Cloudera Runtime 7.0.0

Monitoring Clusters and Apache Hadoop YARN Applications

Date published: 2019-08-21
Date modified:

CLOUDƎRA

Legal Notice

Contents

Monitoring clusters using YARN web user interface

Using the YARN Web User Interface, you can monitor clusters, queues, applications, services, flow activities, and nodes. You can also view the configuration details and check the logs for various applications and services.

Access the YARN web user interface

Access the YARN Web User Interface to monitor clusters, queues, applications, services, and flow activities.

About this task

You can access the YARN Web User Interface from Cloudera Manager to monitor the cluster.

Procedure

1. In Cloudera Manager, select the YARN service.
2. Click the Web UI tab.
3. Select the ResourceManager Web UI.
   The ResourceManager Web UI opens in a new browser window.

   Note, that in a ResourceManager High Availability configuration, if you select the non-active ResourceManager the link directs to the web UI of the active ResourceManager.

Monitor clusters

The Cluster Overview page shows cluster resource usage by applications and queues, information about finished and running applications, and usage of memory and vCores in the cluster.
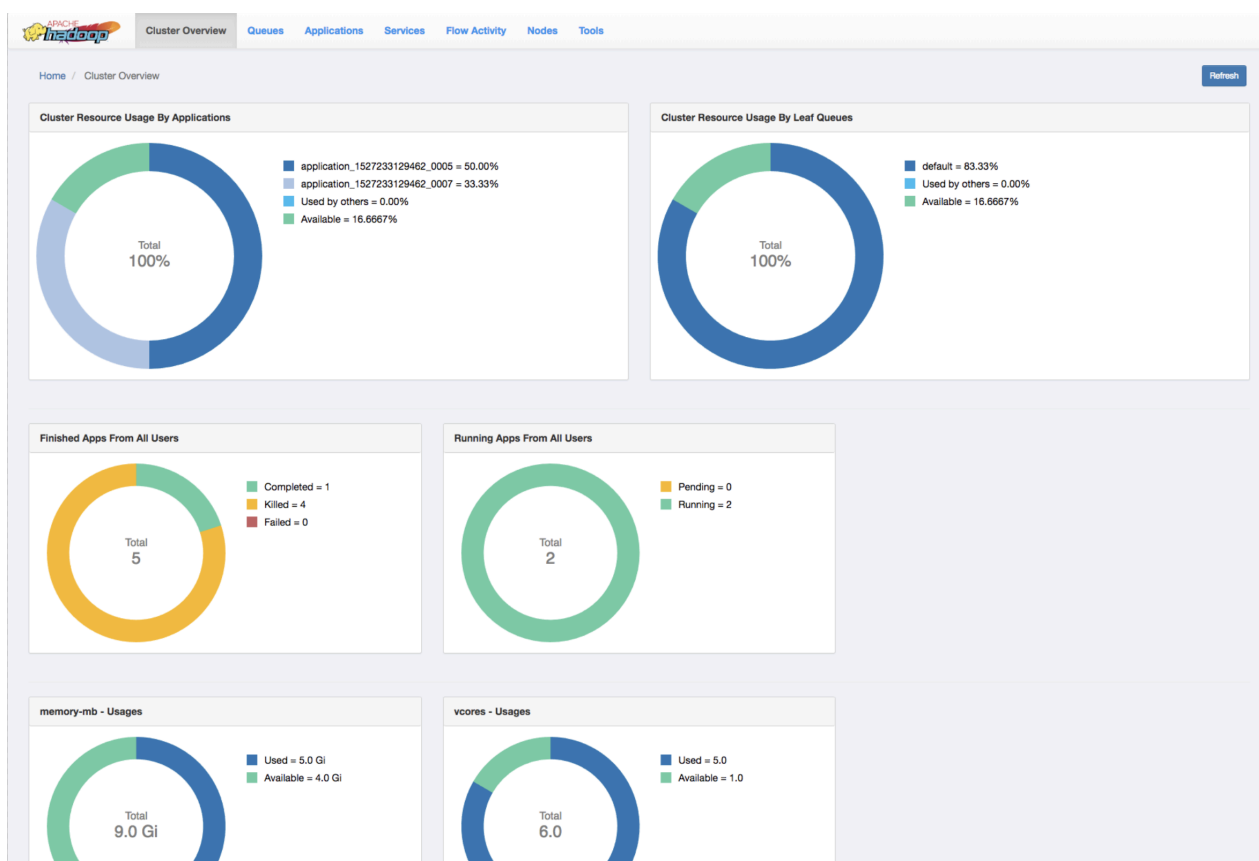
Cluster Resource Usage by Applications

> Displays the percentage of cluster resources in use by applications and the percentage available for usage.

