the cluster.
- Maximum Utilization – Maximum memory consumption for the entire cluster during the reporting window, including resources consumed by user applications and Cloudera Runtime services. If this is high, consider adding more hosts to the cluster. Click the drop-down menu next to the date and select one of the following to view details about jobs running when maximum utilization occurred:
 - View YARN Applications Running at the Time
 - View Impala Queries Running at the Time
- Average Daily Peak – Average daily peak memory consumption by YARN applications and Impala queries during the reporting window. The number is computed by finding the maximum resource consumption per day and then calculating the mean. Click View Time Series Chart to view a chart of peak utilization.

Utilization by Tenant

Displays overall utilization for each tenant. Tenants can be either pools or users.

Utilization by Tenant

Displays overall utilization for each tenant. Tenants can be either pools or users.

Impala Tab

The Impala tab displays CPU and memory utilization for Impala queries using three tabs: Queries, Usage and Spilled Memory tab.

Queries Tab

The Overview tab displays information about Impala queries.

The top part of the page displays summary information about Impala queries for the entire cluster. The lower part displays the same information by tenant. Both sections display the following:

- **Total** – Total number of queries. Click the link with the total to view details and charts about the queries.
- **Avg Wait Time in Queue** – Average time, in milliseconds, spent by a query in an Impala pool while waiting for resources. If this number is high, consider increasing the resources allocated to the pool. If this number is high for several pools, consider increasing the number of hosts in the cluster.
- **Successful** – The number and percentage of queries that finished successfully. Click the link with the total to view details and charts about the queries.
- **Memory Limit Exceeded** – Number and percentage of queries that failed due to insufficient memory. If there are such queries, consider increasing the memory allocated to the pool. If there are several pools with such queries, consider increasing the number of hosts in the cluster.
- **Timed Out in Queue** – Number of queries that timed out while waiting for resources in a pool. If there are such queries, consider increasing the maximum number of running queries allowed for the pool. If there are several pools with such queries, consider increasing the number of hosts in the cluster.
- **Rejected** – Number of queries that were rejected by Impala because the pool was full. If this number is high, consider increasing the maximum number of queued queries allowed for the pool. Click the column header to sort the table by that column.

Peak Memory Usage Tab

This report shows how Impala consumes memory at peak utilization. If utilization is high for several pools, consider adding more hosts to the cluster. The Summary section of this page displays aggregated peak memory usage information for the entire cluster. The Utilization by Tenant section displays peak memory usage by tenant. Both sections display the following:

- **Max Allocated**
 - **Peak Allocation Time** – The time when Impala reserved the maximum amount of memory for queries. Click the drop-down list next to the date and time and select View Impala Queries Running at the Time to see details about the queries.
 - **Max Allocated** – The maximum memory that was reserved by Impala for executing queries. If the percentage is high, consider increasing the number of hosts in the cluster.
 - **Utilized at the Time** – The amount of memory used by Impala for running queries at the time when maximum memory was reserved. Click View Time Series Chart to view a chart of peak memory allocations.
 - **Histogram of Allocated Memory at Peak Allocation Time** – Distribution of memory reserved per Impala daemon for executing queries

at the time Impala reserved the maximum memory. If some Impala daemons have reserved memory close to the configured limit, consider adding more physical memory to the hosts. Note: This histogram is generated from the minute-level metrics for Impala daemons. If the minute-level metrics for the timestamp at which peak allocation happened are no longer present in the Cloudera Service Monitor Time-Series Storage, the histogram shows no data. To maintain a longer history for the minute-level metrics, increase the value of the Time-Series Storage property for the Cloudera Service Monitor. (Go to the Cloudera Management Service Configuration and search for Time-Series Storage.)

