Pivotal™

Pivotal Command Center 2.0 Installation and User Guide

Rev: A03

Use of Open Source

This product may be distributed with open source code, licensed to you in accordance with the applicable open source license. If you would like a copy of any such source code, EMC will provide a copy of the source code that is required to be made available in accordance with the applicable open source license. EMC may charge reasonable shipping and handling charges for such distribution. Please direct requests in writing to EMC Legal, 176 South St., Hopkinton, MA 01748, ATTN: Open Source Program Office.

EMC believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

THE INFORMATION IN THIS PUBLICATION IS PROVIDED "AS IS." EMC CORPORATION MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WITH RESPECT TO THE INFORMATION IN THIS PUBLICATION, AND SPECIFICALLY DISCLAIMS IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Date: 7/1/13

Table of Contents

Preface

This preface includes four sections:

- About Pivotal, Inc.
- About this Guide
- Document Conventions

About Pivotal, Inc.

Greenplum is currently transitioning to a new corporate identity (Pivotal, Inc.). We estimate that this transition will be completed in 2013. During this transition, there will be some legacy instances of our former corporate identity (Greenplum) appearing in our products and documentation. If you have any questions or concerns, please do not hesitate to contact us through our web site, `http://gopivotal.com/about-pivotal/support`.

About this Guide

This guide includes the following topics:

- Chapter 1, "Overview"— An overview of the functionality and architecture of Pivotal Command Center.
- Chapter 2, "Installing Pivotal Command Center 2.0.x" — Basic instructions for installing and configuring Pivotal Command Center to monitor a running instance of Pivotal HD.
- Chapter 3, "Using Pivotal Command Center UI"— An overview of the functionality of the Pivotal Command Center console (user interface).
- Chapter 4, "Pivotal Command Center Performance Monitor"— An overview of the Pivotal Command Center performance monitor known as `nmon`.
- Appendix A, "Creating a YUM EPEL Repository"— Instructions how to setup a local yum repository or point your hosts to an EPEL repository.
- Appendix B, "Best Practises -Troubleshooting".

Document Conventions

The following conventions are used throughout the Pivotal Command Center documentation to help you identify certain types of information.

- Text Conventions
- Command Syntax Conventions

Text Conventions

Table 0.1 Text Conventions

Text Convention	Usage	Examples
italics	New terms where they are defined Database objects, such as schema, table, or columns names	The *master instance* is the `postgres` process that accepts client connections. Catalog information for Pivotal Command Center resides in the *pg_catalog* schema.
`monospace`	File names and path names Programs and executables Command names and syntax Parameter names	Edit the `postgresql.conf` file. Use `gpstart` to start Pivotal Command Center.

`<monospace italics>`	Variable information within file paths and file names
	Variable information within command syntax
		`/home/gpadmin/`*`<config_file>`*
		`COPY` *`tablename`* `FROM '`*`<filename>`*`'`

`monospace bold`	Used to call attention to a particular part of a command, parameter, or code snippet.	Change the host name, port, and database name in the JDBC connection URL: `jdbc:postgresql://`**`host:5432`**`/`**`mydb`**
`UPPERCASE`	Environment variables SQL commands Keyboard keys	Make sure that the Java `/bin` directory is in your `$PATH`. `SELECT * FROM` *`my_table`*`;` Press `CTRL+C` to escape.

Command Syntax Conventions

Table 0.2 Command Syntax Conventions

Text Convention	Usage	Examples
`{ }`	Within command syntax, curly braces group related command options. Do not type the curly braces.	`FROM` **`{`** `'`*`filename`*`' | STDIN` **`}`**
`[ ]`	Within command syntax, square brackets denote optional arguments. Do not type the brackets.	`TRUNCATE` **`[`** `TABLE` **`]`** *`name`*

Table 0.2 Command Syntax Conventions

Text Convention	Usage	Examples
`...`	Within command syntax, an ellipsis denotes repetition of a command, variable, or option. Do not type the ellipsis.	`DROP TABLE` *name* `[, ...]`
`|`	Within command syntax, the pipe symbol denotes an “OR” relationship. Do not type the pipe symbol.	`VACUUM [ FULL | FREEZE ]`

1. Overview

This document provides an overview of Pivotal Command Center 2.0.x and instructions for installing the product. This chapter provides an overview of Pivotal Command Center, then briefly describes each component.

Pivotal Command Center Overview

The Pivotal Command Center allows an administrative user to administer and monitor one or more Pivotal HD clusters. The Command Center has command-line tools to deploy and configure Pivotal HD clusters, as well as, an intuitive graphical user interface that is designed to help the user view the status of the clusters and take appropriate action. This release of Command Center allows only administering and monitoring of Pivotal HD Enterprise 1.0.x clusters.

Pivotal Command Center 2.0.x is comprised of the following:

Pivotal Command Center UI

The Pivotal Command Center UI provides the user with a single web-based graphical interface to monitor and manage one or more Pivotal HD clusters. This web application is hosted on a Ruby-on-Rails application which presents the status and metrics of the clusters. This data comes from multiple sources. All of the Hadoop specific data comes from the Pivotal HD Manager component. The system metrics data is gathered by our Performance Monitor (nmon) component.

See Chapter 3, “Using Pivotal Command Center UI” for more information.

Pivotal HD Manager

Pivotal HD Manager provides complete life cycle management for Pivotal HD Clusters. It performs the following two main groups of functions:

- Cluster installation, configuration and uninstalls
- Cluster monitoring and management

These functions are served through a set of RESTful web services that run as a web application on an Apache-Tomcat server on the Command Center admin host. This is called `gphdmgr-webservices`. This web application stores its metadata and cluster configuration for Pivotal HD cluster nodes and services in the Pivotal Command

Center PostgreSQL database. It makes use of a Puppet Server to perform most of its HD cluster installation and configuration. It also has a polling service that retrieves Hadoop metrics from the cluster and stores them in the Command Center PostgreSQL Database at periodic intervals.