Cluster Resource Usage by Leaf Queues

> Displays the percentage of cluster resources in use by leaf queues and the percentage available for usage.

Finished Apps From All Users

> Displays the number of completed, killed, and failed applications.

Monitor Running Apps

> Displays the number of pending and running applications.

memory-mb - Usages

> Displays the amount of used and available memory.

vcores - Usages

> Displays the number of used and available virtual cores.

Monitor Node Managers

> Displays the status of the Node Managers under the following categories:

- Active
- Unhealthy
- Decommissioning
- Decommissioned

Monitor queues

The Queues page is scheduler dependent and displays details of YARN queues. You can either view queues from all the partitions or filter to view queues of a partition.

Capacity Scheduler
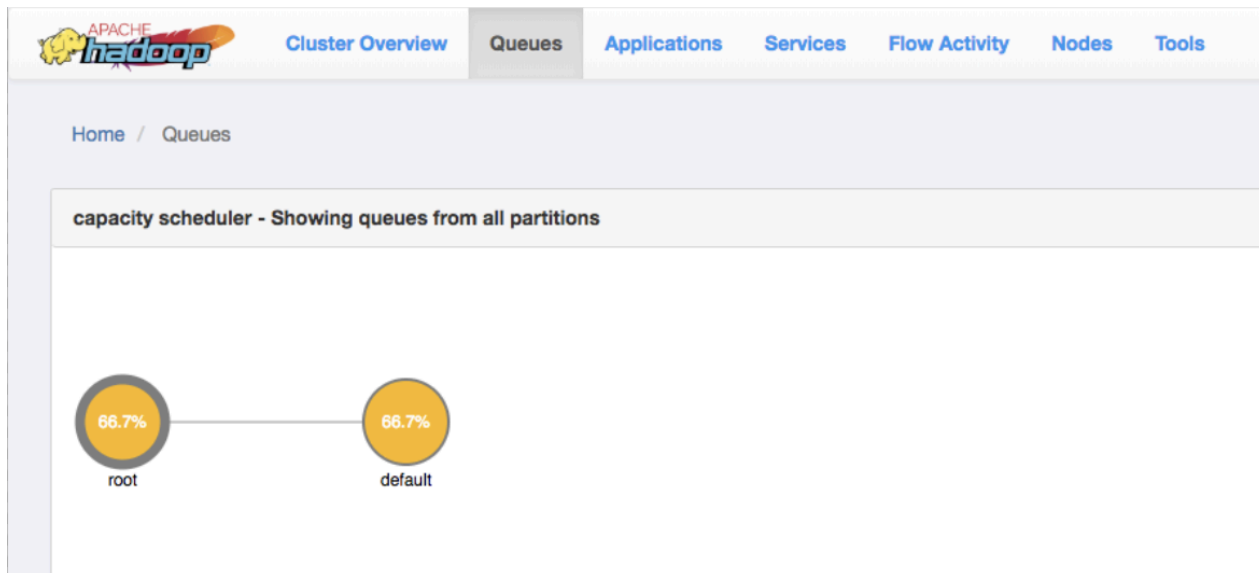
View Queue Details

In the capacity scheduler view, click the circle that represents a particular queue. The right column of the page gets updated with details of that queue.

