# Hortonworks Data Platform

Ambari Views Guide

(Sep 30, 2015)

docs.hortonworks.com

#### Hortonworks Data Platform : Ambari Views Guide

Copyright © 2012-2015 Hortonworks, Inc. All rights reserved.

The Hortonworks Data Platform, powered by Apache Hadoop, is a massively scalable and 100% open source platform for storing, processing and analyzing large volumes of data. It is designed to deal with data from many sources and formats in a very quick, easy and cost-effective manner. The Hortonworks Data Platform consists of the essential set of Apache Hadoop projects including MapReduce, Hadoop Distributed File System (HDFS), HCatalog, Pig, Hive, HBase, Zookeeper and Ambari. Hortonworks is the major contributor of code and patches to many of these projects. These projects have been integrated and tested as part of the Hortonworks Data Platform release process and installation and configuration tools have also been included.

Unlike other providers of platforms built using Apache Hadoop, Hortonworks contributes 100% of our code back to the Apache Software Foundation. The Hortonworks Data Platform is Apache-licensed and completely open source. We sell only expert technical support, training and partner-enablement services. All of our technology is, and will remain free and open source. Please visit the Hortonworks Data Platform page for more information on Hortonworks technology. For more information on Hortonworks services, please visit either the Support or Training page. Feel free to Contact Us directly to discuss your specific needs.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

#### http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

# **Table of Contents**

1. Using Ambari Views	1
2. Preparing Ambari Server for Views	2
3. Running Ambari Server Standalone	3
3.1. Prerequisites	3
3.2. Standalone Server Setup	4
3.3. Reverse Proxy	4
4. Configuring Views for Kerberos	6
5. Using the Tez View	7
5.1. Configuring Your Cluster for Tez View	7
5.2. Creating or Editing the Tez View Instance	8
5.2.1. User Permissions for Tez Views	. 10
5.2.2. Kerberos Setup for Tez Views	11
5.3. Using the Tez View	12
5.3.1. Understanding Directed Acyclic Graphs (DAGs), Vertices, and Tasks	13
5.3.2. Identifying the Tez DAG for Your Job	
5.3.3. Understanding How Your Tez Job Is Executed	. 15
5.3.4. Identifying Causes of Failed Jobs	
5.3.5. Viewing All Failed Tasks	
5.3.6. Using Counters to Identify the Cause of Slow-Performing Jobs	
6. Using the Pig View	
6.1. Configuring Your Cluster for Pig View	
6.2. Creating the Pig View Instance	
6.2.1. Getting Correct Configuration Values for Manually-Deployed Clusters	
	21
6.2.2. User Permissions for Pig Views	22
6.2.3. Kerberos Setup for Pig Views	
6.3. Using the Pig View	
6.3.1. Writing Pig Scripts	
6.3.2. Viewing Pig Script Execution History	
6.3.3. User-Defined Functions (UDFs) Tab	
7. Using the Capacity Scheduler View	
7.1. Configuring your Cluster for the Capacity Scheduler View	
7.2. Creating a Capacity Scheduler View Instance	
7.2.1. User Permissions for Capacity Scheduler Views	
7.3. Using the Capacity Scheduler View	
7.3.1. Setting up Queues	
7.3.2. Configuring Queues	
7.3.3. Configuring Cluster Scheduler Settings	
7.3.4. Applying the Configuration Changes	
7.4. Troubleshooting	
8. Using the Hive View	
8.1. Configuring Your Cluster	
8.1.1. Setup HDFS Proxy User	
8.1.2. Setup HDFS User Directory	
8.2. Creating the Hive View Instance	
8.2.1. Settings and Cluster Configuration	
8.2.2. User Permissions for Hive Views	
8.2.3. Kerberos Setup for Hive Views	
· · · · · · · · · · · · · · · · · · ·	

# **List of Figures**

3.1. Configuring Views with your HDP Cluster	. 3
5.1. Tez View Create Instance Page	8
5.2. Tez View Instance Page	10
5.3. Granting User Permissions to Tez Views	
5.4. SQL Query Execution in Hive	13
5.5. Tez View Column Selector Dialog Box	14
5.6. View Tab in Tez View	
5.7. DAG Details Window	16
5.8. Tez View All Tasks Tab	
5.9. Tez View DAG-Level Counters Tab	
5.10. Tez View Vertex-Level Counters Tab	17
5.11. Tez View Task-Level Counters Tab	
6.1. Pig View Details and Settings	
6.2. Pig View Cluster Configuration	20
6.3. HDFS Service Page in Ambari	22
6.4. Using the Filter to Search Advanced hdfs-site Settings	
6.5. Granting User Permissions to Pig Views	23
6.6. Kerberos Settings for Pig Views	
6.7. Pig Script Running in the Pig View	24
6.8. Pig View Script History Tab	25
6.9. Pig View UDFs Tab	25
8.1. HDFS Service Page in Ambari	46
8.2. Using the Filter to Search Advanced hdfs-site Settings	
8.3. Granting User Permissions to Hive Views	47
8.4. Hive View Kerberos Configuration Example	48
8.5. Hive View Database Explorer	
8.6. Query Editor	49
8.7. Query Results and Logs in Hive View Query Editor	50
8.8. Query Editor Textual Explain Feature	51
8.9. Query Editor Visual Explain Feature	51
8.10. Tez View Query Debugging Option	52
8.11. Query Editor Error Message Summary Window	52
8.12. Query Editor Error Message Details Window	
8.13. Saved Queries Tab	53
8.14. History Tab	53
8.15. UDF Tab	53

# **List of Tables**

1. Cluster Configurations for Tez View	7
2. Cluster Configuration Values for the Tez View in Ambari	9
.3. Kerberos Settings for Tez Views1	1
.4. In core-site.xml 1	1
.5. In yarn-site.xml 1	2
.6. Tez Job Status Descriptions 1	4
.1. Finding Cluster Configuration Values for the Pig View in Ambari	1
.2. Pig View Settings for NameNode High Availability	1
1. Hive View Instance Details 4	5
.2. Finding Cluster Configuration Values for the Hive View in Ambari	5
.3. Hive View Settings for NameNode High Availability	6
.4. Kerberos Settings for Hive Views	7
.5. Troubleshooting Hive Views Errors	4

# **1. Using Ambari Views**

Ambari includes the Ambari Views Framework, which allows for developers to create UI components that "plug into" the Ambari Web interface. Ambari includes a built-in set of Views that are pre-deployed for you to use with your cluster. This guide provides information on configuring the built-in set of Views, as well as information on how to configure Ambari Server for "standalone" operation.

Views can be deployed and managed in the "operational" Ambari Server that is operating your cluster. In addition, Views can be deployed and managed in one or more separate "standalone" Ambari Servers. Running "standalone" Ambari Server instances is useful when users who will access views will not have (and should not) have access to that Ambari Server that is operating the cluster. As well, you can run one or more separate Ambari Server instances "standalone" for a scale-out approach to handling a large number of users. See Running Ambari Standalone for more information.



#### Important

It is critical that you prepare your Ambari Server for hosting views. It is strongly recommended you increase the amount of memory available to your Ambari Server, and that you run additional "standalone" Ambari Servers to host the views. See Preparing Ambari Server for Views and Running Ambari Server Standalone for more information.

View	Description	HDP Stacks	Required Services
Capacity Scheduler	Provides a visual way to configure YARN capacity scheduler queue capacity.	HDP 2.3 or later	YARN
Files	Allows you to browse the HDFS file system.	HDP 2.2 or later	HDFS
Hive	Exposes a way to find, author, execute and debug Hive queries.	HDP 2.3 or later	HDFS, YARN, Hive
Pig	Provides a way to author and execute Pig Scripts.	HDP 2.2 or later	HDFS, Hive ( WebHCat ), Pig
Slider	A tool to help deploy and manage Slider-based applications.	HDP 2.1 or later	HDFS, YARN
Tez	View information related to Tez jobs that are executing on the cluster.	HDP 2.2.4.2 or later	HDFS, YARN, Tez

#### Learning More About Views

You can learn more about the Views Framework at the following resources:

Resource	URL
Administering Views	Ambari Administration Guide - Managing Views
Ambari Project Wiki	https://cwiki.apache.org/confluence/display/AMBARI/Views
Example Views	https://github.com/apache/ambari/tree/trunk/ambari-views/examples
View Contributions	https://github.com/apache/ambari/tree/trunk/contrib/views

# 2. Preparing Ambari Server for Views

When hosting multiple views in Ambari, it is **strongly recommended** you increase the amount of memory available available to the Ambari Server. Since each view requires it's own memory footprint, increasing the Ambari Server maximum allocable memory will help support multiple deployed views and concurrent use.

1. On the Ambari Server host, edit the ambari-env.sh file:

```
vi /var/lib/ambari-server/ambari-env.sh
```

2. For the AMBARI\_JVM\_ARGS variable, replace the default -Xmx2048m with the following:

-Xmx4096m -XX:PermSize=128m -XX:MaxPermSize=128m

3. Restart Ambari Server for this change to take effect.

ambari-server restart

# **3. Running Ambari Server Standalone**

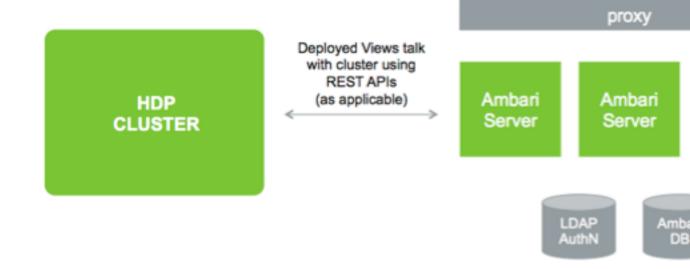
You can run one or more separate Ambari Server instances running in "standalone" mode. Running "standalone" Ambari Server instances is useful when users who will access views will not have (and should not) have access to that Ambari Server that is operating the cluster. As well, you can run one or more separate Ambari Server instances "standalone" for a scale-out approach to handling a large amount of users. See Reverse Proxy for more information.

# **3.1. Prerequisites**

There are several requirements that need to be considered when setting up multiple Ambari Server "standalone" instances:

- Ambari Server instances should be the same version.
- The Ambari Server instances should point to the same underlying database. Ensure that it is **not** the same database that is being used by an Operational Ambari Server managing the HDP cluster.
- Ambari database should be scaled and made highly-available, independent of Ambari Server.
- If using an external authentication source (such as LDAP or Active Directory), Ambari Server authentication should be configured the same for all Ambari Server instances.
- If the cluster you are accessing with Views is Kerberos-enabled, you need to configure Ambari and the Views for Kerberos.
- Run the multiple "standalone" Ambari Server instances behind a Reverse Proxy.

After your standalone Ambari Servers are setup and configured, you can configure the views to communicate with your HDP cluster.



#### Figure 3.1. Configuring Views with your HDP Cluster

### **3.2. Standalone Server Setup**

Setting up a standalone Ambari Server instance is very similar to setting up an operational Ambari Server. Many of the steps are the same, with one key **exception: you do not install a cluster with a standalone Ambari Server**. A standalone Ambari Server does not manage a cluster and does not deploy or communicate with Ambari Agents. The standalone Ambari Server runs as web server instance, serving views for users.



#### Important

Refer to the Ambari Install Guide for the details steps for setting up an Ambari Server. For a standalone Ambari Server instance, you are not required to install a cluster.



#### Important

Refer to Managing Views in the Ambari Administration Guide for information on deploying and configuring Views.

The following table compares the high-level tasks required to setup an operational Ambari Server vs. a standalone Ambari Server.

	Operational Ambari Server	Standalone Ambari Server
1	Install ambari -server package	Install ambari -server package
2	Run ambari -server setup (DB, JDK)	Run ambari -server setup (DB, JDK) Important Do not share the DB with an Operational Ambari Server.
3	Configure external LDAP authentication	Configure external LDAP authentication
4	Install Cluster	NA
5	Deploy views	Deploy views
6	Create + configure view instances	Create + configure view instances
7		(Optional) Repeat for each Ambari Server instance
8		(Optional) Setup proxy for Ambari Server instances

# **3.3. Reverse Proxy**

If you require a larger number of users to access Ambari Views, it may be necessary to "scale-out" the Ambari Server by installing and running multiple Ambari Server standalone instances that host Ambari Views and run those instances behind a reverse proxy.

If a reverse proxy fronts the standalone Ambari Server instances, the only requirement is that the reverse proxy honors session affinity, meaning that once a session has been established the reverse proxy routes each subsequent request to the same Ambari server instance. Depending on the reverse proxy implementation, this can be accomplished in a number of different ways, including hashing client IP and using the JSESSIONID header.



### Important

Using multiple Ambari Server instances and a reverse proxy in front of those instances is **not supported** for an operational Ambari Server. It is only supported for standalone Ambari Server instances (i.e. Ambari instances that are not managing a cluster).

# **4. Configuring Views for Kerberos**

If the cluster your views will communicate with is Kerberos-enabled, you need to configure the Ambari Server instance(s) for Kerberos and be sure to configure the views to work with Kerberos.