Pivotal HD Manager provides a command-line interface (CLI) for installation, configuration and uninstalls. This tool invokes the `gphdmgr-webservice` APIs to install and configure the various Pivotal HD services. The CLI also provides a way to start and stop the clusters. For how to use this CLI, please refer to the *Pivotal HD Enterprise 1.0 Installation and Administrator Guide.*

The Command Center UI also invokes the `gphdmgr-webservice` APIs to retrieve all Hadoop-specific cluster metrics and status information. This includes the Hadoop metrics that was previously retrieved by the polling service.

Performance Monitor (nmon)

Pivotal Command Center comes with a Performance Monitor called `nmon` (for node monitor). This makes use of a highly scalable message passing architecture to gather performance metrics from each node that Command Center monitors. This consists of a `nmon` master daemon that runs on the Command Center admin host and an `nmon` daemon that runs on all the cluster nodes that report system metric information to the `nmon` master. This includes metrics such as CPU, memory, disk I/O and network usage information.

The `nmon` master on the admin host dumps the system metrics it receives from the `nmon` agents on the cluster nodes into a PostgreSQL DB. This is then queried by the Command Center UI application to display its cluster analysis graphs.

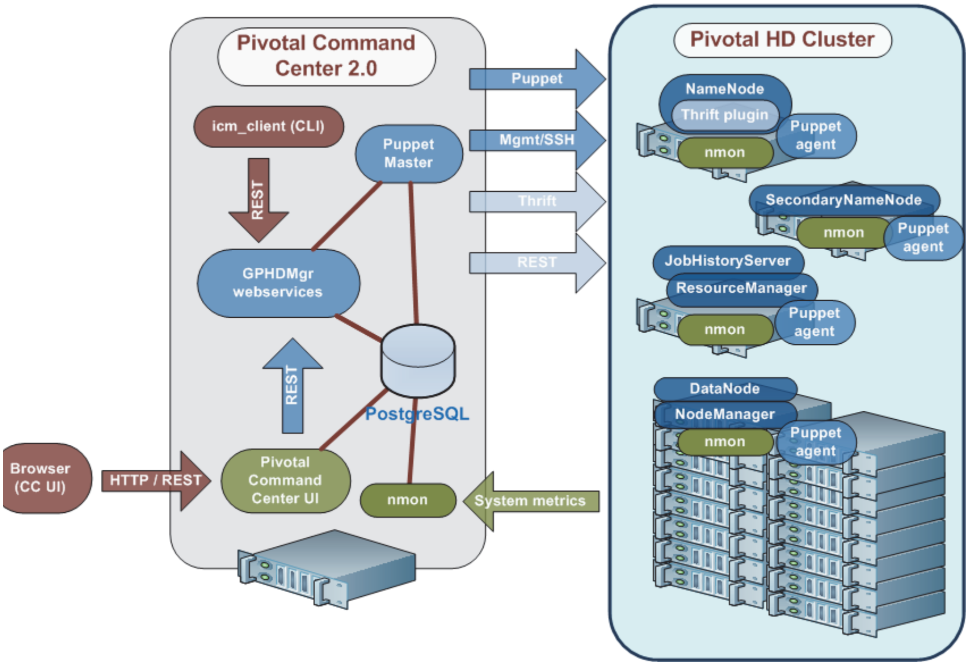
PostgreSQL Database

Pivotal Command Center makes use of a PostgreSQL Database to store the following:

- Cluster configurations
- Hadoop cluster metrics
- System metrics of the cluster
- Pivotal Command Center Metadata

Architectural Overview

For more details about Pivotal HD Enterprise 1.0.x, refer to the *Pivotal HD 1.0 Installation and Administrator Guide*.



2. Installing Pivotal Command Center 2.0.x

This section describes how to install and configure Pivotal HD 2.0.x using the Pivotal Command Center Unified Installer.

This chapter includes the following sections:

Supported Platforms

- RHEL 6.1 64-bit, 6.2 64-bit
- CentOS 6.1 64-bit, 6.2 64-bit

Product Downloads

The following packages are required:

- `PCC-2.0.x.*.version_build_OS.x86_64.tar.gz`

Prerequisites

- Oracle JDK 1.6 installed on the Admin host.
- See Package Accessibility, below, for prerequisite packages.

Installation of Pivotal HD Manager assumes the user has a working knowledge of the following:

- Yum. Yum enables you to install or update software from the command line. See `http://yum.baseurl.org/`
- RPM (Redhat Package Manager).
- SSH (Secure Shell protocol).

Package Accessibility

Pivotal Command Center and Pivotal HD Enterprise expect some prerequisite packages to be pre-installed on each host, depending on the software that gets deployed on a particular host. In order to have a smoother installation it is recommended that each host would have yum access to an EPEL yum repository. If you have access to the Internet, then you can configure your hosts to have access to the external EPEL repositories. However, if your hosts do not have Internet access (or you are deploying onto a large cluster), then having a local yum EPEL repo is highly recommended. This will also give you some control on the package versions you want deployed on your cluster. See Appendix A, "Creating a YUM EPEL Repository" for instructions on how to setup a local yum repository or point your hosts to an EPEL repository.

For Pivotal Command Center 2.0.x, here is a list of pre-requisites that need to either already be installed on the Command Center admin host or on an accessible yum repository:

- httpd
- mod_ssl
- postgresql
- postgresql-devel
- postgresql-server
- compat-readline5
- createrepo
- sigar

You can run the following command on the admin node to make sure that you are able to install the prerequisite packages during installation.

```
$ sudo yum list httpd mod_ssl postgresql postgresql-devel
postgresql-server compat-readline5 createrepo sigar
```

If any of them are not available or not already installed, then you may have not added the repository correctly to your admin host.