The following example shows the details of a queue:
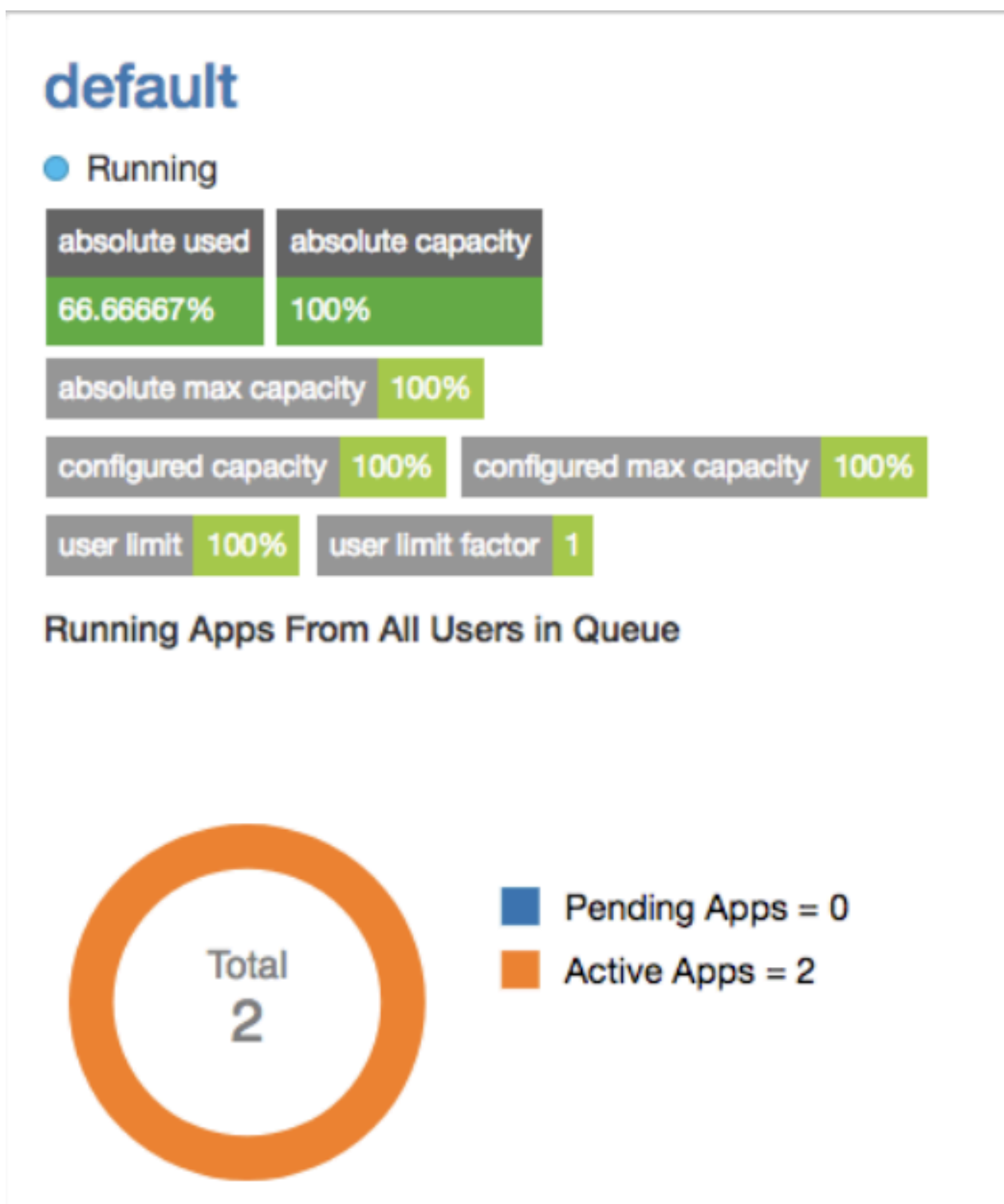
You can double-click the queue to view its details on a separate page. You can also view details of any application submitted to that queue by clicking its corresponding Application ID.

Monitor applications

You can search for applications and view their details using the YARN Web User Interface.

The Applications page displays details of the YARN applications in a tabular form.

- Application ID: The identifier for the application.
- Application Type: Specifies the application type for Mapreduce, YARN, TEZ, or other applications.
- Application Name: Provides the name of the application
- User: The name of the user who is the owner of the application.
- State: The running state of the application.
- Queue: Specifies the name of the queue to which the application belongs.
- Progress: The progress of the application in a percentage display.
- Start Time: The time when an application run started.
- Elapsed Time: The time taken for the application run.
- Finished Time: The time of completion of the application run.
- Priority: The priority of running the application.
- %Cluster: The percentage of cluster resources used by the application run.