- Max Utilized
 - Peak Usage Time – The time when Impala used the maximum amount of memory for queries. Click the drop-down list next to the date and time and select View Impala Queries Running at the Time to see details about the queries.
 - Max Utilized – The maximum memory that was used by Impala for executing queries. If the percentage is high, consider increasing the number of hosts in the cluster.
 - Reserved at the Time – The amount of memory reserved by Impala at the time when it was using the maximum memory for executing queries. Click View Time Series Chart to view a chart of peak memory utilization.
 - Histogram of Utilized Memory at Peak Usage Time – Distribution of memory used per Impala daemon for executing queries at the time Impala used the maximum memory. If some Impala daemons are using memory close to the configured limit, consider adding more physical memory to the hosts. Note: This histogram is generated from the minute-level metrics for Impala daemons. If the minute-level metrics for the timestamp at which peak allocation happened are no longer present in the Cloudera Service Monitor Time-Series Storage, the histogram shows no data. To maintain a longer history for the minute-level metrics, increase the value of the Time-Series Storage property for the Cloudera Service Monitor. (Go to the Cloudera Management Service Configuration and search for Time-Series Storage.)

Spilled Memory Tab

The Spilled Memory tab displays information about Impala spilled memory. These disk spills can significantly impact the performance of Impala queries. This report shows the amount of disk spills for each pool. If disk spill is high for a pool, consider adding resources to the pool. If disk spill is high for the entire cluster, consider adding more hosts to the cluster.

For each tenant, the following are displayed:

- Average Spill – Average spill per query
- Maximum Spill – Maximum memory spilled per hour

Download the Cluster Utilization Report

You can download the Cluster Utilization Reports as a JSON file using the Cloudera Manager Admin Console. See the Cloudera Manager REST API documentation for the following API endpoints:

- Cluster Utilization: `path__clusters-clusterName-__utilization.html`
- Impala Utilization: `path__clusters-clusterName-__impalaUtilization.html`
- YARN Utilization: `path__clusters-clusterName-__yarnUtilization.html`

Creating a Custom Cluster Utilization Report

You can create a custom Cluster Utilization Report using different metrics and queries. Cloudera Manager provides a Cluster Utilization Report that displays aggregated utilization and Impala jobs. If you wish to export the data from this report, you can build custom reports using metrics data using the Cloudera Manager Admin console or the Cloudera Manager API. These reports use the `tsquery` Language to chart time-series data.

Metrics and queries

You can use metrics and queries to create a custom Cluster Utilization Report in Cloudera Manager. Many of the metrics described below use a data granularity of hourly. This is not required, because some of the YARN utilization metrics are only available hourly and using the hourly consistent reporting.

Cluster-Level CPU and Memory Metrics

Total cluster CPU usage

Data Granularity: hourly

Units: percentage

`tsquery:`

```
SELECT
cpu_percent_across_hosts
WHERE
category=CLUSTER
AND clusterName=Cluster_Name
```

Total CPU Cores in the cluster

Data Granularity: hourly

Units: CPU cores

`tsquery:`

```
SELECT
total_cores_across_hosts
WHERE
category=CLUSTER
AND clusterName=Cluster_Name
```

Total cluster memory usage

Data Granularity: hourly

Units: percentage

tsquery:

```
SELECT
100 * total_physical_memory_used_across_hosts/total_physical_m
emory_total_across_hosts
WHERE
category=CLUSTER
AND clusterName=Cluster_Name
```

Total cluster memory usage
Time series of total cluster memory usage.
Data Granularity: hourly
Units: Byte seconds
tsquery:

```
SELECT
total_physical_memory_total_across_hosts
WHERE
category=CLUSTER
AND clusterName=Cluster_Name
```

CPU used by Impala
Time series of total Impala CPU usage in milliseconds.
Data Granularity: hourly
Units: milliseconds

tsquery:

```
SELECT
counter_delta(impala_query_thread_cpu_time_rate)
WHERE
category=CLUSTER
AND clusterName=Cluster_Name
```

Memory used by Impala
Time series of Impala memory usage
Data Granularity: hourly
Units: byte seconds
tsquery:

```
SELECT
counter_delta(impala_query_memory_accrual_rate)
WHERE
category=CLUSTER
AND clusterName=Cluster_Name
```

CPU used by YARN

The yarn_reports_containers_used_cpu metric used in this tsquery is generated per hour, therefore the data granularity used for this query is the raw metric value.
Data Granularity: Raw