Refer to the Set Up Kerberos for Ambari for the instructions on how to configure Ambari Server for Kerberos. Be sure to configure all standalone Ambari Server instances for Kerberos.



#### Important

Be sure to install the Kerberos client utilities on the Ambari Server so that Ambari can kinit.

#### **RHEL/CentOS/Oracle Linux**

yum install krb5-workstation

SLES

zypper install krb5-client

#### Ubuntu/Debian

apt-get install krb5-user krb5-config

Once your Ambari Server is setup for Kerberos, be sure to follow the specific instructions with each view on how to configure the view for Kerberos and the cluster for Kerberos access from the view. Also, if the view requires HDFS or WebHCat to be configured for a proxy user, **instead of using the ambari-server daemon user as the proxy user, you must user primary Kerberos principal**. For example, if you configure Ambari Server for Kerberos principal **ambari-server@EXAMPLE.COM**, this value would be **ambari-server**.

# **5. Using the Tez View**

Tez is an framework for building high performance batch and interactive data processing applications. Apache Hive and Pig use the Tez framework. When you run a job such as a Hive query or Pig script using Tez, you can use the Tez View to track and debug the execution of that job. Topics in this chapter describe how to configure, deploy and use the Tez View to execute jobs in your cluster:

- Configuring Your Cluster for Tez View
- Creating the Tez View Instance
- Using the Tez View

# **5.1. Configuring Your Cluster for Tez View**

When you deploy a cluster with Ambari, a Tez View instance is automatically created. However, you must verify that the configurations listed in the following table have been correctly set.

If you have manually deployed your cluster, you must set the properties listed in the following table to configure your cluster before you create the Tez View on your standalone Ambari server.

#### To configure your cluster for the Tez View:

1. Confirm the following configurations are set:

#### Table 5.1. Cluster Configurations for Tez View

Component	Configuration	Property	Comments
YARN	yarn-site.xml	yarn.resourcemanager. system-metrics- publisher.enabled	Enable the generic history service in the Timeline Server. Verify that this property is set to true.
YARN	yarn-site.xml	yarn.timeline- service.enabled	Enable the Timeline Server for logging details. Verify that this property is set to true.
YARN	yarn-site.xml	yarn.timeline- service.webapp.address	Value must be the IP:PORT on which the Timeline Server is running.

2. If you changed any settings, you must restart the YARN ResourceManager and the Timeline Server for your changes to take effect.



#### Important

If you do not need to reconfigure the Ambari-created Tez View, see Using the Tez View.

# **5.2. Creating or Editing the Tez View Instance**

Depending on whether you must create a new Tez View instance for a manually deployed cluster or modify an Ambari-created Tez View, see one of the following sections:

- Modifying a Tez View Instance on an Ambari-Managed Cluster [8]
- Creating a New Tez View Instance on a Manually-Deployed Cluster [8]

#### To modify a Tez View instance on an Ambari-managed cluster:

- 1. Navigate to the Ambari Administration interface.
- 2. Click **Views** and expand the **Tez View**.
- 3. On the Create Instance page, change the appropriate configuration parameters.
- 4. Select Local Ambari-Managed Cluster:

#### Figure 5.1. Tez View Create Instance Page

w	TEZ			
sion	0.7.0.2.3	0.0-2108 \$		
Details				
	nstance Name			e
	Display Name			
	Description			
		S Visible		
Cluster Con	figuration			
CLocal Ar	nbari Managed Cli	uster		
	Cluster Name	MyCluster 0		
O Custom				
YARN Time	ine Server URL	yam.timeline-service.hostname:8188		
		yam.resourcemanager.hostname:8088		
YARN Re	sourceManager URL			



#### Important

Secure clusters that use wire encryption (SSL/TSL) cannot use the **Local Ambari Managed Cluster** option. Instead you must configure the view as described in the instructions for manually-deployed clusters [8].

5. Click Save, grant Permissions on the view (see User Permissions for Tez Views), and click Go to instance to use the view. See Using the Tez View.

#### To create a new Tez View instance for a manually-deployed cluster:

1. Navigate to the Ambari Administration interface.

- 2. Click Views, expand the Tez View, and click Create Instance.
- 3. On the Create Instance page, select the Version.
- 4. Enter the Details (required). The Instance Name appears in the URI, the Display Name appears in the Views drop-down list, and the Description helps multiple users identify the view.
- 5. Scroll down to the Cluster Configuration, verify that **Custom** is checked and enter the following values, which tell the Tez View how to access resources in the cluster:

Property	Value
YARN Timeline Server URL (required)	The URL to the YARN Application Timeline Server, used to provide Tez information. Typically, this is the yarn.timeline-service.webapp.address property that is specified in the etc/hadoop/conf/ yarn-site.xml.
	When you enter the value in the view definition, pre- pend "http://" to the value you find in the yarn-site.xml file. For example, http:// <timeline server<br="">host&gt;:8188</timeline>
	For wire encryption-enabled clusters:
	Set this based on the value of yarn.timeline- service.webapp.https.address in yarn- site.xml
	When you enter the value in the view definition, pre- pend "https://" to the value. For example, https:// <timeline host="" server="">:8190</timeline>
YARN ResourceManager URL (required)	The URL to the YARN ResourceManager, used to provide YARN Application data. Typically, this is the yarn.resourcemanager.webapp.address property that is specified in the etc/hadoop/conf/ yarn-site.xml.
	When you enter the value in the view definition, pre- pend "http://" to the value you find in the yarn-site.xml file. For example, http:// <resourcemanager host&gt;:8088</resourcemanager 
	For wire encryption-enabled clusters:
	Set this based on the value of yarn.resourcemanager.webapp.https.address in yarn-site.xml
	When you enter the value in the view definition, pre- pend "https://" to the value. For example, https:// <resourcemanager host="">:8090</resourcemanager>

#### Table 5.2. Cluster Configuration Values for the Tez View in Ambari

- 6. Click Save and grant Permissions on the view (see User Permissions for Tez Views).
- 7. At the top of the view instance configuration page, click **Go to instance**.
- 8. When your browser is at the view instance page, copy the URL for the Tez View from your browser address bar:

#### Figure 5.2. Tez View Instance Page



- 9. In tez-site.xml, specify the URL that you copied in Step 8 as the value for the tez.tez-ui.history-url.base property, and save the file.
- 10Restart the HiveServer2 daemon to make sure that your changes to tez-site.xml take effect.

To use the view, see Using the Tez View.



#### Important

If your cluster is configured for Kerberos, you must set up Ambari Server for Kerberos for the Tez View to access the ATS component. See Kerberos Setup for Tez Views.

### 5.2.1. User Permissions for Tez Views

After saving the Tez View instance definition, grant permission on the view for the set of users who can use the view:

#### Figure 5.3. Granting User Permissions to Tez Views

Views / Tez View 😡	to instance Delete Instance
View TEZ	0.0-2108
Details	🖊 Edit
Instance Name	TEZ_CLUSTER_INSTANCE
Display Name	Tez View
Description	Monitor and debug all Tez jobs, submitted by Hive queries and Pig scripts (auto-created)
	2 Visible
Permissions	
	rmission to these users Grant permission to these groups
Permission Grant p	
Permission Grant pr	Add Group
Permission Grant pr Use	Add Group
Permission Grant pr Use Add Cluster Configuration	Add Group Fox
Permission Grant p Use Add Cluster Configuration * Local Amberi Managed C Cluster Name	Add Group Fox



Note

To grant access to all Hive and Pig users, create a group that contains these users, and then grant permission to use the Tez View to that group. See also the "Managing Users and Groups" section in the *Administering Ambari* guide.

### **5.2.2. Kerberos Setup for Tez Views**

To set up basic Kerberos for views, see "Set Up Kerberos for Ambari Server" in the Ambari Security Guide.

After you have set up basic Kerberos for the Tez View, you must set the following configuration properties:

1. On the timeline server host, set the following values for properties in the YARN configuration for Ambari-managed clusters or the <code>yarn-site.xml</code> for manually deployed clusters:

#### Table 5.3. Kerberos Settings for Tez Views

Property	Value
<pre>yarn.timeline-service.http- authentication.proxyuser.\${ambari principal name}.hosts</pre>	*
<pre>yarn.timeline-service.http- authentication.proxyuser.\${ambari principal name}.users</pre>	*
<pre>yarn.timeline-service.http- authentication.proxyuser.\${ambari principal name}.groups</pre>	*

For example, if the Kerberos principal used for the Ambari server is ambariservice@EXAMPLE.COM, replace \${ambari principal name} with ambariservice.

- 2. Restart the Timeline Server so your configuration changes take effect.
- 3. On the ResourceManager host, add the following properties to kerberize and enable proxy-user access to the ResourceManager UI:

#### Table 5.4. In core-site.xml

Property	Value	
hadoop.http.authentication.type	kerberos	
hadoop.http.filter.initializers	org.apache.hadoop.security.AuthenticationF	ilterInitialize
hadoop.http.authentication.kerberos.keytab	<path container="" http="" keytab="" principal="" to=""></path>	
	for example: <etc <br="" keytabs="" security="">spnego.service.keytab&gt;</etc>	
hadoop.http.authentication.kerberos.princi	phitp/_Host@realm	
	for example: HTTP/_HOST@EXAMPLE.COM	
hadoop.http.authentication.signature.secre	t<₽äilheto signature secret file for signing the authentication tokens>	

Property	Value
	The same secret should be used for all nodes in the cluster, NameNode, DataNode, ResourceManager, NodeManager. For example: /etc/hadoop/conf/ secret_http_file
hadoop.http.authentication.cookie.domain	<the cookie<br="" domain="" for="" http="" the="" to="" use="">that stores the authentication token&gt; for example: hortonworks.com. This setting is optional. The default is no domain.</the>
hadoop.proxyuser.\${ambari principal name}.hosts	*
hadoop.proxyuser.\${ambari principal name}.groups	*
hadoop.proxyuser.\${ambari principal name}.users	*

#### Table 5.5. In yarn-site.xml

Property	Value
yarn.resourcemanager.webapp.delegation- token-auth-filter.enabled	true
yarn.timeline-service.http- authentication.proxyuser.\${ambari principal name}.hosts	*
<pre>yarn.timeline-service.http- authentication.proxyuser.\${ambari principal name}.groups</pre>	*
yarn.timeline-service.http- authentication.proxyuser.\${ambari principal name}.users	*
The following settings are optional, if proxyusers settings are added in core-site.xml:	
yarn.resourcemanager.proxyuser.\${ambari principal name}.hosts	*
yarn.resourcemanager.proxyuser.\${ambari principal name}.users	*
yarn.resourcemanager.proxyuser.\${ambari principal name}.groups	*

# 5.3. Using the Tez View

Tez provides a framework that enables human-interactive response times with Apache Hive queries and Apache Pig data transformations. The Tez View enables you to understand and debug submitted Tez jobs, such as Hive queries or Pig scripts, that are executed using the Tez execution engine.

The following sections discuss using the Tez Views to manage Hive and Pig tasks:

- Understanding DAGs, Vertices, and Tasks
- Identifying the Tez DAG for Your Job
- Understanding How Your Tez Job Is Executed
- Identifying Causes of Failed Jobs

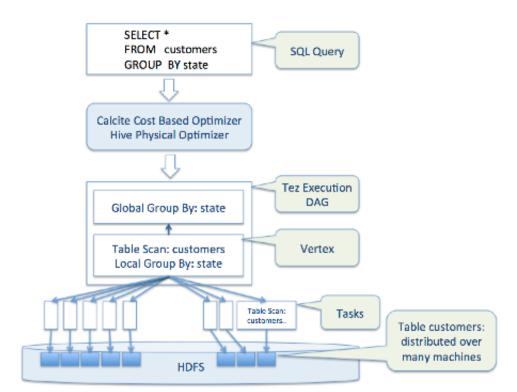
- Viewing All Failed Tasks
- Using Counters to Identify the Cause of Slow-Performing Jobs

### 5.3.1. Understanding Directed Acyclic Graphs (DAGs), Vertices, and Tasks

To explain DAGs, vertices, and tasks, consider how Hive SQL queries are compiled and converted into a Tez execution graph also known as a DAG. A *DAG* is a collection of vertices where each vertex executes a fragment of the query or script. Directed connections between vertices determine the order in which they are executed. For example, the vertex to read a table must be run before a filter can be applied to the rows of that table.

As another example, consider when a vertex reads a user table. This table can be very large and distributed across multiple computers and multiple racks. Reading the table is achieved by running many tasks in parallel. The following figure shows the execution of a SQL query in Hive:

#### Figure 5.4. SQL Query Execution in Hive



### 5.3.2. Identifying the Tez DAG for Your Job

#### To identify the Tez DAG for your job:

1. Navigate to the Tez View instance by clicking **Go to instance** on the Tez View page in Ambari. The Tez View instance page displays a list of jobs sorted by time, listing the latest jobs first. You can search a job using the following fields:

- Dag Name (DAG name for the job)
- Id (DAG identifier)
- Submitter (user who submitted the job)
- Status (job status)
- Application ID
- 2. When you have entered your search criteria, press **Enter**, and search results matching your criteria are returned below.