Search an application

The Applications page displays the list of YARN applications in a tabular form. You can apply search filters on this list to show only those applications that match the search criteria.

About this task

You can specify the search criteria either as regular expressions or SQL statements.

Procedure

1. On the Applications page, select either Regex or SQL from the drop-down list depending on the type of search you want to perform.
2. In the Search box, specify the search criteria.

Search Criteria	Description
Regex	Lists the applications whose details match the search criterion specified as a regular expression.
	For example, if you want to view application runs longer than an hour, mention the regular expression ^hso that the YARN UI shows only those applications that mention the Elapsed Time value in terms of hours, minutes, and seconds.
SQL	Lists the applications whose details match the search criterion specified as a SQL statement.
	For example, if you want to view all the applications submitted by the user yarn, specify "User"='yarn' as the search criterion.

3. Click Search to view details of the applications matching the search criteria.

> **Note:**
> * Apart from specifying search criteria to filter the list of applications, you can also select or clear the State and Queue check-boxes to view a specific set of applications depending on your requirements.
> * You can sort the application entries in ascending or descending order by clicking the corresponding arrow next to any column header in the table.

Related Information

Manage and monitor services

View application details

Clicking a YARN application on the Applications page displays its details.

You can view the following details for the selected application:

* Application Attempts
* Resource Usage
* Diagnostics
* Logs

Application Attempts

You can view the attempts in Graph View and Grid View.

Graph View

> A graph displays the start time and finish time of the attempt. You can also find the details of the attempt such as application attempt ID, started time, finished time, elapsed time, AM Container ID, and AM Node ID in the form of a table. You can access the Node UI using the AM Node Web UI. You can also view the log by clicking on the log link.

Grid View

> A table displays the details of the application attempts. You can find the details of the attempt such as application attempt ID, started time, finished time, elapsed time, AM Container ID, and AM Node ID. You can access the Node UI using the AM Node Web UI. You can also view the log by clicking on the log link.

Resource Usage

This tab displays the resources used by the application attempts.

Diagnostics

Use this tab to view the diagnostics details of the application attempts. You can view any outstanding resource requests, and scheduling information.

Logs

Use this tab to view logs specific to containers. Select an attempt from the dropdown list and select the specific container to view the desired logs.

Manage and monitor services

You can create new YARN services and view the list of existing services on the Services page.

The Services page lists the services in a tabular form. Clicking an individual service displays its details. You can filter the list of services the same way as you filter applications on the Applications page.

Related Information
Search an application

Create new services

The YARN Web User Interface enables you to define new services. You can either create standard services by providing their details or custom services by using JSON files containing the required definitions.

Create a standard service
You can create a standard service as a template and deploy it based on your requirements.

Procedure

1. On the Services page of the YARN Web User Interface, click New Service.
2. In the User name for service field, specify the name of the user who launches the service.
3. Enter the service definition details.

   • Service Name: Enter a unique name for the application.
   • Queue Name: Enter the name of the YARN queue to which this application should belong.
   • Service Lifetime: Life time of the application from the time it is in STARTED state till the time it is automatically destroyed by YARN. If you want to have unlimited life time, do not enter any value.
   • Service Components: Enter the details of the service components such as component name, CPU required, memory, number of containers, artifact ID, and launch command. If it is an application like HBase, the components can be a simple role like master or RegionServer. For a complex application, the components can be other nested applications with their own details.
   • Service Configurations: Set of configuration properties that can be ingested into the application components through environments, files, and custom pluggable helper docker containers. You can upload files of several formats such as properties files, JSON files, XML files, YAML files, and template files.
   • File Configurations: Set of file configurations that needs to be created and made available as a volume in an application component container. You can upload JSON file configurations to add to the service.

4. Click Save.
5. Specify a name for the new service and click Add.

   The newly created service is added to the list of saved templates.

   > **Note:** Click Reset if you do not want to save your changes and specify the service details again.

6. Select the service and click Deploy to deploy it.

Create a custom service

You can define a service in JSON format and save it as a template.

Procedure

1. On the Services page of the YARN Web User Interface, click New Service.
2. Click the Custom tab.
3. In the User name for service field, specify the name of the user who launches the service.
4. In the Service Definition field, specify the service definition in JSON format.

   The following example shows the sleeper service template definition.

   ```
   {
      "name": "sleeper-service",
      "version": "1.0.0",
      "components" :
      [
        {
          "name": "sleeper",
          "number_of_containers": 2,
          "launch_command": "sleep 900000",
          "resource": {
            "cpus": 1,
            "memory": "256"
          }
        }
      ]
   }
   ```

5. Click Save.
6. Specify a name for the new service and click Add.

   The newly created service is added to the list of saved templates.

   > **Note:** Click Reset if you do not want to save your changes and specify the service details again.

7. Select the service and click Deploy to deploy it.

Monitor nodes

The Nodes page on the YARN Web User Interface enables you to view information about the cluster nodes on which the NodeManagers are running.

The Nodes page displays details under the following headers: Information, Node Status and Nodes Heatmap Chart.

Information

The Information tab displays the node details as shown in the following figure:
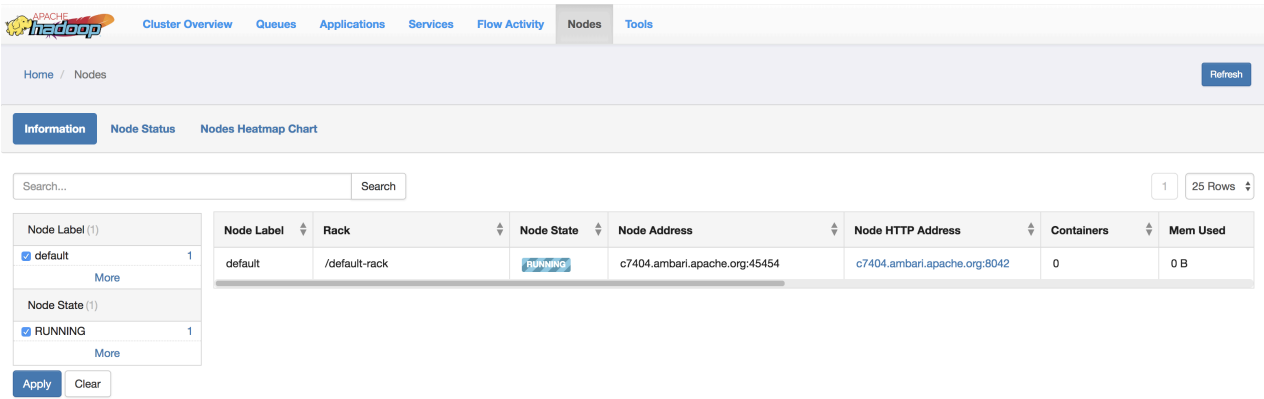
You can sort through any of the columns to view the details of the required nodes. You can also search to find the specific node labels from the entire list.

Node Manager Page

To view the Node Manager page of any node label, click the corresponding Node HTTP address. The Node Manager page displays details of a node as shown in the following figure:
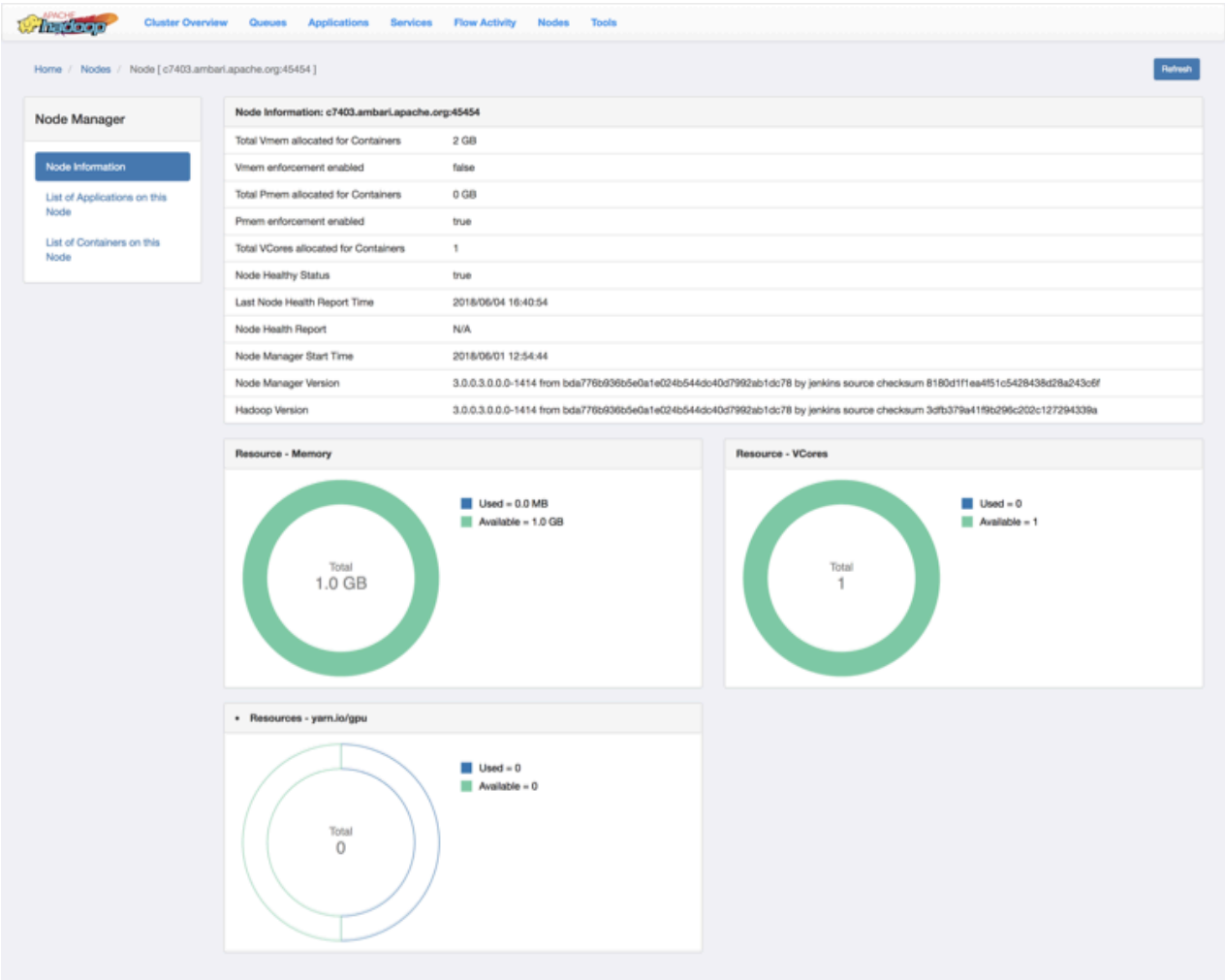
You can also view the resource usage in the following categories in a pie-chart representation:

1. Memory

2. VCores
3. yarn-io/GPU

Node Status

This tab displays the node managers in a pictorial representation. It displays details such as the number of active nodes, number of unhealthy nodes, decommissioning nodes, and the number of decommissioned node managers.

Nodes Heatmap Chart

This tab graphically displays nodes on the basis of their usage of memory. You can enter host or rack details in the search bar to filter nodes.

Tools

You can view the YARN configuration and YARN Daemon logs on the Tools page.

YARN Configuration

You can see the values of the properties defined in the following configuration files:

1. Core Configuration: Details of properties defined in the core-default.xml file.
2. YARN Configuration: Details of properties defined in the yarn-default.xml file.
3. MapReduce Configuration: Details of the properties defined in the mapred-default.xml file.

YARN Daemon Logs

You can view the list of log files. Click on a log file to view its contents in another tab of your browser.

Accessing YARN logs

Use the YARN CLI to view logs for running application. Configure the log aggregation to aggregate and write out logs for all containers belonging to a single Application grouped by NodeManagers to single log files at a configured location in the file system.

Use the YARN CLI to view logs for applications

You can use the YARN CLI (Command Line Interface) to view log files for applications.

You can access container log files using the YARN ResourceManager web UI, but more options are available when you use the yarn logs CLI command.

View all Log Files for an Application

Use the following command format to view all logs for an application:

```
yarn logs -applicationId <Application ID>
```

View a Specific Log Type for an Application

Use the following command format to view all logs of a particular type for an application:

```
yarn logs -applicationId <Application ID> -log_files <log_file_type>
```

For example, to view only the stderr error logs:

```
yarn logs -applicationId <Application ID> -log_files stderr
```

The -logFiles option also supports Java regular expressions, so the following format would return all types of log files:

```
yarn logs -applicationId <Application ID> -log_files .*
```

View ApplicationMaster Log Files

Use the following command format to view all ApplicationMaster container log files for an application:

```
yarn logs -applicationId <Application ID> -am ALL
```

Use the following command format to view only the first ApplicationMaster container log files:

```
yarn logs -applicationId <Application ID> -am 1
```

List Container IDs

Use the following command format to list all container IDs for an application:

```
yarn logs -applicationId <Application ID> -show_application_log_info
```

View Log Files for One Container

Once you have the container IDs, you can use the following command format to list the log files for a particular container:

```
yarn logs -applicationId <Application ID> -containerId <Container ID>
```

Show Container Log File Information

Use the following command format to list all of the container log file names (types) for an application:

```
yarn logs -applicationId <Application ID> -show_container_log_info
```

You can then use the -logFiles option to view a particular log type.

View a Portion of the Log Files for One Container

For large container log files, you can use the following command format to list only a portion of the log files for a particular container:

```
yarn logs -applicationId <Application ID> -containerId <Container ID> -size
<bytes>
```

To view the first 1000 bytes:

```
yarn logs -applicationId <Application ID> -containerId <Container ID> -size
1000
```

To view the last 1000 bytes:

```
yarn logs -applicationId <Application ID> -containerId <Container ID> -size
-1000
```

Download Logs for an Application

Use the following command format to download logs to a local folder:

```
yarn logs -applicationId <Application ID> -out <path_to_local_folder>
```

The container log files are organized in parent folders labeled with the applicable node ID.

Display Help for YARN Logs

To display Help for yarn logs, run the following command:

```
yarn logs -help
```

Use log aggregation

The YARN Log Aggregation feature enables you to move local log files of any application onto HDFS or a cloud-based storage depending on your cluster configuration.

Application logs has great significance: meaningful information can be extracted from them, can be used to debug issues or can be kept for historical analysis. YARN can move local logs securely onto HDFS or a cloud-based storage, such as AWS. This allows the logs to be stored for a much longer time than they could be on a local disk, allows faster search for a particular log file and optionally can handle compression.

Log Aggregation File Controllers

By default, log aggregation supports two file controllers: TFIile and IFile. You can also add your own custom file controller.

> **Note:** TFile and IFile are commonly referred as file formats. However, they do not only define the format of the file, but also how the log aggregation is managed. Therefore referring to them as file controllers specifies them more clearly.

TFile

TFile is the legacy file controller in YARN. It is reliable and well tested. Its buffer and chunk sizes are configurable.

TFile provides the following features:

- Block compression
- Named metadata blocks
- Sorted or unsorted keys
- Seek by key or by file offset

IFile

IFile is a newer file controller than TFile. It also uses TFile internally so it provides the same features as TFile.

In an IFile the files are indexed so it is faster to search in the aggregated log file than in a regular TFile. It uses checksums and temporary files which help to prevent failures. Its buffer sizes and rollover file size are configurable on top of the configuration options of TFile.

Configure log aggregation

Log aggregation is enabled by default. You can configure it using Cloudera Manager.

Procedure

1. In Cloudera Manager, select the YARN service.
2. Click the Configuration tab.
3. Select the Log Aggregation filter under Category.
   Every log aggregation related property is displayed.
4. Find the Enable Log Aggregation property and ensure that it is selected, meaning that log aggregation is enabled.
5. Configure the log aggregation properties as applicable.

   ⚠️ **Important:**

   The remote-app-log-dir and remote-app-log-dir-suffix configuration properties are not backward compatible. If you change their value later, the logs of applications that are already finished cannot be read.

6. Click Save Changes.

Related Information
Log aggregation properties

Log aggregation properties

You can configure YARN log aggregation for your cluster by specifying values of specific properties depending on your requirements.

Name	Property	Default value	Description
Enable Log Aggregation	yarn.log-aggregation-enable	Enabled	Specifies if log aggregation is enabled. If log aggregation is disabled, YARN container logs are not aggregated.
Log Aggregation Retention Period	yarn.log-aggregation.retain-seconds	7 days	Specifies how long aggregation logs are kept before they are deleted.
Log Aggregation Status Timeout	yarn.log-aggregation-status.timeout.ms	10 minutes	Specifies the maximum amount of time that the NodeManager has for reporting a container's log aggregation status. If no log aggregation status is sent by the NodeManager within the allotted time, the ResourceManager reports a TIME_OUT log aggregation status for that container.
Supported Log Aggregation File Formats	yarn.log-aggregation.file-formats	IFile and TFile	Specifies which log file formats are supported. The first file format in the list is used to write the aggregated logs. TFile format is always added to the end of the list, to support backward compatibility.

Name	Property	Default value	Description
Remote App Log Directory for TFile Format	yarn.log-aggregation.TFile.remote-app-log-dir	empty	Specifies the path of the directory where application logs are stored after an application is completed if TFile format is selected for writing. This configuration overwrites the one given in NodeManager level (yarn.nodemanager.remote-app-log-dir). This property is not backward compatible. If you change its value later, the logs of applications that are already finished cannot be read.
Remote App Log Directory Suffix for TFile Format	yarn.log-aggregation.TFile.remote-app-log-dir-suffix	empty	The remote log directory is created at {remote-app-log-dir}/${user}/{thisParam} if TFile format is selected for writing. This configuration overwrites the one given in NodeManager level (yarn.nodemanager.remote-app-log-dir-suffix). This property is not backward compatible. If you change its value later, the logs of applications that are already finished cannot be read.
Remote App Log Directory for IFile Format	yarn.log-aggregation.IFile.remote-app-log-dor	empty	Specifies the path of the directory where application logs are stored after an application is completed if IFile format is given as the file format for writing. This configuration overwrites the one given in NodeManager level (yarn.nodemanager.remote-app-log-dir). This property is not backward compatible. If you change its value later, the logs of applications that are already finished cannot be read.
Remote App Log Directory Suffix for IFile Format	yarn.log-aggregation.IFile.remote-app-log-dir-suffix	empty	The remote log directory is created at {remote-app-log-dir}/${user}/{thisParam} if IFile format is selected for writing. This configuration overwrites the one given in NodeManager level (yarn.nodemanager.remote-app-log-dir-suffix). This property is not backward compatible. If you change its value later, the logs of applications that are already finished cannot be read.
Log Aggregation Compression Type	yarn.nodemanager.log-aggregation.compression-type	none	Specifies the TFile compression type used to compress aggregated logs. Its value can be: gz, lzo or none.

Name	Property	Default value	Description
Remote App Log Directory	yarn.nodemanager.remote-app-log-dir	/tmp/logs	Specifies the path of the directory where application logs are stored after an application is completed. This property is not backward compatible. If you change its value later, the logs of applications that are already finished cannot be read.
Remote App Log Directory Suffix	yarn.nodemanager.remote-app-log-dir-siffux	logs	The remote log dir will be created at {yarn.nodemanager.remote-app-log-dir}/${user}/{thisParam}. This property is not backward compatible. If you change its value later, the logs of applications that are already finished cannot be read.
Log Aggregation Policy	yarn.nodemanager.log-aggregation.policy.class	All	Specifies the types of container logs that are uploaded during the log aggregation. Valid values are: Only AM, AM or Failed, All, Only Failed, Failed or Killed, None, or Sample.

Related Information
Configure log aggregation

Enable debug-delay

You can keep the application's localized files and log directory on the local disk for a predefined time before the NodeManager deletes them.

About this task

This feature can be useful for example when there is a problem with the log aggregation feature or if you want to diagnose YARN application issues.

Procedure

1. In Cloudera Manager, select the YARN service.
2. Click the Configuration tab.
3. Search for deletion delay.
4. Find the Localized Dir Deletion Delay property.
5. Define the delay time in seconds.

   If you want to diagnose YARN application issues, set this property to a large value, for example to 600 seconds, to permit examination of these directories.
6. Click Save Changes.