Units: percent seconds

tsquery:

```
SELECT
yarn_reports_containers_used_cpu FROM REPORTS
WHERE
category=SERVICE
AND clusterName=Cluster_Name
```

Memory used by YARN

Yarn memory usage. The yarn_reports_containers_used_memory metric used in this tsquery is generated per hour, therefore the data granularity used for this query is the raw metric value

Data Granularity: raw metric value

Units: megabyte seconds

tsquery:

```
SELECT
yarn_reports_containers_used_memory
FROM
REPORTS
WHERE
category=SERVICE
AND clusterName=Cluster_Name
```

Pool-Level CPU and Memory Metrics

CPU used by Impala pool

CPU usage for an Impala pool.

Data Granularity: hourly

Units: milliseconds

tsquery:

```
SELECT
counter_delta(impala_query_thread_cpu_time_rate)
WHERE
category=IMPALA_POOL
AND poolName=Pool_Name
```

Memory used by Impala pool

Data Granularity: hourly

Units: byte seconds

tsquery:

```
SELECT
counter_delta(impala_query_memory_accrual_rate)
WHERE
category=IMPALA_POOL
AND poolName=Pool_Name
```

CPU used by YARN pool

Provides CPU metrics per YARN pool and user. You can aggregate a pool-level metric from this query.

Data Granularity: Raw

Units: percent seconds

tsquery:

```
SELECT
yarn_reports_containers_used_cpu FROM REPORTS
WHERE
category=YARN_POOL_USER
```

Memory used by YARN pool

Provides memory metrics per YARN pool and user. You can aggregate a pool-level metric from this query.

Data Granularity: hourly

Units: megabyte seconds

tsquery:

```
SELECT
yarn_reports_containers_used_memory
FROM
REPORTS
WHERE
category=YARN_POOL_USER
```

YARN Metrics

YARN VCore usage

Data Granularity: Raw

Units: VCore seconds

tsquery:

```
SELECT
yarn_reports_containers_used_vcores
FROM
REPORTS
WHERE
category=SERVICE
AND clusterName=Cluster_Name
```

Total VCores available to YARN

Data Granularity: hourly

Units: Number of VCores (Note that this value is not multiplied by the time unit.)

tsquery:

```
SELECT
total_allocated_vcores_across_yarn_pools + total_available_vco
res_across_yarn_pools
WHERE
category=SERVICE
```

AND clusterName=Cluster_Name

YARN Memory usage

Data Granularity: Raw

Units: MB seconds

tsquery:

SELECT

yarn_reports_containers_used_memory FROM REPORTS

WHERE

category=SERVICE

AND clusterName=Cluster_Name

Total memory available to YARN

Data Granularity: hourly

Units: MB (Note that this value is not multiplied by the time unit.)

tsquery:

SELECT

total_available_memory_mb_across_yarn_pools + total_allocated_
memory_mb_across_yarn_pools

WHERE

category=SERVICE

AND clusterName=Cluster_Name

Pool-level VCore usage

The results of this query return the usage for each user in each pool. To see the total usage for a pool, sum all users of the pool.

Data Granularity: Raw

Units: VCore seconds

tsquery:

SELECT

yarn_reports_containers_used_vcores FROM REPORTS

WHERE

category=YARN_POOL_USER

To view metrics for a specific pool, add poolName=Pool Name to the tsquery statement.

Pool-level memory usage

The results of this query return the usage for each user in each pool. To see the total usage for a pool, sum all users of the pool.

Data Granularity: Raw

Units: MB seconds

tsquery:

SELECT

yarn_reports_containers_used_memory FROM REPORTS

WHERE

category=YARN_POOL_USER

To view metrics for a specific pool, add poolName=Pool Name to the tsquery statement.

Pool-level allocated VCores

The results of this query return the usage for each user in each pool. To see the total usage for each pool, sum all users of the pool.