#### Selecting the Columns That Appear in Search Results

To select which columns are included in the Tez View search results, click the gear icon to the right of the search tool bar. A Column Selector dialog box appears where you can select which columns appear in the search results. Select the columns, and click **Ok** to return to the Tez View:

#### Figure 5.5. Tez View Column Selector Dialog Box

	-				
0		Column Selector		an 2015 22:18:55	C Refresh
	d Search	Select All Filter options		First 1	Nows
Dag Name	ld.	El Dag Name		Duration	Applic
hive_2015061421444	dag_1434303000	to id	15:06	9 secs	applica
amberi-qa_20150614	dag_1434230756	Ø Submitter	8:44	8 secs	applica
ambari-ga_20150614	dag_1434230756	Ø Status	7:23	9 secs	applica
OrderedWordCount	dag_1434230750	Ø Start Time	8.35	8 secs	applica
PigLatin:pigSmoke.sh	dag_1434230756	End Time	17:59	6 secs	applica
		Duration			
		Application ID			
		B Queue			
		FileSystem - FILE_BYTES_READ			
		FileSystem - FILE_BYTES_WRITTEN			
		FileSystem - FILE. READ. OPS			



#### Note

To search for columns, use the search well at the top of the Column Selector dialog box. Check **Select All** to include all columns in your search results and uncheck it to clear all of your column selections.

#### **Understanding Tez View Job Status**

The following table explains the job status field that is returned for all search results returned in the Tez View:

#### **Table 5.6. Tez Job Status Descriptions**

Status	Description
Submitted	The DAG is submitted to Tez but is not running.
Running	The DAG is currently running.
Succeeded	The DAG completed successfully.
Failed	The DAG failed to complete successfully.

Status	Description
Killed	The DAG was stopped manually.
Error	An internal error occurred when executing the DAG.

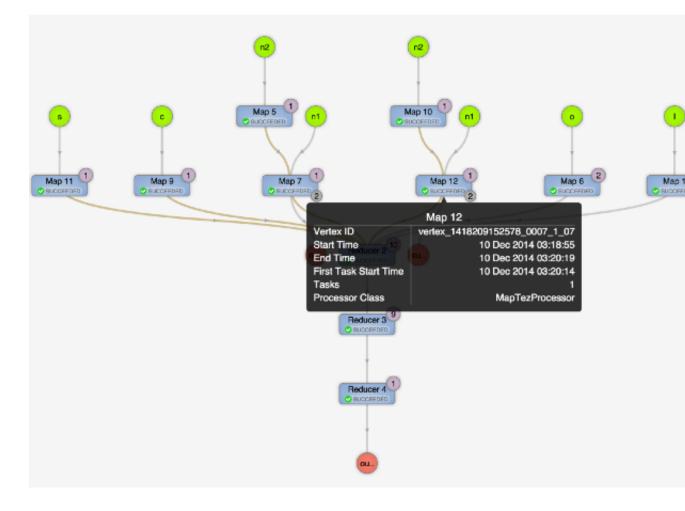
### 5.3.3. Understanding How Your Tez Job Is Executed

The Tez View enables you to gain insight into the complexity and the progress of executing jobs.

The View tab shows the following:

- DAG graphical view
- All vertices
- Tasks per vertex on top right of the vertex
- Failed vertices display in red, successful vertices display in green
- Mouse over vertices to view timeline details

#### Figure 5.6. View Tab in Tez View



The View Tab enables you to investigate the vertices that have failures or are taking a long time.

### 5.3.4. Identifying Causes of Failed Jobs

The Tez View enables you to quickly find and report errors. When a Tez task fails, you must:

- Identify why the task failed
- Capture the reason for task failure

When a Tez task fails, the DAG Details tab explains the failure:

#### Figure 5.7. DAG Details Window

	DAG De	tails DAG Counters	Graphical View	All Vertices	All Tasks	All TaskAttempts
Ð			L	ast refreshed at 1	4 Jun 2015 10	19:01 C Refres
DAG Details						
A Download	data					
Application Id	application_1434230750579_0006					
Entity Id	dag_1434230750579_0006_1					
User	ambari-ga.					
Status	FAILED [ Failed Tasks ] [ Failed TaskAttempts ]					
Start Time	14 Jun 2015 10:17:13					
End Time	14 Jun 2015 10:17:23					
Duration	9 secs					
Nagnostics						
» Task failed, » TaskAtte » Contu » Con Exit Exit Stac a	texhiame-Map 1, vertexki-vertex, 1-34230750578, 0006, tabidi-tabid, 1-43230750578, 0006, 1, 00, 00000, diagen prof balled, Hol- timer contamer, 1-64243075078, 0006, 01, 000000 finale tabinar star consumer, 1-642430750578, 0006, 01, 00000 finale tabinar star consumer, 1-642430750578, 0000, 01, 00000 finale tabinar star consumer, 1-642430750578, 0000, 01, 00000 finale tabinar star consumer, 1-642430750578, 0000, 01, 00000 finale tabinar star consumer, 1-642430750578, 00000, 010000 finale tabinar star consumer, 1-642430750578, 00000, 000000 finale tabinar star consumer, 1-642430750578, 00000, 000000, 000	stics= d with diagnostics set to sch.				

### **5.3.5. Viewing All Failed Tasks**

Multiple task failures may occur. The Tez View All Tasks tab enables you to view all tasks that failed and examine the reason and logs for each failure. Logs for failed tasks, but not for aborted tasks are available to download from this tab:

### Figure 5.8. Tez View All Tasks Tab

Ambari	мусі	uster <b>Cope</b> D					1	Dashboa	rd Services	Hosts	Alerts	Admin		🛦 admin 👻
🐔 All DAGs	/ D	AG [ hive_2015	06142	14445_ec5b3c3	1-96	21-4303-a9	10-950897	abd099	et ]					
					DA	G Details	DAG Cou	iters	Graphical View	All Ver	ices 🚺	I Tasks	All TaskAt	tempts
0										Last refresh	ed at 14 Ju	n 2015 22:4	0:58	Refresh
Status:FAILED				Search							First	1 Las	t - 1 B	
Task Index	0	Vertex Name	0	Status	0	Start Time		End Tim		Duration	0	Actions		Logs
00.000000		Map 3		FAILED		14 Jun 2015			015 14:45:06	7 secs		counters at		Not Avail

### **5.3.6. Using Counters to Identify the Cause of Slow-Performing Jobs**

The Tez View shows counters so you can understand why a task performs more slowly than expected. Counters help you better understand the task size and enable you to locate anomalies. Elapsed time is one of the primary counters to look for.

Counters are available at the DAG, vertex, and task levels:

#### Figure 5.9. Tez View DAG-Level Counters Tab

Ambari MyCluster (Bape Balents		Dashboar	rd Services	Hosts A	lerts Admin		<b>≜</b> admir
All DAGs / DAG [ OrderedWordCount ]							
	DAG Details	DAG Counters	Graphical View	All Vertices	All Tasks	All Ti	askAttempts
0			L	ast refreshed a	t 14 Jun 2015 25	:43:13	C Refresh
Counter Name				Counter Va	Nue		
Search							
org.apache.fez.common.counters.DAGCounter							
NUM_SUCCEEDED_TASKS				3			
TOTAL_LAUNCHED_TASKS				3			
DATA_LOCAL_TASKS				1			
AM_CPU_MILLISECONDS				1,550			
AM_GC_TIME_MILLIS				196			
File System Counters							
FILE_BYTES_READ				225			
FILE_BYTES_WRITTEN				161			

#### Figure 5.10. Tez View Vertex-Level Counters Tab

Ambari MyCluster (Depa Dalerta	Dashboard	Services	Hosts	Alerts	Admin	ш	admin
All DAGs / DAG [OrderedWordCount] / Vertex [Tokenizer]							
	Vertex Details Vert	ex Counters	Tasks	Task A	ttempts	Sour	ces & Sinks
0		L	ist refreshe	d at 14 Ju	n 2015 22:	13:41	© Refresh
Counter Name			Counte	r Value			
Search							
org.apache.tez.common.counters.DAGCounter							
DATA_LOCAL_TASKS			1				
File System Counters							
FILE_BYTES_READ			32				
FILE_BYTES_WRITTEN			89				

#### Figure 5.11. Tez View Task-Level Counters Tab

Ambari MyCluster Coor Coletta	Dashboard	Services	Hosts	Alerts	Admin	ш	<b>≜</b> admin
All DAGs / DAG [ OrderedWordCount ] / Vertex [ Tokenizer ]	/ Task [ 00_000000 ]						
			Task Details	Tasi	k Counters	Tas	k Attempts
Ø		L	ast refreshe	id at 14 Ja	un 2015 22:4	3:59	2 Refresh
Counter Name			Counte	r Value			
Search							
rg.apache.tez.common.counters.DAGCounter							
NATA_LOCAL_TASKS			1				
We System Counters							
ILE_BYTES_READ			32				
FILE_BYTES_WRITTEN			89				

#### **Monitoring Task Progress for Jobs**

The Tez View shows task progress by increasing the count of completed tasks and total tasks. This enables you to identify the tasks that might be "hung" and to understand more about long-running tasks.

# 6. Using the Pig View

Apache Pig is a scripting platform for processing and analyzing large data sets. Pig was designed to perform extract-transform-load (ETL) operations, raw data research, and iterative data processing. The **Pig View** provides a web-based interface to compose, edit, and submit Pig scripts, download results, and view logs and the history of job submissions.

This chapter explains:

- Configuring Your Cluster for the Pig View
- Creating the Pig View Instance
- Using the Pig View

# **6.1. Configuring Your Cluster for Pig View**

#### **Configuring HDFS for Pig**

You must set up an HDFS proxy user for the Ambari daemon account. For example, if ambari-server daemon is running as root, you set up a proxy user for root in core-site by clicking HDFS > Configs > Advanced > Custom core-site > Add Property, and then add the following key-value pairs in the Add Property dialog box:

hadoop.proxyuser.root.groups=\*

hadoop.proxyuser.root.hosts=\*

Click Add to add the property to core-site.

You must also set up an HDFS proxy user for WebHCat. For example, if your WebHCat server is running as hcat, you set up a proxy user for hcat in core-site. In the same HDFS advanced core-site settings pane, click **Add Property**, add the following key-value pairs to the Add Property dialog box, and then click **Add**to add them to core-site:

hadoop.proxyuser.hcat.groups=\*
hadoop.proxyuser.hcat.hosts=\*

Ambari views use the doAs option for commands. This option enables the Ambari *process user* to impersonate the Ambari *logged-in user*. To avoid receiving permissions errors for job submissions and file save operations, you must create HDFS users for all Ambari users that use the views.

#### **Configuring WebHCat for Pig**

You must set up a WebHCat proxy user for the Ambari daemon account. For example, if ambari-server daemon is running as root, you set up a proxy user for root in webhcatsite by clicking **Hive > Configs > Advanced > Custom webhcat-site > Add Property**, add the following key-value pairs in the Add Property dialog box, and then click **Add** to add them to webhcat-site: webhcat.proxyuser.root.groups=\*

webhcat.proxyuser.root.hosts=\*



#### Note

If Ambari Server is running as a non-root user, such as 'ambari', and you are planning on using Ambari Views, the following properties in **Services > HDFS > Configs > Advanced core-site** must be added:

```
hadoop.proxyuser.ambari.groups=*
hadoop.proxyuser.ambari.hosts=*
```

See the Ambari Security Guide for more information about How to Configure Ambari Server for Non-Root.



#### Note

This view stores user metadata in HDFS. By default, the location in HDFS for this metadata is /user/<user name of logged in user>. Since many users leverage the 'admin' account for getting started with Ambari, the /user/ admin folder needs to be created. To do so execute the following commands as the hdfs user:

```
su - hdfs
hadoop fs -mkdir /user/admin
hadoop fs -chown admin:hadoop /user/admin
```

# 6.2. Creating the Pig View Instance

- 1. Browse to the Ambari Administration interface.
- 2. Click Views, expand the Pig View, and click Create Instance.
- 3. On the Create Instance page, select **Version**. If multiple Pig View jars are present, choose one.
- 4. Enter the Details and Settings. The Instance Name appears in the URI, the Display Name appears in the Views drop-down list, and the Description helps multiple users identify the view:

liew	PIG	
lersion	1.0.0	•
Details		
	Instance Name	ETLPig
	Display Name	ETL Pig
	Description	Pig View for ETL team
		E Visible
Settings		Ø Vable
	HDFS Username	Visble
Web	HDFS Username S Authentication	
WebHDF		\$(userrame)
WebHDF WebHDF	S Authentication	\$(userrame)
WebHDFI WebHDFI Web	S Authentication HCat Username	\$(usemame) auth-SIMPLE

#### Figure 6.1. Pig View Details and Settings

5. Scroll down, and enter the Cluster Configuration information, which tells the Pig View how to access resources in the cluster. For a cluster that is deployed and managed by Ambari, select Local Ambari Managed Cluster:

ster Configuration		
Local Ambari Managed Clu	ister	
Cluster Name	MyCluster \$	
Custom		
ebHDFS FileSystem URI*	webhdfs://namenode:50070	
Logical name of the NameNode cluster		
List of NameNodes		
First NameNode RPC Address		
Second NameNode RPC Address		
First NameNode HTTP (WebHDFS) Address		
Second NameNode HTTP (WebHDFS) Address		
Failover Proxy Provider		
WebHCat Hostname*	webhcat-host.example.com	
WebHCat Port*	50111	

#### **Figure 6.2. Pig View Cluster Configuration**

6. Click **Save**, give Permissions to the appropriate users and groups, and click **Go to instance** at the top of the page to go to the view instance.

### 6.2.1. Getting Correct Configuration Values for Manually-Deployed Clusters

If you have manually deployed your cluster, you must enter cluster configuration values in the Pig View Create Instance page. The following table explains where you can find cluster configuration settings in Ambari.

#### Table 6.1. Finding Cluster Configuration Values for the Pig View in Ambari

Property	Value
Scripts HDFS Directory*	/user/\${username}/pig/scripts
Jobs HDFS Directory*	/user/\${username}/pig/jobs
WebHDFS FileSystem URI*	Click HDFS > Configs > Advanced hdfs-site > dfs.namenode.http-address. When you enter the value in the view definition, pre-pend "webhdfs://" to the value you find in the advanced HDFS configuration settings. For example, webhdfs:// c6401.ambari.apache.org:50070
WebHCat Hostname*	Click Hive > Configs > Advanced > WebHCat Server > WebHCat Server host to view the hostname. For example, c6402.ambari.apache.org
WebHCat Port*	Click Hive > Configs > Advanced > Advanced webhcat-site > templeton.port to view the port number. For example, 50111

#### For NameNode High Availability

The following values must be entered for primary and secondary NameNodes:

#### Table 6.2. Pig View Settings for NameNode High Availability

Property	Value
First NameNode RPC Address or Second NameNode RPC Address	Select the primary or secondary NameNode to view settings from that host in the cluster. See how to get the NameNode RPC address [21]. When you enter the value in the view definition, pre-pend "http://" to the value you find in the advanced hdfs-site settings. For example, http://c6401.ambari.apache.org:8020
First NameNode HTTP (WebHDFS) Address or Second NameNode HTTP (WebHDFS) Address	Click HDFS > Configs > Advanced > Advanced hdfs-site > dfs.namenode.http-address. When you enter the value in the view definition, pre-pend "http://" to the value you find in the advanced hdfs-site settings. For example, http://c6401.ambari.apache.org:50070

#### To get First NameNode RPC Address values:

 Navigate to the HDFS service page in Ambari that contains links to individual NameNodes. Click NameNode (primary) or SNameNode (secondary) to view the host page:

#### Figure 6.3. HDFS Service Page in Ambari



- 2. On the host page, click **Configs > Advanced**.
- 3. Enter "rpc" in the Filter search well at the top right corner of the page or navigate to the Advanced hdfs-site settings to find the dfs.namenode.rpc-address value that you can enter into the Pig View definition. Here is an example of using the Filter to locate a value:

#### Figure 6.4. Using the Filter to Search Advanced hdfs-site Settings

c6401.a Back	mbari.ap	ache.org							
Summary	Configs	Alerts 🖸	Versions					Host A	ections •
HDFS MapReduc	#2	Group	HDFS Default (1)	Change			po	0	*
YARN Tez		Setting	Advanced						
Hive Pig		- A	vanced hoffs-site						
ZooKeeper		dfs.nar	menode.rpc-	c6401.ambari.ap	pache.org:8020				

### **6.2.2. User Permissions for Pig Views**

After saving the Pig View instance definition, grant permission on the view for the set of users who can use the view:

Views / M	y Pig \	View Go to Instance	Delete Instance
View Version	PIG 1.0.0		
Details			✓ Edt
Instance	Name	MyPigView	
Display	Name	My Pig View	
Descr	ription	description	
Permissions			
Permission	Grant p	ermission to these users	Grant permission to these groups
Use			Add Group
Settings			✓ Edit

Figure 6.5. Granting User Permissions to Pig Views

### **6.2.3. Kerberos Setup for Pig Views**

To set up basic Kerberos for views, see "Set Up Kerberos for Ambari Server" in the Ambari Security Guide. After you have set up basic Kerberos for the Pig View, Pig requires that WebHDFS Authentication be set to auth=KERBEROS;proxyuser=<ambari-user-principal>

For example, see the following figure:

#### Figure 6.6. Kerberos Settings for Pig Views

Properties		🖌 Edit
WebHDFS FileSystem URI*	webhdfs://erik-views-1.c.pramod-thangail.internal:50070	
WebHDFS Username	\$(usemame)	
WebHDFS Authentication	auth=KERBEROS;proxyuser=ambariuser	
WebHCat URL*	http://erik-views-3.c.pramod-thangaii.internal:50111/templeton/v1	
WebHCat Username	\${username}	
Dataworker Username	\$(username)	
Scripts HDFS Directory*	/user/\$(username)/pig/scripts	
Jobs HDFS Directory*	/user/\${usemame}/pig/jobs	
Meta HDFS Directory	/user/\$(usemame)/pig/store	

### 6.3. Using the Pig View

Use the Pig View to:

- Write Pig scripts
- Execute Pig scripts
- Add user-defined functions (UDFs) to Pig scripts
- View the history of all Pig scripts run by the current user

### 6.3.1. Writing Pig Scripts

Navigate to the Pig View instance Scripts page, and click **New Script** in the upper right corner of the window. Name the script in the New Script dialog box, click **Create**, and enter your script into the editor. After you have written the script, you can use the execute button on the upper right to run it. Check the box that is adjacent to the execute button to use Tez instead of the default MapReduce engine.

The following figure shows a running Pig script:

mbari MyCluster	Test Date:	Dashboard S	ienvices Hosts	Alerts Adr	nin 111	🛦 ambari-qa 🔹
Pig ETL 1	X Soript History					
D Gave	Pig_ETL_1 🥒				Execute on Tez	Execute -
(2) Copy	PIG helper + UDF helper +		/user/anbari-qa	/pig/scripts/p	iget 11-2015	-06-15_02-55.pig
	3 grp_data = GROUP run 4 max_runs = FOREACH g 5 join_max_run = JOIN :	ng GENERATE 10 as playeri a by (year); rp_data GENERATE group as max_runs by (10, max_runs join_max_run GENERATE 10 ;	grp,MAX(runs.r), runs by (yes	ins) os max_r		
			-			
	Arguments		-			

#### Figure 6.7. Pig Script Running in the Pig View

### **6.3.2. Viewing Pig Script Execution History**

The History tab shows the history of Pig scripts run by the current user. A particular script in history can be clicked to open it in a new Script tab to view its details:

#### Figure 6.8. Pig View Script History Tab

🔬 A	mbari MyC	uster (Cope Calenta		Dashboard Se	rvices Hosts	Alerts Admin	III 4	ambari-qa •
$\langle \rangle$	Scripts	History						
ý	UDFs	Date	Script	Status	Duration	Actions		
0	History	2015-06-15 08:00	Pig_ETL_1	RUNNING		B Delete		
						Show: 10	0 1-1 of	1

### 6.3.3. User-Defined Functions (UDFs) Tab

UDFs can be added to Pig scripts by clicking **Create UDF** in the upper right corner of the UDFs window. In the Create UDF dialog box, point to a UDF in the system by specifying the name and path:

#### Figure 6.9. Pig View UDFs Tab

🔬 Am	bari MyCluster 🛛 ops 🛛	alerta	Dashboard	Services	Hosts	Alerts	Admin	 ▲ ambari-qa •
Φ	Scripts	UDFs						+ Create UDF
ø	UDFs	Name	Path		Owne	r		
Ø	History	No UDFs to display						

# 7. Using the Capacity Scheduler View

The Yarn Capacity Scheduler allows for multiple tenants in an HDP cluster to share compute resources according to configurable workload management policies.

The Capacity Scheduler View is designed to help hadoop operators configure these policies for YARN. In the View, operators can create hierarchical queues and tune configurations for each queue to define an overall workload management policy for the cluster.

In this section:

- Configuring your Cluster for the Capacity Scheduler View
- Creating a Capacity Scheduler View Instance
- Using the Capacity Scheduler View
- Troubleshooting

# 7.1. Configuring your Cluster for the Capacity Scheduler View

The Capacity Scheduler View requires that the cluster is managed by Ambari – the view utilizes the Ambari Server API.

# 7.2. Creating a Capacity Scheduler View Instance

When you deploy a cluster using Ambari, a Capacity Scheduler View instance is automatically created. If you do not need to reconfigure the Ambari-created cluster, proceed to the next section, Using the Capacity Scheduler View.

If you have deployed your cluster manually, or if you need to re-configure the Ambaricreated Capacity Scheduler View, you can use the information in this section to create and configure a view instance.

Use the following steps to set up a Capacity Scheduler view instance.

1. Select admin > Manage Ambari in the Ambari Web top menu.

						Cthe Othe	er Bookmarks
Dashboard	Services	Hosts	Alerts	Admin		💄 admin 👻	
_						About	
						Manage An	nbari
						Settings	
						Sign out	
HDFS Links		Memory U	sage	N	etwork U	Isage	

2. On the Manage Ambari page, click Views.

🚕 Ambari		🗰 🔺 admin 🗸
Clusters test_cluster1 Permissions	Welcome to Apache Ambari Monitor your cluster resources, manage who can access the clust	iter, and customize views for Amberi users.
Go to Deshboard Versions	Operate You Manage the configuration of your cluster a	
₩ Views Views & User + Group Management	Manage Permissions	Go to Dashboard
Users Groups	Manage Users + Groups Manage the users and groups that can access Ambari	Deploy Views Create view instances and grant permissions
	Usera Groupa	Views

3. On the Views page, click CAPACITY-SCHEDULER, then click Create Instance.

Clusters	Views		Search Q
test_oluster1 © Permissions Go to Dashboard	View Name	Instances DULER 0.4.0 (1)	
Versions		YARN Queue 0,4.0 Manager	Manage YARN Capacity Scheduler
## Views		+ Create Instance	
Views	> FILES	0.2.0 (0)	
LUser + Group Management	> HIVE	0.4.0 (0)	
Users	> PIG	0.1.0 (0)	
Groups	> SLIDER	2.0.0 (0)	
	> TEZ	0.7.0.2.3.0.0-812 (1)	

4. In the Details box on the Create Instance page, type in an instance name, display name, and a description for the view.



Note

The instance name cannot contain spaces or special characters.

- 5. In the Cluster Configuration box on the Create Instance page, configure the view to communicate with the HDP cluster.
  - For HDP clusters that are local (managed by the local Ambari Server), select the Local Ambari Managed Cluster option, then select the local cluster name.

- To configure the view to work with HDP clusters that are remote (not part of this Ambari Server instance), select the **Custom** option, then specify the remote Ambari cluster API URL and the Ambari cluster user name and password.
- 6. Click **Save** at the bottom of the page.

🔬 Ambari		🗰 🔺 admin 🗸			
Clusters	Views / Create Insta	nce			
test_cluster1 © Permissions Go to Dashboard	View         CAPACITY-SCHEDULER           Version         0.4.0         0				
Versions	Details				
III Views	Instance Name	Capacity_Scheduler_1			
Views	Display Name	Capacity Scheduler 1			
LUser + Group Management	Description	Capacity Scheduler configuration 1			
Users Groups		C Visible			
	Cluster Configuration				
	<ul> <li>Local Ambari Managed Clus</li> </ul>	ster			
	Cluster Name	test_cluster1 \$			
	Oustom				
	Ambari Cluster URL*	http://ambarl.server;8080/api/v1/clusters/MyCluster			
	Operator Username*	djones			
	Operator Password*	····· •			
		Cancel			

7. The Capacity Scheduler View instance is created, and the configuration page for the instance appears.

Clusters	Views / Capacity Sc	Views / Capacity Scheduler 1 Go to Instance		Delete Instance		
test_cluster1 C Permissions Go to Dashboard Versions	View CAPACITY- Version 0.4.0	SCHEDULER				
III Views	Details			✓ Edit		
Views	Instance Name	Capacity_Scheduler_1				
LUser + Group Management	Display Name	Capacity Scheduler 1				
Users Groups	Description	Capacity Scheduler configuration	1			
	⊘ Valble Permissions					
	Permission Grant per	Permission Grant permission to these users Grant per				
	Use Add U					
	Cluster Configuration			✓ Edit		
	<ul> <li>Local Ambari Managed Clu Cluster Name</li> </ul>	test_cluster1 \$				
	Custom					
	Ambari Cluster URL*	Ambari Cluster URL* http://ambari.server:8080/api/v1/clu				

### 7.2.1. User Permissions for Capacity Scheduler Views

Use the following procedure to add users and groups to a Capacity Scheduler view instance.

1. On the Capacity Scheduler view instance configuration page, click the box labeled Add User in the Permissions box.

🝌 Ambari					💷 🔺 admin 🗸
Clusters	Views / Capacity Scheduler 1 Go to instance				Delete Instance
test_cluster1 © Permissions Go to Deshboard Versions	View Version	0.4.0	SCHEDULER		
III Views	Details			🖊 Edit	
Views	Instan	ce Name	Capacity_Scheduler_1		
LUser + Group Management	Displ	ay Name	Capacity Scheduler 1		
Users Groups	Description		Capacity Scheduler configuration 1		
			Visible		
	Permissions				
	Permission Grant permission to these users Grant permission to t		Grant permission to these gro	oups	
	Use	Add U	ser.	Add Group	
	Cluster Configura	ation			🖌 Edit
	Local Ambari Managed Cluster     Cluster Name test_cluster1 \$				
	Custom				

2. Enter user names in the Use box, then click the blue check mark to add the users. You can use the same method to add groups in the Add Group box.

🝌 Ambari					🛓 admin 🕶
Clusters	Views / Cap	acity Schedu	ler 1 Co to instance	D	elete Instance
test_cluster1 C Permissions Go to Dashboard	View Version	0.4.0	JLER		
Versions					
III Views	Details				🖉 Edit
Views	Insta	Cape	city_Scheduler_1		
LUser + Group Management	Disp	Capa Capa	city Scheduler 1		
Users Groups	D	escription Capa	city Scheduler configuration 1		
Groups		☑ Visit	ie		
	Permissions	Grant permission	to these users	Grant permission to these groups	
	Use	bemith × dj	ones ×	Add Group	
	Cluster Configur	ation			✓ Edt
		Managed Cluster ster Name test_c	luster1 ‡		
	Custom				
	Ambari Clu	ster URL* http://	/ambari.server:8080/api/v1/cluster	s/MyCluster	

3. After you have finished adding users and groups, click **Go to instance** at the top of the page to open the Capacity Scheduler view instance.

🝌 Ambari			🔠 🔺 admin 👻
Clusters	Views / Capac	ity Scheduler 1 Go to Instance	Delete Instance
test_cluster1 @	View CA	PACITY-SCHEDULER	
Go to Dashboard Versions	Version	0.4.0	
III Views	Details		🖌 Edit
Views	Instance M	Capacity_Scheduler_1	
LUser + Group Management	Display M	Vame Capacity Scheduler 1	
Users Groups	Descri	ption Capacity Scheduler configuration 1	
Groupa		Visible	
	Permissions		
	Permission 0	Grant permission to these users	Grant permission to these groups
	Use	bamith djones	product_management
	Cluster Configuration	n	🖌 Edit
	<ul> <li>Local Ambari Man Cluster I</li> </ul>		
	Custom		
	Ambari Cluster	URL* http://ambari.server:8060/api/v1/cluste	ns/MyCluster
	Operator Usern	ame* admin	

4. The Capacity Scheduler view instance page appears.

🔬 Ambari te	st_cluster1 10 apr 1	aler1		Dashboard	Services	Hosts	Alerta	Admin	ш	🛓 admin 🕶
+ Add Queue	Action	s <del>-</del>	Click on a queue to the left f	or details.						
root (100%)		~								
default (	100%)	× .								
Scheduler		× .								
Maximum Applications	10000									
Maximum AM Resource	20 %									
Node Locality Delay	40									
Calculator	org.apache.hadoop.)	jam.								
Versions										
v1 Current	45 years ago	load								

# 7.3. Using the Capacity Scheduler View

The Capacity Scheduler View is designed to help hadoop operators configure workload management policies for YARN. In the Capacity Scheduler View, operators can create hierarchical queues and tune configurations for each queue to define an overall workload management policy for the cluster.

## 7.3.1. Setting up Queues

Use the following steps to set up Capacity Scheduler queues on a view instance.

1. On the Capacity Scheduler view instance configuration page, click **Add Queue**. The queue will be added under the top level, or "root" queue. A "default" queue already exists under the root queue.



#### Note

To return to a previously created Capacity Scheduler view instance, click **Views** on the Manage Ambari page, then click **CAPACITY-SCHEDULER**. Click the applicable Capacity Scheduler view instance, then click **Go to instance** at the top of the page.

🝌 Amberi test_cluste	rl - Dage <mark>1 alert</mark>		Dashboard	Services	Hosts	Alerts	Admin	ш	🛦 admin 🔹
+ Add Queue	Actions +	Click on a queue to the left for	r details.						
root (100%)	~								
default (100%)	× .								
Scheduler	× .								
Maximum Applications 1000 Applications 20 Maximum AM 20 Resource 40 Node Locality 40 Delay Calculator org.1	0 El								
Versions VI Const 46 years	ago load								

2. Type in a name for the new queue, then click the green check mark to create the queue. In the following example, we're creating the "Engineering" queue.

Ambari test	_						-
Engineering	×	×.	Click on a queue to the left for de	als.			
root (100%)		1					
default (10	0%)	1					
Scheduler		1					
Maximum	10000						
Maximum AM	20 %						
Resource	20 99						
Node Locality	40						
Delay							
Calculator	org.apache.hadoop.yam						
Versions							
v1 Current 4	6 years ago lo	nd					

3. The "Engineering" queue is added, and its configuration page appears.

Amberi test_cluster1 Copp 1 dert	Dashboard	Services Hosts Alerts Admin III 🔺 admin
Add Queue     C Actions -     cot (100%)	Engineering 2	ж
📥 default (100%)	Capacity	Level Total 100%
Englineering (0%)	Engineering Capacity: 0 56	Enable node labels  Max Capacity:  0  96
Maximum 10000	v 9	how Peer Level Queues
Applications Maximum AM 20 %	Access Control and Status	Resources
Resource Node Locality 40 Delay Caloulator org.apache.hadoop.yarr	State         Running         Stopped           Administer Queue         Anyone         Custom           Submit Applications         Anyone         Custom	User Limit Factor 1 Minimum User Limit 100 56 Maximum Applications Inherited Maximum AM Resource Inhe 56
Versions		Ordering policy fito \$

4. The sum of queue capacities at any level in the Capacity Scheduler configuration must total 100%. Here the default queue is already set to 100%. Therefore, if we try to set the "Engineering" queue capacity to 60%, error messages appear warning that the total at this level is 160%.

Scheduler       Capacity: 00 %         Maximum AM       20 %         Maximum AM       20 %         Node Locality       40         Delay       State         Calculator       Grapacity: Engineering         Calculator       7%         Bubmit       Aryone Custom         Applications       Maximum AM Particulations	🝌 Ambari test_cluster1 🚾 🚛 🖬 🗤	Dashboard	Services Hosts Alerts Admin III 🔺 admin
Engineering   Scheduler   Maximum AM   Applications   Maximum AM   Calculater   Org.apache/hadoop.yerr     Vanions   Administer   Angolications   Maximum AM   Calculater   Org.apache/hadoop.yerr     Vanions     Vanions <td></td> <td></td> <td>3</td>			3
Scheduler       Capacity: 00 %         Maximum Applications       Capacity: 00 %         Maximum AM       20 %         Node Locality       40         Delay       State         Render Stopped         Administer       Anyole Custorn         Submit       Anyole Custorn         Maximum AM Resource       Internet	💼 default (100%) 🛛 🔺 💌	Capacity	Level Total 160%
Maximum Applications     10000     Image: Constraint of Constrain			Max Capacity: 60 %
Maximum AM Resource     20     %       Node Locality Delay     40     State     Running Stopped     User Limit Factor     1       Calculator     org.apache/hadcop.yarr     Administer Queue     Anyone Custom     Minimum User Limit     100     %       Submit     Anyone Custom     Maximum Applications     Maximum Applications     Maximum Applications     Writed		~	Show Peer Level Queues
Delay     Administer     Anyone     Custom     Minimum User Limit     100     %       Calculator     org apache/hadoop.yam     Submit     Anyone     Custom     Maximum Applications     Interted       Namimum User Limit     Anyone     Custom     Maximum Applications     Interted       Namimum User Limit     Maximum Applications     Interted     Interted			
Submit Anyone Custom Maximum Approximate 16	Delay		Minimum User Limit 100 %
Versions Ordering policy the 2	Calculator org.apache.hadoop.yam		
vi Gurrent 46 years ago load			Ordering policy the #

5. If we click the "default" queue and set its capacity to 0%, the error messages no longer appear, and the Level Total bar at the top of the page lists the total queue capacity at this level as 60%.

+ Add Queue	7 Actions -	default root.default	
default (D%)	<b>Ø</b>	Capacity	Level Total 60%
Engineering (60%)	0 V	default Capacity: 0  56	Enable node labels     Max Capacity: 100 %
Maximum 10000			✓ Show Peer Level Queues
Applications Maximum AM Resource Node Locality Deby Calculator org.apache.h		Access Control and Status State Faming Stopped Administer Queue	Hesources User Limit Pactor I Minimum User Limit 100 96 Maximum Applications Informed
Versions		Submit Anyone Custom Applications	Maximum AM Resource Inhe 36
VI Connot 46 years ago	load		Ordering policy ‡

6. To add more queues at the root level, click the **root** queue, then click **Add Queue**. In the following example, we have added a "Support" queue set to 10% of the level capacity, and a "Marketing" queue set to 30%. The root-level queue capacities now total 100%.

root (100%)	1	Markerting C		×
default (0%)	1	Capacity	Level Total 100	n.
Engineering (50%) Markerting (50%) Support (10%)	0 10 0	Markerting Capacity: 30 96	Max Capacity: 30 %	Enable node labels
Scheduler	× .	Access Control and Status	Resources	
Applications Maximum AM Resource Node Locality Delay		State Furning Stopped Administer Anyone Custom Gueue Submit Anyone Custom	User Limit Factor Minimum User Limit Maximum Applications Maximum AM Resource	1 100 56 Inherted
Calculator org.apache.had	loop.yarr		Ordering policy	ffo \$
Versions				

7. To save your configuration, click **Actions > Save Only**. On the **Notes** pop-up, enter an optional description of your changes, then click **Save**. Each version is retained and listed in the Versions box.

+ Add Queue	C Actions -	Markerting C	×
Save and Refresh	Queues	Capacity	Level Total 100%
Download config     Markenting (30%)     Support (10%)	0	Markerting Capacity: 30 56	Capacity: 30 %
			V Show Peer Level Queues
Scheduler	× .	Access Control and Status	Resources
Maximum Applications 10000 Maximum AM 20 55 Resource	0	State Ruming Stopped Administer Queue	User Limit Factor 1 Minimum User Limit 100 95
Node Locality 40 Delay		Submit Anyone Custom Applications	Maximum Applications Inhuman Maximum AM Resource Inhum 56
Calculator org.apact	he.hadoop.yam		Ordering policy file ()
Versions			
vi Gurrent 46 years ago	load		

8. To build a queue hierarchy, click a top level queue, then click Add Queue. In the following example, the "qa" and "development" queues have been added under the "Engineering" queue.

+ Add Queue		stions +	qa 🕜	2
root (100%	i)	~	root.Engineering.qa	
defaul	t (D96)	~	Capacity	Level Total 100%
Engine	ering (60%)	~	qa	Enable node labels
e de	welopment (20%)	c	Capacity: 60 %	D Max Capacity: 80 96 D
q	(80%)	•		A Hide Feer Level Queues
<ul> <li>Marke</li> </ul>	rting (30%)	~	development	Enable node labels
<ul> <li>Suppo</li> </ul>	rt (10%)	~	Capacity: 20 %	D Max Capacity: 20 % D
Scheduler		~		
		2	Access Control and Status	Resources
Maximur Application			State Running Stopped	User Limit Factor
Maximum Al			Administer Anyone Custom	Minimum User Limit 100 %
Resourc			Queue	Minimum User Limit 100 %
Node Localit Dela			Submit Anyone Custom	Maximum Applications Inherited
Calculate	er org.apache.hado	op. yant	Applications	Maximum AM Resource Inhe 96
				Ordering policy 110 \$
Versions				
vi Current	2 hours ago	load		
48	3 hours ago	load		
42	3 hours ago	load		
with the second s	46 years ago	load		

## **7.3.2. Configuring Queues**

To configure a queue, click the queue name, then set the following queue parameters:



#### Note

Hold the cursor over a parameter name to display a description of the parameter.

#### Capacity

- Capacity The percentage of cluster resources available to the queue. For a sub-queue, the percentage of parent queue resources.
- Max Capacity The maximum percentage of cluster resources available to the queue. Setting this value tends to restrict elasticity, as the queue will be unable to utilize idle cluster resources beyond this setting.
- Enable Node Labels Select this check box to enable node labels for the queue.

#### **Access Control and Status**

- State Running is the default state. Setting this to Stopped lets you gracefully drain the queue of jobs (for example, before deleting a queue).
- Administer Queue Click **Custom** to restrict administration of the queue to specific users and groups.
- Submit Applications Click **Custom** to restrict the ability to run applications in the queue to specific users and groups.

#### Resources

- User Limit Factor The default value of "1" means that any single user in the queue can at maximum only occupy the queue's configured capacity. This prevents users in a single queue from monopolizing resources across all queues in a cluster. Setting the value to "2" would restrict the queue's users to twice the queue's configured capacity. Setting it to a value of 0.5 would restrict any user from using resources beyond half of the queue capacity.
- Minimum User Limit This property can be used to set the minimum percentage of resources allocated to each queue user. For example, to enable equal sharing of the queue capacity among five users, you would set this property to 20%.
- Maximum Applications This setting enables you to override the Scheduler Maximum Applications setting (described in Configuring Cluster Scheduler Settings). The default setting is Inherited (no override).
- Maximum AM Resource This setting enables you to override the Scheduler Maximum AM Resource setting (described in Configuring Cluster Scheduler Settings). The default setting is Inherited (no override).
- Ordering Policy You can specify FIFO (First In, First Out) or fair (Fair Scheduler: applications get a fair share of capacity regardless of the order in which they were submitted).

+ Add Que	Act	ions +	Engineerin	g 🕜			×
	0%)	~	root.Engineering				
def	ault (0%)	~	Capacity		Level To	tal 10	55 ,
💶 🖿 Eng	gineering (80%)	<b>~</b>	Engineering				P Enable node labels
<ul> <li>Ma</li> </ul>	rkerting (30%)	~	Capacity: 60	%	- Max	Capacity: 60 %	
e Su	pport (10%)	× .					Node Labels Access
Scheduler		~			v Show Peer I	Level Queues	
Maxir	mum 10000 E	i	Access Contro	and Status		Resources	
Applicat Maximum Reso	n AM 20 %		State Administer Queue	Running Stopped Anyone Custom		User Limit Factor	1 20 %
Node Los	ality 40 Delay		Users	bsmith,djones	40	Maximum Applications	Inherited
Calcu	lator org apache hado	sp.yam	Groups	project_management	삼	Maximum AM Resource	Inhe 56
Versions			Submit Applications	Anyone Custom		Ordering policy	ffo \$
ut Current	a minute ago	load	Users	bsmith,djones	4		
*5	16 minutes ago	load	Groups	project_management	*		
	3 hours ago	load					
43	4 hours ago	load					
#2	4 hours ago	load					
et.	46 years ago	bad					

The following image shows the example "Engineering" queue with these settings specified:

# **7.3.3. Configuring Cluster Scheduler Settings**

You can use the Scheduler box to set global capacity scheduler settings that apply to all queues.

Ambari test_ci			Dashboard	I Service	s Hosts	Alerts Ac	amin 🎫	≜ admin
+ Add Queue	C Actions -	Engineering @						×
root (100%)	~	root.Engineering						
default (0%)	~	Capacity		Level To	otal		00%	
Engineering (	2096) 🧭	Engineering					97 Enab	le node labels
Markerting (3	296) 🗸	Capacity: 60 %	<u> </u>	- Ma	K Capacity: 1	96		0 💿
Support (10%	• •						Node Labels	Access 🔹
Scheduler	~			v Show Peer	Level Queues			
Maximum	10000	Access Control and	Status		Resource	:65		
Applications Maximum AM	20 %	State Ru	ming Stopped		Us	er Limit Factor	1	
Resource	20 76	Administer An Queue	vone Custom		Minin	num User Limit	20 96	
	40	_		40	Maximu	m Applications	Inherited	
Node Locality Delay		Users bi	mith, djones					
	org.apacha.hadoop.yarr		oject_management		Maximum	n AM Resource	inhe 96	
Delay	org.apacha.hadoop.yarr	Groups p		40 20		n AM Resource Ordering policy	Inhe 96	
Delay	org.apachs.hadoop.yarr	Groups p	oject_management					

The following Scheduler global parameters are available:

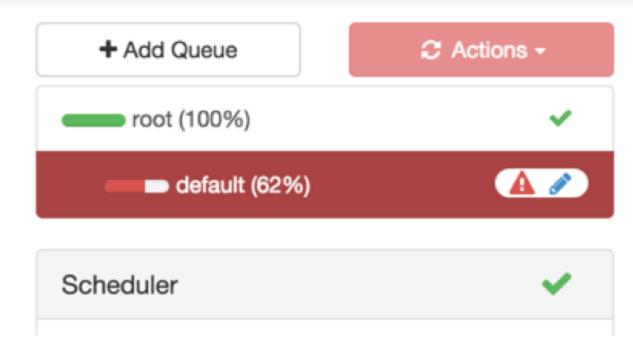
- Maximum Applications To avoid system-thrash due to an unmanageable load caused either by malicious users, or accidentally the Capacity Scheduler enables you to place a static, configurable limit on the total number of concurrently active (both running and pending) applications at any one time. This property is used to set this limit, with a default value of 10,000.
- Maximum AM Resource The limit for running applications in any specific queue is a fraction of this total limit, proportional to its capacity. This is a hard limit, which means that once this limit is reached for a queue, any new applications submitted to that queue will be rejected, and clients will have to wait and retry later.
- Node Locality Delay The number of missed scheduling cycles after which the scheduler attempts to schedule rack-local containers.
- Calculator The method by which the scheduler calculates resource capacity across resource types.

# 7.3.4. Applying the Configuration Changes

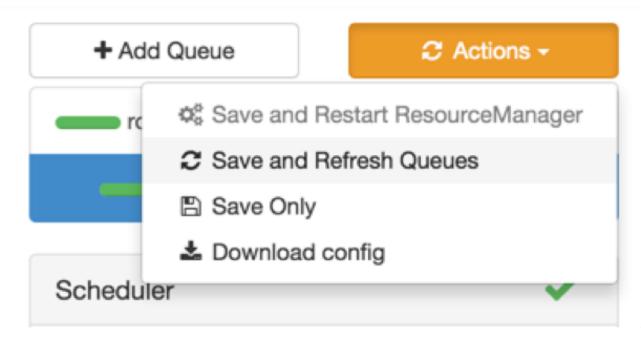
You can use the Actions menu to apply configuration changes made to the queue hierarchy.

Depending on the configuration changes made, the Actions menu will guide you to the options available to apply the changes.

For changes that are not valid and cannot be applied, the **Actions** button will turn red, and the menu will not appear.



For configuration changes that can be applied dynamically (without restarting the YARN ResourceManager), the Actions Menu will guide you to **Save and Refresh Queues**.



For configuration changes that require a restart of the YARN ResourceManager, the Actions Menu will guide you to **Save and Restart ResourceManager**.

+ Add	Queue	🛱 Actions 🗝
en re	¢ Save and R	estart ResourceManager
	C Save and R	efresh Queues
	🖺 Save Only	
	🕹 Download c	onfig

# 7.4. Troubleshooting

If you encounter an issue where the configurations cannot be applied from the View, you should go to the local Ambari Server instance managing the cluster and directly edit the Capacity Scheduler configuration from the YARN configuration page.

In the local Ambari instance, navigate to **Services > YARN**, then select the **Configs** tab. On the **Advanced** tab, expand the Scheduler section.

yarn.resourcemanager. scheduler.class	org.apache.hadoop.yarn.server.resourcemanager.scheduler.capacity.CapacityScheduler.capacity.capacityScheduler.capacity.capacityScheduler.capacity.capacityScheduler.capacity.capacityScheduler.capacity.capacityScheduler.capacity.capacityScheduler.capacity.capacity.capacityScheduler.capacity	<b>a</b>	•	C
Capacity Scheduler	yarn.scheduler.capacity.maximum-am-resource-percent=0.2 yarn.scheduler.capacity.maximum-applications=10000 yarn.scheduler.capacity.node-locality-delay=40 yarn.scheduler.capacity.queue-mappings-override.enable=false yarn.scheduler.capacity.root.accessible-node-labels=* yarn.scheduler.capacity.root.acl_administer_queue=* yarn.scheduler.capacity.root.capacity=100 yarn.scheduler.capacity.root.default.acl_submit_applications=* yarn.scheduler.capacity.root.default.capacity=100 yarn.scheduler.capacity.root.default.capacity=100	0	c	

Here you will be able to edit the underlying configurations for the Capacity Scheduler and fix any issues you may encounter.

# 8. Using the Hive View

Hive is a data warehouse infrastructure built on top of Hadoop. It provides tools to enable data ETL, a mechanism to put structures on the data, and the capability to query and analyze large data sets that are stored in Hadoop. The **Hive View** is designed to help you author, execute, understand, and debug Hive queries.

This chapter explains:

- Configuring Your Cluster
- Creating the Hive View Instance
- Using the Hive View
- Troubleshooting



#### Important

The Tez View integrates with the Hive View. Please install the Tez View when you install the Hive View. See Using the Tez View for more information.



#### Important

It is critical that you prepare your Ambari Server for hosting views. It is strongly recommended you increase the amount of memory available to your Ambari Server, and that you run additional "standalone" Ambari Servers to host the views. See Preparing Ambari Server for Views and Running Ambari Server Standalone for more information.

# 8.1. Configuring Your Cluster

For the Hive View to access HDFS, the Ambari Server daemon hosting the view needs to act as the proxy user for HDFS. This allows Ambari to submit requests to HDFS on behalf of the users using the Hive View. This is critical since the Hive View will store metadata about their user Hive queries in HDFS. This also means users that will access the Hive View must have a user directory setup in HDFS.

- Setup HDFS Proxy User
- Setup HDFS User Directory

## 8.1.1. Setup HDFS Proxy User

To set up an HDFS proxy user for the Ambari Server daemon account, you need to configure the proxy user in the HDFS configuration. This configuration is determined by the account name the **ambari-server** daemon is running as. For example, if your ambari-server is running as **root**, you set up an HDFS proxy user for **root** with the following:

1. In Ambari Web, browse to **Services > HDFS > Configs**.

- 2. Under the Advanced tab, navigate to the Custom core-site section.
- 3. Click Add Property... to add the following custom properties:

hadoop.proxyuser.root.groups=\* hadoop.proxyuser.root.hosts=\*

Notice the **ambari-server** daemon account name root is part of the property name. Be sure to modify this property name for the account name you are running the ambari-server as. For example, if you were running **ambari-server** daemon under an account name of **ambariusr**, you would use the following properties instead:

```
hadoop.proxyuser.ambariusr.groups=*
hadoop.proxyuser.ambariusr.hosts=*
```

Similarly, if you have configured Ambari Server for Kerberos, be sure to modify this property name for the **primary Kerberos principal** user. For example, if ambari-server is setup for Kerberos using principal **ambari-server@EXAMPLE.COM**, you would use the following properties instead:

```
hadoop.proxyuser.ambari-server.groups=*
hadoop.proxyuser.ambari-server.hosts=*
```

4. Save the configuration change and restart the required components as indicated by Ambari.

## 8.1.2. Setup HDFS User Directory

The Hive View stores user metadata in HDFS. By default, the location in HDFS for this metadata is /user/\${username} where \${username} is the username of the currently logged in user that is accessing the Hive View.



#### Important

Since many users leverage the default Ambari admin user for getting started with Ambari, the /user/admin folder needs to be created in HDFS. Therefore, be sure to create the admin user directory in HDFS using these instructions prior to using the view.

To create user directories in HDFS, do the following for each user you plan to have use the Hive View.

- 1. Connect to a host in the cluster that includes the HDFS client.
- 2. Switch to the hdfs system account user.

su - hdfs

3. Using the HDFS client, make an HDFS directory for the user. For example, if your username is admin, you would create the following directory.

hadoop fs -mkdir /user/admin

4. Set the ownership on the newly created directory. For example, if your username is admin, you would make that user the directory owner.

hadoop fs -chown admin:hadoop /user/admin

# **8.2. Creating the Hive View Instance**

- 1. Browse to the Ambari Administration interface.
- 2. Click Views, expand the Hive View, and click Create Instance.
- 3. On the Create Instance page, select the **Version**. If multiple Hive View jars are present, choose one.
- 4. Enter the following view instance Details:

#### **Table 8.1. Hive View Instance Details**

Property	Description	Example Value
Instance Name	This is the Hive view instance name. This value should be unique for all Hive view instances you create. This value cannot contain spaces and is required.	HIVE_1
Display Name	This is the name of the view link displayed to the user in Ambari Web.	Hive
Description	This is the description of the view displayed to the user in Ambari Web.	Author and execute Hive queries.
Visible	This checkbox determines whether the view is displayed to users in Ambari Web.	Visible or Not Visible

- 5. The **Settings** and **Cluster Configuration** options depend on a few cluster and deployment factors in your environment. You can typically leave the default **Settings** unless you are using the Hive View with a Kerberos enabled cluster. Refer to Settings and Cluster Configuration for more information.
- 6. Click Save.

## **8.2.1. Settings and Cluster Configuration**

If you have manually deployed your cluster, you must enter cluster configuration values in the Hive View Create Instance page. The following table explains where you can find cluster configuration settings in Ambari.

#### Table 8.2. Finding Cluster Configuration Values for the Hive View in Ambari

Property	Value
Hive Authentication	auth=NONE;user=\${username}
Scripts HDFS Directory*	/user/\${username}/hive/scripts
Jobs HDFS Directory*	/user/\${username}/hive/jobs
HiveServer2 Host*	Click Hive > Summary > HiveServer2 to view the host name. For example, c6401.ambari.apache.org
HiveServer2 Thrift port*	Click Hive > Configs > Advanced > General > HiveServer2 Port. For example, 10000

Property	Value
WebHDFS FileSystem URI*	Click HDFS > Configs > Advanced > Advanced hdfs-site > dfs.namenode.http-address. When you enter the value in the view definition, pre-pend "webhdfs://" to the value you find in the advanced HDFS configuration settings. For example, webhdfs:// c6401.ambari.apache.org:50070
YARN Application Timeline Server URL*	Click YARN > Configs > Advanced > Application Timeline Server > yarn.timeline-service.webapp.address. When you enter the value in the view definition, pre- pend "http://" to the value you find in the YARN advanced configuration settings. For example, http:// c6401.ambari.apache.org:8188
YARN ResourceManager URL*	Click YARN > Configs > Advanced > Advanced yarn- site > yarn.resourcemanager.webapp.address. When you enter the value in the view definition, pre- pend "http://" to the value you find in the YARN advanced configuration settings. For example, http:// c6401.ambari.apache.org:8088

#### For NameNode High Availability

The following values must be entered for primary and secondary NameNodes:

#### Table 8.3. Hive View Settings for NameNode High Availability

Property	Value
First NameNode RPC Address or Second NameNode RPC Address	Select the primary or secondary NameNode to view settings from that host in the cluster. See how to get the NameNode RPC address [46]. When you enter the value in the view definition, pre-pend "http://" to the value you find in the advanced hdfs-site settings. For example, http://c6401.ambari.apache.org:8020
First NameNode HTTP (WebHDFS) Address or Second NameNode HTTP (WebHDFS) Address	Click HDFS > Configs > Advanced > Advanced hdfs-site > dfs.namenode.http-address. When you enter the value in the view definition, pre-pend "http://" to the value you find in the advanced hdfs-site settings. For example, http://c6401.ambari.apache.org:50070

#### To get First NameNode RPC Address values:

 Navigate to the HDFS service page in Ambari that contains links to individual NameNodes. Click NameNode (primary) or SNameNode (secondary) to view the host page:

#### Figure 8.1. HDFS Service Page in Ambari



- 2. On the host page, click **Configs > Advanced**.
- 3. Enter "rpc" in the Filter search well at the top right corner of the page or navigate to the Advanced hdfs-site settings to find the dfs.namenode.rpc-address value that you

can enter into the Hive View definition. Here is an example of using the Filter to locate a value:

Figure 8.2. Using the Filter to Search Advanced hdfs-site Settings

Amb	ari MyC	lutter (Rope Raints		Dashboard	Services	Hosts	Alerts	Admin	ш	▲ ambari-qa •
o c6401.a Back	mbari.ap	ache.org								
Summary	Configs	Alerts 🖸 Versions								Host Actions •
HDFS MapReduc	w2	Group HDFS Default (1)	Change					po		0 •
YARN Tez		Settings Advanced								
Hive Pig		* Advanced hofs-site								
ZooKeepe		dfs.namenode.rpc- address	c6401.ambari.apa	che.org:8020						

## **8.2.2. User Permissions for Hive Views**

After saving the Hive View instance definition, grant permission on the view for the set of users who can use the view:

**Figure 8.3. Granting User Permissions to Hive Views** 

yw.	HIVE			
rsion	1.0.0			
Details				🖌 Edit
	Instance Name	Hive		
	Display Name	Hive		
	Description	Hive		
Permission	s			
Permission	Grant pe	rmission to these users	Grant permission to these groups	
Use		ri-ga		

## **8.2.3. Kerberos Setup for Hive Views**

To set up basic Kerberos for views, see "Set Up Kerberos for Ambari Server" in the Ambari Security Guide. After you have set up basic Kerberos for the Hive View, Hive requires the following additional settings:

#### Table 8.4. Kerberos Settings for Hive Views

Property	Value
WebHDFS Authentication	auth=KERBEROS;proxyuser= <ambari-principal></ambari-principal>
Hive Authentication	KERBEROS and the principal is set to the same principal that is specified in hive-site.xml for hive.server2.authentication.kerberos.principa . For example, see the following figure.

Figure 8.4. Hive View Kerberos Configuration Example

Settings	/ E
Hive Authentication	auth=KERBEROS;principal=hive/erik-bb2-2.c.pramod-thangal.internal@EXAMPLE.COM
WebHDFS Username	\${username}
WebHDFS Authentication	auth=KERBEROS;proxyuser=ambariuser
Instance name of Tez view	TEZ_CLUSTER_INSTANCE
Scripts HDFS Directory*	/user/\$(username)/hive/scripts
Jobs HDFS Directory*	/user/\$(usemame)/hive/jobs
Default script settings file*	/user/\$(username)/.\$(instanceName).defaultSettings
Charles Configuration	
Cluster Configuration	
<ul> <li>Local Ambari Managed Clu Cluster Name</li> </ul>	views a
Guster Name	views ÷
· Custom	
HiveServer2 Host*	erik-bb2-2.c.pramod-thangail.internal
HiveServer2 Thrift port*	10000
WebHDFS FileSystem URI*	webhdfs://erik-bb2-1.c.pramod-thangaii.internal:50070
Logical name of the NameNode cluster	
List of NameNodes	
First NameNode RPC Address	
Second NameNode RPC	
Address	
First NameNode HTTP (WebHDFS) Address	
Second NameNode HTTP (WebHDFS) Address	
Failover Proxy Provider	
YARN Application Timeline Server URL*	http://erik-bb2-2.c.pramod-thangail.internal.8188
YARN ResourceManager	http://erik-bb2-2.c.pramod-thangali.internal:8088
URL*	

# 8.3. Using the Hive View

Use the Hive View to:

- Browse databases
- Write and execute queries
- Manage query execution jobs and history

## 8.3.1. Query Tab

Click the **Query** tab to browse database tables and columns and to build, execute, and debug queries.

#### **Database Explorer**

The Database Explorer enables you to view all databases and tables in Hive that you have permissions to view. It is designed to navigate a large number of databases, tables, and columns:

#### Figure 8.5. Hive View Database Explorer

Database Explorer	c
consumption	•
Search tables	
Databases	
consumption	
III power	
III power2	
adate	STRING
atime	STRING
global_active_power	DOUBLE
voltage	DOUBLE
global_intensity	DOUBLE
sub_metering_1	DOUBLE
sub_metering_2	DOUBLE
sub_metering_3	DOUBLE
III power3	
III power4	
🛢 default	

#### Features of Database Explorer:

- Click the refresh icon in the top right to view tables that were created since the Hive View session began.
- Select a database from the drop-down list. All queries in the current tab are then run against the selected database. You can also edit the drop-down list to enable substring searches over a large number of databases.
- Use the Search tables and Search columns fields to search when you have a large number of tables and columns.
- Browse the Databases tab to view all of the databases, tables, and columns. This is useful when you are authoring queries. The icon to the right of a table enables you to see sample data within that table.

#### **Query Editor**

You can author and execute queries in the Query Editor:

#### **Figure 8.6. Query Editor**

Ambari MyClust	<b>Gueries</b> He		•
Database Explorer	σ	Query Editor	
consumption	•	LargePowerQuery	0
Search tables. Databases Contumption power power2 edate global_intensity sub_metering_1	STRING STRING DOUBLE DOUBLE DOUBLE DOUBLE	1 Loart into table power Fridmin safet and and a safety jower) 2 Sin power g2 4 Sin power g2 4 Sin power g2 5 Troop by p-safety	0 0 12 10 10 10 10 10 10 10 10 10 10 10 10 10
sub_metering_2 sub_metering_3 power3 power4 default	DOUBLE	Eastern Espain Serve Ma.	

Features and Behavior of Query Editor

- All queries contained in a Worksheet tab execute sequentially, and they run in the same session. Running all queries in one pass requires handling the output of multiple select statements and is not supported in the 1.0 version.
- To run a specific query, highlight it, and click **Execute**.
- When the first query is executed in a Worksheet, a Tez session is opened.
- Click Save as to save your query.
- Double-click the **Worksheet** tab to rename the query, click **OK**, and then **Save as** to save the query with the new name.
- Click **New Worksheet** to open a new worksheet tab. Queries executed from the new worksheet tab will execute in a different session. Queries from different worksheets can execute in parallel.
- Press CTRL + space to autocomplete query statements.
- Click the double arrow icon in the upper right corner of the Query Editor to expand the Worksheet area and cover Database Explorer. Click the icon again to collapse the Worksheet and make Database Explorer available again.
- Click the icon at the bottom of the Worksheet window and drag it down to expand the authoring space.
- Query results and logs display below the query when it is executed.

#### Figure 8.7. Query Results and Logs in Hive View Query Editor

Ambari foo gaga Galera	Dashboard Services Hosts Alerts Admin III 🔺	ambari-qa •
Database Explorer 2	Query Editor	2
consumption	Volume I serv List bis power( I compose p I compose p	Assheet
NU, metering, 2 DOUBLE NU, metering, 3 DOUBLE © powerk © powerk © sample, 0.0 © sample, 0.4 © sample, 0.4	Query Process Reads (Status RUNNING)           Logs         Reads           NO1: Tax season hant to en orested yet. Opening session NO1:         NO1: Status Status Annual Status Stat	

#### **Query Editor Settings**

Click the gear icon on the right margin of the worksheet to access settings for the Query Editor. Then click **Add**, select a setting parameter from the drop-down list, and then select a value for the parameter. Query Editor settings are configured per worksheet.

To save settings as default settings so they are applied each time that a new worksheet is opened, click **Save Default Settings** in the upper right corner of the settings window.

Click **SQL** to the right of the Worksheet window to exit settings and return to the Query Editor authoring pane.

Text Explain and Visual Explain

There are two options that help you understand how your queries are executed. One is a textual explanation of your query and the other form explains the query visually as a diagram. In future releases, column lineage will be added.

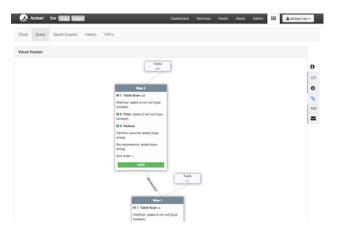
The **Explain** button in the lower left corner of the Worksheet window launches a textual explanation:

#### Figure 8.8. Query Editor Textual Explain Feature

=
Explain Save as
Process Results (Status: Succeeded)
ún -
GE DEPENDENCIES:
AGE PLANS:
ge: Stage-1
liz.
Edges:
Map 1 <- Map 3 (BROADCAST_EDGE)
Reducer 2 <- Map 1 (SIMPLE_EDGE)
DagName: hive_20150614220150_93cfd706-65b8-4baa-8442-9c7126de0856:2
Vertices:
Map 1
Map Operator Tree:
TableScan
alias: p
filterExpr: adate is not null (type: boolean)
Statistics: Num rows: 1231116 Data size: 132960632 Basic stats: COMPLETE Column stats: NONE Filter Operator
predicate: adate is not null (type: boolean)
Statistics: Num rows: 615558 Data size: 66480316 Basic stats: COMPLETE Column stats: NONE Map Join Operator

To launch the Visual Explain diagram, click the link icon to the right of the Worksheet window. If the query is running, Visual Explain shows the query execution progress per vertex:

#### Figure 8.9. Query Editor Visual Explain Feature



Using the Tez View to Debug Query Execution

Query execution can be debugged using the embedded Tez view. To access the Tez view, click **TEZ** in the toolbar on the right of the Worksheet window:

Figure 8.10. Tez View Query Debugging Option

TEZ								
	😤 All DAGs	DAG [ ambari-ga_20150614171835_071eb47	75-8865-41e1-903	e-Sef0c553fe1	3.2]			
		DAG Details	DAG Counters	Graphical View	All Vertices	All Taska	All TaskAttempts	
	0						-	
	0			u	ast refreshed at 14	Jun 2015 12:40	0.20 C Rehwin	
	DAG Details							
	A Download o	data						
	Application Id	application_1434230750579_0006						
	Entity Id	dag_1434230750579_0006_2						
	User	ambari-ga						
	Status	FAILED [Failed Tasks ] [Failed TaskAttempts ]						
	Start Time	14 Jun 2015 10:18:36						
	End Time	14 Jun 2015 10:18:44						
	Duration	8 secs						
	Diagnostics							

When a query fails, the Status field displays **FAILED** and there is a link to Failed Tasks and the error displays on the first page. Click **Download data** to get the data for the task. For further details on debugging, see the Tez View.

#### **Errors and Alerts**

Errors and alerts can be viewed by clicking the envelope icon in the toolbar to the right of the Worksheet window. When the icon is clicked, all the messages are shown with a one-line summary per message:

#### Figure 8.11. Query Editor Error Message Summary Window

Ambari MyCluster (10.998) Bakets	Dashboard	Services	Hosts	Alerts	Admin	ш	▲ ambari-o	a•
Hive Query Saved Queries History UDFs								
Messages								
- Clear All								0
Exception Description: No transaction is currently active							×	<
								0
								٩
								т
								2

If you want to view details of the errors, expand the summary by clicking it. The details text can be copied into a bug report:

#### Figure 8.12. Query Editor Error Message Details Window

<b>\$</b> *	nbari MyCluster Team Editoria Admin Distributer Services Hosts Alarts Admin	III 🔺 ambari-ga	•
Hive	Query Saved Queries History UDFs		
Message			
- Clear Al	1		4
A De	ception Description: No transaction is currently active	×	1
	ava.lang.IllegalStateException:		1
Excep	tion Description: No transaction is currently active		4
	lang.IllegalStateException: ition Description: No transaction is currently active		5
	at org_excluse_persistence_internal.jps.transaction.btilyTransactionEpti.java torg_apote_inderister_interval_ps.transaction.btilyTransactionEpti.java at org_apote_inderister_interval_persistence_Batisforefilmings_inter(Endstechedis)_inder(StransactionEpt).java at org_apote_inderister_interval_persistence_Batisforefilmings_inter(Endstechedis)_inder(StransactionEpt) at org_apote_inderister_inder(StransactionEcd).StransactionEcd) at org_apote_inderister_inder(StransactionEcd).StransactionEcd)		1

### 8.3.2. Saved Queries Tab

The Saved Queries tab shows all the queries that have been saved by the current user. Click the gear icon to the right of the query list to view the history of a query or to delete it:

#### Figure 8.13. Saved Queries Tab

Ambari cn105PerfC	- 10 ops 1 aler	_	Dashboard	Services	Hosts 🚺	Alerts Admin	iii 🔺 admin 🗸
Hive Query Saved Quer	es History UDFs						
preview	1350	•	database	•	owner	•	Clear filters
select dt.d_year .item.i_brand_id	Query3		tpcds_bin_orc_200		admin		0

## 8.3.3. History Tab

You can view the history of all jobs run by the current user in the History tab. It pulls history from the Application Timeline Server database. All queries for which logs are present in that database are displayed here. This means that regardless of the source of the query, (CLI, JDBC/ODBC, Hive View) it will appear here on the History tab. Queries that have not been assigned a name, such as those created in the Hive View, appear as query text. For example, see the insert statement that was submitted by CLI in the following image:

#### Figure 8.14. History Tab

Ambari	foo <b>(Days</b> ) Dater					Dashboard	Services	Hosts	Alerts	Admin	•	ambari-qa •
Hive Query	Saved Queries	History	UDFs									
title	٠	at	atus	•	05/29/2015	05/29	/2015		•		21940	Clear filters
Worksheet		RU	NNING		12 minutes ago				0			
insert into tabl select adate, sum from power p join power2 p2 on p.adate=p2.ada group by p.adate;	n(p.Global_active	_power)										Stop execution

For queries that are submitted from the Hive View, a Stop Execution button is available to enable you to end a currently running query. When you select a query by clicking the title in the first column, that query appears on a new sub-tab in the Query tab where it can be analyzed and debugged.

## 8.3.4. UDF Tab

User-defined functions (UDFs) can be added to queries by pointing to a JAR file on HDFS, which contains the UDF definition. After the UDF is added here, an Insert UDF button appears in the Query Editor that enables you to add the UDF to your query:

#### Figure 8.15. UDF Tab

🔬 Ambari cn105PerfC 🜆 💶		Dashboard Services	Hosts 🚺 Alerts	Admin III 🔺 admin •
Hive Query Saved Queries History	UDFs			
file resource	udi name	udf class name		Citer filters New UDF
Select File Resource	udfiname	udf class name		Cancel Save

# 8.4. Troubleshooting

#### Table 8.5. Troubleshooting Hive Views Errors

Error	Solution
User: root is not allowed to impersonate admin	HDFS has not been configured for Ambari as a proxy user. Refer to Setup HDFS Proxy User.
E090 HDFS020 Could not write file /user/admin/hive/jobs/ hive-job-1-2015-10-30_02-12/query.hql [HdfsApiException]	, · · · · · · · · · · · · · · · · · · ·

# 9. Using the Slider View

Slider is a framework for deploying and managing long-running applications on YARN. When applications are packaged using Slider for YARN, the **Slider View** can be used to help deploy and manage those applications from Ambari.

# 9.1. Deploying the Slider View

Refer to the Ambari Administration guide for general information about Managing Views.

- 1. From the Ambari Administration interface, browse to the Views section.
- 2. Click to expand the Slider view and click Create Instance.
- 3. Enter the instance name, the display name and description.
- 4. Enter the configuration properties for your cluster.

Property	Description	Example
Ambari Server URL (required)	The Ambari REST URL to the cluster resource.	http://ambari.server:8080/api/v1/ clusters/MyCluster
Ambari Server Username (required)	The username to connect to Ambari. Must be an Ambari Admin user.	admin
Ambari Server Password (required)	The password for the Ambari user.	password
Slider User	The user to deploy slider applications as. By default, the applications will be deployed as the "yarn" service account user. To use the current logged-in Ambari user, enter \${username}.	joe.user or \${username}
Kerberos Principal	The Kerberos principal for Ambari views. This principal identifies the process in which the view runs. Only required if your cluster is configured for Kerberos. Be sure to configure the view principal as a proxy user in core-site.	view-principal@EXAMPLE.CO
Kerberos Keytab	The Kerberos keytab for Ambari views. Only required if your cluster is configured for Kerberos.	/path/to/keytab/view- principal.headless.keytab

5. Save the view.

# **10. Using the Files View**

The **Files View** provides a convenient way to access HDFS through a web-based interface. This document provides information on how to configure a view instance and your cluster for browsing HDFS via the **Files View**.

- Configuring Your Cluster
- Creating and Configuring a Files View Instance
- Troubleshooting



#### Important

It is critical that you prepare your Ambari Server for hosting views. It is strongly recommended you increase the amount of memory available to your Ambari Server, and that you run additional "standalone" Ambari Servers to host the views. See Preparing Ambari Server for Views and Running Ambari Server Standalone for more information.

# **10.1. Configuring Your Cluster**

For the Files View to access HDFS, the Ambari Server daemon hosting the view needs to act as the proxy user for HDFS. This allows Ambari to submit requests to HDFS on behalf of the users using the Files View.

To set up an HDFS proxy user for the Ambari Server daemon account, you need to configure the proxy user in the HDFS configuration. This configuration is determined by the account name the **ambari-server** daemon is running as. For example, if your ambari-server is running as **root**, you set up an HDFS proxy user for **root** with the following:

- 1. In Ambari Web, browse to **Services > HDFS > Configs**.
- 2. Under the Advanced tab, navigate to the Custom core-site section.
- 3. Click Add Property... to add the following custom properties:

```
hadoop.proxyuser.root.groups=*
hadoop.proxyuser.root.hosts=*
```

Notice the **ambari-server** daemon account name **root** is part of the property name. Be sure to modify this property name for the account name you are running the ambari-server as. For example, if you were running **ambari-server** daemon under an account name of **ambariusr**, you would use the following properties instead:

```
hadoop.proxyuser.ambariusr.groups=*
hadoop.proxyuser.ambariusr.hosts=*
```

Similarly, if you have configured Ambari Server for Kerberos, be sure to modify this property name for the primary Kerberos principal user. For example, if ambari-server is setup for Kerberos using principal **ambari-server@EXAMPLE.COM**, you would use the following properties instead:

```
hadoop.proxyuser.ambari-server.groups=*
hadoop.proxyuser.ambari-server.hosts=*
```

4. Save the configuration change and restart the required components as indicated by Ambari.

# **10.2. Creating and Configuring a Files View Instance**

- 1. Browse to the Ambari Administration interface.
- 2. Click Views, expand the Files View, and click Create Instance.
- 3. Enter the following View instance Details:

Property	Description	Value
Instance Name	This is the Files view instance name. This value should be unique for all Files view instances you create. This value cannot contain spaces and is required.	FILES_1
Display Name	This is the name of the view link displayed to the user in Ambari Web.	MyFiles
Description	This is the description of the view displayed to the user in Ambari Web.	Browse HDFS files and directories.
Visible	This checkbox determines whether the view is displayed to users in Ambari Web.	Visible or Not Visible

- 4. The **Settings** and **Cluster Configuration** options depend on a few cluster & deployment factors in your environment:
  - Is your cluster Kerberos-enabled?
  - Is NameNode HA configured?
  - Is your **Files View** instance being configured in an **Operational** Ambari Server or a **Standalone** Ambari Server?

Refer to the following table on the instructions to complete the **Files View** configuration:

Kerberos Enabled	NameNode HA Enabled	Operational Ambari Server	Standalone Ambari Server
		see note #1:	see note #2:
No	No	Settings: defaults	Settings: defaults
No	Yes	Cluster Configuration: Local	Cluster Configuration: Custom
Yes	No	Settings : Kerberos	
		Cluster Configuration : Custom	
Yes	Yes	Settings: Kerberos	
		Cluster Configuration: Custom	



#### Note

#1: The Local Ambari Managed Cluster Configuration option is enabled in the Ambari Administration Interface only if you are managing a cluster in an Operational Ambari Server.



#### Note

#2: See Running Ambari Standalone for more information.

# **10.2.1. Kerberos Settings**

You must first set up Kerberos for Ambari by configuring the Ambari Server daemon with a Kerberos principal and keytab. Refer to Configuring Views for Kerberos for instructions. After you have set up Kerberos for Ambari, in the Settings section of the Files View, enter the following:

Property	Description	Example Value
WebHDFS Username	This is the username the view will access HDFS as. Leave this default value intact to represent the authenticated view user.	\${username}
WebHDFS Authorization	This is the semicolon-separated authentication configuration for WebHDFS access.	auth=KERBEROS; proxyuser=ambari-server



#### Note

With a Kerberos setup, the proxy user setting should be the primary value of the Kerberos principal for Ambari Server. For example, if you configured Ambari Server for Kerberos principal **ambari-server@EXAMPLE.COM**, this value would be **ambari-server**. Refer to Configuring Views for Kerberos for more information..

## **10.2.2. Cluster Configuration: Local**

The **Local Ambari Managed Cluster Configuration** option is enabled in the Ambari Administration Interface if you are managing a cluster with Ambari. When enabled, you can choose this option and Ambari will automatically configure the view based on how the cluster is configured.

When you configure the view using the Local option, the Files View will communicate with HDFS based on the **fs.defaultFS** property (for example: hdfs://namenode:8020). The View will also determine if NameNode HA is configured and adjust accordingly.

# 10.2.3. Cluster Configuration: Custom

These properties are required if using Custom configuration.

<b>Required Properties</b>	Description	Example Value
WebHDFS FileSystem URI	The WebHDFS FileSystem URI in the format webhdfs:// <host>:<http_port></http_port></host>	webhdfs://namenode:50070

Property	Description	Example Value
Logical name of the NameNode cluster	Comma-separated list of nameservices.	hdfs-site/dfs.nameservices
		For example:
		nameservice
List of NameNodes	Comma-separated list of NameNodes for a given nameservice.	hdfs-site/dfs.ha.namenodes
	a given nameservice.	For example:
		namenode1, namenode2
First NameNode RPC Address	RPC address for first name node.	hdfs-site/dfs.namenode.rpc-address. [nameservice].[namenode1]
Second NameNode RPC Address	RPC address for second NameNode.	hdfs-site/dfs.namenode.rpc-address. [nameservice].[namenode2]
First NameNode HTTP (WebHDFS) Address	WebHDFS address for first NameNode.	hdfs-site/dfs.namenode.http-address. [nameservice].[namenode1]
Second NameNode HTTP (WebHDFS) Address	WebHDFS address for second NameNode.	hdfs-site/dfs.namenode.http-address. [nameservice].[namenode2]
Failover Proxy Provider	The Java class that HDFS clients use to contact the Active NameNode.	hdfs-site/ dfs.client.failover.proxy.provider. [nameservice]

These properties are required if your cluster is configured for NameNode HA.

# 10.2.4. Troubleshooting

Error	Solution
<b>500</b> Usernames not matched: name=root != expected=ambari- server	If your cluster is configured for Kerberos, double-check WebHDFS Authorization setting and confirm the "proxyuser=" part of the string is set to the Ambari Server principal name. For example:
	auth=KERBEROS;proxyuser=ambari-server
	Refer to Kerberos Settings.
<b>500</b> User: ambari-server is not allowed to impersonate admin	HDFS has not been configured for Ambari as a proxy user. Refer to Configuring Your Cluster.
<b>500</b> SIMPLE authentication is not enabled. Available:[TOKEN, KERBEROS]	If your cluster is configured for Kerberos, you cannot use the Local Cluster Configuration option. You must use the Custom Cluster Configuration option and enter the WebHDFS FileSystem URI.
	For example:
	webhdfs://namenode:50070
	Refer to Cluster Configuration: Custom