For the cluster hosts (where you plan to install the cluster), the prerequisite packages depend on the software you will eventually install there, but you may want to verify that the following two packages are installed or accessible by yum on all hosts:

- nc
- postgresql-devel

System Checks

Important: Avoid using hostnames with capital letters in them because Puppet has an issue generating certificates for domains with capital letters.

- Check that SELinux is disabled by running the following command:

  ```
  # sestatus
  ```

  Accepted return values are:

  ```
  SELinuxstatus: disabled
  ```

  or:

  ```
  SELinux status: permissive
  ```

  If SELinux is `enabled`, you can temporarily disable it or make it permissive (this meets requirements for installation) by running the following command:

  ```
  # echo 0 >/selinux/enforce
  ```

  This only temporarily disables SELinux; once the host is rebooted, SE Linux will be re-enabled therefore we recommend disabling SELinux, described below, while running Pivotal HD/HAWQ.

  Note: You can permanently disable SE Linux by editing the `/etc/selinux/config` file as follows, however disabling SELinux like this requires a system reboot for the change to take effect.

  Change the value for the `SELINUX` parameter to:

  ```
  SELINUX=disabled
  ```

  Reboot the system.

- Every cluster node must be able to perform a forward and reverse DNS look-up for every other node.

- Verify that iptables is turned off, for example:

  ```
  # chkconfig iptables off
  # service iptables stop
  # service iptables status
  iptables: Firewall is not running.
  ```

Installation Instructions

Once you have met the prerequisites, you are ready to begin the installation. Perform the following installation steps as a root user.

Upgrade Note: If you are upgrading from Pivotal Command Center 2.0 to version 2.0.1, you must first stop the earlier version by running the following command:

```
$ service commander stop
```

1. Copy the Command Center tar file to your host. For example:

   ```
   # scp ./PCC-2.0.x.version.build.os.x86_64.tar.gz
   host:/root/phd/
   ```

2. Log into the Command Center admin host as root user. `cd` to the directory where the Command Center tar files are located and untar. For example:

   ```
   # cd /root/phd
   # tar --no-same-owner -zxvf
   PCC-2.0.x.version.build.os.x86_64.tar.gz
   ```

3. Still as root user, run the installation script. This installs the required packages and configures both Pivotal Command Center and Pivotal HD Manager, and starts services.

   Important: You must run the installation script from the directory where it is installed, for example: `PCC-2.0.x.version`

   For example:

   ```
   # ls
   PCC-2.0.x.version
   PCC-2.0.x.version.build.os.x86_64.tar.gz
   # cd PCC-version
   # ./install
   ```

   You will see installation progress information on the screen. Once the installation successfully completes, you will see the following:

   ```
   You have successfully installed PCC 2.0.x
   You now need to install a GPHD cluster to monitor or sync
   PCC to monitor an existing GPHD cluster. You can view your
   cluster statuses here:
   http ://node0781.ic.analyticsworkbench.com:5000/status
   ```

4. Verify that your PCC instance is running by executing the following command:

   ```
   $ service commander status
   ```

Starting, Stopping, and Restarting Command Center Services

To stop or restart Command Center services, run the following commands on the Pivotal Command Center admin host:

```
$ service commander stop
$ service commander start
$ service commander restart
```

Launching Command Center

Launch a browser and navigate to the host on which you installed Command Center. For example:

```
http: //CommandCenterHost:5000
```

The Command Center login page is launched in your browser. The default username/password is `gpadmin/gpadmin`.

See Chapter 3, “Using Pivotal Command Center UI” for more details about the application, including how to change the default password.

Next Steps

See the *Pivotal HD 1.0 Enterprise Installation and AdministratorGuide* for instructions for using the command-line interface of Pivotal Command Center to deploy and configure a HD cluster.

Uninstalling Pivotal Command Center

Follow the steps below to uninstall Pivotal Command Center and the Pivotal HD cluster:

1. Stop services on all your clusters (See the *Pivotal HD Enterprise 1.0 Installation and Administrator Guide* for detailed steps).
2. Uninstall all your clusters (See the *Pivotal HD Enterprise 1.0 Installation and Administrator Guide* for detailed steps).
3. From the directory where you untarred the Pivotal Command Center, run the uninstall script:

   ```
   # cd /root/phd/PCC-2.0.x.version/
   # ./uninstall
   ```

Upgrading Pivotal Command Center

Follow the steps below to upgrade your Pivotal CC to a newer version:

Note: We recommend backing up your data before performing any upgrades.

1. As gpadmin, stop the cluster:

   ```
   # icm_client -l CLUSTERNAME stop
   ```

2. As root, stop the Command Center services:

   ```
   # service commander stop
   ```

3. Make sure that the tomcat server and puppet are not running any more (check for processes as well).
4. Download the new PCC tarball and untar.
5. Run the installer from the new PCC location:

   ```
   # ./install
   ```

Note: In the nmon configuration file (`/etc/nmon/conf/nmon-site.xml`), make sure the port number shown in `<nmon-database-config>` -> `<uri>` is 10432. If not update it to 10432 and restart the nmon service (`service nmon restart`).

Important:

- The upgrade replaces the Tomcat web server with Jetty.
- After this upgrade, the PostgreSQL database will run on port 10432 (pre-upgrade the default port was 5432)

3. Using Pivotal Command Center UI

This section provides an overview of the Pivotal Command Center 2.0 user interface.

Overview

Pivotal Command Center UI is a browser-based application for viewing the status and performance of Pivotal HD clusters. At a high level, the screens consist of:

- Dashboard—Provides an overview of your Pivotal HD cluster. This screen shows at one glance the most important states and metrics that an administrator needs to know about the Pivotal HD cluster.
- Cluster Analysis—Provides detailed information about various metrics of your Pivotal HD cluster. This provides cluster-wide metrics all the way down to host-level metrics. This has hadoop-specific metrics such as MapReduce slot utilization and NameNode performance, as well as system metrics such as CPU, memory, disk and network statistics.
- MapReduce Job Monitor—Provides details about all, or a filtered set of MapReduce jobs.
- YARN App Monitor—Provides details about all, or a filtered set of YARN applications.
- HAWQ Query Monitor—When HAWQ (a revolutionary MPP database on Hadoop solution) is deployed on the cluster, Command Center can show the progress of all actively running queries on HAWQ.

Status indicators

Note that throughout the user interface the following indicators are used to indicate the status of nodes:

- Green: Succeeded
- Blue: Running
- Grey: Stopped/Pending
- Red: Killed/failed

Logging In

The URL to access Pivotal Command Center UI from a browser is

```
http://CommandCenterHost:5000/login
```

To change the default port (5000), update the `port` settings in the following file:

```
/usr/local/greenplum-cc/config/app.yml
```

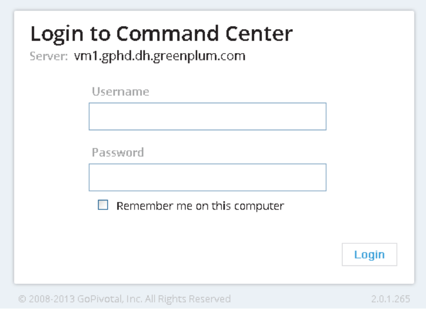
Browser Support

The following browsers are supported by Pivotal Command Center 2.0:

- Firefox 16, 19
- IE 8, IE 9, both with Google Chrome Frame
- Chrome 25.0.1364.172

Login Screen

The first time you launch the Command Center UI, a login screen appears showing the hostname of the host for the Command Center.



Login to Command Center

Server: vm1.gphd.dh.greenplum.com

© 2008-2013 GoPivotal, Inc. All Rights Reserved 2.0.1.265

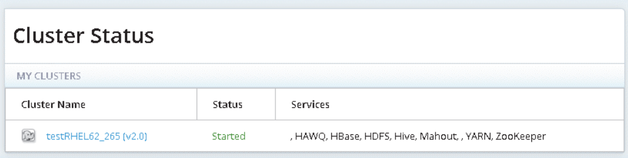
The default admin user/password is `gpadmin/gpadmin`. You can change this password via the Settings Menu.

Click the `Login` button to launch the Command Center UI.

Selecting a Cluster

Once you have launched Command Center, the Cluster Status screen appears, displaying a list of available clusters to monitor, the status of each cluster (**started**, **stopped**), and a list of services running on that cluster (Hive, mahout, and so on).

- Click the cluster name in the table to select a cluster.
- From any point within Command Center UI, you can always select a different cluster by using the **Select Cluster** drop-down menu in the upper right corner of the screen.



Cluster Status

MY CLUSTERS

Cluster Name	Status	Services
testRHEL62_265 (v2.0)	Started	, HAWQ, HBase, HDFS, Hive, Mahout, , YARN, ZooKeeper

Settings Menu

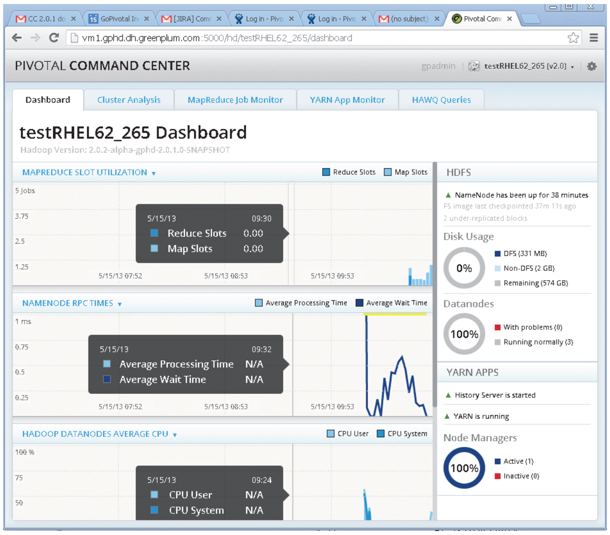
Click the **gear** icon in the upper right corner of the screen at any time to display the **Settings** menu. From the settings menu you can:

- **Cluster Status**. Click this to go back to a list of available clusters.
- **Change Password**. Click this to change your password.
- **Logout**.

Dashboard

The dashboard gives you a high level view of a cluster at a glance. You are able to view the status of the most important cluster services, such as HDFS and YARN. It also shows you how the most important cluster metrics are trending in a visual way.

The graphs provide a unified view of the state of your system. They are also useful in detecting outliers and pinpointing specific problems that may be present in your system.



vm1.gphd.dh.greenplum.com:5000/hd/testRHEL62_265/dashboard

testRHEL62_265 Dashboard

Hadoop Version: 2.0.2-alpha-gphd-2.0.1.0-SNAPSHOT

NameNode has been up for 38 minutes

FS image last checkpointed 37m 11s ago

2 under-replicated blocks

History Server is started

YARN is running

The right side of the Dashboard displays the state of both HDFS and YARN services. It answers the following questions:

- Is HDFS up?
- When did the last NameNode checkpoint occur?
- What percentage of cluster storage is used by HDFS and how much is free?
- How many DataNodes are up and are they running normally or with problems?
- Is YARN up?
- Is the History Serverup?

  Note: The History Server stores a history of the mapreduce jobs run on the cluster.
- How many NodeManagers are up?

The Dashboard provides metrics about:

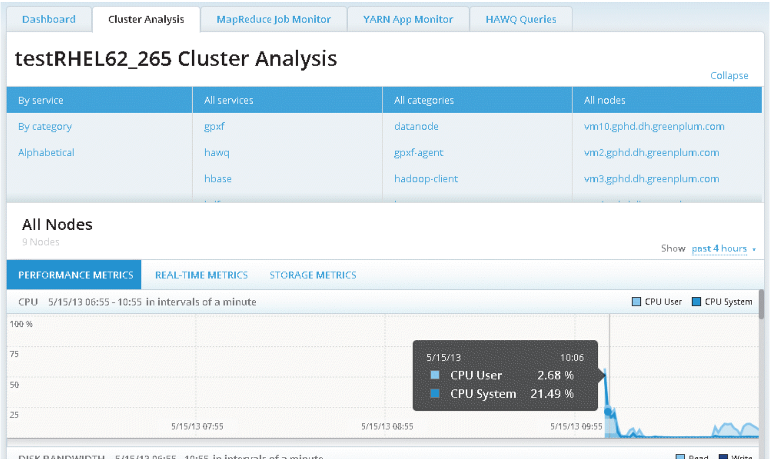
- **Mapreduce Slot Utilization**
- **Namenode RPC Times**
- **Hadoop Datanodes Average CPU**
- **Hadoop Datanodes Average Bandwidth**
- **Namenode Operations Per Second**
- **Hadoop Datanodes Average Disk Bandwidth**
- **Hadoop Datanodes Average Memory**
- **Mapreduce Jobs By Status**

Cluster Analysis

The Cluster Analysis screen provides detailed metrics on your Pivotal HD cluster.

It provides cluster-wide metrics all the way down to host-level metrics. It provides Hadoop-specific metrics, as well as system metrics that you can drill down to if needed.

The Cluster Analysis screen displays the same data that is shown in the dashboard but in greater detail.



testRHEL62_265 Cluster Analysis

By service	All services	All categories	All nodes
By category	gpxf	datanode	vm10.gphd.dh.greenplum.com
Alphabetical	hawq	gpxf-agent	vm2.gphd.dh.greenplum.com
	hbase	hadoop-client	vm3.gphd.dh.greenplum.com

All Nodes
9 Nodes

CPU 5/15/13 06:55 - 10:55 in intervals of a minute

By default the Cluster Analysis screen displays the metrics for all services, all categories, and all nodes. You can filter the information displayed by combinations of the following filters:

- **By Service**
  Metrics can be filtered by services such as HDFS, YARN, or HAWQ.
- **By Category**
  Metrics can be filtered by categories such as:
  - namenode
  - secondarynamenode
  - datanode

  - yarn-resourcemanager
  - yarn-nodemanager
  - mapreduce-historyserver
  - hawq-master
  - hawq-segment
- **Alphabetically**

  Metrics can be filtered alphabetically.

Based on the filters you select, the lower part of the Cluster Analysis screen provides detailed graphs that display data related to:

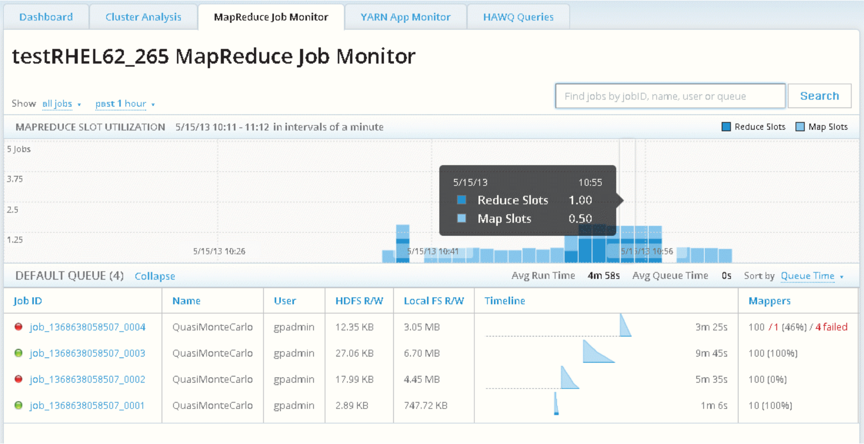
- **Mapreduce Slot Utilization**
- **Namenode RPC Times**
- **Avg Namenode File Operations Per Second**
- **Mapreduce Jobs by Status**
- **Segment CPU**
- **Disk Bandwidth**
- **Network Bandwidth**
- **Segment Memory**
- **Load**
- **Swap Usage**
- **Swap I/O**
- **Network Operations**
- **Disk Operations**

You can view either the **Performance Metrics**, which show the cluster/node utilization over-time, the **Real-time Metrics** which show the current metrics in real-time, or **Storage Metrics**, which show metrics about cluster storage.

If you select Cluster Analysis for **All Nodes** (the default), the Trending Metrics graph for the cluster is displayed:

MapReduce Job Monitor

The Job Monitor screen tracks the MapReduce jobs that are executed in the Pivotal HD cluster when the YARN MapReduce service is running. It provides details about all, or a filtered set of MapReduce jobs.



testRHEL62_265 MapReduce Job Monitor

MAPREDUCE SLOT UTILIZATION 5/15/13 10:11 - 11:12 in intervals of a minute

DEFAULT QUEUE (4) Avg Run Time 4m 58s Avg Queue Time 0s

Job ID	Name	User	HDFS R/W	Local FS R/W	Timeline	Mappers
● job_1368638058507_0004	QuasiMonteCarlo	gpadmin	12.35 KB	3.05 MB	3m 25s	100 /1 (46%) / 4 failed
● job_1368638058507_0003	QuasiMonteCarlo	gpadmin	27.06 KB	6.70 MB	9m 45s	100 (100%)
● job_1368638058507_0002	QuasiMonteCarlo	gpadmin	17.99 KB	4.45 MB	5m 35s	100 (0%)
● job_1368638058507_0001	QuasiMonteCarlo	gpadmin	2.89 KB	747.72 KB	1m 6s	10 (100%)

The MapReduce jobs displayed can be filtered by state and/or time range.

- **By state**:
  - **all jobs** (set by default)
  - **currently pending jobs**
  - **currently running jobs**
  - **succeeded jobs**
  - **failed jobs**
  - **killed jobs**
- **By time range**:

  By selecting a preset time range in hours, weeks, months, year, or by specifying a custom time range.

The MapReduce jobs can also be filtered by searching for values for the following:

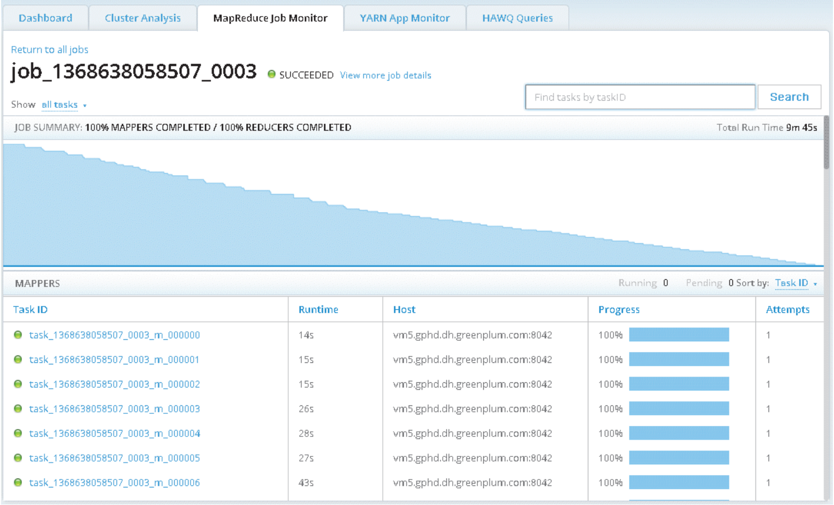
- **jobID**
- **name**
- **user**
- **queue**

Enter your search value in the search bar in the following format: searchKey=searchValue, where searchKey is one of **jobID**, **name**, **user**, or **queue**.

These are substring searches. For example: **jobID=1363920466130** will locate a job with **jobID=job_1363920466130_0002**

Job Details

When you click on any of the jobs in the Job Monitor more details of the job are shown.



job_1368638058507_0003 ● SUCCEEDED

Show all tasks

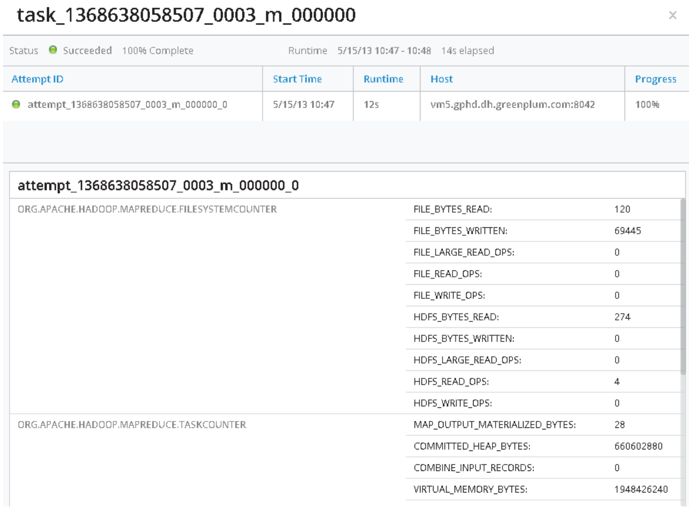
JOB SUMMARY: 100% MAPPERS COMPLETED / 100% REDUCERS COMPLETED Total Run Time 9m 45s

MAPPERS Running 0 Pending 0 Sort by: Task ID

Task ID	Runtime	Host	Progress	Attempts
● task_1368638058507_0003_m_000000	14s	vm5.gphd.dh.greenplum.com:8042	100%	1
● task_1368638058507_0003_m_000001	15s	vm5.gphd.dh.greenplum.com:8042	100%	1
● task_1368638058507_0003_m_000002	15s	vm5.gphd.dh.greenplum.com:8042	100%	1
● task_1368638058507_0003_m_000003	26s	vm5.gphd.dh.greenplum.com:8042	100%	1
● task_1368638058507_0003_m_000004	28s	vm5.gphd.dh.greenplum.com:8042	100%	1
● task_1368638058507_0003_m_000005	27s	vm5.gphd.dh.greenplum.com:8042	100%	1
● task_1368638058507_0003_m_000006	43s	vm5.gphd.dh.greenplum.com:8042	100%	1

This screen displays all the tasks that are have been allocated for the selected job and their progress. You can see the mapper and the reducer tasks separately. In the above screen capture, the bars in the JOB SUMMARY section represent the two Mapper tasks that have run, one took 19 seconds, the other, 20 seconds.

Clicking on each task ID will show even more details about that particular task. You can also filter on a particular task ID in the search bar.



task_1368638058507_0003_m_000000

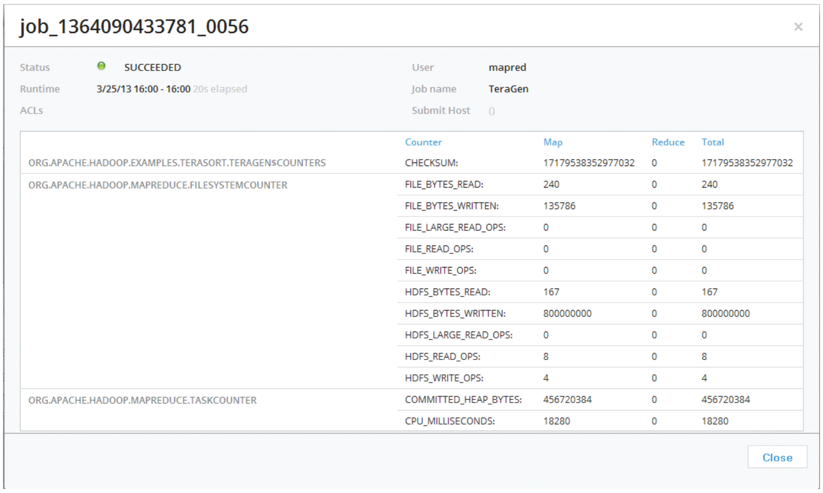
Status ● Succeeded 100% Complete Runtime 5/15/13 10:47 - 10:48 14s elapsed

Attempt ID	Start Time	Runtime	Host	Progress
● attempt_1368638058507_0003_m_000000_0	5/15/13 10:47	12s	vm5.gphd.dh.greenplum.com:8042	100%

attempt_1368638058507_0003_m_000000_0

ORG.APACHE.HADOOP.MAPREDUCE.FILESYSTEMCOUNTER	FILE_BYTES_READ:	120
	FILE_BYTES_WRITTEN:	69445
	FILE_LARGE_READ_OPS:	0
	FILE_READ_OPS:	0
	FILE_WRITE_OPS:	0
	HDFS_BYTES_READ:	274
	HDFS_BYTES_WRITTEN:	0
	HDFS_LARGE_READ_OPS:	0
	HDFS_READ_OPS:	4
	HDFS_WRITE_OPS:	0
ORG.APACHE.HADOOP.MAPREDUCE.TASKCOUNTER	MAP_OUTPUT_MATERIALIZED_BYTES:	28
	COMMITTED_HEAP_BYTES:	660602880
	COMBINE_INPUT_RECORDS:	0
	VIRTUAL_MEMORY_BYTES:	1948426240

To see job related counters click on **View more job details** next to the job ID:



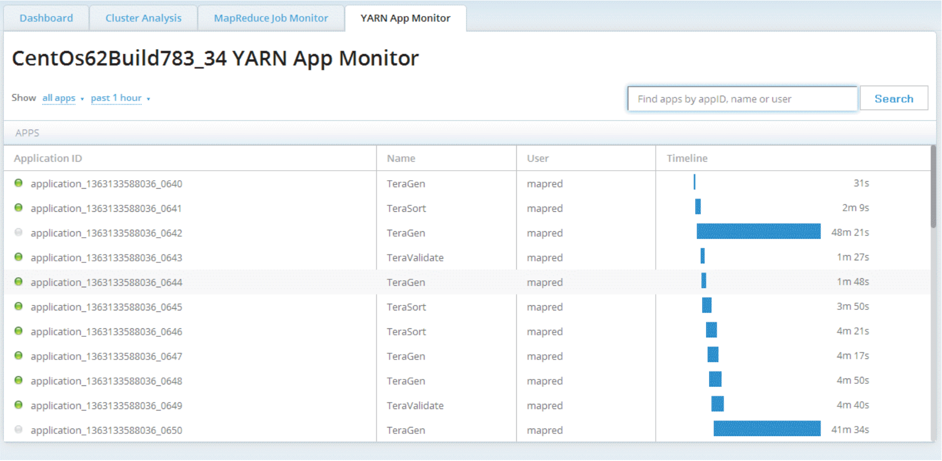
job_1364090433781_0056

Status	● SUCCEEDED	User	mapred
Runtime	3/25/13 16:00 - 16:00 20s elapsed	Job name	TeraGen
ACLs		Submit Host	()

	Counter	Map	Reduce	Total
ORG.APACHE.HADOOP.EXAMPLES.TERASORT.TERAGEN$COUNTERS	CHECKSUM:	17179538352977032	0	17179538352977032
ORG.APACHE.HADOOP.MAPREDUCE.FILESYSTEMCOUNTER	FILE_BYTES_READ:	240	0	240
	FILE_BYTES_WRITTEN:	135786	0	135786
	FILE_LARGE_READ_OPS:	0	0	0
	FILE_READ_OPS:	0	0	0
	FILE_WRITE_OPS:	0	0	0
	HDFS_BYTES_READ:	167	0	167
	HDFS_BYTES_WRITTEN:	800000000	0	800000000
	HDFS_LARGE_READ_OPS:	0	0	0
	HDFS_READ_OPS:	8	0	8
	HDFS_WRITE_OPS:	4	0	4
ORG.APACHE.HADOOP.MAPREDUCE.TASKCOUNTER	COMMITTED_HEAP_BYTES:	456720384	0	456720384
	CPU_MILLISECONDS:	18280	0	18280

YARN App Monitor

The YARN App Monitor screen tracks YARN applications that are executed in the Pivotal HD Cluster.



CentOs62Build783_34 YARN App Monitor

Show all apps ▾ past 1 hour ▾

APPS

Application ID	Name	User	Timeline
● application_1363133588036_0640	TeraGen	mapred	31s
● application_1363133588036_0641	TeraSort	mapred	2m 9s
○ application_1363133588036_0642	TeraGen	mapred	48m 21s
● application_1363133588036_0643	TeraValidate	mapred	1m 27s
● application_1363133588036_0644	TeraGen	mapred	1m 48s
● application_1363133588036_0645	TeraSort	mapred	3m 50s
● application_1363133588036_0646	TeraSort	mapred	4m 21s
● application_1363133588036_0647	TeraGen	mapred	4m 17s
● application_1363133588036_0648	TeraGen	mapred	4m 50s
● application_1363133588036_0649	TeraValidate	mapred	4m 40s
○ application_1363133588036_0650	TeraGen	mapred	41m 34s

The YARN applications displayed can be filtered by category and/or time range:

- **By Category**:
  - **all apps** (set by default)
  - **currently pending apps**
  - **currently running apps**
  - **succeeded apps**
  - **failed apps**
  - **killed apps**
- **By Time Range**:

  By selecting a preset time range in hours, weeks, months, year, or by specifying a custom time range.

The YARN applications can also be filtered by the following fields by entering it in the search bar in the following format: `searchKey=searchValue`:

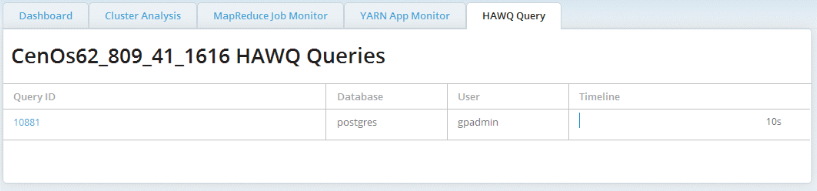
- **appID**
- **name**
- **user**

These are substring searches. For example: **appID=1363920466130** will locate the application with **appID=application_1363920466130_0002**

HAWQ Query Monitor

The HAWQ Query monitor is only displayed when HAWQ is installed on the cluster.

This screen displays all **active** queries running on the HAWQ cluster:



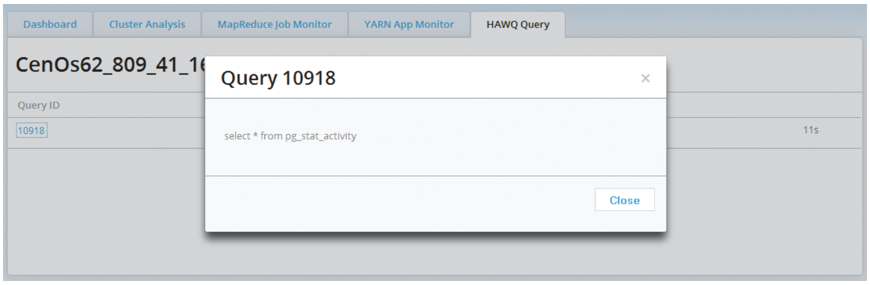
CenOs62_809_41_1616 HAWQ Queries

Query ID	Database	User	Timeline
10881	postgres	gpadmin	10s

In this release, this screen only displays **active** queries as can be seen when you run:
`SELECT * FROM pg_stat_activity;`

on the HAWQ cluster.

Click on a Query ID to get the syntax of that query:



Query 10918

```
select * from pg_stat_activity
```

4. Pivotal Command Center Performance Monitor

This section provides an overview of the Pivotal Command Center performance monitor known as `nmon`.

Overview

Pivotal Command Center comes with a Performance monitor called `nmon` (for node monitor). This makes use of a highly scalable message passing architecture to gather performance metrics from each node that Command Center monitors. This consists of a `nmon` master daemon that runs on the Command Center admin host and an `nmon` daemon that runs on all the cluster nodes that report system metric information to the `nmon` master. This includes metrics such as CPU, memory, disk I/O and network usage information.

The `nmon` master on the admin host dumps the system metrics it receives from the `nmon` agents on the cluster nodes into a PostgreSQL DB. This is then queried by the Command Center UI application to display its cluster analysis graphs.

The `nmon` agents hosts are deployed throughout the cluster during Pivotal HD cluster deployment itself (see *Pivotal HD Enterprise 1.0 Installation and Administrator Guide* for details).

The agents are deployed as services on each host, including on the Pivotal Command Center admin host. To stop or start the `nmon` service run the following as root:

```
# service nmon stop
# service nmon start
```

A. Creating a YUM EPEL Repository

Pivotal Command Center and Pivotal HD Enterprise expect some prerequisite packages to be pre-installed on each host, depending on the software that gets deployed on a particular host. In order to have a smoother installation it is recommended that each host would have yum access to an EPEL yum repository. If you have access to the Internet, then you can configure your hosts to have access to the external EPEL repositories. However, if your hosts do not have Internet access (or you are deploying onto a large cluster), then having a local yum EPEL repo would be highly recommended. This will also give you some control on the package versions you want deployed on your cluster.

Following are the steps to create a local yum repo:

1. Mount the RHEL/CentOS DVD on a machine that will act as the local yum repo

2. Install a webserver on that machine (e.g. httpd), making sure that HTTP traffic can reach this machine

3. Install the following packages on the machine:

   ```
   yum-utils
   createrep
   ```

4. Go to the directory where the DVD is mounted and run the following command:

   ```
   # createrepo .
   ```

5. Create a repo file on each host with a descriptive filename in the `/etc/yum.repos.d/` directory of each host (for example, *`CentOS-6.1.repo`*) with the following contents:

   ```
   [CentOS-6.1]
   name=CentOS 6.1 local repo for OS RPMS
   baseurl=http://172.254.51.221/centos/$releasever/os/
   $basearch/
   enabled=1
   gpgcheck=1
   gpgkey=http://172.254.51.221/centos/$releasever/os/$basearch
   /RPM-GPG-KEY-CentOS-6
   ```

6. Validate that you can access the local yum repos by running the following command:

   ```
   Yum list
   ```

B. Best Practises -Troubleshooting

nmon Issues

- If you have to restart the Admin node, ensure that the nmon service is started.
- If you notice any of the clusters are not being fully monitored, perform the following on the Admin node:
  - Make sure the nmon configuration (`/etc/nmon/nmon-site.xml`) includes all the clusters and their hosts. If it doesn't, update it and distribute the updated configuration to all the cluster hosts, then restart nmon on the Admin node as well as on the cluster hosts:

    ```
    sudo service nmon restart
    ```

    ```
    massh clusterHosts verbose 'sudo service nmon restart'
    ```

    Where *`clusterHosts`* contains all the cluster hosts.

Job Monitor Page

- If an application is completed, but on the job monitoring page in the Command Center User Interface, it shows app/job as still running, then check whether History Server is running or not. If it is not running, start it.

  Check using: `http://<HistoryServerHost>:19888`