Data Granularity: raw metric value

Units: VCore seconds

tsquery:

```
SELECT
yarn_reports_containers_allocated_vcores FROM REPORTS
WHERE
category=YARN_POOL_USER
```

To view metrics for a specific pool, add poolName=Pool Name to the tsquery statement.

Pool-level allocated memory

The results of this query return the usage for each user in each pool. To see the total usage for each pool, sum all users of the pool.

Data Granularity: raw metric value

Units: megabyte seconds

tsquery:

```
SELECT
yarn_reports_containers_allocated_memory
FROM
REPORTS
WHERE
category=YARN_POOL_USER
```

To view metrics for a specific pool, add poolName=Pool Name to the tsquery statement.

Pool-level steady fair share VCore

Data Granularity: hourly

Units: VCores

tsquery:

```
SELECT
steady_fair_share_vcores
WHERE
category=YARN_POOL
```

To view metrics for a specific pool, add poolName=Pool Name to the tsquery statement.

Pool-level fair share VCore

Data Granularity: hourly

Units: VCores

tsquery:

```
SELECT
fair_share_vcores
WHERE
category=YARN_POOL
```

Pool-level steady fair share memory

Data Granularity: hourly

Units: MB

tsquery:

```
SELECT
steady_fair_share_mb
WHERE
category=YARN_POOL
```

To view metrics for a specific pool, add poolName=Pool Name to the tsquery statement.

Pool-level fair share memory

Data Granularity: hourly

Units: MB

tsquery:

```
SELECT
fair_share_mb
WHERE
category=YARN_POOL
```

To view metrics for a specific pool, add poolName=Pool Name to the tsquery statement.

Metric indicating contention

Data Granularity: hourly

Units: percentage

tsquery:

```
SELECT
container_wait_ratio
WHERE
category=YARN_POOL
```

To view metrics for a specific pool, add poolName=Pool Name to the tsquery statement.

YARN Contention-Related Metrics

Use the following metrics to monitor resource contention.

Pool-level allocated VCores when contention occurs

Data Granularity: hourly

Units: VCores

tsquery:

```
SELECT
allocated_vcores_with_pending_containers
WHERE
category=YARN_POOL
```

To view metrics for a specific pool, add poolName=Pool Name to the tsquery statement.

Pool level steady fair share VCores when contention occurs

Data Granularity: hourly

Units: VCores

tsquery:

```
SELECT
steady_fair_share_vcores_with_pending_containers
WHERE
category=YARN_POOL
```

To view metrics for a specific pool, add poolName=Pool Name to the tsquery statement.

Pool level fair share VCores when contention occurs

Data Granularity: hourly

Units: VCores

tsquery:

```
SELECT
fair_share_vcores_with_pending_containers
WHERE
category=YARN_POOL
```

To view metrics for a specific pool, add poolName=Pool Name to the tsquery statement.

Pool level allocated memory when contention occurs

Data Granularity: hourly

Units: MB

tsquery:

```
SELECT
allocated_memory_mb_with_pending_containers
WHERE
category=YARN_POOL
```

To view metrics for a specific pool, add poolName=Pool Name to the tsquery statement.

Pool level steady fair share memory when contention occurs

Data Granularity: hourly

Units: MB

tsquery:

```
SELECT
steady_fair_share_mb_with_pending_containers
WHERE
category=YARN_POOL
```

To view metrics for a specific pool, add poolName=Pool Name to the tsquery statement.

Pool level fair share memory when contention occurs

Data Granularity: hourly

Units: MB

tsquery:

```
SELECT
fair_share_mb_with_pending_containers
WHERE
category=YARN_POOL
```

To view metrics for a specific pool, add poolName=Pool Name to the tsquery statement.

Impala-Specific Metrics

To view metrics for a specific pool, add poolName=Pool Name to the tsquery statement.

Total reserved memory

Data Granularity: hourly

Units: MB seconds

tsquery:

```
SELECT
total_impala_admission_controller_local_backend_mem_reserved_a
cross_impala_daemon_pools
WHERE
category=CLUSTER
AND clusterName=Cluster_Name
```

Total used memory

Data Granularity: hourly

Units: MB seconds

tsquery:

```
SELECT
total_impala_admission_controller_local_backend_mem_usage_acro
ss_impala_daemon_pools
WHERE
category=CLUSTER
AND clusterName=Cluster_Name
```

Total available memory

Data Granularity: hourly

Units: MB seconds

tsquery:

```
SELECT
total_mem_tracker_process_limit_across_impalads
WHERE
category=CLUSTER
AND clusterName=Cluster_Name
```

Note: To query for pool-level metrics, change the category to IMPALA-POOL in the above tsquery statements.

Impala query counter metrics

Use Impala query counter metrics to get information about the rate of Impala queries. Include the following in the SELECT statement of the tsquery:

- counter_delta(queries_ingested_rate)
- counter_delta(queries_successful_rate)
- counter_delta(queries_rejected_rate)

- counter_delta(queries_oom_rate)
- counter_delta(queries_timed_out_rate)
- counter_delta(impala_query_admission_wait_rate)
- counter_delta(impala_query_memory_spilled_rate) For example:

```
SELECT
counter_delta(queries_ingested_rate)
WHERE
category=IMPALA_POOL
AND clusterName=Cluster_Name
AND serviceName=Service_Name
```

Calculations for reports

Learn about how to correctly perform calculation using metrics values.

All the metrics return a time series of metric values. Depending on the collection frequency and data granularity you use when issuing tsquery statements, the results return metric values and therefore there are different ways to handle the metric values.

Note the following about how to correctly perform calculations using metric values:

- YARN container metrics are generated once per hour resulting in one raw metric value every hour. Therefore, the most detailed results possible for YARN CPU and memory usage are hourly reports.
- Hourly aggregates are summarized from raw metric values. These aggregates include a set of statistics that include the sum, maximum, minimum, count and other statistics that summarize the raw metric values. When you use the hourly granularity, you lose the single values of the raw metric values. However, you can still get peak usage data for such metrics.
- For some of the YARN metrics described in this topic, the tsquery statement aggregates from the pool and user level to pool level in the Cloudera Manager Cluster Utilization reports. For these queries, because the maximum and minimum for different pool and user combinations are not likely to happen at the same time, there is no way to get the peak usage across pool and user combinations, or at the pool level. The only meaningful results possible are average and sum.
- When calculating CPU/Memory usage percentage, pay attention to the units for each metric. For example, if the cluster consistently has 8 VCores, the total VCore seconds for each hour would be 8 * 3600 VCore seconds. You can then use this adjusted number to compare with the VCore seconds used by YARN or YARN pools.

Retrieving metric data

You can view the Cloudera Manager Service Monitor data storage granularities in Cloudera Manager. There is a Time series endpoint exposed by the Cloudera Manager REST API. The API accepts tsquery as input for which metrics need to be retrieved during the specified time window. The API provides

to specify the desired data granularity (for example, raw metric values, TEN_MINUTES, HOURLY). The desired granularity level of data is maintained in a leveldb table. This data is aggregated from raw data into minimum, maximum, etc. within the corresponding data window.

For example, if you do not need the metric data at a specific timestamp but care more about trends, HOURLY data should be good enough. In general, the longer the granular window it is, the less data is stored and thus the longer period of time you are able to keep that level of data without being purged. It reaches the configured limit. In the case of Cloudera Manager Cluster Utilization Reports, Cloudera Manager stores the reports based on an hourly window.

To view the Cloudera Manager Service Monitor data storage granularities, go to ClustersCloud Manager>ServiceMonitorCharts Library>Service Monitor Storage and scroll down to see the Data Granularity table to see the earliest available data points for each level of granularity. The value in the Age column indicates the age of the oldest data in the table.

To configure the Time series storage used by the Service Monitor, go to ClustersCloud Manager>ServiceConfiguration and search for "Time-Series Storage".

Querying metric data

You can build charts that query time series data using the Cloudera Manager Admin console. Go to Charts Chart Builder. When building charts, it may be useful to choose the data granularity. Click the Show additional options link on the chart builder page and then selecting the Data Granularity. Selecting data granularity in chart builder: