Cloudera Manager Enterprise Edition Installation Guide

Important Notice

The information in this document is subject to change without notice. Cloudera shall not be liable for any damages resulting from technical errors or omissions which may be present in this document, or from use of this document.

Cloudera, Inc.
220 Portage Avenue
Palo Alto, CA 94306
info@cloudera.com
US: 1-888-789-1488
Intl: 1-650-362-0488
www.cloudera.com

Release Information

Version: 4.5.3
Date: May 23, 2013

Table of Contents

Installing CDH and Cloudera Manager..37

About this Guide

This guide explains how to install CDH and Cloudera Manager. This guide also explains how to use Cloudera Manager to install, configure, manage, and monitor CDH on your cluster. Cloudera Manager 4 supports managing CDH3 and CDH4.

Other Cloudera Manager Guides

Guide	Available Here
Cloudera Manager 4.5.x Release Notes	Cloudera Manager Enterprise Edition 4.5.x Release Notes
Cloudera Manager User Guide	Cloudera Manager Enterprise Edition User Guide
Configuring Hadoop Security with Cloudera Manager	Configuring Hadoop Security with Cloudera Manager Enterprise Edition
Configuring TLS Security for Cloudera Manager	Configuring TLS Security for Cloudera Manager Enterprise Edition
Configuring Ports for Cloudera Manager	Configuring Ports for Cloudera Manager Enterprise Edition

Introduction to Cloudera Manager Installation

Cloudera Manager automates the installation and configuration of CDH on an entire cluster, requiring only that you have root SSH access to your cluster's machines, and access to the internet or a local repository with installation files for all these machines. Cloudera Manager consists of:

- A small self-executing Cloudera Manager installation program to install the Cloudera Manager Server and other packages in preparation for cluster host installation
- Cloudera Manager wizard for automating CDH installation and configuration on the cluster hosts
- Cloudera Manager features for monitoring and configuring the cluster after installation is completed

Cloudera Manager provides two methods for installing CDH and its associated components: traditional packages (RPMs or Debian packages) or parcels. Parcels are a new packaging format that simplifies the installation process, and more importantly allows you to download, distribute, and activate a new CDH version all from within Cloudera Manager. Parcels are available for CDH (4.1.2 and onwards) and Impala 1.0.

About the Cloudera Manager Installation Program

The Cloudera Manager installation program, which you will install on the host where you want to the Cloudera Manager Server to run, automatically:

- Installs the package repositories for Cloudera Manager and the Oracle Java Development Kit (JDK)
- Installs the Oracle JDK if it is not already installed
- Installs the Cloudera Manager Server
- Installs and configures an embedded PostgreSQL database for use by the Cloudera Manager server

About the Cloudera Manager Wizard

After you have installed the Cloudera Manager Server and when you run it for the first time, you can use the Cloudera Manager wizard to do the following on the cluster hosts automatically.

- Using SSH, discover the cluster hosts you specify via IP address ranges or hostnames
- Configure the parcel or package repositories for Cloudera Manager, CDH, Impala, and the Oracle JDK
- Install the Cloudera Manager Agent, CDH, and Impala on the cluster hosts
- Install the Oracle JDK if it is not already installed on the cluster hosts
- Determine mapping of services to hosts
- Suggest a Hadoop configuration and start the Hadoop services

Note:

- If you will use external databases, you must install and configure those databases before you start the wizard. These are the databases that will be used by Cloudera Manager, Service Monitor, Activity Monitor, Host Monitor, Report Manager, Cloudera Navigator, and the Hive Metastore. See Installing and Configuring Databases for more information. If you will use the embedded PostgreSQL database, you do not have to prepare databases in advance.
- When you install or upgrade Cloudera Manager and/or CDH using parcels, only the Cloudera Manager server host needs access to a remote parcel repository. The other cluster hosts only need access to the local repository (by default `/opt/cloudera/parcels`) on the Cloudera Manager server.
- When you use the Cloudera Manager Wizard to install or upgrade Cloudera Manager and/or CDH on your cluster hosts using packages, all of those hosts need access to installation files.
- Installation files are available on the Internet at `archive.cloudera.com` or you can download installation files and create a local repository. For more information, see the individual installation and upgrade procedures, and the Cloudera Manager Enterprise Edition Frequently Asked Questions.

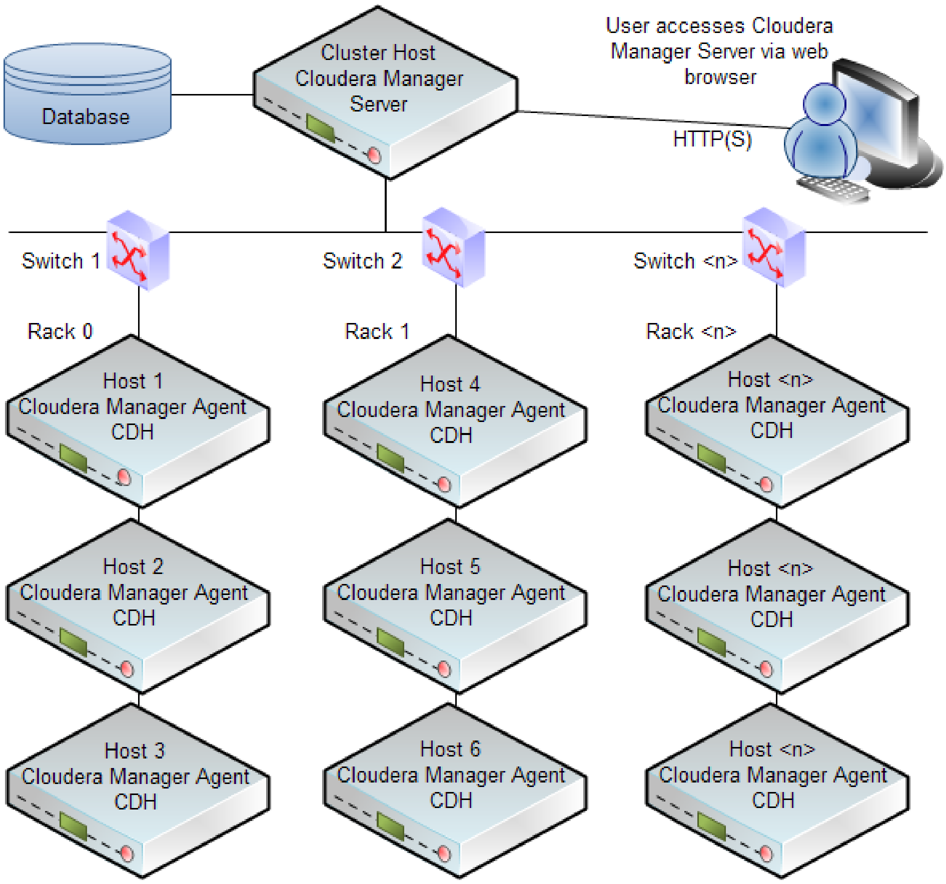
You can choose to abort the Cloudera Manager Agent and CDH installation process and Cloudera Manager wizard will automatically revert and completely rollback the installation process for any uninstalled components. (Installation that has completed successfully on a given host is not rolled back on that host.)

Installing Cloudera Manager for the First Time

To install Cloudera Manager, you will:

- Install a database application on the Cloudera Manager Server host machine or on a host machine that the Cloudera Manager Server can access, and (depending on the configuration you decide on) on other hosts as well.
- Install the Cloudera Manager Server on one cluster host machine.
- Install CDH and the Cloudera Manager Agents on the other cluster host machines.

The following diagram illustrates a simple example of where each component is installed.



Note:

If you want to configure security for Cloudera Manager, see the following sections:

- Configuring Hadoop Security with Cloudera Manager Enterprise Edition
- Configuring TLS Security for Cloudera Manager Enterprise Edition

For overview and usage information, see the Cloudera Manager Enterprise Edition User Guide.

Requirements for Cloudera Manager

Cloudera Manager interacts with a diversity of entities such as operating systems, databases, and browsers. Cloudera provides information about which major release version and minor release version is supported, where available. In some cases, such as some browsers, a minor version may not be provided. After installing each element, upgrade to the latest patch version and apply any other appropriate updates. Note that the available updates may be specific to the operating system on which it is installed.

For example, you might be using CentOS in your environment. You could choose 6 as the major version and 2 as the minor version. These choices would mean you would be using CentOS 6.2. After installing this operating system, you would then apply any and all relevant CentOS 6.2 upgrades and patches.

For information on CDH4 requirements, see "Supported Operating Systems for CDH4" in the CDH4 Installation Guide.

For information on CDH3 requirements, see "Supported Operating Systems for CDH3" in the CDH3 Installation Guide.

Supported Operating Systems for Cloudera Manager

Cloudera Manager supports a range of operating systems including:

- Red Hat-compatible systems
  - Red Hat Enterprise Linux 5.7 and CentOS 5.7, 64-bit
  - Red Hat Enterprise Linux 6.2 and CentOS 6.2, 64-bit
  - Oracle Enterprise Linux 5.6 with Unbreakable Enterprise Kernel, 64-bit
- SLES systems
  - SUSE Linux Enterprise Server 11, 64-bit. Service Pack 1 or later is required. Also, the SUSE Linux Enterprise Software Development Kit 11 SP1 is required on cluster hosts running the Cloudera Manager Agents (not required on the Cloudera Manager Server host); you can download the SDK here.
- Debian systems
  - Debian 6.0 (Squeeze), 64-bit
- Ubuntu systems
  - Ubuntu 10.04 (Lucid Lynx), 64-bit
  - Ubuntu 12.04 (Precise Pangolin), 64-bit

Note:

For Hadoop to work properly, using the same version of the same operating system on all cluster hosts is strongly recommended.

Supported Browsers for Cloudera Manager Admin console

The Cloudera Manager Admin console, which you use to configure, manage, and monitor CDH, supports the following browsers:

- Firefox 11 or later
- Google Chrome

- Internet Explorer 9
- Safari 5 or later

Other Requirements

Cloudera Manager supports a variety of services and depends on resources being available.

Version Support

- Cloudera Manager 4.5 supports CDH3 Update 1 (cdh3u1) or later and CDH4.0 or later. CDH3 Update 2 or later is strongly recommended.

> **Warning:**
>
> The latest released versions of CDH3 and CDH4 are strongly recommended.

- If you want to use Cloudera Manager to manage Oozie, CDH3 Update 2 or later is required.
- Cloudera Manager uses Python. Python is part of the default installation for all operating systems that Cloudera Manager supports, so there is no need to complete any installation tasks to make Python available. Cloudera Manager is tested with the default installation. Modifying the Python installation available on systems on which you install Cloudera Manager is not supported.
- Cloudera Manager supports Impala 1.0 or later with CDH4.1 or CDH4.2.

Resources

Cloudera Manager requires sufficient:

- Disk space. A minimum of:
- 5 GB on the partition hosting `/var`.
- 500 MB on the partition hosting `/usr`.
- RAM. 4GB is appropriate for most cases, and is required when using Oracle databases. 2GB may be sufficient for non-Oracle deployments involving fewer than 100 hosts.

Networking and Security

- Cluster hosts must have a working network name resolution system. Properly configuring DNS and reverse DNS meets this requirement. If you use `/etc/hosts` instead of DNS, all hosts files must contain consistent information about host names and addresses across all nodes. For example, `/etc/hosts` might contain something of the form:

```
127.0.0.1 localhost.localdomain localhost
192.168.1.1 cluster-01.domain cluster-01
192.168.1.2 cluster-02.domain cluster-02
192.168.1.3 cluster-03.domain cluster-03
```

- In most cases, the Cloudera Manager Server must have SSH access to the cluster hosts when you run the installation or upgrade wizard. This does not apply if you install Cloudera Manager using Path B.

Note:

You must log in using a root account or an account that has password-less sudo permission. For authentication during the installation and upgrade procedures, you will need to either enter the password or upload a public and private key pair for the root or sudo user account. If you want to use a public and private key pair, the public key must be installed on the cluster hosts before you use Cloudera Manager. Authentication is not supported for accounts that have password-protected sudo permission. Cloudera Manager uses SSH only during the initial install or upgrade. Once your cluster is set up, you can safely disable root SSH access or change the root password. Cloudera Manager does not save SSH credentials and all credential information is discarded once the installation is complete.

- No blocking by iptables or firewalls; make sure port 7180 is open because it is the port used to access Cloudera Manager after installation. Cloudera Manager communicates using specific ports, which must be open. For additional port information, see Configuring Ports for Cloudera Manager Free Edition.
- No blocking by Security-Enhanced Linux (SELinux).
- Cloudera Manager and CDH use several user accounts and groups to complete their tasks. The set of user accounts and groups varies according to which components you choose to install. Do not delete these accounts or groups and do not modify their permissions and rights. Ensure no existing systems obstruct the functioning of these accounts and groups. For example, if you have scripts that delete user accounts not

in a white-list, add these accounts to the list of permitted accounts. Cloudera Manager and CDH create and use the following accounts and groups:

Account	Type	Product	Comment
`cloudera-scm`	User and group	Cloudera Manager	
`mapred`	User and group	CDH3 and CDH4	MapReduce
`hdfs`	User and group	CDH3 and CDH4	Distributed file system
`zookeeper`	User and group	CDH3 and CDH4	Distributed system coordination service
`yarn`	User and group	CDH4	MapReduce2.0 or MRv2
`httpfs`	User and group	CDH4	HTTP gateway to HDFS
`hbase`	User and group	CDH3 and CDH4	Hadoop database
`hive`	User and group	CDH3 and CDH4	Hadoop data warehouse
`hue`	User and group	CDH3 and CDH4	Web interface to hadoop
`oozie`	User and group	CDH3 and CDH4	Workflow coordination system
`flume`	User and group	CDH3 and CDH4	Log collection system
`hadoop`	Group	CDH3 and CDH4	
`impala`	User and group	CDH4.1	Interactive query tool

- The Cloudera Manager Agent runs as root so that it can make sure the required directories are created and that processes and files are owned by the appropriate user (for example, the `hdfs` user and `mapred` user).

For additional port information, see Configuring Ports for Cloudera Manager Enterprise Edition.

Supported Databases for Cloudera Manager

Cloudera Manager requires several databases. The Cloudera Manager server stores information about configured services, role assignments, configuration history, commands, users, and running processes in a database of its own. The Activity Monitor, Service Monitor, Report Manager, and Host Monitor also each use a database to store information.

The database you choose to use must be configured to support UTF8 character set encoding. The embedded PostgreSQL database that is installed using Path A automatically provides UTF8 encoding. If you install a custom database, you may need to enable UTF8 encoding. The commands for enabling UTF8 encoding are described in each database's section under Installing and Configuring Databases.

After installing a database, upgrade to the latest patch version and apply any other appropriate updates. Note that the available updates may be specific to the operating system on which it is installed.

Cloudera Manager and its supporting services can use the following database systems and releases:

- MySQL:
- 5.0
- 5.1
- 5.5

- Oracle
- 10g Release 2
- 11g Release 2

- PostgreSQL
- 8.1
- 8.3
- 8.4
- 9.1

Installing and Configuring Databases

Cloudera Manager uses databases to store information about the Cloudera Manager configuration, as well as information such as the health of the system or task progress. Cloudera Manager supports using a variety of databases to store required information. To facilitate rapid completion of simple installations, the Cloudera Manager can install and configure a PostgreSQL database as part of the broader Cloudera Manager installation process. This automatically installed database is sometimes referred to as an embedded PostgreSQL database. While the embedded database is a useful option for getting started quickly, Cloudera Manager also allows you to use other databases. You can opt to use your own PostgreSQL database or MySQL or Oracle databases.

If you plan to use the embedded database provided during the Cloudera Manager installation for all databases, you can skip ahead to Installation Path A - Automated Installation by Cloudera Manager. To learn more about database options or if you are unsure whether or not using the embedded database is right for your environment, continue reading.

What Databases Must Be Installed

The Cloudera Manager Server and the server's Activity Monitor, Service Monitor, Report Manager, and Host Monitor all require databases, as does Cloudera Navigator and the Hive Metastore. Cloudera Manager does support deploying different types of databases in a single environment, but doing so may create unexpected complications. Cloudera recommends choosing one of the three database providers to use for all of the Cloudera Manager databases.

Cloudera provides two install paths:

- Path A automatically installs embedded PostgreSQL databases to meet the requirements of the services. This path reduces the number of installation tasks you must complete, as well as the number of choices to make.
- Path B requires you have databases in your environment for use by Cloudera Manager, Cloudera Management Services, and Hive metastore. This path requires more input and intervention as you either install databases or gather information about existing databases. This path also provides greater flexibility in choosing database types and configurations.

A service works with a database. In most cases, you should install databases and services on the same host. For example, if you create the database for Activity Monitor on `myhost1`, then you should typically assign the Activity Monitor role to `myhost1`. You will assign the Service Monitor, Activity Monitor, Report Manager, and Host Monitor roles in the Cloudera Manager wizard during the install or upgrade process. After completing the install or upgrade process, you can also modify role assignments in the Management services pages of Cloudera Manager. While it is true that database location is changeable, before beginning an installation or upgrade, you should decide which hosts you will use. Note that the JDBC connector for your database **must** be installed on the hosts where you assign the Service Monitor, Activity Monitor, Report Manager, and Host Monitor roles. Installing JDBC connectors is described later in this guide.

It is possible to install the database and services on different hosts. Separating databases from services is more likely to occur in larger deployments and in cases where more sophisticated database administrators actively choose to establish such a configuration. For example, databases and services might be separated if your environment includes Oracle databases that will be separately managed by Oracle database administrators (DBAs).

The table that follows provides a summary; details are in the sections that follow.

Install or Upgrade Path	Install Supported Database For	Typically Install Databases on Systems That Will Host
Installation Path A - Automated Installation by Cloudera Manager	No installations required. Automated installation	No manual installation required.

Install or Upgrade Path	Install Supported Database For	Typically Install Databases on Systems That Will Host
	automatically creates embedded PostgreSQL databases for all Cloudera Manager and all services.	
Installation Path B - Installation Using Your Own Method	The Cloudera Manager Server configuration and for Activity Monitor, Service Monitor, Report Manager, Host Monitor, Hive Metastore, and Cloudera Navigator.	The Cloudera Manager Server, Activity Monitor, Service Monitor, Report Manager, Host Monitor, and Cloudera Navigator roles, and the Hive Metastore. Alternately, you may install these databases on other systems, assuming those systems are accessible to the Cloudera Manager Server.
Upgrade from Cloudera Manager Free Edition 3.7.x, Cloudera Manager Enterprise Edition 3.7.x, or SCM Express 3.6 to Cloudera Manager 4.5 Enterprise Edition	Host Monitor and Cloudera Navigator roles, and the Hive Metastore.	Host Monitor and Cloudera Navigator roles, and the Hive Metastore.

About the Cloudera Manager Server Database

This database, which is used for storing information about services' configurations, is independent of the databases used by the Activity Monitor, Service Monitor, Report Manager, Host Monitor, Cloudera Navigator, or the Hive Metastore.

Installation type	Process
Automatic installation: Installation Path A	The wizard automatically installs, configures, and uses embedded PostgreSQL databases to store information about service configuration, as well as the Activity Monitor, Service Monitor, Report Manager, Host Monitor, Cloudera Navigator, and the Hive Metastore. This functionality is provided by the `cloudera-scm-server-db` package, and you can start and stop these databases using the `service cloudera-scm-server-db [start|stop]` command. If you are using Installation Path A, you can proceed directly to Installation Path A.
Manual installation: Installation Path B	You must install a supported database. This database can be installed on the machine where you install the Cloudera Manager Server or on a machine accessible to the Cloudera Manager Server. You will need to configure the connection between Cloudera Manager and the database, as is described in the documentation for Path B.

The installation instructions for the database containing the Cloudera Manager Server's configuration are also included under Installation Path B - Installation Using Your Own Method.

Important:

- To prevent deadlocks, Cloudera Manager requires the isolation level to be set to read committed. This is the default setting for PostgreSQL and Oracle. The instructions for using a MySQL database with Cloudera Manager describe how to configure this setting. Contact your DBA to find out what your isolation level currently is set to.
- If you use a MySQL database to store information about service configuration, make sure that the `InnoDB` engine is configured, not the `MyISAM` engine. Cloudera Manager will not start if its tables are configured with the MyISAM engine. (Typically, tables revert to MyISAM if the InnoDB engine is misconfigured.) To check what engine your tables are using, run the following command from the MySQL shell:

```
mysql> show table status;
```

Using an External Database for Oozie, Hue, and Hive Metastore

Configuring an External Database for Oozie

By default, Cloudera Manager uses Derby for Oozie's database. If you want to use an external database for Oozie, you would do the configuration after Cloudera Manager is installed. For more information, see Using an External Database for Oozie.

Configuring an External Database for Hue

By default, Cloudera Manager uses SQLite for Hue's database. If necessary, you can configure Cloudera Manager to use an external database such as MySQL as the database for Hue - do this after Cloudera Manager is installed. For more information, see Using an External Database for Hue.

Configuring an External Database for Hive Metastore

By default, Cloudera Manager uses the embedded PostgreSQL database for the Hive metastore. If necessary, you can configure Cloudera Manager to use an external database such as MySQL or Oracle as the database for Hive Metatstore - do this before Cloudera Manager is installed. For more information, see "Configuring the Metastore Database" in the CDH4 Installation Guide.

Backing up the Databases

It's important that you periodically back up the databases that Cloudera Manager uses to store configuration, monitoring, and reporting data. Be sure to back up all of the databases you are using with Cloudera Manager:

- Cloudera Manager database: This is the most important database to back up. This database contains all the information about what services you have configured, their role assignments, all configuration history, commands, users, and running processes. This is a relatively small database, typically smaller than 100MB.
- Activity Monitor database: Contains information about past activities. In large clusters, this database can become very large.
- Service Monitor database: contains monitoring information about daemons. In large clusters, this database can become very large.
- Report Manager database: Keeps track of disk utilization over time. This database is typically medium-sized.
- Host Manager database: Contains information about host status. The number of hosts in the cluster affects this database's size, so the database size varies, but the database is typically large in deployments with many hosts.

Installing and Configuring a MySQL Database

You can use MySQL databases to store information for Cloudera Manager. Cloudera Manager requires databases to store information, so you may need to create databases for each of those entities. See your MySQL documentation for more information about installation and configuration.

Installing MySQL

To install MySQL on a Red Hat system:

```
$ sudo yum install mysql-server
```

To install MySQL on a SLES system:

```
$ sudo zypper install mysql
$ sudo zypper install libmysqlclient_r15
```

> **Note:**
>
> Some SLES systems encounter errors when using the preceding `zypper install` command. For more information on resolving this issue, see Novell's Knowledgebase topic, error running chkconfig.

To install MySQL on an Debian/Ubuntu system:

```
$ sudo apt-get install mysql-server
```

After issuing the command to install MySQL, you may need to respond to prompts to confirm that you do want to complete the installation. After installation completes, start the mysql daemon.

On Red Hat systems

```
$ sudo service mysqld start
```

On SLES and Debian/Ubuntu systems

```
$ sudo service mysql start
```

Configuring MySQL for the Cloudera Manager Databases

The default settings in the MySQL installations in most distributions are very conservative with regards to buffer sizes and memory usage. In addition, Cloudera Manager requires InnoDB. Cloudera Managment services need high write throughput as, based on cluster size, they may insert a lot of records in the database. Therefore Cloudera recommends that you set `innodb_flush_method` to `O_DIRECT`.

For the databases used by Cloudera Manager, Cloudera recommends that you update `/etc/my.cnf` or `/etc/mysql/my.cnf` to at least the values shown below. It is important that the `datadir` directory, which, by default, is `/var/lib/mysql`, is on a partition that has plentiful free space.

Recommended Settings

1. Determine the version of MySQL.

Important:

- For a fresh MySQL installation on Red Hat or SLES systems, apply the settings in the next step before you start MySQL for the first time.
- For an existing installation and a new installation on Ubuntu, you must perform the steps in Reconfiguring InnoDB Settings for an Existing MySQL Installation*before* changing InnoDB settings.

2. Recommended MySQL configurations settings are as follows. You should incorporate these changes as appropriate into your configuration settings.

```
[mysqld]
transaction-isolation=READ-COMMITTED
# Disabling symbolic-links is recommended to prevent assorted security risks;
# to do so, uncomment this line:
# symbolic-links=0

key_buffer                = 16M
key_buffer_size           = 32M
max_allowed_packet        = 16M
thread_stack              = 256K
thread_cache_size         = 64
query_cache_limit         = 8M
query_cache_size          = 64M
query_cache_type          = 1
# Important: see Configuring the Databases and Setting max_connections
max_connections           = 550

# log-bin should be on a disk with enough free space
log-bin=/x/home/mysql/logs/binary/mysql_binary_log

# For MySQL version 5.1.8 or later. Comment out binlog_format for older versions.
binlog_format             = mixed

read_buffer_size = 2M
read_rnd_buffer_size = 16M
sort_buffer_size = 8M
join_buffer_size = 8M

# InnoDB settings
innodb_file_per_table = 1
innodb_flush_log_at_trx_commit  = 2
innodb_log_buffer_size          = 64M
innodb_buffer_pool_size         = 4G
innodb_thread_concurrency       = 8
innodb_flush_method             = O_DIRECT
innodb_log_file_size = 512M

[mysqld_safe]
log-error=/var/log/mysqld.log
pid-file=/var/run/mysqld/mysqld.pid
```

Configuring the Databases and Setting max_connections

The definition of a small or large cluster is not absolute, so this information is intended as general guidance. For the purposes of this discussion, clusters with fewer than 50 hosts can be considered small clusters and clusters with more than 50 hosts can be considered large clusters.

Follow these guidelines:

- In a small cluster, you can store more than one database (for example, both the Activity Monitor and Service Monitor) on the same host. If you do this, you should:
- Put each database on its own storage volume.
- Allow 100 maximum connections for each database and then add 50 extra connections. For example, for two databases set the maximum connections to 250. If you store seven databases on one host (the databases

for Activity Monitor, Service Monitor, Report Manager, Host Monitor, Cloudera Manager Server, Cloudera Navigator, and Hive Metastore), set the maximum connections to 750.
- In a large cluster, do not store more than one database on the same host. In such a case, use a separate host for each database/host pair. The hosts need not be reserved exclusively for databases, but each database should be on a separate host.

Reconfiguring InnoDB Settings for an Existing MySQL Installation

To update InnoDB settings on all hosts that are using an existing MySQL installation, proceed as follows.

1. Stop MySQL.

   On Red Hat systems

   ```
   $ sudo service mysqld stop
   ```

   On SLES and Debian/Ubuntu systems

   ```
   $ sudo service mysql stop
   ```

2. Edit the InnoDB entries in `/etc/my.cnf` as shown in the previous section.
3. Move the old InnoDB log files to a backup location. The two files to move are `/var/lib/mysql/ib_logfile0` and `/var/lib/mysql/ib_logfile1`. Make sure you move these files out of the `/var/lib/mysql/` directory (don't copy them and leave the originals in place).
4. Start MySQL.

   On Red Hat systems

   ```
   $ sudo service mysqld start
   ```

   On SLES and Debian/Ubuntu systems

   ```
   $ sudo service mysql start
   ```

Installing the MySQL JDBC Connector

Install the JDBC connector on the Cloudera Manager Server host, as well as hosts to which you assign the Activity Monitor, Service Monitor, Report Manager, Host Monitor, Cloudera Navigator, and Hive Metastore Server roles.

Cloudera recommends that you assign all roles that require databases on the same host and install the connector on that host. While putting all such roles on the same host is recommended, it is not required. You could install a role, such as Activity Monitor on one host and other roles on a separate host. In such a case you would install the JDBC connector on each host running roles that access the database.

On Red Hat 6 systems:

```
$ sudo yum install mysql-connector-java
```

On Red Hat 5 systems:

1. Download the MySQL JDBC connector from http://www.mysql.com/downloads/connector/j/5.1.html.
2. Extract the JDBC driver JAR file from the downloaded file; for example:

   ```
   tar zxvf mysql-connector-java-5.1.18.tar.gz
   ```

3. Add the JDBC driver, renamed, to the relevant server; for example:

   ```
   $ sudo cp mysql-connector-java-5.1.18/mysql-connector-java-5.1.18-bin.jar
   /usr/share/java/mysql-connector-java.jar
   ```

If the target directory does not yet exist on this host, you can create it before copying the `.jar` file; for example:

```
$ sudo mkdir -p /usr/share/java/
$ sudo cp mysql-connector-java-5.1.18/mysql-connector-java-5.1.18-bin.jar
/usr/share/java/mysql-connector-java.jar
```

On SLES systems:

```
$ sudo zypper install mysql-connector-java
```

On Debian/Ubuntu systems:

```
$ sudo apt-get install libmysql-java
```

Configuring MySQL

Configure MySQL to use a strong password and to start at boot.

1. Set the MySQL root password. Note that in the following procedure, your current `root` password is blank. Press the **Enter** key when you're prompted for the root password.

```
$ sudo /usr/bin/mysql_secure_installation
[...]
Enter current password for root (enter for none):
OK, successfully used password, moving on...
[...]
Set root password? [Y/n] y
New password:
Re-enter new password:
Remove anonymous users? [Y/n] Y
[...]
Disallow root login remotely? [Y/n] N
[...]
Remove test database and access to it [Y/n] Y
[...]
Reload privilege tables now? [Y/n] Y
All done!
```

2. Ensure the MySQL server starts at boot.

On Red Hat systems:

```
$ sudo /sbin/chkconfig mysqld on
$ sudo /sbin/chkconfig --list mysqld
mysqld          0:off   1:off   2:on    3:on    4:on    5:on    6:off
```

On SLES systems:

```
$ sudo chkconfig --add mysql
```

On Debian/Ubuntu systems:

```
$ sudo chkconfig mysql on
```

Creating the MySQL Databases for Cloudera Manager

The next step involves creating databases and user accounts for all database-backed services in Cloudera Manager.

Create databases for each of the following features that are part of the Cloudera Management Services:

- Activity Monitor

- Service Monitor
- Report Manager
- Host Monitor
- Cloudera Navigator

In addition add a database for each Hive service.

You can create these databases on the host where the Cloudera Manager Server will run, or on any other nodes in the cluster. For performance reasons, you should typically install each database on the host on which the service runs, as determined by the roles you will assign during installation or upgrade. In larger deployments or in cases where database administrators (DBAs) are managing the databases the services will use, databases may be separated from services, but do not undertake such an implementation lightly.

Note the values you enter for database names, user names, and passwords. The Cloudera Manager installation wizard requires this information to correctly connect to these databases.

The database must be configured to support UTF-8 character set encoding. The sample commands below include the required options to enable UTF-8 support.

To create the MySQL databases for Cloudera Manager:

1. Log into MySQL as the root user:

   ```
   $ mysql -u root -p
   Enter password:
   ```

2. Create a database for the Activity Monitor. The database name, user name, and password can be anything you want. For example:

   ```
   mysql> create database amon DEFAULT CHARACTER SET utf8;
   Query OK, 1 row affected (0.00 sec)

   mysql> grant all on amon.* TO 'amon'@'%' IDENTIFIED BY 'amon_password';
   Query OK, 0 rows affected (0.00 sec)
   ```

3. Create a database for the Service Monitor. The database name, user name, and password can be anything you want. For example:

   ```
   mysql> create database smon DEFAULT CHARACTER SET utf8;
   Query OK, 1 row affected (0.00 sec)

   mysql> grant all on smon.* TO 'smon'@'%' IDENTIFIED BY 'smon_password';
   Query OK, 0 rows affected (0.00 sec)
   ```

4. Create a database for the Report Manager. The database name, user name, and password can be anything you want. For example:

   ```
   mysql> create database rman DEFAULT CHARACTER SET utf8;
   Query OK, 1 row affected (0.00 sec)

   mysql> grant all on rman.* TO 'rman '@'%' IDENTIFIED BY 'rman_password';
   Query OK, 0 rows affected (0.00 sec)
   ```

5. Create a database for the Host Monitor. The database name, user name, and password can be anything you want. For example:

   ```
   mysql> create database hmon DEFAULT CHARACTER SET utf8;
   Query OK, 1 row affected (0.00 sec)

   mysql> grant all on hmon.* TO 'hmon'@'%' IDENTIFIED BY 'hmon_password';
   Query OK, 0 rows affected (0.00 sec)
   ```

6. Create a database for the Hive metastore. The database name, user name, and password can be anything you want. For example:

   ```
   mysql> create database hive DEFAULT CHARACTER SET utf8;
   Query OK, 1 row affected (0.00 sec)

   mysql> grant all on hive.* TO 'hive'@'%' IDENTIFIED BY 'hive_password';
   Query OK, 0 rows affected (0.00 sec)
   ```

7. Create a database for Cloudera Navigator. The database name, user name, and password can be anything you want. For example:

   ```
   mysql> create database nav DEFAULT CHARACTER SET utf8;
   Query OK, 1 row affected (0.00 sec)

   mysql> grant all on nav.* TO 'nav'@'%' IDENTIFIED BY 'nav_password';
   Query OK, 0 rows affected (0.00 sec)
   ```

Backing Up the MySQL Databases

To back up the MySQL database, run the `mysqldump` command on the MySQL host, as follows:

```
$ mysqldump -h<hostname> -u<username> -p<password> <database> >
/tmp/<database-backup>.sql
```

For example, to back up database `scm_database` on the local host as the root user, with the password `mypasswd`:

```
$ mysqldump -pmypasswd scm_database > /tmp/scm_database-backup.sql
```

To back up database `scm_database` on remote host `myhost.example.com` as the root user, with the password `mypasswd`:

```
$ mysqldump -hmyhost.example.com -uroot -pcloudera scm_database >
/tmp/scm_database-backup.sql
```

Installing and Configuring an External PostgreSQL Database

Use the following instructions to install PostgreSQL and set up a database on the appropriate hosts. It's useful to set a password for the root user of PostgreSQL. Note the host name and port number where you install PostgreSQL because you will need to specify them when you install the JDBC connector to PostgreSQL in a later step. Note that PostgreSQL does not have an accepted default port. You must determine the port used in your environment. You will also need to create a PostgreSQL database and user account for Cloudera Manager to use to store data. See your PostgreSQL documentation for more information about installation and configuration.

To install PostgreSQL on a Red Hat system:

```
$ sudo yum install postgresql-server
```

To install PostgreSQL on a SLES system:

```
$ sudo zypper install postgresql-server
```

To install PostgreSQL on an Ubuntu system:

```
$ sudo apt-get install postgresql
```

Configuring Your Systems to Support PostgreSQL

You must configure the PostgreSQL database to run as expected. This involves:

- Configuring PostgreSQL to accept network connections.
- Initializing the database to work with Cloudera Manager.
- Configuring the operating system to start PostgreSQL.

Configuring PostgreSQL to accept network connections By default, PostgreSQL only accepts connections on the loopback interface. Remember to reconfigure PostgreSQL to accept connections from the Fully Qualified Domain Name (FQDN) of the machines hosting the management roles. If you do not make these changes, the management processes will not be able to connect to and use the database on which they depend.

Initializing and configuring the external PostgreSQL database

1. Prepare the external PostgreSQL database for use with the Cloudera Manager Server.
   - On Red Hat and SLES systems:

     ```
     $ sudo service postgresql initdb
     ```

   - On Debian/Ubuntu systems:

     ```
     $ sudo service postgresql initdb
     ```

2. Enable MD5 authentication. Edit `pg_hba.conf`, which is usually found in `/var/lib/pgsql/data` or `/etc/postgresql/8.4/main`. Add the following line:

   ```
   host all all 127.0.0.1/32 md5
   ```

   Add this line before another line in the configuration file that references ident authentication. You can modify the contents of the previous line to support different configurations. For example, if you want to access PostgreSQL from a different host, replace 127.0.0.1 with your IP address and update `postgresql.conf`, which is typically found in the same place as `pg_hba.conf` to include:

   ```
   listen_addresses = '*'
   ```

3. Start the PostgreSQL database.
   - On Red Hat and SLES systems:

     ```
     $ sudo service postgresql start
     ```

   - On Debian/Ubuntu systems:

     ```
     $ sudo service postgresql start
     ```

4. Configure the PostgreSQL server to start at boot.
   - On Red Hat systems:

     ```
     $ sudo /sbin/chkconfig postgresql on
     $ sudo /sbin/chkconfig --list postgresql
     postgresql          0:off   1:off   2:on    3:on    4:on    5:on    6:off
     ```

   - On SLES systems:

     ```
     $ sudo chkconfig --add postgresql
     ```

- On Debian/Ubuntu systems:

```
$ sudo chkconfig postgresql on
```

Creating the PostgreSQL Databases for Cloudera Manager

The next step involves creating databases and user accounts for all database-backed services in Cloudera Manager.

Create databases for each of the following features that are part of the Cloudera Management Services:

- Activity Monitor
- Service Monitor
- Report Manager
- Host Monitor
- Cloudera Navigator

In addition add a database for each Hive service.

You can create these databases on the host where the Cloudera Manager Server will run, or on any other nodes in the cluster. For performance reasons, you should typically install each database on the host on which the service runs, as determined by the roles you will assign during installation or upgrade. In larger deployments or in cases where database administrators (DBAs) are managing the databases the services will use, databases may be separated from services, but do not undertake such an implementation lightly.

Note the values you enter for database names, user names, and passwords. The Cloudera Manager installation wizard requires this information to correctly connect to these databases.

The database must be configured to support UTF-8 character set encoding. The sample commands below include the required options to enable UTF-8 support.

To create the PostgreSQL Databases for Cloudera Manager:

1. Connect to PostgreSQL.

```
$ sudo -u postgres psql
```

2. Create a database for the Activity Monitor feature and assign permissions to a database user. The database name, user name, and password can be anything you want.

```
postgres=# CREATE ROLE amon LOGIN PASSWORD 'amon_password';
postgres=# CREATE DATABASE amon OWNER amon ENCODING 'UTF8';
```

3. Create a database for the Service Monitor feature and assign permissions to a database user. The database name, user name, and password can be anything you want.

```
postgres=# CREATE ROLE smon LOGIN PASSWORD 'smon_password';
postgres=# CREATE DATABASE smon OWNER smon ENCODING 'UTF8';
```

4. Create a database for the Report Manager feature and assign permissions to a database user. The database name, user name, and password can be anything you want.

```
postgres=# CREATE ROLE rman LOGIN PASSWORD 'rman_password';
postgres=# CREATE DATABASE rman OWNER rman ENCODING 'UTF8';
```

5. Create a database for the Host Monitor feature and assign permissions to a database user. The database name, user name, and password can be anything you want.

```
postgres=# CREATE ROLE hmon LOGIN PASSWORD 'hmon_password';
postgres=# CREATE DATABASE hmon OWNER hmon ENCODING 'UTF8';
```

6. Create a database for the Hive Metastore feature and assign permissions to a database user. The database name, user name, and password can be anything you want.

   ```
   postgres=# CREATE ROLE hive LOGIN PASSWORD 'hive_password';
   postgres=# CREATE DATABASE hive OWNER hive ENCODING 'UTF8';
   ```

   For PostgreSQL 8.2.23 or later, also do:

   ```
   postgres=# ALTER DATABASE hive SET standard_conforming_strings = off;
   ```

7. Create a database for the Cloudera Navigator feature and assign permissions to a database user. The database name, user name, and password can be anything you want.

   ```
   postgres=# CREATE ROLE nav LOGIN PASSWORD 'nav_password';
   postgres=# CREATE DATABASE nav OWNER nav ENCODING 'UTF8';
   ```

Configuring PostgreSQL Settings

There are several settings you should update to ensure your system performs as expected. Update these settings in the `/etc/postgresql.conf` file. Settings vary based on cluster size and resources.

Large Clusters

Large clusters may contain up to 1000 hosts. For large clusters consider the following suggestions as a starting point for settings.

- `max_connection`: For large clusters, each database is typically hosted on a different machine. The general rule is to allow each database on a host 100 maximum connections and then add 50 extra connections. As a result, in the normal case for large clusters, configure each of the five machines that hosts a single database for 150 connections. You may have to increase the system resources available to PostgreSQL, as described at Connection Settings.
- `shared_buffers`: 1024MB. Note that this requires that the operating system can allocate sufficient shared memory. See Postgres information on Managing Kernel Resources for more information on setting kernel resources.
- `wal_buffers`: 16MB. This value is derived from the `shared_buffers` value. Setting `wal_buffers` to be approximately 3% of `shared_buffers` up to a maximum of approximately 16MB works well in most case.
- `checkpoint_segments`: 128. The PostgreSQL Tuning Guide recommends values between 32 and 256 for write-intensive systems, such as this one.
- `checkpoint_completion_target`:0.9. This setting is only available in PostgreSQL 8.3 and later. These versions are highly recommended.

Small to Mid-sized Clusters

For small to mid-sized clusters, consider the following suggestions as a starting point for settings. If resources are especially limited, consider reducing the buffer sizes and checkpoint segments further. Ongoing tuning may be required based on each machine's resource utilization. For example, if Cloudera Manager is running on the same machine as other roles, the following values may be acceptable:

- `shared_buffers`: 256MB
- `wal_buffers`: 8MB
- `checkpoint_segments`: 16
- `checkpoint_completion_target`: 0.9

Configuration Settings for Postgres 8.1

Cloudera recommends using PostgreSQL 8.4 or later. While more recent versions provide better results, earlier versions may be used. For example, Cloudera supports PostgreSQL 8.1, which is bundled with some older Linux distributions. If you use PostgreSQL 8.1, settings such as `checkpoint_completion_target` are not available. Consequently, consider using the following recommended settings:

- `shared_buffers`: 131072
- `wal_buffers`: 4096
- `checkpoint_segments`: 256

Note that because PostgreSQL 8.1 does not support entering parameters in MB, the preceding values are provided in buffers or segments. For example, each buffer is 8KB, so 131072 is equivalent to 1024 MB.

After updating database settings, you must restart PostgreSQL for the new settings to take effect.

Restarting PostgreSQL

After making database configuration changes, you must restart the database for the changes to be applied.

To restart PostgreSQL:

```
$ pg_ctl restart
```

Backing up the Databases

Cloudera recommends that you periodically back up the databases that Cloudera Manager uses to store configuration, monitoring, and reporting data. Be sure to back all of the databases you are using with Cloudera Manager:

- Cloudera Manager database: Contains all the information about what services you have configured, their role assignments, all configuration history, commands, users, and running processes. This is a relatively small database (<100MB), and is the most important to back up.
- Activity Monitor database: Contains information about past activities. In large clusters, this database can grow large.
- Service Monitor database: Contains monitoring information about daemons. In large clusters, this database can grow large.
- Report Manager database: Keeps track of disk utilization over time. Medium-sized.
- Host Manager database: Contains information about host status. In large clusters, this database can grow large.
- Cloudera Navigator database: Contains auditing information. In large clusters, this database can grow large.
- Hive Metastore database: Contains Hive metadata. Relatively small.

Backing Up the PostgreSQL Database

It's important that you periodically back up the external or embedded PostgreSQL database that Cloudera Manager uses to store configuration information.

To back up the embedded PostgreSQL database, you can simply backup the `/var/lib/cloudera-scm-server-db` directory.

To back up the external PostgreSQL database, you can use the pg_dump utility.

To use the pg_dump utility:

1. Log in to the host where the Cloudera Manager Server is installed.
2. Run the following command as root:

```
cat /etc/cloudera-scm-server/db.properties.
The db.properties file contains:
# Auto-generated by scm_prepare_database.sh
# Mon Jul 27 22:36:36 PDT 2011
com.cloudera.cmf.db.type=postgresql
com.cloudera.cmf.db.host=localhost:7432
com.cloudera.cmf.db.name=scm
com.cloudera.cmf.db.user=scm
com.cloudera.cmf.db.password=NnYfWIjlbk
```

3. Run the following command as root using the parameters from the preceding step:

```
# pg_dump -h localhost -p 7432 -U scm > /tmp/scm_server_db_backup.$(date +%Y%m%d)
```

4. Enter the password specified for the `com.cloudera.cmf.db.password` property on the last line of the `db.properties file`. If you are using the embedded database, Cloudera Manager generated the password for you during installation. If you are using an external database, enter the appropriate information for your database.

For more information about using the pg_dump utility, see the pg_dump reference.

Installing an Embedded PostgreSQL Database

You can complete a manual installation that uses an embedded PostgreSQL server. If you are using Installation Path A - Automated Installation by Cloudera Manager, an embedded PostgreSQL database is automatically installed, so you do not need to complete this procedure. If, however, you are using Installation Path B - Installation Using Your Own Method, and you want to use an embedded PostgreSQL database, use this procedure.

Use this procedure to manually install PostgreSQL and set up an embedded PostgreSQL database on the appropriate hosts. Note the host name and port number where you install PostgreSQL because you will need to specify them when you install the JDBC connector. Note that PostgreSQL does not have an accepted default port. You must determine the port used in your environment. You will also need to create a PostgreSQL database and user account for Cloudera Manager to use to store data. See your PostgreSQL documentation for more information about installation and configuration.

To install the embedded PostgreSQL database package on the Cloudera Manager Server host: On a Red Hat system if you have a yum repo configured:

```
$ sudo yum install cloudera-manager-server-db
```

On a Red Hat system if you're transferring RPMs manually:

```
$ sudo yum --nogpgcheck localinstall cloudera-manager-server-db.noarch.rpm
```

On a SUSE system:

```
$ sudo zypper install cloudera-manager-server-db
```

On a Debian/Ubuntu system:

```
$ sudo apt-get install cloudera-manager-server-db
```

Using an Oracle Database

To use an Oracle database, several conditions must be met.

- You should collect information about the Oracle database you will use.
- You should install the Oracle JDBC.
- For larger CDH clusters, adjust Oracle settings or ask your DBA to do this for you.
- Ensure your Oracle database supports UTF8 character set encoding.

Collect Oracle Database Information

Installing, configuring, and maintaining an Oracle database should be completed by your organization's database administrator. In preparation for configuring Cloudera Manager to work with Oracle databases, gather the following information from your Oracle DBA:

- Host Name - The DNS name or the IP address of the host where the Oracle database is installed.
- SID - the name of the database that will store Cloudera Manager information. This database could contain schemas that would store information for the Cloudera Manager Server, Activity Monitor, Service Monitor, Report Manager, Host Monitor, and Cloudera Navigator.
- User name - a user name for each schema that is storing information. This means you might have six unique usernames for the six schemas.
- Password - a password corresponding to each user name.

You will use the Oracle database information that you have gathered to configure the external database to work with the Cloudera Manager Server.

Installing the Oracle JDBC Connector

You must install the JDBC connector on the Cloudera Manager Server host, as well as hosts to which you assign the Activity Monitor, Service Monitor, Report Manager, Host Monitor, Cloudera Navigator, and Hive Metastore Server roles.

Cloudera recommends that you assign all roles that require databases on the same host and install the connector on that host. While putting all such roles on the same host is recommended, it is not required. You could install a role, such as Activity Monitor on one host and other roles on a separate host. In such a case you would install the JDBC connector on each host running roles that access the database.

Download and install the ojdbc6.jar file, which contains the JDBC driver. There are different versions of the ojdbc6.jar file. You must download the version that is designed for:

- Java 6
- The Oracle database version used in your environment For example, for an environment using Oracle 11g R2, the jar file can be downloaded from http://www.oracle.com/technetwork/database/enterprise-edition/jdbc-112010-090769.html.

Copy the appropriate JDBC `jar` file to `/usr/share/cmf/lib/ojdbc6.jar` for use with the Cloudera Manager databases (e.g. for the Activity Monitor etc.), and to `/usr/share/java/oracle-connector-java.jar` for use with Hive.

```
$ mkdir /usr/share/java (if necessary)
$ cp /tmp/ojdbc6.jar /usr/share/java/oracle-connector-java.jar
$ cp /tmp/ojdbc6.jar /usr/share/cmf/lib/ojdbc6.jar
```

Adjust Oracle Settings to Accommodate Larger Clusters

Cloudera Management services require high write throughput. Depending on the size of your deployments, your DBA may need to modify Oracle settings for monitoring services. Note that these guidelines are for larger clusters and do not apply to Cloudera Manager configuration database and to smaller clusters. Many factors contribute to whether to reconfigure your database settings, but in most cases, if your cluster has more than 100 hosts, you should consider making the following changes:

- Enable direct and asynchronous I/O by setting the FILESYSTEMIO_OPTIONS parameter to SETALL.
- Increase the RAM available to Oracle by changing the MEMORY_TARGET parameter. The amount of memory to assign depends on the size of Hadoop cluster.
- Create more redo log groups and spread the redo log members across separate disks/LUNs.
- Increase the size of redo log members to be at least 1 gigabyte.

Adjust Oracle System Settings for Sufficient Database Connectivity

Work with your Oracle database administrator to ensure appropriate values are applied for your Oracle database settings. You must determine the number of connections, transactions, and sessions to be allowed. Allow 100 maximum connections for each database and then add 50 extra connections. For example, for two databases set the maximum connections to 250. If you store seven databases on one host (the databases for Activity

Monitor, Service Monitor, Report Manager, Host Monitor, Cloudera Manager Server, Cloudera Navigator, and Hive Metastore), set the maximum connections to 750.

From the maximum number of connections, you can determine the number of anticipated sessions using the following formula:

```
sessions = (1.1 * maximum_connections) + 5
```

For example, if a host has two databases, you anticipate 250 maximum connections. If you anticipate a maximum of 250 connections, plan for 280 sessions.

Once you know the number of sessions, you can determine the number of anticipated transactions using the following formula:

```
transactions = 1.1 * sessions
```

Continuing with the previous example, if you anticipate 280 sessions, you can plan for 308 transactions.

Work with your Oracle database administrator to apply these derived values to your system.

Using the sample values above, Oracle attributes would be set as follows:

```
alter system set processes=250;
alter system set transactions=308;
alter system set sessions=280;
```

Ensure your Oracle Database Supports UTF8

The database you use must be configured to support UTF8 character set encoding. One way your DBA might implement UTF8 character set encoding in Oracle databases is using the `dbca` utility. In such a case, when creating a database, the `characterSet AL32UTF8` option might be used to specify proper encoding. Consult with your DBA to ensure UTF8 encoding is properly configured.

Having collected information about your Oracle database, installed the Oracle JDBC, considered having database settings adjusted, and ensured UTF-8 encoding is enabled, proceed to Install CDH and Cloudera Manager.

Installing CDH and Cloudera Manager

There are several paths to installing this version of CDH and Cloudera Manager:

- Installation Path A: Automated installation of Cloudera Manager and CDH through an installation wizard
- Installation Path B: Installation of Cloudera Manager and CDH using your own method
- Installation Path C: Installation of Cloudera Manager using tarballs
- Upgrading Cloudera Manager from a previous version

> **Important:**
>
> Follow the instructions for only one of the installation paths.

Any database type may be used with any of the paths.

Installation Path A

In Installation Path A, Cloudera Manager can automate the installation of databases for Cloudera Manager and its services, the packages for CDH, Cloudera Manager, and the Oracle JDK. This option is available if your cluster deployment meets the following requirements:

- Uniform SSH access to cluster hosts on the same port from Cloudera Manager Server host.
- All hosts must have access to standard package repositories.
- All hosts must have access to the either `archive.cloudera.com` on the internet or to a local repository with the necessary installation files.

For Installation Path A instructions, click here.

Installation Path B

Follow Installation Path B if your cluster does not meet the requirements for Installation Path A, or if you want or need to manage the installation of the CDH, Cloudera Manager, and the Oracle JDK packages yourself using whatever method you currently use to install software and configuration files on your cluster hosts.

For Installation Path B instructions, click here.

Installation Path C

Follow Installation Path C to install Cloudera Manager using tarballs provided by Cloudera.

For Installation Path C instructions, click here.

Upgrading to Cloudera Manager and CDH

Follow one of the Upgrade Paths if your cluster already has an installation of a previous version of the Cloudera Management Suite or SCM Express Edition.

To upgrade to Cloudera Manager 4.5, see Upgrading to Cloudera Manager 4.5.

You can also upgrade CDH from within Cloudera Manager 4.5: see Upgrading CDH in a Cloudera Manager Deployment

Installation Path A - Automated Installation by Cloudera Manager

If your cluster meets the requirements for Installation Path A, follow the instructions in this section for automated installation by Cloudera Manager. The requirements for Path A are:

- Uniform SSH access to cluster hosts on the same port from Cloudera Manager Server host.
- All hosts must have access to standard package repositories.
- All hosts must have access to the either `archive.cloudera.com` on the internet or to a local repository with the necessary installation files.

The Cloudera Manager configuration, as well as the other monitoring and management information is stored in databases. As part of the process of Installation Path A, Cloudera Manager installs embedded PostgreSQL databases. It is simplest to use these automatically installed and configured databases. During the installation, you are provided with the option to select databases other than the automatically installed databases. If you intended to customize the installation to use other databases, install and configure them before beginning to use Installation Path A.

Using custom databases is a more advanced process, which is more often a part of an Installation Using Your Own Method. For more information on installing custom databases, see Installing and Configuring Databases. Otherwise, use the embedded PostgreSQL database, which the installer creates.

The general steps in this procedure for Installation Path A are:

- Step 1: Download and Run the Cloudera Manager Installer
- Step 2: Start the Cloudera Manager Admin Console
- Step 3: Use Cloudera Manager for Automated CDH Installation and Configuration
- Step 4: Change the Default Administrator Password
- Step 5: Test the Installation

Step 1: Download and Run the Cloudera Manager Installer

Important:

For installation purposes, the Cloudera Manager Server must have SSH access to the cluster hosts and you must log in using a root account or an account that has password-less sudo permission. See Requirements for Cloudera Manager for more information.

Cloudera Manager accesses `archive.cloudera.com` by using yum on Red Hat systems, zypper on SUSE systems, or apt-get on Debian/Ubuntu systems. If your hosts access the Internet through an HTTP Proxy, you can configure yum, zypper, or apt-get, system-wide, to access archive.cloudera.com through a proxy. To do so, modify the system configuration on the Cloudera Manager Server host and on every cluster host where you want to install CDH. This is not required in all cases.

To configure your system to use a proxy

On Red Hat systems, add the following property to `/etc/yum.conf`:

```
proxy=http://server:port/
```

On SUSE systems, add the following property to `/root/.curlrc`:

```
--proxy=http://server:port/
```

On Debian/Ubuntu systems, add the following property to `/etc/apt/apt.conf`:

```
Acquire::http::Proxy "http://server:port";
```

To download and run the Cloudera Manager installer:

1. Download `cloudera-manager-installer.bin` from the [Cloudera Downloads page]() to the host where you want to install the Cloudera Manager Server that is on your cluster or is accessible to your cluster over your network. Install Cloudera Manager on a single host.
2. After downloading `cloudera-manager-installer.bin`, change it to have executable permission.

   ```
   $ chmod u+x cloudera-manager-installer.bin
   ```

3. Run `cloudera-manager-installer.bin`.

   > **Note:**
   >
   > This installer's default behavior is to install packages from the Internet. If you have created a local repository and configured your machine to recognize that repository, you can instruct the installer to use local repositories by running the `cloudera-manager-installer.bin` with the `--skip_repo_package=1` option.

   ```
   $ sudo ./cloudera-manager-installer.bin
   ```

4. Read the Cloudera Manager Readme and then press **Enter** to choose **Next**.
5. Read the Cloudera Manager License and then press **Enter** to choose **Next**. Use the arrow keys and press **Enter** to choose **Yes** to confirm you accept the license.
6. Read the Oracle Binary Code License Agreement and then press **Enter** to choose **Next**. Use the arrow keys and press **Enter** to choose **Yes** to confirm you accept the Oracle Binary Code License Agreement. The Cloudera Manager installer begins installing the Oracle JDK and the Cloudera Manager repo files and then installs the packages. The installer also installs the Cloudera Manager Server.

   > **Note:**
   >
   > If an error message "Failed to start server" appears while running `cloudera-manager-installer.bin`, exit the installation program. If the Cloudera Manager Server log file `/var/log/cloudera-scm-server/cloudera-scm-server.log` contains the following message, then it's likely you have SELinux enabled
   >
   > ```
   > Caused by: java.lang.ClassNotFoundException: com.mysql.jdbc.Driver
   >         at java.net.URLClassLoader$1.run(Unknown Source)
   >         at java.security.AccessController.doPrivileged(Native Method)
   >         at java.net.URLClassLoader.findClass(Unknown Source)
   >         at java.lang.ClassLoader.loadClass(Unknown Source)
   >         ...
   > ```
   >
   > You can disable SELinux by running the following command on the Cloudera Manager Server host:
   >
   > ```
   > $ sudo setenforce 0
   > ```
   >
   > To disable it permanently, edit `/etc/selinux/config`.

7. Note the complete URL provided for the Cloudera Manager Admin Console, including the port number, which is 7180 by default. Click **OK** to continue.
8. Click **OK** to exit the installer.

> **Note:**
>
> If the installation is interrupted for some reason, you may need to clean up before you can re-run it. See [Uninstalling Cloudera Manager]().

Step 2: Start the Cloudera Manager Admin Console

The Cloudera Manager Admin Console enables you to use Cloudera Manager to configure, manage, and monitor Hadoop on your cluster. Before using the Cloudera Manager Admin Console, gather information about the server's URL and port.

The server URL takes the following form:

```
http://<Server host>:<port>
```

`<Server host>` is the fully-qualified domain name or IP address of the host machine where the Cloudera Manager Server is installed. `<port>` is the port configured for the Cloudera Manager Server. The default port is 7180. For example, use a URL such as the following:

```
http://myhost.example.com:7180/
```

Cloudera Manager does not support changing the `admin` username for the installed account. You can change the password using Cloudera Manager after you run the wizard in the next section. While you cannot change the `admin` username, you can add a new user, assign administrative privileges to the new user, and then delete the default `admin` account.

To start the Cloudera Manager Admin Console:

In a web browser, enter the URL, including the port, for the Cloudera Server. The login screen for Cloudera Manager appears.

Log into Cloudera Manager. The default credentials are: **Username:** `admin` **Password:** `admin`

Step 3: Use Cloudera Manager for Automated CDH Installation and Configuration

The following instructions show you how to use the Cloudera Manager wizard to do an initial installation and configuration. The wizard helps you to install and set up Cloudera packages across your cluster and will:

- Install and validate your Cloudera Manager License
- Find the cluster hosts you specify via hostname and IP-address ranges
- Connect to each host with SSH to install the Cloudera Manager Agent and CDH (including Hue)
- Install the Oracle JDK on the cluster hosts (if not already installed)
- Install CDH packages or parcels, optionally including the Cloudera Impala package or parcel
- Configure Hadoop automatically and start the Hadoop services

> **Important:**
>
> All hosts in the cluster must have some way to access installation files. This can be done one of two ways:
>
> - Internet access to allow the wizard to install software packages or parcels from `archive.cloudera.com`.
> - An internal repository that each host can access. For example, for Red Hat hosts, you could set up a Yum repository. See Creating and Using your own Repository for more information.

To use Cloudera Manager:

1. The first time you start the Cloudera Manager Admin Console, the install wizard starts up.
2. Browse to your Cloudera Manager License file. If you don't install the license now, Cloudera Manager Free Edition will be installed.

> **Note:**
>
> The instructions that follow assume you have installed a Cloudera Manager license. If you are not yet ready to install a Cloudera Manager license, and want to proceed with a Free Edition installation, stop here and use the Cloudera Manager Free Edition Installation Guide instead. If you install the Free Edition, and later need to upgrade to the full version of Cloudera Manager, follow the instructions under Upgrading from Cloudera Manager Free Edition 4.5 to the Cloudera Manager Enterprise Edition.

3. After you install the Cloudera Manager license, restart the Cloudera Manager server. **On Red Hat/CentOS/SUSE systems:**

   ```
   $ sudo service cloudera-scm-server restart
   ```

   On Debian/Ubuntu systems:

   ```
   $ sudo service cloudera-scm-server restart
   ```

4. After the Cloudera Manager server restarts, use your web browser to connect to the Cloudera Manager Admin Console URL again and log in, as described in Step 2.

   > **Note:**
   >
   > After restarting the server, wait a few seconds for the server to finish initializing before you try to reconnect to the Admin Console.

5. Information is displayed indicating what the CDH installation includes. Click **Continue**.
6. To enable Cloudera Manager to automatically discover your cluster hosts where you want to install CDH, enter the cluster hostnames or IP addresses. You can also specify hostname and IP address ranges: For example:

Use this Expansion Range	To Specify these Hosts
10.1.1.[1-4]	10.1.1.1, 10.1.1.2, 10.1.1.3, 10.1.1.4
host[1-3].company.com	host1.company.com, host2.company.com, host3.company.com
host[07-10].company.com	host07.company.com, host08.company.com, host09.company.com, host10.company.com

   You can specify multiple addresses and address ranges by separating them by commas, semicolons, tabs, or blank spaces, or by placing them on separate lines. Use this technique to make more specific searches instead of searching overly wide ranges. The scan results will include all addresses scanned, but only scans that reach hosts running SSH will be selected for inclusion in your cluster by default.

   > **Note:**
   >
   > If you don't know the IP addresses of all of the hosts, you can enter an address range that spans over unused addresses and then deselect the hosts that do not exist (and are not discovered) later in this procedure. However, keep in mind that wider ranges will require more time to scan.

7. Click **Search**. Cloudera Manager identifies the hosts on your cluster to allow you to configure them for CDH. If there are a large number of hosts on your cluster, wait a few moments to allow them to be discovered and shown in the wizard. If the search is taking too long, you can stop the scan by clicking **Abort Scan**. To find additional hosts, add their host name or IP address and click **Search** again.

Note:

Cloudera Manager scans hosts by checking for network connectivity. If there are some hosts where you want to install CDH that are not shown in the list, make sure you have network connectivity between the Cloudera Manager Server host and those hosts. Common causes of loss of connectivity are firewalls and interference from SELinux.

8. Verify that the number of hosts shown matches the number of hosts where you want to install CDH. Deselect host entries that do not exist and deselect the hosts where you do not want to install CDH. Click **Install CDH On Selected Hosts.**

   Click **Continue**

9. Select the repository type you want to use for the installation.

   - To install using **Parcels**, select Parcels, and follow the directions at Installation using Parcels.
   - To install using **Packages**, select Packages, and follow the directions at Installation using Packages.

   Installing from parcels is recommended, if they are available for the version you want to install.

Note:

Parcels are available for CDH4.1.2 or later, and for Impala. To install CDH3 or to install an earlier version of CDH4, select **Packages**.

Installation using Parcels

1. Choose the parcel you want to install. The choices you see depend on the repositories you have chosen – a repository may contain multiple parcels.

   If you have parcels in a custom repository, you can specify the repository and Cloudera Manager will add those parcels to the list shown on this page.

   a. Click **More Options** to show the custom repository field.
   b. Enter the URL of the repository you want into the field provided, and click the **+ Add** button. The URL you specify here will also be added to the list of remote repositories referenced in the **Remote Parcel Repository URLs** property. If you have multiple repositories configured, you will see all the unique parcels contained in all your repositories.

2. Select the specific release of Cloudera Manager to install on your hosts. You may choose either the version that matches with the Cloudera Manager Server you are currently using, or you can specify an installation from a custom repository.

3. If available, select the specific release of Impala to install on your hosts. You may choose either the latest version or use a custom repository. If you do not want to install Impala, select **None**.

4. If you opted to use custom repositories for installation files, you may provide a GPG key URL that will apply for all repositories.

5. Click **Continue**. You are now asked to provide your credentials, following the instructions at Provide credentials for authenticating with hosts.

Installation using Packages

1. Choose the CDH version to install.
2. Select the major release of CDH to install. This is often CDH4.
3. Select the specific release of CDH to install from within the major version you selected. You may choose a custom repository.

4. Select the specific release of Impala to install on your hosts. You may choose either the latest version or use a custom repository.
5. Select the specific release of Cloudera Manager to install on your hosts. You may choose either the version that matches with the Cloudera Manager Server you are currently using or you can specify an installation at a custom repository.
6. If you opted to use custom repositories for installation files, you may provide a GPG key URL that will apply for all repositories.
7. Click **Continue**.

Provide credentials for authenticating with hosts

1. Select **root** or enter the user name for an account that has password-less sudo permissions.
2. Select an authentication method.
   - If you choose to use password authentication, enter and confirm the password.
   - If you choose to use public-key authentication provide a passphrase and path to the required key files.
   - You can choose to specify an alternate SSH port. The default value is 22.
   - You can specify the maximum number of host installations to run at once. The default value is 10.
3. Click **Continue** to begin installing the Cloudera Manager Agent and Daemons on the cluster hosts. If you are installing from packages, the process also installs CDH (and Impala, if you've selected it) on your hosts.

Install Cloudera Manager and CDH components

The status of installation on each host is displayed in the following screen. The Cloudera Manager wizard uses SSH to access the cluster hosts and follows a sequence of steps to download and install the Oracle JDK, Cloudera Manager Agents and Daemons. If you are installing from packages rather than parcels, CDH is also installed at this step. .

> **Note:**
>
> Clicking **Abort Installation** while installation is in progress halts any pending or in-progress installations and rolls back any in-progress installations to a clean state. Clicking **Abort Installation** does not affect completed or failed host installations.

If installation fails on a host, you can click the **Uninstall** link next to the failed host. This will give you the choice of uninstalling the failed hosts, or to try installation on that host again. To uninstall, click **Uninstall Failed Hosts**. To retry installation on all failed hosts, click **Retry Failed Hosts**.

To avoid excessive network load, the wizard runs a limited number of installations in parallel, based on the value indicated on the page where you provided your authentication credentials. The default is 10 simultaneous installations.

1. **If you are installing from packages**, the wizard configures package repositories, installs the Oracle JDK, CDH, and the Cloudera Manager Agent, and then starts the Cloudera Manager Agent. The status of installation on each host is displayed. You can also click the **Details** link for individual hosts to view detailed information about the installation and error messages if installation fails on any hosts.
   a. When the **Continue** button appears at the bottom of the screen, the installation process is completed. If the installation has completed successfully on some hosts but failed on others, you can click **Continue** if you want to skip installation on the failed hosts and continue to the next screen to start configuring CDH on the successful hosts.
2. **If you are installing from parcels**, the wizard installs the Oracle JDK and the Cloudera Manager Agent using packages, as described above. The status of installation on each host is displayed.
   a. When the Cloudera Manager Agent, the JDK etc. have been installed, click **Continue** to proceed to the cluster installation section. During the parcel installation, progress is indicated for the three phases of

the parcel installation process (Download, Distribution, and Activation) in a single progress bar. If you are installing multiple parcels (e.g. CDH and Impala) you will see a progress bar for each parcel.

   b. When the **Continue** button appears at the bottom of the screen, the installation process is completed.

3. When you continue, the Host Inspector runs to validate the installation, and provides a summary of what it finds, including all the versions of the installed components. If the validation is successful, click **Continue**.

Choose the services you want to start on your cluster

1. Choose which version of CDH to use.
2. Choose the combination of services to install: Core Hadoop, Real-Time Delivery (previously known as HBase Services), Real-Time Query (which includes HDFS, Hive and Impala), All Services, or Custom Services.

   > **Note:**
   >
   > - Some services depend on others; for example, HBase requires HDFS and ZooKeeper.
   > - Most of the combinations install MapReduce v1. Choose the **Custom Services** option to install MapReduce v2 (YARN) or use the Add Service functionality to add YARN after installation completes.

3. Choose whether to install Cloudera Navigator. Cloudera Navigator is independently licensed from the core Cloudera Enterprise offering.
4. Click **Inspect Role Assignments** to see how the wizard will assign roles for the services you have chosen, and change them if you need to. These assignments are typically acceptable, but you can reassign services to nodes of your choosing, if desired. The wizard evaluates the hardware configurations of the cluster hosts to determine the best machines for each role. For example, the wizard assigns the NameNode role to the machine that best meets the NameNode requirements. The wizard also configures other options, such as the number of map and reduce slots for TaskTracker, on the basis of the size of the cluster and the physical characteristics of each machines, such as the number of CPUs, amount of RAM, and disk space. These assignments are typically acceptable, but you can reassign services to nodes of your choosing, if desired.
5. Click **Continue** when you are satisfied with the assignments.
6. On the Database Setup page, configure settings for the Activity Monitor, Service Monitor, Report Manager, Host Monitor, and Hive metastore databases.

   - Leave the default settings of **Use Embedded Database** to have Cloudera Manager create and configure all required databases.
   - Select **Custom** to specify external databases, and enter the required information for the databases that you created when you set up your databases for Cloudera Manager. You must provide the Database host, database type, database name, username, and password.
   - Click **Test Connection** to confirm that Cloudera Manager can communicate with the databases using the information you have supplied. If the test succeeds in all cases, click **Continue**; otherwise check and correct the information you have provided for the databases and then try the test again. (Note that for Hive, if you are using the embedded database, you may see a message saying the connection will be created at a later point in the installation process.)

7. Review the Configuration Changes to be applied.

   Confirm the settings entered for file system paths. The file paths required vary based on the services to be installed. For example, you might confirm the NameNode Data Directory and the DataNode Data Directory for HDFS or confirm the TaskTracker Local Data Directory List or JobTracker Local Data Directory for MapReduce.

8. Click **Continue**. The wizard starts the services on your cluster.
9. When all of the services are started, click **Continue**. You will see a success message indicating that your cluster has been successfully started.
10. Click **Continue** to proceed to the Cloudera Manager Services page.

Step 4: Change the Default Administrator Password

As soon as possible after running the wizard and beginning to use Cloudera Manager, you should change the default administrator password.

To change the administrator password:

1. Click the gear icon ⚙ to display the **Administration** page.
2. Click the **Users** tab.
3. Click the **Change Password** button next to the **admin** account.
4. Enter a new password twice and then click **Submit**.

Step 5: Test the Installation

Now that you have finished with the CDH and Cloudera Manager installation, you are ready to test the installation. For testing instructions, see Testing the Installation.

> **Note:**
>
> If you change the hostname or port where the Cloudera Manager is running, or you enable TLS security, you must restart the Cloudera Management Services to update the URL to the Server. For instructions, see Restarting a Service.

Installation Path B - Installation Using Your Own Method

To manage the installation of packages yourself, follow the instructions in this section. If you have already installed Cloudera Manager and CDH, skip this section and continue on to other installation tasks. For example, if you have already completed a Path A installation, you might Specify the Racks for Hosts or Test the Installation next.

> **Note:**
>
> You can also use Puppet or Chef to install packages on your cluster.

The general steps in this procedure for Installation Path B are:

- Step 1: Establish Your Repository Strategy
- Step 2: Install CDH
- Step 3: Install the Cloudera Manager Server
- Step 4: Configure a Database for the Cloudera Manager Server
- Step 5: Install the Cloudera Manager Agents
- Step 6: Start the Cloudera Manager Server
- Step 7: Start the Cloudera Manager Agents
- Step 8: Start the Cloudera Manager Admin Console
- Step 9: Configure Services
- Step 10: Change the Default Administrator Password
- Step 11: Test the Installation

Before You Begin

Cloudera Manager and Cloudera Distribution of Hadoop (CDH) are comprised of a set of services. These services interact among each other and use databases to complete tasks. The parts that make up this system are very flexible, so you could deploy these services and resources in many different ways, though the process is greatly simplified by following Cloudera's installation and configuration guidelines.

Considering this, Cloudera recommends you begin by establishing a foundation of database resources that can be utilized as they become necessary throughout the installation process. Begin by deploying the necessary supporting services and then proceeding through the installation process.

Install the Oracle JDK

Install the Oracle Java Development Kit (JDK) on each of your cluster hosts where you want to run Hadoop before installing Cloudera's packages. Cloudera Manager can manage both CDH3 and CDH4 hosts, and the required JDK version varies accordingly.

- For installation instructions and recommendations for CDH3, see the topic on Java Development Kit Installation in the CDH3 Installation Guide.
- For installation instructions and recommendations for CDH4, see the topic on Java Development Kit Installation in the CDH4 Installation Guide.

Install Databases for the Cloudera Manager Services

Create and configure databases for the Cloudera Manager Monitoring Services (Activity Monitor, Service Monitor, Report Manager, and Host Monitor), and for the Hive Metastore and Cloudera Navigator. Cloudera supports various database solutions including the PostgreSQL embedded database, PostgreSQL external databases, MySQL databases, or Oracle databases.

Information about how these databases are set up in your environment is required to complete the CDH and Cloudera Manager configuration. The details of what is required varies among database types. Gather this information either as you complete the installations or from database administrators who have the information required. A list of what information is required for each database type is provided in each database section.

Follow the instructions at Installing and Configuring Databases to complete this task.

Database choices	Notes and Instructions
Option A: Embedded PostgreSQL	This is the same PostgreSQL application and database that the Cloudera Manager wizard installs. For installation and configuration instructions, see Installing an Embedded PostgreSQL Database.
Option B: External PostgreSQL	After PostgreSQL is installed, you need to run a script to prepare a database for the Cloudera Manager Server as described in Installing and Configuring an External PostgreSQL Database.
Option C: External MySQL	You can use the same MySQL application that is used for the monitoring and reporting features, as described in Installing and Configuring a MySQL Database. After MySQL is installed, you need to run a script to prepare a database for the Cloudera Manager Server, as is described later in this topic.
Option D: External Oracle	You can use an external Oracle database for monitoring and reporting features, as described in Using an Oracle Database.

Step 1: Establish Your Repository Strategy

Cloudera recommends installing products using package management tools such as `yum` for Red Hat Enterprise Linux (RHEL) or CentOS, `zypper` for SUSE, or `apt-get` for Debian/Ubuntu. These tools depend on access to repositories to install software. For example, Cloudera maintains Internet-accessible repositories for CDH and Cloudera Manager installation files. Strategies for installing CDH and Cloudera Manager include using:

- Standard Cloudera repositories. For this method, ensure you have added the required repository information to your systems. For Cloudera repository locations and client `.repo` files, see Cloudera Manager Version and Download Information.
- Internally hosted repositories. You might use internal repositories for environments where machines do not have access to the Internet. In such a case, ensure your environment is properly prepared. For more information, see Appendix A - Understanding Custom Installation Solutions.

> **Note:**
>
> The instructions that follow assume you are adding a CDH4 repository; if you are installing CDH3, modify the commands accordingly with the correct standard repository and repo file paths as found in Cloudera Manager Version and Download Information.

On Red Hat-compatible Systems

To add the CDH4 Yum repository:

Click the entry in the table at Cloudera Manager Version and Download Information that matches your Red Hat or CentOS system, navigate to the repo file (`cloudera-cdh4.repo`) for your system and save it in the `/etc/yum.repos.d/` directory.

To add the Cloudera Manager Yum repository: Navigate to the Cloudera Manager repo file (`cloudera-manager.repo`) for your system and save it in the `/etc/yum.repos.d/` directory.

For OS Version	Click this Link
Red Hat/CentOS/Oracle 5	http://archive.cloudera.com/cm4/redhat/5/x86_64/cm/cloudera-manager.repo
Red Hat/CentOS 6 (64-bit)	http://archive.cloudera.com/cm4/redhat/6/x86_64/cm/cloudera-manager.repo

To optionally add a Repository key:

- **For Red Hat/CentOS/Oracle 5 systems:**

```
$ sudo rpm --import
http://archive.cloudera.com/cdh4/redhat/5/x86_64/cdh/RPM-GPG-KEY-cloudera
```

- **For Red Hat/CentOS 6 systems:**

```
$ sudo rpm --import
http://archive.cloudera.com/cdh4/redhat/6/x86_64/cdh/RPM-GPG-KEY-cloudera
```

On SLES Systems

To add the CDH4 and Cloudera Manager repositories:

1. Run the following commands:

   ```
   $ sudo zypper addrepo -f
   http://archive.cloudera.com/cdh4/sles/11/x86_64/cdh/cloudera-cdh4.repo
   $ sudo zypper addrepo -f
   http://archive.cloudera.com/cm4/sles/11/x86_64/cm/cloudera-manager.repo
   ```

2. Update your system package index by running:

   ```
   $ sudo zypper refresh
   ```

To optionally add a Repository key:

```
$ sudo rpm --import
http://archive.cloudera.com/cdh4/sles/11/x86_64/cdh/RPM-GPG-KEY-cloudera
```

On Ubuntu or Debian Systems

To add the CDH4 and Cloudera Manager repositories:

Create a new file ccd with the following contents:

- For Ubuntu systems:

```

deb [arch=amd64] http://archive.cloudera.com/cdh4/<OS-release-arch-cdh> <RELEASE>-cdh4
 contrib
deb-src http://archive.cloudera.com/cdh4/<OS-release-arch-cdh> <RELEASE>-cdh4 contrib
deb http://archive.cloudera.com/cm4/<OS-release-arch-cm> <RELEASE>-cm4 contrib
deb-src http://archive.cloudera.com/cm4/<OS-release-arch-cm> <RELEASE>-cm4 contrib
```

- For Debian systems:

```

deb http://archive.cloudera.com/cdh4/<OS-release-arch-cdh> <RELEASE>-cdh4 contrib
deb-src http://archive.cloudera.com/cdh4/<OS-release-arch-cdh> <RELEASE>-cdh4 contrib
deb http://archive.cloudera.com/cm4/<OS-release-arch-cm> <RELEASE>-cm4 contrib
deb-src http://archive.cloudera.com/cm4/<OS-release-arch-cm> <RELEASE>-cm4 contrib
```

where: <*OS-release-arch-cdh*> is `debian/squeeze/amd64/cdh`, `ubuntu/lucid/amd64/cdh`, or `ubuntu/precise/amd64/cdh`, <*OS-release-arch-cm*> is `debian/squeeze/amd64/cm`, `ubuntu/lucid/amd64/cm`, or `ubuntu/precise/amd64/cm`, and <*RELEASE*> is the name of your distribution, which you can find by running `lsb_release -c`.

For example, to install CDH4 for 64-bit Ubuntu Lucid:

```
deb [arch=amd64] http://archive.cloudera.com/cdh4/ubuntu/lucid/amd64/cdh lucid-cdh4
contrib
deb-src http://archive.cloudera.com/cdh4/ubuntu/lucid/amd64/cdh lucid-cdh4 contrib
```

To optionally add a Repository key:

- **For Ubuntu Lucid systems:**

```
$ curl -s http://archive.cloudera.com/cdh4/ubuntu/lucid/amd64/cdh/archive.key | sudo
apt-key add -
```

- **For Ubuntu Precise systems:**

```
$ curl -s http://archive.cloudera.com/cdh4/ubuntu/precise/amd64/cdh/archive.key | sudo
 apt-key add -
```

- **For Debian Squeeze systems:**

```
$ curl -s http://archive.cloudera.com/cdh4/debian/squeeze/amd64/cdh/archive.key | sudo
 apt-key add -
```

Step 2: Install CDH

This section describes how to install CDH on RHEL, CentOS, SUSE, and Debian/Ubuntu systems. This installation is done in preparation for using Cloudera Manager to configure and manage your cluster. For information about installing CDH, see the CDH4 Installation Guide.

Important:

Cloudera Manager requires Hadoop to be installed on all hosts, but Hadoop must **not** be configured and must **not** be running.

Important:

The Activity Monitor in Cloudera Manager 4.0 requires the `hue-plugins` package to be installed on the JobTracker host, regardless of whether you are using Hue. If you are using Hue, the `hue-plugins` package must be installed on all hosts.

On Red Hat-compatible Systems (RHEL/CentOS/Oracle)

1. Install packages on every host in your cluster: **For CDH4:**

```
$ sudo yum -y install bigtop-utils bigtop-jsvc bigtop-tomcat hadoop hadoop-hdfs
hadoop-httpfs hadoop-mapreduce hadoop-yarn hadoop-client hadoop-0.20-mapreduce
hue-plugins hbase hive oozie oozie-client pig zookeeper
```

   For CDH3:

```
$ sudo yum -y install hadoop-0.20 hadoop-0.20-native.x86_64 hadoop-0.20-sbin.x86_64
 hue-plugins hadoop-zookeeper hadoop-hbase oozie oozie-client
```

2. (**Requires CDH 4.1 or later**) Install Impala and the Impala Shell on Impala machines:

```
$ sudo yum -y install impala impala-shell
```

3. To install the `hue-common` package and all Hue applications on the Hue machine, install the hue meta-package:

```
$ sudo yum install hue
```

4. Disable autostart for Hue on the Hue machine, and for Oozie on every machine on which it is installed.

   Note:

   Preventing Hue from starting is only required on CDH3u5 and earlier. If you are running CDH4 or a version of CDH3 newer than CDH3u5, you do not need to prevent Hue from starting. While stopping Hue from starting is not required in those cases, executing the `chkconfig` command generates an error, but there are no other negative effects.

```
$ sudo /sbin/chkconfig hue off
$ sudo /sbin/chkconfig oozie off
$ sudo /sbin/chkconfig hadoop-httpfs off
```

On SLES Systems

1. Install packages on every host in your cluster:

   For CDH4:

```
$ sudo zypper install bigtop-utils bigtop-jsvc bigtop-tomcat hadoop hadoop-hdfs
hadoop-httpfs hadoop-mapreduce hadoop-yarn hadoop-client hadoop-0.20-mapreduce
hue-plugins hbase hive oozie oozie-client pig zookeeper
```

For CDH3:

```
$ sudo zypper install hadoop-0.20 hadoop-0.20-native.x86_64 hadoop-0.20-sbin.x86_64
 hue-plugins hadoop-zookeeper hadoop-hbase oozie oozie-client
```

2. (**Requires CDH 4.1 or later**) Install Impala and the Impala Shell on Impala machines: **On 64-bit SUSE systems:**

```
$ sudo zypper install impala impala-shell
```

3. To install the `hue-common` package and all Hue applications on the Hue machine, install the hue meta-package:

```
$ sudo zypper install hue
```

4. Disable autostart for Hue on the Hue machine, and for Oozie on every machine on which it is installed.

Note:

Preventing Hue from starting is only required on CDH3u5 and earlier. If you are running CDH4 or a version of CDH3 newer than CDH3u5, you do not need to prevent Hue from starting. While stopping Hue from starting is not required in those cases, executing the `chkconfig` command generates an error, but there are no other negative effects.

```
$ sudo /sbin/chkconfig hue off
$ sudo /sbin/chkconfig oozie off
$ sudo /sbin/chkconfig hadoop-httpfs off
```

On Ubuntu or Debian Systems

1. Install packages on every host in your cluster:

For CDH4:

```
$ sudo apt-get install bigtop-utils bigtop-jsvc bigtop-tomcat hadoop hadoop-hdfs
hadoop-httpfs hadoop-mapreduce hadoop-yarn hadoop-client hadoop-0.20-mapreduce
hue-plugins hbase hive oozie oozie-client pig zookeeper
```

For CDH3:

```
$ sudo apt-get install hadoop-0.20 hadoop-0.20-native.x86_64 hadoop-0.20-sbin.x86_64
 hue-plugins hadoop-zookeeper hadoop-hbase oozie oozie-client
```

2. (**Requires CDH 4.1 or later**) Install Impala and the Impala Shell on Impala machines: **On 64-bit systems:**

```
$ sudo apt-get install impala impala-shell
```

3. To install the `hue-common` package and all Hue applications on the Hue machine, install the hue meta-package:

```
$ sudo apt-get install hue
```

4. Stop `hadoop-httpfs`.

```
service hadoop-httpfs stop
```

5. Disable autostart for Hue on the Hue machine, and for Oozie on every machine on which it is installed.

■ **Note:**

Preventing Hue from starting is only required on CDH3u5 and earlier. If you are running CDH4 or a version of CDH3 newer than CDH3u5, you do not need to prevent Hue from starting. While stopping Hue from starting is not required in those cases, executing the `chkconfig` command generates an error, but there are no other negative effects.

```
$ sudo update-rc.d -f hue remove
$ sudo update-rc.d -f oozie remove
$ sudo update-rc.d -f hadoop-httpfs remove
```

Step 3: Install the Cloudera Manager Server

Install the Cloudera Manager Server either on the machine where the database is installed, or on a machine that has access to the database. This machine need not be a host in the cluster that you want to manage with Cloudera Manager. The Cloudera Manager Server does not require CDH4 to be installed on the same machine.

On the Cloudera Manager Server machine, type the following commands to install the Cloudera Manager packages.

To install Cloudera Manager on a RHEL system if you have a yum repo configured:

```
$ sudo yum install cloudera-manager-daemons
$ sudo yum install cloudera-manager-server
```

To install Cloudera Manager on a RHEL system if you're transferring RPMs manually:

```
$ sudo yum --nogpgcheck localinstall cloudera-manager-daemons-*.rpm
$ sudo yum --nogpgcheck localinstall cloudera-manager-server-*.rpm
```

To install Cloudera Manager Server on a SUSE system:

```
$ sudo zypper install cloudera-manager-daemons cloudera-manager-server
```

To install Cloudera Manager Server on a Debian/Ubuntu system:

```
$ sudo apt-get install cloudera-manager-daemons cloudera-manager-server
```

Step 4: Configure a Database for the Cloudera Manager Server

To manage the services, Cloudera Manager Agents, and configurations in your cluster, the Cloudera Manager Server stores data in a database. You can either use an existing database or install a new database. After installing the database, you must then run a script to prepare that database for use with the Cloudera Manager Server.

■ **Note:**

The Cloudera Manager Server database is separate from the databases used by the Cloudera Manager Activity Monitor, Service Monitor, Report Manager, Host Monitor, Hive Metastore, and Cloudera Navigator. You should have installed these services' databases in the prerequisites.

In this release, you can use any **one** of the database options listed in the table below.

■ **Important:**

You do not need to complete all options listed below. After establishing one database for the Cloudera Manager server, move onto the next steps. Do not install all the database options.

Using the Embedded PostgreSQL Database for the Cloudera Manager Server

To use the embedded PostgreSQL database:

1. Install the embedded PostgreSQL database package on the Cloudera Manager Server host:

   ```
   $ sudo yum install cloudera-manager-server-db
   ```

2. Prepare and start the embedded PostgreSQL database by running this command:

   ```
   $ sudo service cloudera-scm-server-db start
   ```

The Cloudera Manager Server can now use the embedded PostgreSQL database. You can skip to the next step, Install the Cloudera Manager Agents.

Preparing a Custom Database for the Cloudera Manager Server

If you have elected not to use the embedded database, the Cloudera Manager server database configuration can be completed using the `scm_prepare_database.sh` script, which is installed in the `/usr/share/cmf/schema` directory on the host where the Cloudera Manager Server package is installed. You must run the script on the Cloudera Manager Server host.

Note: If you are not using the embedded database, remove `/etc/cloudera-scm-server/db.mgmt.properties>` and restart the Cloudera Manager server.

After you have installed your database application or collected information about an existing Oracle installation, use the `scm_prepare_database.sh` script to prepare the database for use with the Cloudera Manager Server. This script enables Cloudera Manager Server to connect to an external database in MySQL, PostgreSQL, or Oracle. The script prepares the database by:

- Creating the Cloudera Manager Server database configuration file.
- Creating a database for the Cloudera Manager Server to use. This is optional and is only completed if options are specified.
- Setting up a user account for the Cloudera Manager Server. This is optional and is only completed if options are specified.

Script syntax

```
scm_prepare_database.sh [options] database-type database-name username password
```

Required Parameter	Description
`database-type`	To connect to a MySQL database, specify `mysql` as the database type. To connect to an Oracle database, specify `oracle`. To connect to an external PostgreSQL database, specify `postgresql`.
`database-name`	The name of the Cloudera Manager Server database you want to create.
`username`	The username for the Cloudera Manager Server database you want to create.
`password`	The password for the Cloudera Manager Server database you want to create. If you don't specify the password on the command line, the script will prompt you to enter it.

Option	Description
-h or --host	The IP address or hostname of the host where MySQL or Oracle is installed. The default is to use the local host.
-P or --port	The port number to use to connect to MySQL or Oracle. The default port is 3306. This option is used for a remote connection only.
-u or --user	The username for the MySQL or Oracle application. The default is `root`.
-p or --password	The password for the MySQL or Oracle application. The default is no password.
--scm-host	The hostname where the Cloudera Manager Server is installed. Omit if the Cloudera Manager server and MySQL or Oracle are installed on the same host.
--config-path	The path to the Cloudera Manager Server configuration files. The default is `/etc/cloudera-scm-server`.
--schema-path	The path to the Cloudera Manager schema files. The default is `/usr/share/cmf/schema` (the location of the script).
-f	The script will not stop if an error is encountered.
-? or --help	Display help.

Note:

You can also run `scm_prepare_database.sh` without options to see the syntax.

Run the script that applies to the type of database you have installed.

Example 1: Running the script when MySQL is installed on another host

This example explains how to run the script on the Cloudera Manager Server machine (myhost2) and create and use a temporary MySQL user account to connect to MySQL remotely on the MySQL machine (myhost1).

1. On myhost1's MySQL prompt, create a temporary user who can connect from myhost2:

```
mysql> grant all on *.* to 'temp'@'%' identified by 'temp' with grant option;
Query OK, 0 rows affected (0.00 sec)
```

2. On the Cloudera Manager Server host (myhost2), run the script:

   ```
   $ sudo /usr/share/cmf/schema/scm_prepare_database.sh mysql -h
   myhost1.sf.cloudera.com -u temp -ptemp --scm-host myhost2.sf.cloudera.com scm scm
    scm
   Looking for MySQL binary
   Looking for schema files in /usr/share/cmf/schema
   Verifying that we can write to /etc/cloudera-scm-server
   Connecting to mysql at myhost1 as 'temp'
   Creating Cloudera Manager database 'scm'
   Setting up Cloudera Manager user 'scm'@'myhost2.sf.cloudera.com'
   Installing Cloudera Manager schema from file
   /usr/share/cmf/schema/cmf_schema_00001.ddl
   Installing Cloudera Manager schema from file
   /usr/share/cmf/schema/cmf_schema_00002.ddl
   Creating Cloudera Manager configuration file in /etc/cloudera-scm-server
   All done, your Cloudera Manager database is ready to go!
   ```

3. On myhost1, delete the temporary user:

   ```
   mysql> drop user 'temp'@'%';
   Query OK, 0 rows affected (0.00 sec)
   ```

Example 2: Running the script to configure Oracle

This shows an example of running the script to configure an Oracle database.

```
[root@rhel55-6 ~]# /usr/share/cmf/schema/scm_prepare_database.sh -h cm-oracle.example.com
 oracle orcl sample_user sample_pass
Verifying that we can write to /etc/cloudera-scm-server
Creating SCM configuration file in /etc/cloudera-scm-server
Executing: /usr/java/jdk1.6.0_31/bin/java -cp
/usr/share/java/mysql-connector-java.jar:/usr/share/cmf/schema/../lib/*
com.cloudera.enterprise.dbutil.DbCommandExecutor /etc/cloudera-scm-server/db.properties
 com.cloudera.cmf.db.
[ main] DbCommandExecutor INFO Successfully connected to database.
All done, your SCM database is configured correctly!
```

Example 3: Running the script when PostgreSQL is collocated with the Cloudera Manager Server

This example explains how to run the script on the Cloudera Manager Server machine when you have installed PostgreSQL on the same machine.

```
$ /usr/share/cmf/schema/scm_prepare_database.sh postgresql -u postgres scm scm scm
```

Retrieving the Database Host, User Name, or Password

After you are finished installing Cloudera Manager, you can retrieve the Cloudera Manager server database host, user name or password, if necessary, by reading the `/etc/cloudera-scm-server/db.properties` file:

```
# cat /etc/cloudera-scm-server/db.properties

Auto-generated by scm_prepare_database.sh
#
Sat Oct 1 12:19:15 PDT 201
#
com.cloudera.cmf.db.type=mysql
com.cloudera.cmf.db.host=localhost:7432
com.cloudera.cmf.db.name=scm
com.cloudera.cmf.db.user=scm
com.cloudera.cmf.db.password=TXqEESuhj5
```

Step 5: Install the Cloudera Manager Agents

Important:

It is recommended that you install CDH4 before installing the Cloudera Manager Agents.

In this step, you will install the Cloudera Manager Agents and the `/etc/cloudera-scm-agent/config.ini` configuration file on every machine in your cluster that you want to manage using Cloudera Manager. You can use whatever method you currently use to install software and configuration files on your cluster nodes.

On every Cloudera Manager Agent host machine (including those that will run one or more of the Cloudera Manager Services: Service Monitor, Activity Monitor, Event Server, Alert Publisher, Report Manager), use the following commands to install the Cloudera Manager packages: **To install the Cloudera Manager Agent and Services on a RHEL system if you have a yum repo configured:**

```
$ sudo yum install cloudera-manager-agent cloudera-manager-daemons
```

To install the Cloudera Manager Agent and Services on a RHEL system if you're transferring RPMs manually:

```
$ sudo yum --nogpgcheck localinstall cloudera-manager-agent-package.*.x86_64.rpm
cloudera-manager-daemons
```

To install the Cloudera Manager Agent and Services on a SUSE system:

```
$ sudo zypper install cloudera-manager-agent cloudera-manager-daemons
```

To install the Cloudera Manager Agent and Services on a Debian/Ubuntu system:

```
$ sudo apt-get install cloudera-manager-agent cloudera-manager-daemons
```

On every Cloudera Manager Agent host machine, configure the Cloudera Manager Agent to point to the Cloudera Manager Server by setting the following properties in the `/etc/cloudera-scm-agent/config.ini` configuration file:

Property	Description
server_host	Name of host machine where the Server is running
server_port	Port on host machine where the Server is running

Note:

The Cloudera Manager Agent configures its hostname automatically. However, if your cluster machines are multi-homed (that is, they have more than one hostname), and you want to specify which hostname the Cloudera Manager Agent uses, you can update the `listening_hostname=` property in the `/etc/cloudera-scm-agent/config.ini` configuration file on the cluster machines. If you want to specify which IP address the Cloudera Manager Agent uses, you can update the `listening_ip=` property in the same file.

Step 6: Start the Cloudera Manager Server

Important:

When you start the Cloudera Manager Server and Agents, Cloudera Manager assumes you are not already running HDFS and MapReduce. If you are, shut down HDFS and MapReduce (`service hadoop-0.20-<daemon> stop`), and configure the init scripts to not start on boot (for example, `chkconfig hadoop-0.20-<daemon> off`). Contact Cloudera Support for help converting your existing Hadoop configurations for use with Cloudera Manager.

To start the Cloudera Manager Server:

1. To start the Cloudera Manager Server, type this command on the Cloudera Manager Server machine:

```
$ sudo service cloudera-scm-server start
```

2. If you have problems starting the Server, such as database permissions problems, you can use the Server's log `/var/log/cloudera-scm-server/cloudera-scm-server.log` to troubleshoot the problem.

Note:

If the Server fails to start, and you are using MySQL to store information about service configuration, check that the InnoDB engine is configured, not the MyISAM engine; the server will not start if its tables are configured with the MyISAM engine, and an error such as the following will appear in the log file:

```
Tables ... have unsupported engine type ... .  InnoDB is required.
```

For more information, see Installing and Configuring a MySQL Database.

Step 7: Start the Cloudera Manager Agents

To start the Cloudera Manager Agents:

To start the Cloudera Manager Agent, run this command on each Agent machine:

```
$ sudo service cloudera-scm-agent start
```

When the Agent starts up, it contacts the Cloudera Manager Server. When the Agent machines reboot, `cloudera-scm-agent` will start automatically.

Troubleshooting Cloudera Manager Agent Connection Problems

If there is a communication failure between a Cloudera Manager Agent and Cloudera Manager Server, you can use the Cloudera Manager Server log file `/var/log/cloudera-scm-server/cloudera-scm-server.log` and the Cloudera Manager Agent log files `/var/log/cloudera-scm-agent/cloudera-scm-agent.log` to troubleshoot the problem. The following is a common error.

Error message	Description
`error: (113, 'No route to host')` in `cloudera-scm-agent.log`.	This indicates that the agent is unable to connect to the Cloudera Manager Server. Make sure that iptables and SELinux are both turned off.

Step 8: Start the Cloudera Manager Admin Console

The Cloudera Manager Admin Console enables you to use Cloudera Manager to configure, manage, and monitor Hadoop on your cluster. Before using the Cloudera Manager Admin Console, gather information about the server's URL and port.

The server URL takes the following form:

```
http://<Server host>:<port>
```

`<Server host>` is the fully-qualified domain name or IP address of the host machine where the Cloudera Manager Server is installed. `<port>` is the port configured for the Cloudera Manager Server. The default port is 7180. For example, use a URL such as the following:

```
http://myhost.example.com:7180/
```

Cloudera Manager does not support changing the `admin` username for the installed account. You can change the password using Cloudera Manager after you run the wizard in the next section. While you cannot change the `admin` username, you can add a new user, assign administrative privileges to the new user, and then delete the default `admin` account.

To start the Cloudera Manager Admin Console:

1. In a web browser, enter the URL, including the port, for the Cloudera Server. The login screen for Cloudera Manager appears.
2. Log into Cloudera Manager. The default credentials are: **Username:** `admin` **Password:** `admin`

Step 9: Configure Services

The following instructions describe how to use the Cloudera Manager wizard to configure and start the Hadoop services.

To configure services:

1. When you start the Cloudera Manager Admin Console, the install wizard starts up. Click **Continue** to get started.
2. Browse to your Cloudera Manager License file. If you don't install it now, Cloudera Manager Free Edition will be installed.

> ■ **Note:**
>
> The instructions that follow assume you have installed a Cloudera Manager license. If you are not yet ready to install a Cloudera Manager license, and want to proceed with a Free Edition installation, stop here and use the Cloudera Manager Free Edition Installation Guide instead. If you install the Free Edition, and later need to upgrade to the full version of Cloudera Manager, follow the instructions under Upgrade from Cloudera Manager Free Edition 4 to Cloudera Manager 4.

3. After you install the Cloudera Manager license, you must restart the Cloudera Manager server. From the command line, enter:

```
$ sudo service cloudera-scm-server restart
```

4. After the Cloudera Manager server restarts, log in again.

> ■ **Note:**
>
> After restarting the server, wait a few seconds for the server to finish initializing before you try to reconnect to the Admin Console.

5. Click **Continue** in the next screen.
6. On the "Specify hosts..." page, click the **Currently Managed Hosts** tab (you have already installed CDH4 and Cloudera Manager components) and choose the hosts to add to your cluster.
7. On the "Select Repository" page, choose **Use Packages**, as you have already used packages earlier in this process.
8. On the "Host Inspector" page, review the validation results and correct any issues that may have been identified.
9. Choose the Hadoop services you want to start. You can choose one of the standard combinations: Core Hadoop, HBase Services, or All Services; these combinations take into account the dependencies between the Hadoop services. Alternatively, you can choose Custom Services, and select the services individually.

   > **Note:**
   >
   > Some services depend on others; for example, HBase requires HDFS and ZooKeeper.

   The Cloudera Management Services, which are added to each package, are Cloudera Manager processes that run to support monitoring and management features in Cloudera Manager. Cloudera Navigator is a system to support enforcement of compliance with company policies for data stored in a Hadoop Distributed File System (HDFS) deployment.

10. On the Database Setup page, enter the information requested. If the installation you are upgrading includes existing roles, those roles will not require configuration information. At most you will need to provide information for up to the Activity Monitor, Service Monitor, Report Manager, and Host Monitor databases.

    > **Important:**
    >
    > The value you enter as the database hostname **must** match the value you entered for the hostname (if any) when you created the database (see Installing and Configuring Databases).

    For example, if you entered the following for the Activity Monitor database

    ```
    mysql> grant all on amon.* TO 'amon'@'localhost' IDENTIFIED BY 'amon_password';
    ```

    the value you enter here for the database hostname must be `localhost`. On the other hand, if you had entered the following when you created the database

    ```
    mysql> grant all on amon.* TO 'amon'@'myhost1.myco.com' IDENTIFIED BY
    'amon_password';
    ```

    the value you enter here for the database hostname must be myhost1.myco.com. If you did not specify a host, or used a wildcard to allow access from any host, you can enter either the fully-qualified domain name (FQDN) here, or `localhost`. For example, if you entered

    ```
    mysql> grant all on amon.* TO 'amon'@'%' IDENTIFIED BY 'amon_password';
    ```

    the value you enter here for the database hostname can be either the FQDN or `localhost`. Similarly, if you entered

    ```
    mysql> grant all on amon.* TO 'amon' IDENTIFIED BY 'amon_password';
    ```

    the value you enter here for the database hostname can be either the FQDN or `localhost`.

11. Click **Test Connection** to confirm that Cloudera Manager can communicate with the databases using the information you have supplied. This atypical transaction takes two heartbeats to complete (about 30 seconds with the default heartbeat interval). If the test succeeds in all cases, click **Continue**; otherwise check and correct the information you have provided for the databases and then try the test again.

12. Confirm the settings entered for file system paths, such as the NameNode Data Directory and the DataNode Data Directory.
13. Supply the name of the mail server (it can be `localhost`), the mail server user, and the mail recipients.
14. Click **Continue**. The wizard starts the services on your cluster.
15. When all of the services are started, click **Continue**.
16. Click **Continue**.

Step 10: Change the Default Administrator Password

As soon as possible after running the wizard and beginning to use Cloudera Manager, you should change the default administrator password.

To change the administrator password:

1. Click the gear icon ⚙ to display the **Administration** page.
2. Click the **Users** tab.
3. Click the **Change Password** button next to the **admin** account.
4. Enter a new password twice and then click **Submit**.

Step 11: Test the Installation

Now that you have finished the CDH4 and Cloudera Manager installation, you are ready to test the installation. For testing instructions, see Testing the Installation.

> **Note:**
>
> If you change the hostname or port where the Cloudera Manager is running, or you enable TLS security, you must restart the Cloudera Management Services to update the URL to the Server. For instructions, see Restarting a Service.

Installation Path C - Installation Using Tarballs

To avoid using system packages, and to use tarballs (and parcels) instead, follow the instructions in this section.

Before You Begin

Cloudera Manager and Cloudera Distribution of Hadoop (CDH) are comprised of a set of services. These services interact among each other and use databases to complete tasks. The parts that make up this system are very flexible, so you could deploy these services and resources in many different ways, though the process is greatly simplified by following Cloudera's installation and configuration guidelines.

Considering this, Cloudera recommends you begin by establishing a foundation of database resources that can be utilized as they become necessary throughout the installation process. Begin by deploying the necessary supporting services and then proceeding through the installation process.

Install the Oracle JDK

Install the Oracle Java Development Kit (JDK) on each of your cluster hosts where you want to run Hadoop before installing Cloudera's packages. Cloudera Manager can manage both CDH3 and CDH4 hosts, and the required JDK version varies accordingly.

- For installation instructions and recommendations for CDH3, see the topic on Java Development Kit Installation in the CDH3 Installation Guide.
- For installation instructions and recommendations for CDH4, see the topic on Java Development Kit Installation in the CDH4 Installation Guide.

Install Databases for the Cloudera Manager Services

Create and configure databases for the Cloudera Manager Activity Monitor, Service Monitor, Report Manager, and Host Monitor. Cloudera supports various database solutions including MySQL databases or Oracle databases.

Information about how these databases are set up in your environment is required to complete the CDH and Cloudera Manager configuration. The details of what is required varies among database types. Gather this information either as you complete the installations or from database administrators who have the information required. A list of what information is required for each database type is provided in each database section.

Follow the instructions at Installing and Configuring Databases to complete this task.

Database choices	Notes and Instructions
Option A: External PostgreSQL	After PostgreSQL is installed, you need to run a script to prepare a database for the Cloudera Manager Server as described in Installing and Configuring an External PostgreSQL Database.
Option B: External MySQL	You can use the same MySQL application that is used for the monitoring and reporting features, as described in Installing and Configuring a MySQL Database. After MySQL is installed, you need to run a script to prepare a database for the Cloudera Manager Server, as is described later in this topic.
Option C: External Oracle	You can use an external Oracle database for monitoring and reporting features, as described in Using an Oracle Database.

Step 1: Install the Cloudera Manager Server and Agents from Tarballs

Tarballs provide both the Cloudera Manager Server and Cloudera Manager Agents in a single file. Download tarballs from Cloudera Manager Version and Download Information. The files can be unpacked to any location of your choosing. Copy the tarballs, unpack them on all machines on which you intend to install Cloudera Manager Server and Cloudera Manager Agents. If necessary, create a new directory to accommodate the files you extract from the tarball. For instance if `/opt/cloudera-manager` does not exist, create that using a command similar to:

```
$ sudo mkdir /opt/cloudera-manager
```

In the preceding example, files are extracted to a subdirectory named according to the Cloudera Manager version being extracted. For example, files could extract to `/opt/cloudera-manager/cm-4.5/`. This full path is needed later and is referred to as `<tarball root>` directory. The `<tarball root>` directory includes the created Cloudera Manager version number.

When you have a location to which to extract the contents of the tarball, extract the contents. For example, to copy a tar file to your home directory and extract the contents of all tar files to the `/opt/` directory, you might use a command similar to the following:

```
$ tar xzf cloudera-manager*.tar.gz -C /opt/cloudera-manager
```

Creating Users

The Cloudera Manager Server and managed services need a user account to complete tasks. When installing Cloudera Manager from tarballs, you much create this user account manually. Because Cloudera Manager Server and managed services are configured to use the user account `cloudera-scm` by default, creating a user with this name is the simplest approach. After creating such a user, it is automatically used after installation is complete.

To create a user `cloudera-scm`, use a command such as the following:

```
$ useradd --system --home=/opt/cloudera-manager/cm-4.5/run/cloudera-scm-server
--no-create-home --shell=/bin/false --comment "Cloudera SCM User" cloudera-scm
```

For the preceding `useradd` command, ensure the `--home` argument path matches your environment. This argument varies according to where you place the tarball and the version number varies among releases. For example, the `--home` location could be `/opt/cm-4.5/run/cloudera-scm-server`.

Configuring Cloudera Manager Agents

On every Cloudera Manager Agent host machine, configure the Cloudera Manager Agent to point to the Cloudera Manager Server by setting the following properties in the `<tarball root>/etc/cloudera-scm-agent/config.ini` configuration file:

Property	Description
server_host	Name of host machine where the Server is running
server_port	Port on host machine where the Server is running

Custom Cloudera Manager Users and Directories

Cloudera Manager is built to use a default set of directories and user accounts. You can use the default locations and accounts, but there is also the option to change these settings. In some cases, changing these settings is required for Cloudera Manager to work. For most installations, you can skip ahead to Step 2: Configure a Database for the Cloudera Manager Server.

Changing Directories that Cloudera Manager Uses

By default, Cloudera Manager services creates directories in /var/log and /var/lib. The directories the Cloudera Manager installer attempts to create are:

- `/var/log/cloudera-scm-headlamp`
- `/var/log/cloudera-scm-firehose`
- `/var/log/cloudera-scm-alertpublisher`
- `/var/log/cloudera-scm-eventserver`
- `/var/lib/cloudera-scm-headlamp`
- `/var/lib/cloudera-scm-firehose`
- `/var/lib/cloudera-scm-alertpublisher`
- `/var/lib/cloudera-scm-eventserver`

If any of those directories already exist, the Cloudera Manager installer makes no changes to the existing directory. In such a case, Cloudera Manager is unable to write to any existing directories for which it does not have proper permissions. In such a case, services may not perform as expected.

Two ways to resolve this situation are: Changing the ownership of existing directories or specifying alternate directories for agents. You do not need to complete both procedures.

To change ownership for existing directories:

1. Change the directory owner to the Cloudera Manager user. If the Cloudera Manager user and group are cloudera-scm and you needed to take ownership of the headlamp log directory, you would Issue a command similar to the following:

   ```
   $ chown -R cloudera-scm:cloudera-scm /var/log/cloudera-scm-headlamp
   ```

2. Repeat the process of using chown to change ownership for all existing directories to the Cloudera Manager user.

To use alternate directories for services:

> **Note:**
>
> If you changed ownership of existing directories so the Cloudera Manager user can use them, you do not need to use alternate directories.

1. If the directories you plan to use do not exist, create them now. For example to create `/var/cm_logs/cloudera-scm-headlamp` for use by the cloudera-scm user, you might use the following commands:

   ```
   mkdir /var/cm_logs/cloudera-scm-headlamp
   chown cloudera-scm /var/cm_logs/cloudera-scm-headlamp
   ```

2. Connect to the Cloudera Manager admin console.
3. Under the Cloudera Managed Services, click the name of the service.
4. In the service status page, click **Configuration**.
5. In the settings page, enter a term in the **Search** field to find the settings to be change. For example, you might enter "/var" or "directory".
6. Update each value with the new locations for Cloudera Manager to use.
7. Click **Save Changes**.

Step 2: Configure a Database for the Cloudera Manager Server

To manage the services, Cloudera Manager Agents, and configurations in your cluster, the Cloudera Manager Server stores data in a database. You can either use an existing database or install a new database. After installing the database, you must then run a script to prepare that database for use with the Cloudera Manager Server.

> **Note:**
>
> The Cloudera Manager Server database is separate from the databases used by the Cloudera Manager Activity Monitor, Service Monitor, Report Manager, Host Monitor, Hive Metastore, and Cloudera Navigator. You should have installed these services' databases in the prerequisites.

In this release, you can use any **one** of the database options listed in the table below.

> **Important:**
>
> You do not need to complete all options listed below. After establishing one database for the Cloudera Manager server, move onto the next steps. Do not install all the database options.

Example 1: Running the script when MySQL is installed on another host

This example explains how to run the script on the Cloudera Manager Server machine (myhost2) and create and use a temporary MySQL user account to connect to MySQL remotely on the MySQL machine (myhost1).

1. On myhost1's MySQL prompt, create a temporary user who can connect from myhost2:

   ```
   mysql> grant all on *.* to 'temp'@'%' identified by 'temp' with grant option;
   Query OK, 0 rows affected (0.00 sec)
   ```

2. On the Cloudera Manager Server host (myhost2), run the script:

```
$ sudo <tarball root>/share/cmf/schema/scm_prepare_database.sh mysql -h
myhost1.sf.cloudera.com -u temp -ptemp --scm-host myhost2.sf.cloudera.com scm scm
 scm
Looking for MySQL binary
Looking for schema files in /usr/share/cmf/schema
Verifying that we can write to /etc/cloudera-scm-server
Connecting to mysql at myhost1 as 'temp'
Creating Cloudera Manager database 'scm'
Setting up Cloudera Manager user 'scm'@'myhost2.sf.cloudera.com'
Installing Cloudera Manager schema from file
/usr/share/cmf/schema/cmf_schema_00001.ddl
Installing Cloudera Manager schema from file
/usr/share/cmf/schema/cmf_schema_00002.ddl
Creating Cloudera Manager configuration file in /etc/cloudera-scm-server
All done, your Cloudera Manager database is ready to go!
```

3. On myhost1, delete the temporary user:

```
mysql> drop user 'temp'@'%';
Query OK, 0 rows affected (0.00 sec)
```

Example 2: Running the script to configure Oracle

This shows an example of running the script to configure an Oracle database.

```
[root@rhel55-6 ~]# <tarball root>/share/cmf/schema/scm_prepare_database.sh -h
cm-oracle.example.com oracle orcl sample_user sample_pass
Verifying that we can write to /etc/cloudera-scm-server
Creating SCM configuration file in /etc/cloudera-scm-server
Executing: /usr/java/jdk1.6.0_31/bin/java -cp
/usr/share/java/mysql-connector-java.jar:/usr/share/cmf/schema/../lib/*
com.cloudera.enterprise.dbutil.DbCommandExecutor /etc/cloudera-scm-server/db.properties
 com.cloudera.cmf.db.
[ main] DbCommandExecutor INFO Successfully connected to database.
All done, your SCM database is configured correctly!
```

Preparing the Database for the Cloudera Manager Server

Cloudera Manager configuration can be completed using the `scm_prepare_database.sh` script, which is installed in the `<tarball root>/share/cmf/schema/`. You must run the script on the Cloudera Manager Server host. After you have installed your database application or collected information about an existing Oracle installation, use the `scm_prepare_database.sh` script to prepare the database for use with the Cloudera Manager Server. This script enables Cloudera Manager Server to connect to an external database in MySQL, PostgreSQL, or Oracle. The script prepares the database by:

- Creating the Cloudera Manager Server database configuration file.
- Creating a database for the Cloudera Manager Server to use. This is optional and is only completed if options are specified.
- Setting up a user account for the Cloudera Manager Server. This is optional and is only completed if options are specified.

Script syntax

```
scm_prepare_database.sh database-type [options] database-name username password
```

Required Parameter	Description
`database-type`	To connect to a MySQL database, specify `mysql` as the database type. To connect to an Oracle database,

Required Parameter	Description
	specify `oracle`. To connect to an external PostgreSQL database, specify `postgresql`.
`database-name`	The name of the Cloudera Manager Server database you want to create.
`username`	The username for the Cloudera Manager Server database you want to create.
`password`	The password for the Cloudera Manager Server database you want to create. If you don't specify the password on the command line, the script will prompt you to enter it.

Option	Description
-h or --host	The IP address or hostname of the host where MySQL or Oracle is installed. The default is to use the local host.
-P or --port	The port number to use to connect to MySQL or Oracle. The default port is 3306. This option is used for a remote connection only.
-u or --user	The username for the MySQL or Oracle application. The default is `root`.
-p or --password	The password for the MySQL or Oracle application. The default is no password.
--scm-host	The hostname where the Cloudera Manager Server is installed. Omit if the Cloudera Manager server and MySQL or Oracle are installed on the same host.
--config-path	The path to the Cloudera Manager Server configuration files. The default is `/etc/cloudera-scm-server`.
--schema-path	The path to the Cloudera Manager schema files. The default is `/usr/share/cmf/schema` (the location of the script).
-f	The script will not stop if an error is encountered.
-? or --help	Display help.

■ **Note:**

You can also run `scm_prepare_database.sh` without options to see the syntax.

Step 3: Start the Cloudera Manager Server

■ **Important:**

When you start the Cloudera Manager Server and Agents, Cloudera Manager assumes you are not already running HDFS and MapReduce. If you are, shut down HDFS and MapReduce (`service hadoop-0.20-<daemon> stop`), and configure the init scripts to not start on boot (for example, `chkconfig hadoop-0.20-<daemon> off`). Contact Cloudera Support for help converting your existing Hadoop configurations for use with Cloudera Manager.

The way in which you start the Cloudera Manager Server varies according to what account you want the server to run under.

To run as the user invoking the script:

```
$ <tarball root>/etc/init.d/cloudera-scm-server start
```

To explicitly run the server as root:

```
$ sudo <tarball root>/etc/init.d/cloudera-scm-server start
```

To run as another user:

If you run as another user, ensure the user you created for Cloudera Manager owns the location to which you extracted the tarball including the newly created database files. If you followed the earlier examples and created the directory `/opt/cloudera-manager` and the user cloudera-scm, you could use the following command to change ownership of the directory:

```
$ sudo chown -R cloudera-scm:cloudera-scm /opt/cloudera-manager
```

Once you have established proper ownership of directory locations, you can start Cloudera Manager Server using the user account you chose. For example, you might run the Cloudera Manager Server as `cloudera-service`.

In such a case there are two options:

- Use the following command:

```
$ sudo -u <user> <tarball root>/etc/init.d/cloudera-scm-server start
```

- Edit the configuration files so the script internally changes the user. Then run the script as root. To make this possible, complete the following steps:

1. Remove the following line from `<tarball root>/etc/default/cloudera-scm-server`:

   ```
   export CMF_SUDO_CMD=" "
   ```

   Change the user and group in `<tarball root>/etc/init.d/cloudera-scm-server` to the user you want the server to run as. For example, to run as cloudera-service, change the user and group as follows:

   ```
   USER=cloudera-service
   GROUP=cloudera-service
   ```

2. Run the server script as root:

   ```
   $ sudo <tarball root>/etc/init.d/cloudera-scm-server start
   ```

Step 4: Start the Cloudera Manager Agents

To start the Cloudera Manager Agents:

To start the Cloudera Manager Agent, run this command on each Agent machine:

```
$ sudo <tarball root>/etc/init.d/cloudera-scm-agent start
```

When the Agent starts, it contacts the Cloudera Manager Server. If the Agent machines reboot, `cloudera-scm-agent` starts automatically.

Troubleshooting Cloudera Manager Agent Connection Problems

If there is a communication failure between a Cloudera Manager Agent and Cloudera Manager Server, you can use the Cloudera Manager Server log file `/var/log/cloudera-scm-server/cloudera-scm-server.log` and the Cloudera Manager Agent log files `/var/log/cloudera-scm-agent/cloudera-scm-agent.log` to troubleshoot the problem. The following is a common error.

Error message	Description
`error: (113, 'No route to host')` in `cloudera-scm-agent.log`.	This indicates that the agent is unable to connect to the Cloudera Manager Server. Make sure that iptables and SELinux are both turned off.

Step 5: Start the Cloudera Manager Admin Console

The Cloudera Manager Admin Console enables you to use Cloudera Manager to configure, manage, and monitor Hadoop on your cluster. Before using the Cloudera Manager Admin Console, gather information about the server's URL and port.

The server URL takes the following form:

```
http://<Server host>:<port>
```

`<Server host>` is the fully-qualified domain name or IP address of the host machine where the Cloudera Manager Server is installed. `<port>` is the port configured for the Cloudera Manager Server. The default port is 7180. For example, use a URL such as the following:

```
http://myhost.example.com:7180/
```

Cloudera Manager does not support changing the `admin` username for the installed account. You can change the password using Cloudera Manager after you run the wizard in the next section. While you cannot change the `admin` username, you can add a new user, assign administrative privileges to the new user, and then delete the default `admin` account.

To start the Cloudera Manager Admin Console:

1. In a web browser, enter the URL, including the port, for the Cloudera Server. The login screen for Cloudera Manager appears.
2. Log into Cloudera Manager. The default credentials are:

   Username: `admin`

   Password: `admin`

Install the License File

1. When you start the Cloudera Manager Admin Console, the install wizard starts up. Click **Continue** to get started.
2. Browse to your Cloudera Manager License file. If you don't install it now, Cloudera Manager Free Edition will be installed.

> **Note:**
>
> The instructions that follow assume you have installed a Cloudera Manager license. If you are not yet ready to install a Cloudera Manager license, and want to proceed with a Free Edition installation, stop here and use the Cloudera Manager Free Edition Installation Guide instead. If you install the Free Edition, and later need to upgrade to the full version of Cloudera Manager, follow the instructions under Upgrade from Cloudera Manager Free Edition 4 to Cloudera Manager 4.

3. After you install the Cloudera Manager license, you must restart the Cloudera Manager server. From the command line, enter:

   ```
   $ sudo <tarball root>/etc/init.d/cloudera-scm-server restart
   ```

4. After the Cloudera Manager server restarts, log in again.

   > **Note:**
   >
   > After restarting the server, wait a few seconds for the server to finish initializing before you try to reconnect to the Admin Console.

5. Click **Continue** in the next screen.

Step 6: Install CDH Using Parcels

Parcels are a package-less way to install CDH on your cluster. This section walks through installing CDH using parcels, and will let Cloudera Manager download the parcels automatically. See the section on Managing Parcels for alternatives.

Once logged into the Cloduera Manager Admin Console, you will be prompted for a set of hosts for your CDH cluster installation.

Choose the "Cluster Managed Hosts" tab, to select the hosts already running the Cloudera Manager Agents started in Step 4. Select all the hosts, and press Continue.

Choose to use an installation using Parcels. Create the parcel repository directory by using:

```
$ mkdir /opt/cloudera/parcel-repo
```

The server settings allow you to use a different directory.

You must also chown the directory ownership to be the username you are using to run Cloudera Manager:

```
$ chown <username> /opt/cloudera/parcel-repo
```

Choose a version of CDH (and, optionally, Impala) to install on your cluster.

Once you continue, CDH will be downloaded and distributed to all the hosts in your cluster.

Step 7. Configure Services

The following instructions describe how to use the Cloudera Manager wizard to configure and start the Hadoop services.

Note:

After configuring your services, the installation wizard attempts to automatically start the Cloudera Managed Services under the assumption that services will run using `cloudera-scm`. If you configured these services to run using a user other than `cloudera-scm`, then the Cloudera Managed Services do not start automatically. In such a case, change the service configuration to use the user account that you selected. After making this configuration change, manually start the roles, and then begin the process of configuring services.

Changing the Cloudera Manager User

If you specify a user name other than `cloudera-scm` as the Cloudera Manager user, you must update the places where the Cloudera Manager user name is specified.

To update the Cloudera Manager user name

1. Connect to the Cloudera Manager admin console.
2. Under the Cloudera Managed Services, click the name of the service. The status page for services appears.
3. From the **Configuration** tab select **View and Edit** and use the search box to find the property to be changed. For example, you might enter "system" to find the **System User** and **System Group** properties.
4. Make any changes required to the System User and System Group to ensure Cloudera Manager uses the proper user accounts.

To configure services:

1. On the "Specify hosts..." page, select the hosts you configured and then verify that these hosts are reported as managed. Then click **Use Only Currently Managed Hosts** (you have already installed CDH4 and Cloudera Manager components).
2. Choose the Hadoop services you want to start. You can choose one of the standard combinations: Core Hadoop, HBase Services, or All Services; these combinations take into account the dependencies between the Hadoop services. Alternatively, you can choose Custom Services, and select the services individually.

   Note:

   Some services depend on others; for example, HBase requires HDFS and ZooKeeper.

   The Cloudera Management Services, which are added to each package, are Cloudera Manager processes that run to support monitoring and management features in Cloudera Manager.

3. On the Database Setup page, enter the information requested. If the installation you are upgrading includes existing roles, those roles will not require configuration information. At most you will need to provide information for up to the Activity Monitor, Service Monitor, Report Manager, and Host Monitor databases.

   Important:

   The value you enter as the database hostname **must** match the value you entered for the hostname (if any) when you created the database (see Installing and Configuring Databases).

   For example, if you entered the following for the Activity Monitor database

   ```
   mysql> grant all on activity_monitor.* TO 'amon_user'@'localhost' IDENTIFIED BY
   'amon_password';
   ```

the value you enter here for the database hostname must be `localhost`. On the other hand, if you had entered the following when you created the database

```
mysql> grant all on activity_monitor.* TO 'amon_user'@'myhost1.myco.com' IDENTIFIED
 BY 'amon_password';
```

the value you enter here for the database hostname must be myhost1.myco.com. If you did not specify a host, or used a wildcard to allow access from any host, you can enter either the fully-qualified domain name (FQDN) here, or `localhost`. For example, if you entered

```
mysql> grant all on activity_monitor.* TO 'amon_user'@'%' IDENTIFIED BY
'amon_password';
```

the value you enter here for the database hostname can be either the FQDN or `localhost`. Similarly, if you entered

```
mysql> grant all on activity_monitor.* TO 'amon_user' IDENTIFIED BY 'amon_password';
```

the value you enter here for the database hostname can be either the FQDN or `localhost`.

4. Click **Test Connection** to confirm that Cloudera Manager can communicate with the databases using the information you have supplied. This atypical transaction takes two heartbeats to complete (about 30 seconds with the default heartbeat interval). If the test succeeds in all cases, click **Continue**; otherwise check and correct the information you have provided for the databases and then try the test again.
5. Confirm the settings entered for file system paths, such as the NameNode Data Directory and the DataNode Data Directory.
6. Supply the name of the mail server (it can be `localhost`), the mail server user, and the mail recipients.
7. Click **Continue**. The wizard starts the services on your cluster.
8. When all of the services are started, click **Continue**.
9. Click **Continue**.

Step 8. Change the Default Administrator Password

As soon as possible after running the wizard and beginning to use Cloudera Manager, you should change the default administrator password.

To change the administrator password:

1. Click the gear icon ⚙ to display the **Administration** page.
2. Click the **Users** tab.
3. Click the **Change Password** button next to the **admin** account.
4. Enter a new password twice and then click **Submit**.

Step 9. Test the Installation

Now that you have finished the CDH4 and Cloudera Manager installation, you are ready to test the installation. For testing instructions, see Testing the Installation.

Note:

If you change the hostname or port where the Cloudera Manager is running, or you enable TLS security, you must restart the Cloudera Management Services to update the URL to the Server. For instructions, see Restarting a Service.

Installing Impala with Cloudera Manager

To use Cloudera Impala, you must install CDH and Impala (Hive is required, and gets installed with CDH). Install CDH and Impala on the nodes that will run Impala. Use **only one** of the following ways to deploy CDH and Impala:

- Installation Path A - Automated Installation by Cloudera Manager: Installs Cloudera Manager, CDH, and Impala as part of the process of using the pre-packaged installer. This method will install all the necessary software to run Cloudera Impala, will handle setting up the Hive metastore using the default PostgreSQL database, and will start the Impala Service along with the other CDH and Cloudera Manager services. Within the installation wizard you can install Impala using either packages or parcels.
- Installation Path B - Installation Using Your Own Method: Installs Cloudera Manager, CDH, and Impala, specifying each package individually using package management tools. If you follow this method, then you will prepare for the installation by installing the Oracle JDK, creating databases, and determining how you will download packages. Once you have completed these prerequisites, you will install CDH and Cloudera Manager Server using packages. Next you will configure a database for Cloudera Manager and then install Cloudera Manager Agents. Finally you will start the Cloudera Manager Server and Agents and then configure services using the Cloudera Manager Admin console.

> **Note:** Impala 1.0 is supported with CDH4.1 and CDH4.2.

Once you have installed Impala, you can coordinate its use of cluster resources in relation to MapReduce needs for the same resources. See Setting up a Multi-tenant Cluster for Impala and MapReduce below, as well as Resource Management in the Cloudera Manager User Guide.

> **Note:** If you are running CDH 4.1, and the Bypass Hive Metastore Server option is enabled, you must add the following to the Impala Safety Valve for `hive-site.xml`, replacing `<hive_metastore_server_host>` with the name of your Hive metastore server host:
>
> ```
> <property>
>   <name> hive.metastore.local</name>
>   <value>false</value>
> </property>
> <property>
>   <name> hive.metastore.uris</name>
>   <value>thrift://<hive_metastore_server_host>:9083</value>
> </property>
> ```
>
> Otherwise, Impala queries will fail.

Configuring Hive Table Stats

Configuring Hive Table Stats is highly recommended when using Impala. It allows Impala to make optimizations that can result in significant (over 10x) performance improvements for some joins. If these are not available, Impala will still function, but at lower performance.

To configure Hive Table Stats:

- Set up a MySQL server for transient Stats data.

  Note that there is no PostgreSQL or Oracle option. This database will be heavily loaded, so it should not be installed on the same host as anything critical such as the Hive Metastore Server, the database hosting the Hive Metastore, or Cloudera Manager Server. When collecting stats on a large table and/or in a large cluster, this host may become slow or unresponsive.

  For instructions on setting up MySQL, see Installing and Configuring a MySQL Database

- Add the following into the HiveServer2 Safety Valve for `hive-site.xml`:

```
<property>
  <name>hive.stats.dbclass</name>
  <value>jdbc:mysql</value>
</property>
<property>
  <name>hive.stats.jdbcdriver</name>
  <value>com.mysql.jdbc.Driver</value>
</property>
<property>
  <name>hive.stats.dbconnectionstring</name>
  <value>jdbc:mysql://<stats_mysql_host>:3306/<stats_db_name>?useUnicode=true&amp;
characterEncoding=UTF-8&amp;user=<stats_user>&amp;password=<stats_password></value>
</property>
<property>
  <name>hive.aux.jars.path</name>
  <value>file:///usr/share/java/mysql-connector-java.jar</value>
</property>
```

- Restart HiveServer2

Collect stats on a particular table by running the following from a beeline client connected to your HiveServer2:

```
analyze table MY_TABLE compute statistics;
```

Setting up a Multi-tenant Cluster for Impala and MapReduce

MapReduce and Impala often work with the same data set and run side-by-side on the same physical hardware. We call such clusters with multiple active compute frameworks "multi-tenant" clusters. It is important to control the amount of resources assigned to each compute framework. Otherwise, Impala queries and MapReduce jobs may suffer from conflicting resource demands, leading to poor performance for both components. There is no standard resource allocation that fits all requirements. This document illustrates the mechanisms and considerations behind resource management. You should determine what resource allocation your workloads need to meet your production SLAs.

Our resource management controls cover CPU, disk IO, and memory. For simplicity, we assign fraction *x* of all resources to Impala, and the rest (fraction 1-*x*) to MapReduce. With this assignment, we roughly expect Impala multi-tenant performance to be fraction *x* of Impala stand-alone performance, i.e., Impala running by itself on the cluster. Likewise, we roughly expect MapReduce multi-tenant performance to be fraction 1-*x* of MapReduce stand-alone performance.

We make the following assumptions for the examples in the rest of the document:

- MapReduce and Impala are the only two active compute frameworks.
- Each slave node is running a Datanode, Tasktracker, and an Impala daemon.
- The cluster uses homogenous hardware.
- Each slave node has 60 GB of RAM.
- Each slave node has 16 cores.

The actual resource management mechanisms are

- Memory: Impala Daemon Memory Limit, Maximum Number of Simultaneous Map Tasks, Maximum Number of Simultaneous Reduce Tasks.
- CPU: Cgroup CPU shares, Maximum Number of Simultaneous Map Tasks, Maximum Number of Simultaneous Reduce Tasks.

- Disk: Cgroup IO weight, Maximum Number of Simultaneous Map Tasks, Maximum Number of Simultaneous Reduce Tasks.

For instructions on configuring and enabling Cgroups, see Resource Management in the Cloudera Manager Enterprise Edition User Guide.

Example: Assign 50% of resources to Impala

We need to set numerical values for the following parameters to assign 50% of resources to Impala and to MapReduce:

- Memory: Impala Daemon Memory Limit, Maximum Number of Simultaneous Map Tasks, Maximum Number of Simultaneous Reduce Tasks.
- CPU: Cgroup CPU Shares, Maximum Number of Simultaneous Map Tasks, Maximum Number of Simultaneous Reduce Tasks.
- Disk: Cgroup IO Weight, Maximum Number of Simultaneous Map Tasks, Maximum Number of Simultaneous Reduce Tasks.

These parameters are associated with the configurations of the Impala Daemon, DataNode, and TaskTracker roles in Cloudera Manager.

We compute the numerical values using the rationale below:

Memory

We set Impala Daemon Memory Limit to 50% of the effective memory size, which we assume to be (RAM size) / 1.5. The divisor 1.5 is a conservative estimate to adjust for OS accounting overhead. For our numerical example, this works out to 60GB / 1.5 * 50% = 20GB.

We indirectly control MapReduce memory consumption by cutting to 50% the maximum number of simultaneous map and reduce tasks. For CM, the default maximum number of simultaneous map tasks is set to the number of CPU cores, 16 in our example, and maximum number of simultaneous reduce tasks is set to half the number of CPU cores. This works out to 16 * 50% = 8 max map tasks, and 16 / 2 * 50% = 4 max reduce tasks.

CPU

The more Cgroup CPU Shares given to a role, the larger its share of the CPU when under contention. Until processes on the host (including both roles managed by Cloudera Manager and other system processes) are contending for all of the CPUs, this will have no effect. When there is contention, those processes with higher CPU shares will be given more CPU time. The effect is linear: a process with 4 CPU shares will be given roughly twice as much CPU time as a process with 2 CPU shares.

In a multi-tenant workload, all three daemons - Impala, DataNodes & TaskTrackers - may be simultaneously consuming CPU. We assign 50% of the total CPU shares to the Impala daemon. We assign 25% each to the DataNodes and the TaskTracker. This is a very conservative setting. It assumes that DataNode and task computational activity can simultaneously maximize CPU activity. If your MapReduce workload is far below cluster capacity, this is unlikely to be true.

Disk

The more the Cgroup I/O Weight, the higher priority will be given to I/O requests made by the role when I/O is under contention (either by roles managed by Cloudera Manager or by other system processes). The effect is much like Cgroup CPU Shares. Note that Cgroup I/O Weight only prioritizes read requests; write requests remain unprioritized.

The rationale for the numerical values for Cgroup I/O Weight is identical to that for Cgroup CPU Shares, i.e., we ensure Impala daemon has 50% of the sum Cgroup I/O Weight for (Impala daemon + DataNode + TaskTracker), and DataNode and TaskTracker have equal weights that sum up to the remaining 50%.

Follow the steps below to set the numerical values:

Impala daemon

1. Go to the Impala service configuration edit screen.
2. Select **Impala Daemon (base)**, **Resource Management** tab.
3. Set **Impala Daemon Memory Limit** to 20GB.
4. Set **Cgroup CPU Shares** to 2048.
5. Set **Cgroup I/O Weight** to 1000.
6. Click the **Save Changes** button.
7. Repeat Steps 2-6 for all Impala Daemon role configuration groups, e.g., **Impala Daemon (1)**.

DataNode

1. Go to the HDFS service configuration edit screen.
2. Select **DataNode (base)**, **Resource Management** tab.
3. Set **Cgroup CPU Shares** to 1024.
4. Set **Cgroup I/O Weight** to 500.
5. Click the **Save Changes** button.
6. Repeat Steps 2-5 for all DataNode role configuration groups, e.g., **DataNode (1)**.

TaskTracker

1. Go to the MapReduce service configuration edit screen.
2. Select **TaskTracker (base)**, **Resource Management** tab.
3. Set **Cgroup CPU Shares** to 1024.
4. Set **Cgroup I/O Weight** to 500.
5. Select **TaskTracker (base)**, **Performance** tab.
6. Set **Maximum Number of Simultaneous Map Tasks** to 8.
7. Set **Maximum Number of Simultaneous Reduce Tasks** to 4.
8. Click the **Save Changes** button.
9. Repeat Steps 2-8 for all TaskTracker role configuration groups, e.g., **TaskTracker (1)**.

Restart all services for these changes to take effect.

Further Examples

For convenience, we provide here some numerical tables for configurations that assign to Impala 25%, 50%, and 75% of all resources, while MapReduce receives the rest.

Again, there is no standard resource allocation that fits all requirements. You should determine what resource allocation your workloads need to meet your production SLAs. For example, as Impala is often memory-bound and MapReduce often IO-bound, you can consider increasing the Impala memory shares and MapReduce IO-shares from the starting points below.

Impala gets 25%, MapReduce gets 75%

	Impala daemon	DataNode	TaskTracker
Cgroup CPU Shares	683	1024	1024
Cgroup I/O Weight	333	500	500
Impala Daemon Memory Limit	(RAM size) / 1.5 * 0.25	n/a	n/a
Maximum Map Tasks	n/a	n/a	(# cores) * 0.75
Maximum Reduce Tasks	n/a	n/a	(# cores) / 2 * 0.75

Impala gets 50%, MapReduce gets 50%

	Impala daemon	DataNode	TaskTracker

Cgroup CPU Shares	2048	1024	1024
Cgroup I/O Weight	1000	500	500
Impala Daemon Memory Limit	(RAM size) / 1.5 * 0.5	n/a	n/a
Maximum Map Tasks	n/a	n/a	(# cores) * 0.5
Maximum Reduce Tasks	n/a	n/a	(# cores) / 2 * 0.5

Impala gets 75%, MapReduce gets 25%

	Impala daemon	DataNode	TaskTracker
Cgroup CPU Shares	6144	1024	1024
Cgroup I/O Weight	1000	167	167
Impala Daemon Memory Limit	(RAM size) / 1.5 * 0.75	n/a	n/a
Maximum Map Tasks	n/a	n/a	(# cores) * 0.25
Maximum Reduce Tasks	n/a	n/a	(# cores) / 2 * 0.25

More About Cgroups

See Resource Management in the Cloudera Manager User Guide for more information about Cgroups, including Linux distribution support and known issues.

Installing Cloudera Manager and CDH on EC2

The following procedure leads you through setting up Cloudera Manager, CDH and Impala on a cluster of Amazon Web Services (AWS) EC2 instances.

- The Cloudera Manager installation wizard launches the EC2 version of the wizard when Cloudera Manager is started on EC2.
- The EC2 wizard installs Cloudera Manager Free Edition. If you have a license for the Enterprise Edition, you can upgrade from the Cloudera Manager Admin Console after installation. (See the Upgrade from Cloudera Manager Free Edition 4 to Cloudera Manager 4 for details.)
- The resulting installation will use the embedded PostgreSQL data base; there is no option for setting up other databases.
- This wizard will install and start all the latest Cloudera Manager-managed CDH 4 services as well as Impala.

> **Note:** This setup is not recommended for production use.

Step 1: Set up an AWS EC2 instance for the Cloudera Manager server.

1. Log into the Amazon Web Services (AWS) console.
2. Go to EC2.
3. Create a security group:
   - In the left menu, select **Security Groups**.
   - Click **Create Security Group**.
   - When prompted, enter a name and description, and click OK.
   - Select the group you created in the list of groups.
   - In the bottom panel, go to the **Inbound** tab.
   - Authorize TCP ports 7180, 7182, 7183 and 7432.
   - Authorize ICMP (Echo Reply).
4. Create (or import) an SSH key pair:
   - In the left menu, select **Key Pairs**.
   - Click **Create Key Pair**.
   - When prompted, enter a key pair name and click OK
   - Your private key `< keypair-name >`.pem will be downloaded automatically. AWS does not store the private keys – if you lose this file, you won't be able to SSH into instances you provision with this key pair.
5. Launch an EC2 instance:
   a. Select the Ubuntu12.04 AMI or other Operating System supported by Cloudera Manager. See Requirements for Cloudera Manager.
   b. Use the key pair and security group you prepared in the previous steps.
   c. Select the Instance Type: Cloudera recommends using at least "M1 Large" instances.
   d. Look at the instance details, and copy the public hostname.
   e. Then SSH into the instance:

   ```
   $ ssh -i <private-key-file> <username>@ec2-xx-xx-xx-xx.compute-1.amazonaws.com
   ```

   The <username> is usually "ubuntu" on Ubuntu systems, and "ec2-user" on most other Linux images on EC2.

f. Download the Cloudera Manager installer:

```
$ wget
http://archive.cloudera.com/cm4/installer/latest/cloudera-manager-installer.bin
```

g. Execute the installer:

```
$ sudo su
$ chmod +x cloudera-manager-installer.bin
$ ./cloudera-manager-installer.bin
```

6. When the installer finishes, navigate to `http://<public hostname>:7180` and log into the Cloudera Manager Admin console.
7. (optional) Configure TLS encryption. (See Configuring TLS Security for Cloudera Manager Free Edition).

> **Note:** You will need to upload your AWS account credentials to launch the EC2 instances in the installation wizard, and Cloudera strongly recommends configuring TLS connection.

Step 2: Install Cloudera Manager and CDH packages using the Cloud Wizard.

1. Log into the Cloudera Manager admin console on your EC2 instance: `<public hostname>:7180`.
    - The initial user name and password are `admin`.
2. The Welcome Page appears.
    - Please note the Warning that is presented.

> **Warning:**
>
> Instances provisioned on AWS EC2 by this wizard are instance store-based, so all data will be lost when an instance is stopped or terminated.

3. Provide the instance specifications:

    a. Choose your OS: currently supported are Ubuntu 12.04 or Centos 6.3.
    b. Alternatively, you may use a custom AMI:
        - Make sure the AMI is in the same region as Cloudera Manager server!
        - Specify the username Cloudera Manager should use to SSH in. This is usually "ubuntu" on Ubuntu systems, and "ec2-user" on most other Linux images on EC2.

    c. Choose the type of EC2 instances you want to provision:
        - Note that instances not matching the minimum requirements were deliberately removed from the list.

    d. Specify the number of instances you wish to provision.
    e. Specify the group name (string).
        - This string will be included in the name of your instances and the security group and key pair, which will be created by Cloudera Manager.

4. Provide credentials:

    a. Enter the AWS Access and Secret key. To create new ones, follow these instructions:

        a. Go to https://console.aws.amazon.com/iam/.
        b. Click **Users**.

   c. Check the box next to the desired user, scroll down and click **Manage Access Keys**.
   d. Copy the new keys and paste them to the inputs.

   b. Choose the instance authentication method: Either:

      - Let Cloudera Manager create a new SSH key pair for your instances. You will be able to download the private key later to SSH into the new instances. --- **OR** —
      - Import and upload your own key:

        1. In the console, go to **Key Pairs**.
        2. Click **Import Key**.
        3. Select your private key file, specify the name and click **Yes, Import**.

5. Review the Installation Settings:

   - You may go back if you want to correct any information you provided in the previous steps.
   - Once the instances are provisioned, you must terminate them if you need to modify the installation settings.
   - Click **Start Installation**.

6. Provision new instances. Once instances are provisioned:

   - Download the private SSH key if you chose to create one.
   - Remember to terminate the instances (through the AWS EC2 console).

7. The wizard leads you through the installation steps:

   a. Install Cloudera Manager and CDH packages.
   b. Run the Host Inspector.
   c. Start all Services.

How to Terminate your EC2 instances

> ■ **Warning:**
>
> Cloudera Manager will only terminate instances if the installation fails. Terminate the instances manually when you are done using the CDH cluster.

1. Sign into the AWS EC2 console.
2. In the left menu, select **Instances**.
3. Select the instances you want to terminate. You may use the string you entered as "group name" to filter the instances provisioned by Cloudera Manager.
4. From **Actions** select **Terminate**.

Upgrading to Cloudera Manager 4.5

You can upgrade existing Cloudera Manager and Cloudera's Distribution Including Apache Hadoop (CDH) to this version. Upgrading preserves existing data and settings, while enabling the use of the new features provided with the latest product versions. To enable new features, some new settings are added, but nothing is removed.

Understanding Upgrades

The process for upgrading to Cloudera Manager varies based on the starting point. The categories of tasks to be completed include the following:

- Install any databases that are newly required for this release.
- Upgrade Cloudera Manager.
- Upgrade hosts in the cluster.

Before Upgrading

- The Cloudera Manager Server must have SSH access to the cluster hosts and you must log in using a root account or an account that has password-less sudo permission. See Requirements for Cloudera Manager for more information.
- Ensure there are no running commands. Use the Admin Console's main navigation bar to check for any running commands. You can either wait for commands to complete or abort any running commands. For more information on viewing and aborting running commands, see Viewing Running and Recent Commands.
- Ensure you have completed any required process for preparing databases, as described in Database Considerations for Cloudera Manager Upgrades.

During the Upgrade

During the upgrade process, the following changes occur:

- The database schemas are modified for any databases storing information for Cloudera Manager Server, Activity Monitor, Service Monitor, Report Manager, and Host Monitor.
- Configuration information is reorganized.

After Upgrading

After completing an upgrade to 4.5, the following is true:

- You should deploy client configurations to ensure client services have the most current configuration.
- Required databases are established to store information for Cloudera Manager Server, Activity Monitor, Service Monitor, Report Manager, and Host Monitor.
- The database schemas reflect the current version.
- The Host Monitor service is added and active.
- The Cloudera Manager Server and all supporting services, such as the Activity Monitor, Service Monitor, Report Manager, and Host Monitor are updated.

Upgrade Paths

In some cases, completing an upgrade requires changes to your environment, and in other cases, elements are already in place. For example, if you are upgrading your environment from 3.7 to 4.5, you must add Host Monitor, but if you are upgrading from 4.0 to 4.5, this is not required, as Host Monitor is included in 4.0. The specific steps required vary based on the path taken.

To upgrade from an older version of Cloudera Manager, begin by upgrading to 3.7, and then proceed to upgrade from there.

> **Warning:**
>
> Cloudera Manager 4.5 works with CDH3 and CDH4, but does not work with CDH4.0 beta. You must upgrade any installations of CDH4.0 beta.

Begin the upgrade process by evaluating Database Considerations for Cloudera Manager Upgrades.

Database Considerations for Cloudera Manager Upgrades

Cloudera Manager uses databases to store information about system configurations and tasks. Before upgrading, complete the pre-upgrade database tasks that apply in your environment.

After you have completed these steps, the upgrade processes automatically complete any additional updates to database schemas and service data stored. You do not need to complete any data migration.

Back up Databases

Before beginning the upgrade process, shut down the services that are using databases. This includes Cloudera Manager Server, Activity Monitor, Service Monitor, Report Manager, Host Monitor, Cloudera Navigator, and Hive Metastore. Cloudera recommends that you then back up all databases. This is especially important if you are upgrading from 3.7.x and there is any possibility you may want to revert to using 3.7.x. For information on backing up databases:

- For MySQL, see Backing up the MySQL Database.
- For PostgreSQL, see Backing up the PostgreSQL Database.
- For Oracle, work with your database administrator to ensure databases are properly backed up.

If any additional database will be required as a result of the upgrade, complete any required preparatory work to install and configure those databases. For example, Cloudera Manager 4.0 offers a Host Monitoring service that requires a database. To enable the Host Monitoring service, you must install a database. The upgrade instructions assume all required databases have been prepared. For more information on using databases, see Installing and Configuring Databases.

Modify Databases to Support UTF-8

Cloudera Manager 4.0 adds support for UTF-8 character sets. Update any existing databases in your environment that are not configured to support UTF-8.

Modifying MySQL to Support UTF-8

To modify a MySQL database to support UTF-8, the default character set must be changed and then you must restart the mysql service. Use the following commands to complete these tasks:

```
mysql> alter database default character set utf8;
mysql> quit
$ sudo service mysql restart
```

Modifying PostgreSQL to Support UTF-8

There is no single command available to modify an existing PostgreSQL database to support UTF-8. As a result, you must complete the following process:

1. Use pg_dump to export the database to a file. This creates a backup of the database that you will import into a new, empty database that supports UTF-8.

2. Drop the existing database. This deletes the existing database.
3. Create a new database that supports unicode encoding and that has the same name as the old database. Use a command of the following form, replacing the database name and user name with values that match your environment:

   ```
   CREATE DATABASE scm_database WITH OWNER scm_user ENCODING 'UTF8'
   ```

4. Review the contents of the exported database for non-standard characters. If you find unexpected characters, modify these so the database backup file contains the expected data.
5. Import the database backup to the newly created database.

Modifying Oracle to Support UTF-8

Work with your Oracle database administrator to ensure any Oracle databases support UTF-8.

Modify Databases to Support Appropriate Maximum Connections

Check existing databases configurations to ensure the proper maximum number of connections is supported. Update the maximum configuration values, as required.

Modifying the Maximum number of MySQL Connections

Allow 100 maximum connections for each database and then add 50 extra connections. For example, for two databases set the maximum connections to 250. If you store seven databases on one host (the databases for Activity Monitor, Service Monitor, Report Manager, Host Monitor, Cloudera Manager Server, Cloudera Navigator, and Hive Metastore), set the maximum connections to 750.

Modifying the Maximum number of PostgreSQL Connections

Update the `max_connection` parameter in the `/etc/postgresql.conf` file.

You may have to increase the system resources available to PostgreSQL, as described at http://www.postgresql.org/docs/9.1/static/kernel-resources.html.

Modifying the Maximum number of Oracle Connections

Work with your Oracle database administrator to ensure appropriate values are applied for your Oracle database settings. You must determine the number of connections, transactions, and sessions to be allowed. Allow 100 maximum connections for each database and then add 50 extra connections. For example, for two databases set the maximum connections to 250. If you store seven databases on one host (the databases for Activity Monitor, Service Monitor, Report Manager, Host Monitor, Cloudera Manager Server, Cloudera Navigator, and Hive Metastore), set the maximum connections to 750.

From the maximum number of connections, you can determine the number of anticipated sessions using the following formula:

```
sessions = (1.1 * maximum_connections) + 5
```

For example, if a host has two databases, you anticipate 250 maximum connections. If you anticipate a maximum of 250 connections, plan for 280 sessions.

Once you know the number of sessions, you can determine the number of anticipated transactions using the following formula:

```
transactions = 1.1 * sessions
```

Continuing with the previous example, if you anticipate 280 sessions, you can plan for 308 transactions.

Work with your Oracle database administrator to apply these derived values to your system.

Using the sample values above, Oracle attributes would be set as follows:

```
alter system set processes=250;
alter system set transactions=308;
alter system set sessions=280;
```

Next Steps

After you have completed any required database preparatory tasks, continue to the upgrade path that is appropriate for your environment. Supported paths include:

- Upgrade from Cloudera Manager 3.7.x to Cloudera Manager 4.5
- Upgrade from Cloudera Manager 4 to the Latest Cloudera Manager 4
- Upgrade from Cloudera Manager Free Edition 4 to Cloudera Manager 4

Upgrade from Cloudera Manager 3.7.x to Cloudera Manager 4.5

Upgrading from Cloudera Manager 3.7 to the latest version of Cloudera Manager primarily involves upgrading Cloudera Manager Server packages.

To complete the upgrade, you stop the Cloudera Management Service, upgrade the packages (and database tables, if necessary), and then start the Cloudera Management Service again. After upgrading Cloudera Manager you may also want to upgrade CDH or add Impala.

> **Important:**
>
> Cloudera Manager 4.5 adds support for Hive, and includes a new role type called the Hive Metastore Server. This role manages the metastore process when Hive is configured with a Remote Metastore. When upgrading from a previous Cloudera Manager version, Cloudera Manager automatically creates new Hive service(s) to capture the previous implicit Hive dependency from Hue and Impala. Your previous services will continue to function without impact. Note that if Hue is using a Hive metastore of type Derby, then the newly created Hive service will also use Derby. But since Derby does not allow concurrent connections, Hue will continue to work, while the new Hive Metastore Server will fail to run. The failure is harmless (because nothing uses this new Hive Metastore Server at this point) and intentional, to preserve the set of cluster functionality as it was before upgrade. Cloudera discourages the use of a Derby metastore due to its limitations. You should consider switching to a different supported database type (PostgreSQL, MySQL, Oracle). Cloudera Manager provides a Hive configuration option to bypass the Hive Metastore server. When this configuration is enabled, Hive clients, Hue, and Impala connect directly to the Hive Metastore Database. Prior to Cloudera Manager 4.5, Hue and Impala talked directly to the Hive Metastore Database, so the Bypass mode is enabled by default when upgrading to Cloudera Manager 4.5. This is to ensure the upgrade doesn't disrupt your existing setup. You should plan to disable the Bypass Hive Metastore Server mode, especially when using CDH 4.2 or later. Using the Hive Metastore Server is the recommended configuration. After changing this configuration, you must re-deploy your client configurations, restart Hive, and restart any Hue or Impala services configured to use that Hive.

Cloudera Manager 4 adds an optional Host Monitor, which requires a database. If you intend to deploy Host Monitor or any other additional agents, you must establish a database for them. For information on establishing databases for agents such as Host Monitor, see Installing and Configuring Databases.

> **Important:**
>
> You will need to restart your clusters after you have finished the Cloudera Manager upgrade.

Summary: What You are Going to Do

Upgrading from Cloudera Manager 3.7.x to Cloudera Manager 4.5 involves the following broad steps:

Step 1. Stop the Cloudera Management Service - Stop the Cloudera Management Service.

Step 2. Upgrade Cloudera Manager Server - Stop Cloudera Manager services, copy files to the Cloudera Manager server, upgrade the server, and restart the server.

Step 3. Upgrade the Cluster Hosts – Use the upgrade wizard to upgrade hosts in the cluster.

Step 4. Verify the Upgrade – You can choose to check the versions of installed components.

Step 5. Add Hive Gateway Roles – Add Hive gateway roles on Hive client hosts to ensure Hive client configurations are deployed on those hosts.

Step 6. Restart the Cloudera Management Service – Restart the Cloudera Manager Management Service. If you have ZooKeeper installed, you will also need to restart the ZooKeeper service so that the ZooKeeper health checks will succeed.

Step 7. Deploy Updated Client Configurations - Update client configurations to ensure clients operate as expected with the upgraded systems.

Step 8. Restart Your Cluster(s) – You must restart your cluster to ensure compatibility with the updated JDK.

Step 9. (Optional) Upgrade CDH – You may choose to upgrade CDH installations. Cloudera Manager 4 can manage both CDH 3 and CDH 4.

Before beginning the upgrade, follow the guidelines described in Database Considerations for Cloudera Manager Upgrades.

After completing the upgrade from Cloudera Manager 3.7.x to Cloudera Manager 4.5, as described in this topic, all required updates to database schemas and service data is completed automatically. You do not need to complete any additional database updates or data migration.

Step 1. Stop the Cloudera Management Service

The Cloudera Manager Service must be stopped before upgrades can occur.

To stop the Cloudera Management Service

1. Click the **Services** tab in Cloudera Manager Admin Console.
2. Choose **Stop** on the **Actions** menu for the Cloudera Management Services.

Step 2. Upgrade Cloudera Manager Server

This process involves stopping running Cloudera Manager service, downloading and applying updates to Cloudera Manager, and restarting the Cloudera Manager service. Valid licenses from Cloudera Manager 3.7.x continue to work with Cloudera Manager 4.

You can use package management software to download and apply updates from Cloudera's software repository. The default name of the repo file is `cloudera-manager`. This name is also typically in square brackets on the first line of the Cloudera Manager repo file. For example, you could view the contents of the repo file, including the repo name in brackets. This file might be at `/etc/yum.repos.d/cloudera-manager.repo` and its contents could be viewed using the `more` command as follows:

```
[user@system yum.repos.d]$ more cloudera-manager.repo
[cloudera-manager]
...
```

The location of the repo files varies by operating system and package management solution.

- For yum the repo file is at `/etc/yum.repos.d/cloudera-manager.repo`.
- For zypper the repo file is at `/etc/zypp/repos.d/cloudera-manager.repo`.

Find Cloudera's repo file for your distribution by starting at `http://archive.cloudera.com/cm4/` and navigating to the directory that matches your operating system. For example, for Red Hat 6, you would navigate to `http://archive.cloudera.com/cm4/redhat/6/x86_64/cm/`. Within that directory, find the repo file that contains information including the repository's base URL and gpgkey. In the preceding example, the contents of the `cloudera-manager.repo` file might appear as follows:

```
[cloudera-manager]
# Packages for Cloudera Manager, Version 4, on RedHat or CentOS 5 x86_64
name=Cloudera Manager
baseurl=http://archive.cloudera.com/cm4/redhat/5/x86_64/cm/4/
gpgkey = http://archive.cloudera.com/cm4/redhat/5/x86_64/cm/RPM-GPG-KEY-cloudera
gpgcheck = 1
```

Copy this repo file to the configuration location for the package management software for your system. Continuing with the preceding example, on Red Hat 6, you would copy the `cloudera-manager.repo` file to `/etc/yum.repos.d/`.

Before beginning the upgrade process, it can be best to clean all yum's cache directories using the command `yum clean all`. Doing so ensures that you download and install the latest versions of the packages. If your system is not up to date, and any underlying system components need to be upgraded before this `yum update` can succeed, yum will tell you what those are.

To upgrade to the new server

1. Stop the server on the 3.7.x Server host:

   ```
   $ sudo service cloudera-scm-server stop
   ```

2. If you are using the embedded PostgreSQL database, stop `cloudera-manager-server-db` on the host on which it is running:

   ```
   $ sudo service cloudera-scm-server-db stop
   ```

3. On every Cloudera Manager Agent host, run this command to stop the Cloudera Manager Agents:

   ```
   $ sudo service cloudera-scm-agent stop
   ```

4. Install the new version of the server. You can run commands on the Cloudera Manager Server host to update only the Cloudera Manager components.

   For a Red Hat system:

   To upgrade from Cloudera's repository run the following commands on the Cloudera Manager Server host:

   ```
   $ sudo yum clean all
   $ sudo yum update 'cloudera-*'
   ```

   On a SLES system:

   Use the following commands to clean the cached repository information and update only the Cloudera Manager components:

   ```
   $ sudo zypper clean --all
   $ sudo zypper up -r http://archive.cloudera.com/cm4/sles/11/x86_64/cm/4/
   ```

   At the end of this process you should have the 4.5 versions of the following packages installed on the host that will become the Cloudera Manager Server host. For example,

   ```
   $ rpm -qa 'cloudera-manager-*'
   cloudera-manager-daemons-4.5.0-1.cm450.p0.235.x86_64
   cloudera-manager-server-4.5.0-1.cm450.p0.235.x86_64
   cloudera-manager-agent-4.5.0-1.cm450.p0.235.x86_64
   ```

You may also see additional packages for plugins, depending on what was previously installed on the Server host.

5. Start the server. If you are using the embedded PostgreSQL database, start `cloudera-scm-server-db` on the Cloudera Manager Sever host:

   ```
   $ sudo service cloudera-scm-server-db start
   ```

   You will see it upgrade and create tables and databases. On the Cloudera Manager Server host (the system on which you installed the `cloudera-manager-server` package) do the following:

   ```
   $ sudo service cloudera-scm-server start
   ```

   You should see the following:

   ```
   Starting cloudera-scm-server:                              [  OK  ]
   ```

   Note:

   If you have problems starting the server, such as database permissions problems, you can use the server's log `/var/log/cloudera-scm-server/cloudera-scm-server.log` to troubleshoot the problem.

Step 3. Upgrade the Cluster Hosts

After updating Cloudera Manager, connect to Cloudera Manager and use the wizard to continue the upgrade process. In this part of the process, the Cloudera Manager agents and their databases are updated. The Host Monitor role is a new addition for Cloudera Manager 4, so upgrading includes adding this role and its supporting database. If you are adding new agents, such as the Host Monitor, you must have a database available to support these roles. For more information, see Installing and Configuring Databases.

Important:

All hosts in the cluster must have access to the Internet if you plan to use `archive.cloudera.com` as the source for installation files. If you do not have Internet access, create a custom repository.

1. Log in to the Cloudera Manager Admin Console. If you have just restarted the Cloudera Manager server, you may need to log in again.
2. On the Welcome screen, you'll be asked about allowing Cloudera to collect regular usage statistics, and you can review the Cloudera's **Terms and Conditions** and **Privacy Policy**.

   Click **Start Upgrade** to proceed to the Upgrade cluster hosts screen.

   You can click **Skip Host Upgrades** if you want to do this at a later time.
3. Select the release of the Cloudera Manager Agent to install. Normally, this will be the **Matched Release for this Cloudera Manager Server**. However, if you used a custom repository for the Cloudera Manager server, select **Custom Repository** and provide the required information Click **Continue** to proceed.
4. Provide credentials for authenticating with hosts.
   a. Select **root** or enter the user name for an account that has password-less sudo permissions.
   b. Select an authentication method.
      - If you choose to use password authentication, enter and confirm the password.
      - If you choose to use public-key authentication provide a passphrase and path to the required key files.
      - You can choose to specify an alternate SSH port. The default value is 22.
      - You can specify the maximum number of host installations to run at once. The default value is 10.

5. Click **Start Installation** to install and start Cloudera Manager Agents. The status of installation on each host is displayed on the page that appears after you click **Start Installation**. You can also click the **Details** link for individual hosts to view detailed information about the installation and error messages if installation fails on any hosts.

   > **Note:**
   >
   > If you click the **Abort Installation** button while installation is in progress, it will halt any pending or in-progress installations and roll back any in-progress installations to a clean state. The **Abort Installation** button does not affect host installations that have already completed successfully or already failed.

   If installation fails on a host, you can click the **Retry** link next to the failed host to try installation on that host again. To retry installation on all failed hosts, click **Retry Failed Hosts** at the bottom of the screen.

6. When the **Continue** button appears at the bottom of the screen, the installation process is complete. If the installation has completed successfully on some hosts but failed on others, you can click **Continue** if you want to skip installation on the failed hosts and continue to the next screen to start installing the Cloudera Management services on the successful hosts.
7. On the next screen, click **Continue** to install the Cloudera Management services.
8. The Host Inspector runs to validate your installation. This should show your currently installed components as CDH3, with CDH4 components shown as **Not installed**. Note that the Version will be shown as **Unavailable** for all components.
9. Select the Cloudera Management Service roles you want to install. The wizard evaluates the hardware configurations of the cluster hosts to recommend the best machines for each role. The Host Monitor is a new role introduced in Cloudera Manager 4.1. Navigator is a new, indepednetly-licensed feature introduced in Cloudera Manager 4.5.

   > **Important:**
   >
   > For best performance, make sure the Host Monitor role is assigned to the host on which you installed the corresponding databases. For example, if you created the Host Monitor database on `myhost1`, then you should assign the Activity Monitor role to `myhost1`. The JDBC connector **must** be installed and configured on any machine to which any of these roles is assigned.

   Click **Continue** to proceed.

10. On the Database Setup page, enter any required information for Host Monitor and Navigator databases.

    > **Important:**
    >
    > The value you enter as the database hostname **must** match the value you entered for the hostname (if any) when you created the database (see Installing and Configuring Databases).

    a. Enter the fully-qualified domain name for the server that is hosting the database in **Database Host Name**.
    b. Select the proper database type from the choices provided in **Database Type**.
    c. Enter the name you specified when you created the database in **Database Name**.
    d. Enter the user name you specified when you created the database in **Username**.
    e. Enter the password you specified when you created the database in **Password**.

    > **Note:**
    >
    > Problems may occur if a database with a blank password is used.

11. Click **Test Connection** to confirm that Cloudera Manager can communicate with the databases using the information you have supplied. This transaction takes two heartbeats to complete (about 30 seconds with

the default heartbeat interval). If the test succeeds in all cases, click **Continue**; otherwise check and correct the information you have provided for the databases and then try the test again.

12. Review the configuration changes to be applied during the upgrade and click **Accept**.
13. On the next page, select the hosts where the Hive Metastore Server role should be installed. The Hive service is now managed by Cloudera Manager; you must select the host for the Hive MetaStore Server. You should assign the Hive Metastore server to a single host. Click **Continue** to proceed.
14. Review the configuration values for your Hive roles, and click **Accept** to continue.

> ■ **Note:** If Hue is using a Hive metastore of type Derby (the default), then the newly created Hive service will also use Derby. However, since Derby does not allow concurrent connection, Hue will continue to work but the new Hive Metastore Server will fail to start. The failure is harmless (because nothing uses this new Hive Metastore Server at this point) and intentional, to preserve the cluster functionality that existed before the upgrade. Hive's Bypass Metastore Server mode is enabled by default when upgrading to Cloudera Manager 4.5. This is to ensure the upgrade doesn't disrupt your existing setup. You should plan to disable the Bypass Hive Metastore Server mode, especially when using CDH 4.2 or later. Using the Hive Metastore Server is the recommended configuration. After changing this configuration, you must re-deploy your client configurations, restart Hive, and restart any Hue or Impala services configured to use Hive.

15. You are now taken to the Hive service **Instances** page: The Hive metastore server will be stopped.
16. Under the **Services** tab, click the **All Services** link to go to the service overview page. All the services except for Hive and the Cloudera Management Service should now be running.

Step 4. Verify the Upgrade

You can use the host inspector to verify the upgrade completed.

To verify the upgrade has completed as expected

1. Connect to the Cloudera Manager Admin Console.
2. Click the **Hosts** tab.
3. Click **Host Inspector**.
4. Click **Show Inspector Results**. All results from the host inspector process are displayed including the currently installed versions. If this includes listings of current component versions, the installation completed as expected.

Step 5. Add Hive Gateway Roles to Hosts

1. Add Hive Gateway roles to any hosts where Hive clients should run.
2. In the Cloudera Manager Admin console, pull down the **Services** tab and select the Hive service.
3. Go to the **Instances** tab, and click the **Add** button. This opens the **Add Role Instances** page.
4. Select the hosts on which you want a Hive Gateway role to run. This will ensure that the Hive client configurations are deployed on these hosts.

Step 6. Restart the Cloudera Management Service

The Cloudera Management Services are not started automatically after an upgrade — you must restart them.

To start the Cloudera Management Service

1. Click the **Services** tab and select **All Services** in the Cloudera Manager Admin Console.
2. Choose **Start** on the **Actions** menu for the Cloudera Management Services. If you are running more than one cluster, you should do this for each one.

Step 7. Deploy Updated Client Configurations

During upgrades between major versions, resource locations may change. To ensure clients have current information about resources, update client configuration as described in Deploying Client Configuration Files.

Step 8. Restart Your Cluster(s)

From the **Actions** menu for each cluster, click **Restart**.

Step 9. (Optional) Upgrade CDH

Cloudera Manager 4.x can manage both CDH3 and CDH4, so upgrading existing CDH3 installations is not required, but to get the benefits of CDH4, you may want to upgrade to the latest version. See the following topics for more information on upgrading CDH:

- Upgrading CDH3 to CDH4 in a Cloudera Managed Deployment - Follow this path to upgrade existing installations of CDH3 to the latest version of CDH4. You can also install Impala when you upgrade to CDH4 version 4.1.2 or later.
- CDH3 Installation Guide - Follow the path describe in the Upgrading CDH3 topic to upgrade existing installations of CDH3 to the latest version of CDH3. Consider upgrading to CDH4 instead of upgrading to the latest version of CDH3.

Upgrade from Cloudera Manager 4 to the Latest Cloudera Manager 4

Upgrading from Cloudera Manager 4 to the latest version of Cloudera Manager is a relatively simple process, that primarily involves upgrading Cloudera Manager Server packages. This process applies to upgrading from a version Cloudera Manager 4 to a newer version of Cloudera Manager 4. For example, this process applies to upgrading Cloudera Manager 4.1.2 to Cloudera Manager 4.5.

To complete the upgrade, you stop the Cloudera Management Service, upgrade the packages (and database tables, if necessary), and then start the Cloudera Management Service again. After upgrading Cloudera Manager you may also want to upgrade CDH or add Impala.

It is possible to complete the following upgrade without shutting down the Hadoop services. Hadoop daemons can continue running, unaffected, while Cloudera Manager is upgraded.

> **Important:**
>
> Cloudera Manager 4.5 adds support for Hive, and includes a new role type called the Hive Metastore Server. This role manages the metastore process when Hive is configured with a Remote Metastore. When upgrading from a previous Cloudera Manager version, Cloudera Manager automatically creates new Hive service(s) to capture the previous implicit Hive dependency from Hue and Impala. Your previous services will continue to function without impact. Note that if Hue is using a Hive metastore of type Derby, then the newly created Hive service will also use Derby. But since Derby does not allow concurrent connections, Hue will continue to work, while the new Hive Metastore Server will fail to run. The failure is harmless (because nothing uses this new Hive Metastore Server at this point) and intentional, to preserve the set of cluster functionality as it was before upgrade. Cloudera discourages the use of a Derby metastore due to its limitations. You should consider switching to a different supported database type (PostgreSQL, MySQL, Oracle). Cloudera Manager provides a Hive configuration option to bypass the Hive Metastore server. When this configuration is enabled, Hive clients, Hue, and Impala connect directly to the Hive Metastore Database. Prior to Cloudera Manager 4.5, Hue and Impala talked directly to the Hive Metastore Database, so the Bypass mode is enabled by default when upgrading to Cloudera Manager 4.5. This is to ensure the upgrade doesn't disrupt your existing setup. You should plan to disable the Bypass Hive Metastore Server mode, especially when using CDH 4.2 or later. Using the Hive Metastore Server is the recommended configuration. After changing this configuration, you must re-deploy your client configurations, restart Hive, and restart any Hue or Impala services configured to use that Hive.

- **Warning:**

  Cloudera Manager 4.x can manage CDH3 and CDH4, but cannot manage CDH4.0 beta. If you upgrade to Cloudera Manager 4.x, you must upgrade any existing installations of CDH4.0 beta, as well.

Summary: What You are Going to Do

Upgrading from Cloudera Manager 4.0 to the latest version of Cloudera Manager involves the following broad steps:

Step 1. Stop the Cloudera Management Service and Hive

Step 2. Upgrade the Cloudera Manager Server and Agent Packages

Step 3. Start the Server

Step 4. Upgrade the Cluster Hosts

Step 5. Verify the Upgrade Succeeded

Step 6. Add Hive Gateway roles

Step 7. Restart the Services You Stopped

Step 8. Deploy Updated Client Configurations

Step 9. (Optional) Upgrade CDH

Step 1. Stop the Cloudera Management Service and (optionally) services that depend on the Hive metastore.

The Cloudera Manager Service must be stopped before upgrades can occur.

To stop the Cloudera Management Service

1. From the **Services** tab select **All Services** in the Cloudera Manager Admin Console.
2. Choose **Stop** on the **Actions** menu for the Cloudera Management Services.
3. If you are upgrading from Cloudera Manager 4.5 to a newer version, *and* you are using the embedded PostgreSQL database, you must stop the services that have a dependency on the Hive Metastore. You will not be able to stop the Cloudera Manager server's database while these services are running.
   - Choose **Stop** on the **Actions** menus for the Hive and Hue services. Do the same for Impala if you have it running.

Step 2. Upgrade the Cloudera Manager Server and Agent Packages

In this step, you upgrade the Cloudera Manager Server packages to the latest version.

1. Stop the server and the server's database on the Cloudera Manager Server host using the following commands:

   ```
   $ sudo service cloudera-scm-server stop
   ```

2. **If you are using the embedded PostgreSQL database for Cloudera Manager**, stop the database on the Cloudera Manager Server host:

   ```
   $ sudo service cloudera-scm-server-db stop
   ```

   If you are not using the embedded database, you should skip this step.
3. Install the new version of the server. To install the new version, you can upgrade from Cloudera's repository at `http://archive.cloudera.com/cm4/`. Alternately, you can create your own repository, as described in

Appendix A - Understanding Custom Installation Solutions. Creating your own repository is necessary if you are upgrading a cluster that does not have access to the Internet.

On a Red Hat system

1. Find Cloudera's repo file for your distribution by starting at `http://archive.cloudera.com/cm4/` and navigating to the directory that matches your operating system. For example, for RedHat or CentOS 6, you would navigate to `http://archive.cloudera.com/cm4/redhat/6/x86_64/cm/`. Within that directory, find the repo file that contains information including the repository's base URL and gpgkey. In the preceding example, the contents of the `cloudera-manager.repo` file might appear as follows:

   ```
   [cloudera-manager]
   # Packages for Cloudera Manager, Version 4, on RedHat or CentOS 5 x86_64
   name=Cloudera Manager
   baseurl=http://archive.cloudera.com/cm4/redhat/5/x86_64/cm/4/
   gpgkey = http://archive.cloudera.com/cm4/redhat/5/x86_64/cm/RPM-GPG-KEY-cloudera
   gpgcheck = 1
   ```

   Copy this repo file to the configuration location for the package management software for your system. For example, with Red Hat 6, you would copy the `cloudera-manager.repo` file to `/etc/yum.repos.d/`.

2. After verifying that you have the correct repo file, run the following commands:

   ```
   $ sudo yum clean all
   $ sudo yum update 'cloudera-*'
   ```

   Note:

   - `yum clean all` cleans up yum's cache directories, ensuring that you download and install the latest versions of the packages.
   - If your system is not up to date, and any underlying system components need to be upgraded before this `yum update` can succeed, yum will tell you what those are.

On a SLES system:

- To install the latest version from Cloudera's repository, run the following commands:

```
$ sudo zypper clean --all
$ sudo zypper up -r http://archive.cloudera.com/cm4/sles/11/x86_64/cm/4/
```

To download from your own repository:

```
$ sudo zypper clean --all
$ sudo zypper up -r http://myhost.example.com/path_to_cm_repo
```

On a Debian/Ubuntu system;

- Use the following commands to clean cached repository information and update Cloudera Manager components:

```
$ sudo apt-get clean
$ sudo apt-get update
$ sudo apt-get install cloudera-manager-server cloudera-manager-agent
cloudera-manager-daemons
```

As this process proceeds, you may be prompted concerning your configuration file version:

```
Configuration file `/etc/cloudera-scm-agent/config.ini'
==> Modified (by you or by a script) since installation.
==> Package distributor has shipped an updated version.
What would you like to do about it ? Your options are:
Y or I : install the package maintainer's version
N or O : keep your currently-installed version
D : show the differences between the versions
Z : start a shell to examine the situation
The default action is to keep your current version.
```

You will receive a similar prompt for `/etc/cloudera-scm-server/db.properties`. *Answer* **N** *to both these prompts.*

At the end of this process you should have the 4.5 versions of the following packages installed on the host that will become the Cloudera Manager Server host. For example,

```
$ rpm -qa 'cloudera-manager-*'
cloudera-manager-daemons-4.5.0-1.cm450.p0.235.x86_64
cloudera-manager-server-4.5.0-1.cm450.p0.235.x86_64
cloudera-manager-agent-4.5.0-1.cm450.p0.235.x86_64
```

You may also see additional packages for plugins, depending on what was previously installed on the Server host. If the commands to update the server complete without errors, you can assume the upgrade has completed as desired. For additional assurance, you will have the option to check that the server versions have been updated after you start the server. The process of checking the server version is described in Step 5. Verify the Upgrade Succeeded.

Step 3. Start the Server

To start the server

On the Cloudera Manager Server host (the system on which you installed the `cloudera-manager-server` package) do the following:

If you are using the embedded PostgreSQL database for Cloudera Manager:

```
$ sudo service cloudera-scm-server-db start
```

This will set up the new database for Cloudera Navigator.

> ■ **Note:**
>
> The `sudo service cloudera-scm-server-db start` command is not necessary if you are not using the embedded PostgreSQL database.

```
$ sudo service cloudera-scm-server start
```

You should see the following:

```
Starting cloudera-scm-server:                              [  OK  ]
```

> ■ **Note:**
>
> If you have problems starting the server, such as database permissions problems, you can use the server's log `/var/log/cloudera-scm-server/cloudera-scm-server.log` to troubleshoot the problem.

Step 4. Upgrade the Cluster Hosts

Cloudera Manager can automatically upgrade existing agents. After you upgrade Cloudera Manager, when it is started for the first time, it checks for any older versions of agents. If older agents are detected, Cloudera Manager provides the opportunity to automatically update agents, which is recommended.

To upgrade the agents

> **Important:**
>
> All hosts in the cluster must have access to the Internet if you plan to use `archive.cloudera.com` as the source for installation files. If you do not have Internet access, create a custom repository.

1. Log in to the Cloudera Manager Admin Console. If you have just restarted the Cloudera Manager server, you may need to log in again.
2. On the Welcome screen, click **Continue** to proceed to the Upgrade cluster hosts screen.
3. On the Upgrade cluster hosts screen, click **Start Upgrade** to upgrade the existing managed hosts. Click **Skip Host Upgrades** to skip this step.
4. Select the release of the Cloudera Manager Agent to install. Normally, this will be the **Matched Release for this Cloudera Manager Server**. However, if you used a custom repository for the Cloudera Manager server, select **Custom Repository** and provide the required information

   Click **Continue** to proceed.

5. Provide credentials for authenticating with hosts.

   a. Select **root** or enter the user name for an account that has password-less sudo permissions.
   b. Select an authentication method.

      - If you choose to use password authentication, enter and confirm the password.
      - If you choose to use public-key authentication provide a passphrase and path to the required key files.
      - You can choose to specify an alternate SSH port. The default value is 22.
      - You can specify the maximum number of host installations to run at once. The default value is 10.

6. Click **Start Installation** to install and start Cloudera Manager Agents. The status of installation on each host is displayed on the page that appears after you click **Start Installation**. You can also click the **Details** link for individual hosts to view detailed information about the installation and error messages if installation fails on any hosts.

   > **Note:**
   >
   > If you click the **Abort Installation** button while installation is in progress, it will halt any pending or in-progress installations and roll back any in-progress installations to a clean state. The **Abort Installation** button does not affect host installations that have already completed successfully or already failed.

   If installation fails on a host, you can click the **Retry** link next to the failed host to try installation on that host again. To retry installation on all failed hosts, click **Retry Failed Hosts** at the bottom of the screen.

7. When the **Continue** button appears at the bottom of the screen, the installation process is complete. If the installation has completed successfully on some hosts but failed on others, you can click **Continue** if you want to skip installation on the failed hosts and continue to the next screen to start installing the Cloudera Management services on the successful hosts.
8. The Host Inspector runs to inspect your managed hosts for correct versions and configurations. If there are problems, you can make changes and them re-run the inspector. When you are satisfied with the inspection results, click **Continue** to install the Cloudera Management services.

9. On the next page, select the hosts where the Hive Metastore Server role should be installed. The Hive service is now managed by Cloudera Manager; you must select the host for the Hive MetaStore Server. You should assign the Hive Metastore server to a single host.
10. Review the configuration values for your Hive roles, and click **Accept** to continue.

    Note:

    If Hue is using a Hive metastore of type Derby (the default), then the newly created Hive service will also use Derby. However, since Derby does not allow concurrent connection, Hue will continue to work but the new Hive Metastore Server will fail to start. The failure is harmless (because nothing uses this new Hive Metastore Server at this point) and intentional, to preserve the cluster functionality that existed before the upgrade. As mentioned previously, if you are upgrading to CM 4.5 or later from a release prior to 4.5 (i.e. 4.1 or earlier) Hive's metastore bypass mode is enabled by default. You should plan to disable the Bypass Hive Metastore Server mode, especially when using CDH 4.2 or later. Using the Hive Metastore Server is the recommended configuration. After changing this configuration, you must re-deploy your client configurations, restart Hive, and restart any Hue or Impala services configured to use that Hive.
11. Your services (except for Hive and the services you stopped in Step 1) should now be running.

Step 5. Verify the Upgrade Succeeded

If the commands to update and start the server complete without errors, you can assume the upgrade has completed as desired. For additional assurance, you can check that the server versions have been updated.

To verify the server upgrade succeeded

1. In the Cloudera Manager Admin console, click the **Hosts** tab.
2. Click **Host Inspector**. On large clusters, the host inspector may take some time to finish running. You must wait for the process to complete before proceeding to the next step.
3. Click **Show Inspector Results**.

   All results from the host inspector process are displayed including the currently installed versions. If this includes listings of current component versions, the installation completed as expected.

Step 6. Add Hive Gateway roles to hosts where Hive clients should run.

1. In the Cloudera Manager Admin console, pull down the **Services** tab and select the Hive service.
2. Go to the **Instances** tab, and click the **Add** button. This opens the **Add Role Instances** page.
3. Select the hosts on which you want a Hive Gateway role to run. This will ensure that the Hive client configurations are deployed on these hosts.

Step 7. Restart the services you stopped in Step 1.

You must restart the Management Service and any other services (Hive, Hue, Impala) that you stopped at the beginning of this procedure.

In addition, as of Cloudera Manager 4.1, health checks were introduced for the ZooKeeper service. If you are upgrading from a Cloudera Manager version older than 4.1 and have ZooKeeper installed, those new health checks will fail until you restart the ZooKeeper service.

To restart the ZooKeeper Service

1. From the **Services** tab select **All Services** in the Cloudera Manager Admin Console.
2. Choose **Restart** on the **Actions** menu for the ZooKeeper Service.

> **Note:**
>
> If for some reason you do not want to restart the ZooKeeper service at this point, you can disable the alerts for the failing health checks, or disable the health checks themselves. See Configuring Monitoring Settings. However, be sure to re-enable any checks you have disabled when you eventually restart the service. It is strongly recommended that you restart the service as soon as possible.

To start the services you stopped in Step 1

1. From the **Services** tab select **All Services** in the Cloudera Manager Admin Console.
2. Choose **Start** on the **Actions** menu for the each service you need to start.

To start the Cloudera Management Service

1. From the **Services** tab select **All Services** in the Cloudera Manager Admin Console.
2. Choose **Start** on the **Actions** menu for the Cloudera Management Services.

> **Note:**
>
> If you change the hostname or port where the Cloudera Manager is running, or you enable TLS security, you must restart the Cloudera Management Services to update the URL to the Server.

Test the Installation

When you have finished the upgrade to Cloudera Manager, you can test the installation to verify that the monitoring features are working as expected; follow instructions under Testing the Installation.

Step 8. Deploy Updated Client Configurations

During upgrades between major versions, resource locations may change. To ensure clients have current information about resources, update client configuration as described in Deploying Client Configuration Files.

Step 9. (Optional) Upgrade CDH

Cloudera Manager 4.x can manage both CDH3 and CDH4, so upgrading existing CDH3 installations is not required, but to get the benefits of CDH4, you may want to upgrade to the latest version. See the following topics for more information on upgrading CDH:

- Upgrading to the Latest Version of CDH4 in a Cloudera Managed Deployment - Follow this path to upgrade existing installations of CDH4 to the latest version of CDH4.
- Upgrading CDH3 to CDH4 in a Cloudera Managed Deployment - Follow this path to upgrade existing installations of CDH3 to the latest version of CDH4. You can also install Impala when you upgrade to CDH4 version 4.1.2 or later.
- Upgrading to the Latest Version of CDH3 in a Cloudera Managed Deployment - Follow this path to upgrade existing installations of CDH3 to the latest version of CDH3. Consider upgrading to CDH4 instead of upgrading to the latest version of CDH3.

Upgrade from Cloudera Manager Free Edition 4 to Cloudera Manager 4

Upgrading from Cloudera Manager Free Edition 4 to Cloudera Manager 4 involves installing the Cloudera Manager license.

Note:

Cloudera Manager 4 can continue to use the databases that were established for Cloudera Manager 4 Free Edition. Cloudera Manager 4 also supports other database types. To learn about using additional databases for Cloudera Manager, review the guidelines provided under Installing and Configuring Databases. Unless you want to change databases, no database installations are required.

Install the Cloudera Manager license

To install the license:

1. Log in to the Admin Console of Cloudera Manager Free Edition 4.
2. Click the gear icon ⚙ to display the **Administration** page.
3. On the **License** tab, browse to your Cloudera Manager License file to upgrade to the full version of Cloudera Manager. After you upload your license file, a message is displayed instructing you to restart Cloudera Manager to allow the new license to take effect.
4. Restart Cloudera Manager. This may be done by running `service cloudera-scm-server restart` from the command line.

   Note:

   After restarting the server, wait a few seconds for the server to finish initializing before you try to reconnect to the Admin Console.

5. Click **Continue** after restarting the server.
6. Log in to the Cloudera Manager Admin Console again.

   When you log back in, the Cloudera Manager upgrade is complete.

Upgrading CDH

Cloudera Manager 4 can manage both CDH3 and CDH4, so upgrading existing CDH3 installations is not required, but to get the benefits of CDH4, you may want to upgrade CDH. See the following topics for more information on upgrading CDH:

- Upgrading to the Latest Version of CDH4 in a Cloudera Managed Deployment - Follow this path to upgrade existing installations of CDH4 to the latest version of CDH4.
- Upgrading CDH3 to CDH4 in a Cloudera Managed Deployment - Follow this path to upgrade existing installations of CDH3 to the latest version of CDH4. You can also install Impala when you upgrade to CDH4 version 4.1.2 or later.
- Upgrading to the Latest Version of CDH3 in a Cloudera Managed Deployment - Follow this path to upgrade existing installations of CDH3 to the latest version of CDH3. Consider upgrading to CDH4 instead of upgrading to the latest version of CDH3.

Upgrading CDH in a Cloudera Manager Deployment

Cloudera Manager supports managing CDH4 and CDH3. To ensure the highest level of functionality and stability, consider upgrading to the most recent version of CDH4 or CDH3. The process of upgrading CDH to the most recent version is different when those CDH installations are managed by Cloudera Manager.

Upgrading CDH4 or CDH3 to the most recent version is described in the following topics:

- Upgrading CDH3 to CDH4 in a Cloudera Managed Deployment
- Upgrading to the Latest Version of CDH4 in a Cloudera Managed Deployment
- Upgrading to the Latest Version of CDH3 in a Cloudera Managed Deployment

Upgrading CDH3 to CDH4 in a Cloudera Managed Deployment

Important:

The following instructions describe how to upgrade components managed by Cloudera Manager from a CDH3 release to the latest CDH4 release. This involves uninstalling the CDH3 packages and installing the CDH4 packages.

If you are upgrading from a CDH4 release, use the instructions under Upgrading to the Latest Version of CDH4 in a Cloudera Managed Deployment instead.

For instructions on upgrading components, such as Flume, that Cloudera Manager does not manage, see the CDH4 Installation Guide.

Note:

As of Cloudera Manager 4.5, you can upgrade to CDH 4.1.3 (or later) within the Cloudera Manager Admin Console, using parcels and an upgrade wizard. This vastly simplifies the upgrade process. In addition, this will enable Cloudera Manager to automate the deployment and rollback of CDH versions. Electing to upgrade using packages means that future upgrades and rollbacks will still need to be done manually.

If you are running Cloudera Manager 4.5, and want to upgrade to CDH 4.1.3 or later, see Upgrading from CDH 3 to CDH 4.1.2 or Later Using the Upgrade Wizard for instructions. If you want to upgrade to a version of CDH4 earlier than 4.1.3, you will still need to follow the instructions below.

Before You Begin

Important:

Hive underwent major version changes at CDH 4.0, CDH 4.1 and CDH 4.2. (CDH 4.0 had Hive 0.8.0, CDH 4.1 used Hive 0.9.0, and 4.2 has 0.10.0). This requires the user to manually back up and upgrade their Hive metastore database when upgrading between a major Hive version. --- If you are upgrading between major versions, you must follow the steps in Step 4. "Upgrading Your Metastore" in the Hive Installation section in the CDH4 Installation Guide for upgrading the metastore **BEFORE** you start the Hive service. This applies whether you are upgrading using packages or parcels.

> **Important:**
>
> Before upgrading, be sure to read about the latest Incompatible Changes and Known Issues and Workarounds in the CDH4 Release Notes.

Plan Downtime

If you are upgrading a cluster that is part of a production system, be sure to plan ahead. As with any operational work, be sure to reserve a maintenance window with enough extra time allotted in case of complications. The Hadoop upgrade process is well understood, but it is best to be cautious. For production clusters, Cloudera recommends allocating up to a full day maintenance window to perform the upgrade, depending on the number of hosts, the amount of experience you have with Hadoop and Linux, and the particular hardware you are using.

If Security is Enabled, put the NameNode into Safe Mode

> **Important:**
>
> If you have security enabled, you must do this prior to stopping services (if you are upgrading with packages) or prior to starting the Upgrade Wizard to upgrade with parcels.

If security is enabled, put the NameNode into Safemode and save the Namespace (see the CDH3 Security Guide for more information about CDH3 security):

1. In the Cloudera Manager Admin console, go the the HDFS service, NameNode role instance.
2. From the NameNode role's **Actions** menu, click **Enter Safemode...** and confirm that you want to do this.
3. After the NameNode has successfully entered Safemode, from the **Actions** menu, click **Save Namespace...** and confirm that you want to do this.

   This will result in a new `fsimage` being written out with no edit log entries.
4. Leave the NameNode in safe mode while you proceed with the instructions to do the upgrade.

To upgrade CDH in multiple clusters, repeat this process for each cluster.

Upgrading from CDH 3 to CDH 4.1.2 or Later Using the Upgrade Wizard

> **Important:**
>
> If you have security enabled, make sure you have run the commands to put the NameNode into safe mode, as described above in Before You Begin.

> **Important:**
>
> If you set up your secure CDH3 cluster using Cloudera Manager 4.1, and have subsequently upgraded your Cloudera Manager to version 4.5, upgrading your CDH3 cluster will fail due to a missing HTTP principal in the NameNode's keytab. This is not a problem if you installed CDH3 using Cloudera Manager 4.5. See Known Issues and Workarounds in Cloudera Manager Enterprise Edition 4 for more information.

You can upgrade from CDH3 packages to CDH4.1.2 or CDH4.2 parcels from within the Cloudera Manager Admin Console. Not only is this process more streamlined than doing it manually with packages, but it also allows Cloudera Manager to automate the deployment and rollback of CDH versions in the future.

1. In the Cloudera Manager Admin Console, click the Parcels indicator in the top navigation bar (1) to go to the Parcels page.

2. In the parcels page, click **Download** for the version you want to download.
3. When the download has completed, click **Distribute** for the version you downloaded.
4. When the parcel has been distributed and unpacked, the button will change to say **Upgrade**.
5. **BEFORE YOU CLICK UPGRADE** verify that the `/user/oozie` directory exists.

   - If it does not exist, create it before continuing with the upgrade wizard.

6. Click the Upgrade button and in the pop-up that appears, read the information and click **Upgrade Cluster** to proceed. -- You can also select whether this process should start your services and deploy the client configuration as part of the upgrade.

   > **Important:**
   >
   > If you are using Hive,**DO NOT** elect to have the upgrade wizard start the services – you must upgrade your Hive metastore before you restart Hive.

   The upgrade process will execute the commands to stop your services, convert your configuration parameters, upgrade your HDFS metadata and the Oozie database and ShareLib. When the upgrade has finished, the **All Services** page will appear.

7. If Hue is configured to use SQLite as its database, back up the `desktop.db` kept at `/usr/share/hue/desktop/desktop.db` to a temporary location.

   > **Important:** Removing the Hue Common package will remove your Hue database; if you do not back it up you may lose all your Hue user account information.

   Start the new Hue service before you remove your CDH3 packages.

8. Uninstall CDH3 on each host.

   - On Red Hat-compatible systems:

   ```
   $ sudo yum remove hadoop-0.20 hue-common hadoop-pig oozie-client hive hadoop-hbase
    hadoop-zookeeper bigtop-utils
   ```

   - On SUSE systems:

   ```
   $ sudo zypper remove hadoop-0.20 hue-common hadoop-pig oozie-client hive
   hadoop-hbase hadoop-zookeeper bigtop-utils
   ```

   - On Ubuntu and Debian systems:

   ```
   $ sudo apt-get purge hadoop-0.20 hue-common hadoop-pig oozie-client hive
   hadoop-hbase hadoop-zookeeper bigtop-utils
   ```

9. Force an update of the symlinks to point to the newly installed components. There are two ways you can do this:

   - Restart all the Cloudera Manager agents, *OR*
   - Disable and re-enable the "Create symlinks" property through the Cloudera Manager Admin console. Restarting the agents re-triggers symlink creation at agent start up. Unsetting/resetting the symlink property can be done via the Admin console, but it requires waiting at each step until all the agents have synced (at least one heartbeat – harder to verify) for this to be effective. Do one of the following:

     To restart the Cloudera Manager agents:

     On each host:

     ```
     $ sudo service cloudera-scm-agent restart
     ```

To force a re-sync via the Admin Console:

1. In the Cloudera Manager Admin Console, click the gear icon at the right end of the navigation bar to go to the **Admin** page.
2. From the **Properties** tab, **Parcels** section, uncheck the property **Create System-Wide Symlinks for Active Parcels** and **Save** your change.
3. Now, **wait for at least one heartbeat**, then restore the check to the **Create System-Wide Symlinks for Active Parcels** property and **Save** your change.

10. Go to Step 9. Upgrade your Hive Metastore below and follow the instructions there to upgrade the Hive metastore.
11. Restart the Services you Stopped
    a. In the Cloudera Manager Admin Console, click the **Services** tab.
    b. Click the top **Actions** button that corresponds to the cluster and choose **Restart**. The **Command Details** window shows the progress of starting services.
12. Redeploy the client configuration files.
    a. From the top **Actions** button that corresponds to the cluster and choose **Deploy Client Configuration...**.
    b. Click the **Deploy Client Configuration** button in the confirmation pop-up that appears.
13. Go to Step 12: Finalize the HDFS Metadata Upgrade and follow the instructions there to finish the installation process.

Upgrading Using Packages

Use the instructions that follow to upgrade to CDH4.

Step 1: Back Up Important Items

- **Important:**

  Do this step now if your cluster includes any Ubuntu or Debian systems running CDH3u3 or earlier.

  Otherwise, you can perform this step later if you prefer – any time before you use Cloudera Manager to upgrade the cluster.

1. Back up the databases. For instructions, see Database Considerations for Cloudera Manager Upgrades.
2. If the directory `/usr/lib/oozie/libext` exists, move it to a temporary location before you proceed.

Step 2: Stop All CDH Components

- **Important:**

  If you have security enabled, make sure you have run the commands to put the NameNode into safe mode, as described above in Before You Begin.

To stop all services

1. In the Cloudera Manager Admin Console, select **Services > All Services**.
2. Click the top **Actions** button that corresponds to the cluster and choose **Stop...**. Click **Stop** in the confirmation screen. The **Command Details** window shows the progress of stopping services. When **All services successfully stopped** appears, the task is complete and you may close the **Command Details** window.
3. For each Cloudera Management Service entry, click **Actions** and click **Stop...**. Click **Stop** in the confirmation screen. The **Command Details** window shows the progress of stopping services. When **All services successfully stopped** appears, the task is complete and you may close the **Command Details** window.

Step 3: Back up the HDFS Metadata

Back up the HDFS metadata on the NameNode machine. Back up the HDFS metadata on the NameNode machine.

> ■ **Important:**
>
> Do the following when you are sure that all Hadoop services have been shut down. **It is particularly important that the NameNode service is not running so that you can make a consistent backup.**

Cloudera recommends backing up HDFS metadata on a regular basis, as well as before a major upgrade.

1. On the **Services** page of Cloudera Manager, click the link for the HDFS service. Click the **Configuration** tab and click **Edit**. On that page find the name of the NameNode Data Directories (in **NameNode (Base)**).
2. From the command line on the NameNode machine, back up that directory; for example, if the data directory is `/mnt/hadoop/hdfs/name`, do the following as root:

```
# cd /mnt/hadoop/hdfs/name
# tar -cvf /root/nn_backup_data.tar .
```

   You should see output like this:

```
./
./current/
./current/fsimage
./current/fstime
./current/VERSION
./current/edits
./image/
./image/fsimage
```

3. Check the output.

> ■ **Warning:**
>
> If you see a file containing the word *lock*, the NameNode is probably still running. Repeat the preceding steps, starting by shutting down the Hadoop services.

If you need to restore HDFS metadata, refer to Cloudera's Knowledge Base article, "How do I recover a failed NameNode?"

Step 4: Uninstall CDH3

Uninstall CDH3 on each host.

- On Red Hat-compatible systems:

```
$ sudo yum remove hadoop-0.20 hue-common hadoop-pig oozie-client hive hadoop-hbase
hadoop-zookeeper bigtop-utils
```

- On SUSE systems:

```
$ sudo zypper remove hadoop-0.20 hue-common hadoop-pig oozie-client hive hadoop-hbase
 hadoop-zookeeper bigtop-utils
```

- On Ubuntu and Debian systems:

```
$ sudo apt-get purge hadoop-0.20 hue-common hadoop-pig oozie-client hive hadoop-hbase
 hadoop-zookeeper bigtop-utils
```

■ **Warning:**

If you are upgrading an Ubuntu or Debian system from CDH3u3 or earlier, you **must** use `apt-get purge` (rather than `apt-get remove`) to make sure the re-install succeeds, but be aware that `apt-get purge` removes all your configuration data. If you have modified any configuration files, DO NOT PROCEED before backing them up.

Step 5: Download CDH4

On Red Hat-compatible systems:

1. Download the CDH4 Package:

   a. Click the entry in the table below that matches your Red Hat or CentOS system, choose **Save File**, and save the file to a directory to which you have write access (it can be your home directory).

For OS Version	Click this Link
Red Hat/CentOS/Oracle 5	Red Hat/CentOS/Oracle 5 link
Red Hat/CentOS 6 (32-bit)	Red Hat/CentOS 6 link (32-bit)
Red Hat/CentOS 6 (64-bit)	Red Hat/CentOS 6 link (64-bit)

   b. Install the RPM. For Red Hat/CentOS/Oracle 5:

```
$ sudo yum --nogpgcheck localinstall cloudera-cdh-4-0.x86_64.rpm
```

   For Red Hat/CentOS 6 (32-bit):

```
$ sudo yum --nogpgcheck localinstall cloudera-cdh-4-0.i386.rpm
```

   For Red Hat/CentOS 6 (64-bit):

```
$ sudo yum --nogpgcheck localinstall cloudera-cdh-4-0.x86_64.rpm
```

■ **Note:**

For instructions on how to add a CDH4 yum repository or build your own CDH4 yum repository, see the topic Installing CDH4 On Red Hat-compatible systems in the CDH4 Installation Guide.

2. (Optionally) add a repository key on each system in the cluster. Add the Cloudera Public GPG Key to your repository by executing one of the following commands:

   - **For Red Hat/CentOS/Oracle 5 systems:**

```
$ sudo rpm --import
http://archive.cloudera.com/cdh4/redhat/5/x86_64/cdh/RPM-GPG-KEY-cloudera
```

   - **For Red Hat/CentOS 6 systems:**

```
$ sudo rpm --import
http://archive.cloudera.com/cdh4/redhat/6/x86_64/cdh/RPM-GPG-KEY-cloudera
```

On SUSE systems:

1. Download the CDH4 Package:

   a. Click this link, choose **Save File**, and save it to a directory to which you have write access (it can be your home directory).
   b. Install the RPM:

      ```
      $ sudo rpm -i cloudera-cdh-4-0.x86_64.rpm
      ```

   > **Note:**
   >
   > For instructions on how to add a repository or build your own repository, see the topic on Installing CDH4 on SUSE Systems in the CDH4 Installation Guide.

2. Update your system package index by running:

   ```
   $ sudo zypper refresh
   ```

3. (Optionally) add a repository key on each system in the cluster. Add the Cloudera Public GPG Key to your repository by executing the following command:

- **For all SLES systems:**

```
$ sudo rpm --import
http://archive.cloudera.com/cdh4/sles/11/x86_64/cdh/RPM-GPG-KEY-cloudera
```

On Ubuntu and Debian systems:

1. Download the CDH4 Package:
   a. Click one of the following: this link for a Squeeze system, *or* this link for a Lucid system this link for a Precise system.
   b. Install the package. Do one of the following: Choose **Open with** in the download window to use the package manager, *or* Choose **Save File**, save the package to a directory to which you have write access (it can be your home directory) and install it from the command line, for example:

      ```
      sudo dpkg -i cdh4-repository_1.0_all.deb
      ```

   > **Note:**
   >
   > For instructions on how to add a repository or build your own repository, see the topic on Installing CDH4 on Ubuntu Systems in the CDH4 Installation Guide.

   (Optionally) add a repository key on each system in the cluster. Add the Cloudera Public GPG Key to your repository by executing one of the following commands:

- **For Ubuntu Lucid systems:**

```
$ curl -s http://archive.cloudera.com/cdh4/ubuntu/lucid/amd64/cdh/archive.key | sudo
apt-key add -
```

- **For Ubuntu Precise systems:**

```
$ curl -s http://archive.cloudera.com/cdh4/ubuntu/precise/amd64/cdh/archive.key | sudo
 apt-key add -
```

- **For Debian Squeeze systems:**

```
$ curl -s http://archive.cloudera.com/cdh4/debian/squeeze/amd64/cdh/archive.key | sudo
 apt-key add -
```

Step 6: Re-Install HDFS, MapReduce, and the CDH4 Components

- Use one of the following commands to install CDH4 packages on every host in your cluster: **On Red Hat/CentOS/Oracle systems:**

```
$ sudo yum -y install bigtop-utils bigtop-jsvc bigtop-tomcat hadoop hadoop-hdfs
hadoop-httpfs hadoop-mapreduce hadoop-yarn hadoop-client hadoop-0.20-mapreduce hbase
hive oozie oozie-client pig zookeeper mahout
```

On SUSE systems:

```
$ sudo zypper install bigtop-utils bigtop-jsvc bigtop-tomcat hadoop hadoop-hdfs
hadoop-httpfs hadoop-mapreduce hadoop-yarn hadoop-client hadoop-0.20-mapreduce hbase
hive oozie oozie-client pig zookeeper mahout
```

On Debian/Ubuntu systems:

```
$ sudo apt-get install bigtop-utils bigtop-jsvc bigtop-tomcat hadoop hadoop-hdfs
hadoop-httpfs hadoop-mapreduce hadoop-yarn hadoop-client hadoop-0.20-mapreduce hbase
hive oozie oozie-client pig zookeeper mahout
```

- To install the `hue-common` package and all Hue applications on the Hue machine, install the hue meta-package.

> ■ **Important:**
>
> If you used the Hue Authorization Manager with CDH3, you must remove the `hue-userman` package, and disable or remove the Authorization Manager repository before installing the new version of Hue. The repository is the one you installed when you configured the Authorization Manager in Hue. For example, on a Red Hat system, the repository file is `/etc/yum.repos.d/cloudera-authman.repo` by default. Either remove this file or add a line that reads `enabled=0` (or, if there is already a line that reads `enabled=1`, change the 1 to a 0).

To install the hue meta-package on Red Hat/CentOS/Oracle systems:

```
$ sudo yum install hue
```

To install the hue meta-package on SUSE systems:

```
$ sudo zypper install hue
```

To install the hue meta-package on Debian/Ubuntu systems:

```
$ sudo apt-get install hue
```

- If you moved `/usr/lib/oozie/libext` to a temporary location in Step 1, copy **its contents** (not the directory itself) back to the new `/usr/lib/oozie/libext` now.

Step 7: Disable Start on Boot for Hue, Oozie, and HttpFS:

- To prevent Hue from starting on the Hue machine:

Note:

Preventing Hue from starting is only required on CDH3u5 and earlier. If you are running newer versions of CDH3u5 or later or CDH4, you do not need to stop Hue from starting. While stopping Hue from starting is not required in those cases, executing the `chkconfig` command generates an error, but there are no other negative effects.

```
$ sudo /sbin/chkconfig hue off
```

- To prevent Oozie from starting on system boot on every machine on which it is installed:
- On Red Hat-compatible and SUSE systems:

```
$ sudo /sbin/chkconfig oozie off
```

- On Ubuntu and Debian systems:

```
$ sudo /usr/sbin/update-rc.d oozie disable
```

- To prevent HttpFS from starting on system boot:
- On Red Hat-compatible and SUSE systems:

```
$ sudo /sbin/chkconfig hadoop-httpfs off
$ sudo service hadoop-httpfs stop
```

- On Ubuntu and Debian systems:

```
$ sudo /usr/sbin/update-rc.d hadoop-httpfs disable
$ sudo service hadoop-httpfs stop
```

Step 8: Upgrade the HDFS Metadata and the Cluster Configuration

Important:

- If you have not already backed up your configuration data, do so now.
- Before you proceed, click on the **Hosts** tab in Cloudera Manager and make sure that all hosts are up and running CDH4.

To upgrade the cluster

1. In the Cloudera Manager Admin Console, click the **Services** tab, click **Actions** and click **Upgrade Cluster**.
2. Click **Upgrade Cluster** to confirm you want to upgrade the cluster.

Note: If you are already on the **Services** page, the **Upgrade Cluster** may not be available. If this occurs, refresh the page.

--- You can also select whether this process should start your services and deploy the client configuration as part of the upgrade.

Important: If you are using Hive, **DO NOT** elect to have the upgrade wizard start the services – you must upgrade your Hive metastore before you restart Hive.

The upgrade process will execute the commands to stop your services, convert your configuration parameters, upgrade your HDFS metadata and the Oozie database and ShareLib. Cloudera Manager updates the configuration, upgrades HDFS metadata, and upgrades the Oozie database. Upgrading CDH from CDH3 or CDH4 Beta 1 requires these changes.

Step 9. Upgrade your Hive Metastore

1. (Strongly recommended) Make a backup copy of your Hive metastore database.
2. Follow the instructions at Step 4. "Upgrading Your Metastore" in the Hive Installation section in the CDH4 Installation Guide to run the metastore upgrade script.

   - If you are installing to packages, the upgrade script is at `/usr/lib/hive/scripts/metastore/upgrade/<database>`
   - If you are upgrading to parcels, then the upgrade script is located at `/opt/cloudera/parcels/<parcel_name>/lib/hive/scripts/metastore/upgrade/<database>`. `<parcel_name>` should be the name of the parcel to which you have upgraded. `<database>` is the type of database you are running (i.e. mysql, postgres, etc.) For example, if you are installing a CDH 4.2.0 parcel using the default location for the local repository, and using the default database (PostgreSQL) the script will be at: `/opt/cloudera/parcels/CDH-4.2.0-1.cdh4.2.0.p0.10-e16.parcel/lib/hive/scripts/metastore/upgrade/postgres`
   - You must `cd` to the directory the scripts are in.
   - Execute the script in the appropriate DB command shell.

   > **Important:**
   >
   > You must know the password for the database; if you installed Cloudera Manager using the default (embedded PostgreSQL) database, the password was displayed on the Database Setup page during the Cloudera Manager installation wizard. If you do not know the password for your Hive metastore database, you can find it as follows:
   >
   > - `cat /etc/cloudera-scm-server/db.properties` This shows you Cloudera Manager's internal database credentials.
   > - Run the following command:
   >
   > ```
   > psql -p 7432 -U scm scm -c "select s.display_name as hive_service_name,
   >  s.name as hive_internal_name, c.value as metastore_password from configs
   >  c, services s where attr='hive_metastore_database_password' and
   > c.service_id = s.service_id"
   > ```
   >
   > - Use the password from `com.cloudera.cmf.db.password`. This will output the passwords for the hive service metastore as follows:
   >
   > ```
   >  hive_service_name | hive_internal_name | metastore_password
   > -------------------+--------------------+--------------------
   >  hive1             | hive1              | lF3Cv2zsvI
   > (1 row)
   > ```

3. If you have multiple instances of Hive, run the upgrade script on each metastore database.

Step 10. Start the Services you Stopped

1. In the Cloudera Manager Admin Console, click the **Services** tab.
2. Click the top **Actions** button that corresponds to the cluster and choose **Start**. The **Command Details** window shows the progress of starting services. When **All services successfully started** appears, the task is complete and you may close the **Command Details** window.

Step 11: Redeploy the Client Configuration Files

1. From the top **Actions** button that corresponds to the cluster and choose **Deploy Client Configuration...**.
2. Click the **Deploy Client Configuration** button in the confirmation pop-up that appears.

Step 12: Finalize the HDFS Metadata Upgrade

After ensuring that the CDH4 upgrade has succeeded and that everything is running smoothly, finalize the HDFS metadata upgrade. It is not unusual to wait days or even weeks before finalizing the upgrade. **To finalize the HDFS metadata upgrade**

1. In the Cloudera Manager Admin Console, pull down the **Services** tab and go to the HDFS service.
2. Go the the **Instances** tab and click on the **NameNode** instance.
3. From the NameNode Status page, from the **Actions** menu click **Finalize Metadata Upgrade**.
4. Click **Finalize Metadata Upgrade** to confirm you want to complete this process.

   Cloudera Manager finalizes the metadata upgrade. The upgrade is now complete.

Upgrading to the Latest Version of CDH4 in a Cloudera Manager Deployment

■ **Important:**

Use the right instructions: the following instructions describe how to upgrade to the latest CDH4 release from an earlier CDH4 release in a Cloudera Managed Deployment. **If you are upgrading from a CDH3 release, use the instructions under Upgrading CDH3 to CDH4 in a Cloudera Managed Deployment instead.**

■ **Note:**

As of Cloudera Manager 4.5, you can upgrade to CDH 4.1.2 (or later) within the Cloudera Manager Admin Console, using parcels and an upgrade wizard. This vastly simplifies the upgrade process. In addition, this will enable Cloudera Manager to automate the deployment and rollback of CDH versions. Electing to upgrade using packages means that future upgrades and rollbacks will still need to be done manually.

If you are running Cloudera Manager 4.5, and want to upgrade to CDH 4.1.2 or later, see Upgrading Using the Upgrade Wizard for instructions. If you want to upgrade to a version of CDH4 earlier than 4.1.2, you will still need to follow the instructions below.

■ **Note:**

Upgrading Impala: If you have CDH 4.1.x and **Cloudera Impala** installed, and you plan to upgrade to CDH 4.2, you must also upgrade Impala from the beta version to version 1.0. With a parcel installation you can download and activate both parcels before you proceed to restart the cluster.

You will need to change the remote parcel repo URL to point to the location of the released product. Instructions are included below.

Before You Begin

- Before upgrading, be sure to read about the latest Incompatible Changes and Known Issues and Workarounds in the CDH4 Release Notes.
- If you are upgrading a cluster that is part of a production system, be sure to plan ahead. As with any operational work, be sure to reserve a maintenance window with enough extra time allotted in case of complications. The Hadoop upgrade process is well understood, but it is best to be cautious. For production clusters, Cloudera recommends allocating up to a full day maintenance window to perform the upgrade, depending on the number of hosts, the amount of experience you have with Hadoop and Linux, and the particular hardware you are using.

> ■ **Important:**
>
> Hive underwent a major version change between CDH 4.0 to 4.1 and between CDH 4.1 and 4.2. (CDH 4.0 had Hive 0.8.0, CDH 4.1 used Hive 0.9.0, and 4.2 has 0.10.0). This requires the user to manually back up and upgrade their Hive metastore database when upgrading between a major Hive version. --- If you are upgrading between major versions, you must follow the steps in Step 4. "Upgrading Your Metastore" in the Hive Installation section in the CDH4 Installation Guide for upgrading the metastore **BEFORE** you start the Hive service. This applies whether you are upgrading using packages or parcels.

Rolling Upgrade from CDH 4.x to CDH 4.1.2 or later or later Using Parcels

If you are performing an upgrade within CDH 4, you may optionally follow the Rolling Upgrade procedure if you have enabled HDFS High Availability.

Upgrading to CDH 4.1.2 or Later Using the Upgrade Wizard

If you want to upgrade to CDH 4.1.2 or later, you can do the upgrade using parcels from within the Cloudera Manager Admin Console.

If you plan to upgrade Impala:

You must point the Impala parcel repo URL to the released Impala parcel:

1. From the **Administration** tab, select **Properties**.
2. Go to the **Parcels** category.
3. Under the **Remote Parcel Repository URLs** property, find the entry `http://beta.cloudera.com/impala/parcels/` and replace it with `http://archive.cloudera.com/impala/parcels/`.
4. Save your change.

Now you can proceed to upgrade your installation.

Step 1. Download, Distribute, and Activate the CDH4 (and Impala) Parcels.

1. In the Cloudera Manager Admin Console, click the Parcels indicator in the top navigation bar (⊞ or ⊞1) to go to the Parcels page.
2. In the parcels page, click **Download** for the version(s) you want to download. If you want to run both CDH and Cloudera Impala, you should download both the CDH and Impala parcels.
3. When the download has completed, click **Distribute** for the version you downloaded.
4. When the parcel has been distributed and unpacked, the button will change to say **Activate**.
5. Click **Activate**. This will display a pop-up that will offer to restart your services. **DO NOT RESTART** services at this point – click **Close** to remove the pop-up.

> ■ **Important:**
>
> If you are upgrading between major Hive versions (i.e from CDH 4.0 to 4.1 or 4.2, or from CDH 4.1 to 4.2) **DO NOT** restart the services – you must upgrade your Hive metastore before you restart Hive.

Step 2. Upgrade the Hive Metastore

Go to Step 4. Upgrade your Hive Metastore below (under "Upgrading using Packages") and follow the instructions there to upgrade the Hive metastore.

Step 3. (If Upgrading to CDH 4.2) Upgrade the Oozie Sharelib

1. In the Cloudera Manager Admin Console, select **Oozie** from the **Services** tab.
2. If you have already restarted service, then stop the Oozie service. From the **Actions** button, choose **Stop**.
3. When the service has stopped, from the **Actions** button choose **Install Oozie Sharelib**. The commands to perform this function are run.

Step 4. Restart the Services

1. In the Cloudera Manager Admin Console, select **All Services** from the **Services** tab.
2. Click the top **Actions** button that corresponds to the cluster and choose **Restart**. The **Command Details** window shows the progress of starting services.

Step 5. Deploy the new client configuration files

1. From the top **Actions** button that corresponds to the cluster and choose **Deploy Client Configuration...**.
2. Click the **Deploy Client Configuration** button in the confirmation pop-up that appears.

Step 6. Remove the previous CDH version packages.

If your previous installation of CDH 4 (4.0.x or 4.1.x) was done using packages, you must remove those packages and refresh the symlinks so that clients will run the new software versions.

1. If Hue is configured to use SQLite as its database, back up the `desktop.db` to a temporary location before deleting the old Hue Common package. The location of the database can be found in the Hue service **Configuration** tab under **Service** > **Database** > **Hue's Database Directory**.

   > ■ **Important:** Removing the Hue Common package will remove your Hue database; if you do not back it up you may lose all your Hue user account information.

   Make sure the new Hue service is running before you remove the old packages.

2. To uninstall the CDH packages (*not including Impala*):

   On Red Hat systems:

   ```
   $ sudo yum remove hadoop hue-common bigtop-jsvc bigtop-tomcat
   ```

   On SLES systems:

   ```
   $ sudo zypper remove hadoop hue-common bigtop-jsvc bigtop-tomcat
   ```

   On Debian/Ubuntu systems:

   ```
   $ sudo apt-get purge hadoop hue-common bigtop-jsvc bigtop-tomcat
   ```

3. To uninstall the CDH packages *including Impala*:

   On Red Hat systems:

   ```
   $ sudo yum remove hadoop hue-common 'bigtop-*'
   ```

   On SLES systems:

   ```
   $ sudo zypper remove hadoop hue-common 'bigtop-*'
   ```

   On Debian/Ubuntu systems:

   ```
   $ sudo apt-get purge hadoop hue-common 'bigtop-*'
   ```

Step 7. Update symlinks for the newly installed components.

There are two ways you can do this:

- Restart all the Cloudera Manager agents, *OR*
- Disable and re-enable the "Create symlinks" property through the Cloudera Manager Admin console.

Restarting the agents re-triggers symlink creation at agent start up. Unsetting/resetting the symlink property can be done via the Admin console, but it requires waiting at each step until all the agents have synced (at least one heartbeat – harder to verify) for this to be effective.

Do one of the following:

To restart the Cloudera Manager agents:

On each host:

```
$ sudo service cloudera-scm-agent restart
```

To force a re-sync via the Admin Console:

1. In the Cloudera Manager Admin Console, click the gear icon at the right end of the navigation bar to go to the **Administration** page.
2. From the **Properties** tab, **Parcels** section, uncheck the property **Create System-Wide Symlinks for Active Parcels** and **Save** your change.
3. Now, **wait for at least one heartbeat**, then restore the check to the **Create System-Wide Symlinks for Active Parcels** property and **Save** your change.

Upgrading Using Packages

Upgrading Unmanaged Components

Upgrading unmanaged components is a process that is separate from upgrading managed components. Upgrade the unmanaged components before proceeding to upgrade managed components. For example, if you have unmanaged Flume installed, upgrade that before proceeding to upgrade managed components. Components that you might have installed that are not managed by Cloudera Manager include:

- Flume 1.x
- Sqoop
- Pig
- Whirr
- Mahout

For information on upgrading these unmanaged components, see CDH4 Installation Guide.

Step 1. Stop all the CDH Services on All Hosts

You must stop all Hadoop services before upgrading CDH.**To stop all services**

1. In the Cloudera Manager Admin Console, select **Services > All Services**.
2. Click the top **Actions** button that corresponds to the cluster and choose **Stop...**.

   Click **Stop** in the confirmation screen. The **Command Details** window shows the progress of stopping services.

   When **All services successfully stopped** appears, the task is complete and you may close the **Command Details** window.
3. For each Cloudera Management Service entry, click **Actions** and click **Stop...**. Click **Stop** in the confirmation screen.

   The **Command Details** window shows the progress of stopping services.

When **All services successfully stopped** appears, the task is complete and you may close the **Command Details** window.

Repeat this process for all clusters hosting CDH4 machines to be upgraded.

Step 2. Back up the HDFS Metadata on the NameNode

> **Important:**
>
> Do the following when you are sure that all Hadoop services have been shut down. **It is particularly important that the NameNode service is not running so that you can make a consistent backup.**

> **Note:** Cloudera recommends backing up HDFS metadata on a regular basis, as well as before a major upgrade.

1. On the **Services** page of Cloudera Manager, click the HDFS service, then the **Configuration** tab. Navigate to the **NameNode** category and find **NameNode Data Directory**.
2. From the command line on the NameNode machine, back up that directory; for example, if the data directory is `/mnt/hadoop/hdfs/name`, do the following as root:

   ```
   # cd /mnt/hadoop/hdfs/name
   # tar -cvf /root/nn_backup_data.tar .
   ```

   You should see output like this:

   ```
   ./
   ./current/
   ./current/fsimage
   ./current/fstime
   ./current/VERSION
   ./current/edits
   ./image/
   ./image/fsimage
   ```

3. Check the output.

   > **Warning:**
   >
   > If you see a file containing the word *lock*, the NameNode is probably still running. Repeat the preceding steps, starting by shutting down the Hadoop services.

Step 3. Upgrade Managed Components

There are a variety of strategies that you can use to upgrade to the latest version of CDH4.

- You can use your operating system's package management tools to update all packages to the latest version using standard repositories. This approach works well because it minimizes the amount of configuration required and uses the simplest commands. Be aware that this can take a considerable amount of time if you have not upgraded the system recently.
- You can target the `cloudera.com` repository that is added during a typical install, only updating Cloudera components. This limits the scope of updates to be completed, so the process takes less time. This will not work if you created and used a custom repository.
- You can use a custom repository. This process can be more complicated, but enables updating Cloudera components for CDH machines that are not connected to the Internet.

Updating Everything

You can update all components on your system, including Cloudera components. Note that this may take a significant amount of time. To update all packages on your system, use the following command:

- On Red Hat systems:

```
$ sudo yum update
```

- On SLES systems:

```
$ sudo zypper up
```

- On Ubuntu/Debian systems:

```
$ sudo apt-get upgrade
```

Once you complete the process of updating all components, proceed to Start the Services you Stopped.

Updating Cloudera Components Using Default Repositories

To install the new version, you can upgrade from Cloudera's repository by adding an entry to your operating system's package management configuration file. The repository location varies by operating system.

Operating System	Configuration File Repository Entry
Red Hat	http://archive.cloudera.com/cdh4/redhat/6/x86_64/cdh/4/
SLES	http://archive.cloudera.com/cdh4/sles/11/x86_64/cdh/4/
Debian Squeeze	[arch=amd64] http://archive.cloudera.com/cdh4/debian/squeeze squeeze-cdh4 contrib
Ubuntu Lucid	[arch=amd64] http://archive.cloudera.com/cdh4/ubuntu/lucid/amd64/cdh lucid-cdh4 contrib
Ubuntu Precise	[arch=amd64] http://archive.cloudera.com/cdh4/ubuntu/precise/amd64/cdh precise-cdh4 contrib

For example, under Red Hat, to upgrade from Cloudera's repository you can run commands such as the following on the CDH host to update only CDH:

```
$ sudo yum clean all
$ sudo yum update 'cloudera-*'
```

Note:

– cloudera-cdh4 is the name of the repository on your system; the name is usually in square brackets on the first line of the repo file, in this example `/etc/yum.repos.d/cloudera-cdh4.repo`:

```
[chris@ca727 yum.repos.d]$ more cloudera-cdh4.repo
[cloudera-cdh4]
...
```

– `yum clean all` cleans up yum's cache directories, ensuring that you download and install the latest versions of the packages. – If your system is not up to date, and any underlying system components need to be upgraded before this yum update can succeed, yum will tell you what those are.

On a SLES system, use commands like this to clean cached repository information and then update only the CDH components. For example:

```
$ sudo zypper clean --all
$ sudo zypper up -r http://archive.cloudera.com/cdh4/sles/11/x86_64/cdh/4
```

To verify the URL, open the Cloudera repo file in `/etc/zypp/repos.d` on your system (for example `/etc/zypp/repos.d/cloudera-cdh4.repo`) and look at the line beginning

```
baseurl=
```

Use that URL in your `sudo zypper up -r` command.

On a Debian/Ubuntu system, use commands like this to clean cached repository information and then update only the CDH components. First:

```
$ sudo apt-get clean
```

After cleaning the cache, use one of the following upgrade commands to upgrade CDH.

Precise:

```
$ sudo apt-get upgrade -t precise-cdh4
```

Lucid:

```
$ sudo apt-get upgrade -t lucid-cdh4
```

Squeeze:

```
$ sudo apt-get upgrade -t squeeze-cdh4
```

At the end of this process you should have the most recent versions of the CDH packages installed on the host and you can now proceed to Start the Services you Stopped.

Updating Cloudera Components Using Custom Repositories

You can create your own repository, as described in Appendix A - Understanding Custom Installation Solutions. Creating your own repository is necessary if you are upgrading a cluster that does not have access to the Internet.

If you used a custom repository to complete the installation of current files and now you want to update using a custom repository, the details of the steps to complete the process are variable.

In general, begin by updating any existing custom repository that you will use with the installation files you wish to use. This can be completed in a variety of ways. For example, you might use `wget` to copy the necessary installation files. Once the installation files have been updated, use the custom repository you established for the initial installation to update CDH.

Red Hat

On a Red Hat system ensure you have a custom repo that is configured to use your internal repository. For example, if you could have custom repo file in `/etc/yum.conf.d/` called `cdh_custom.repo` in which you specified a local repository. In such a case, you might use the following commands:

```
$ sudo yum clean all
$ sudo yum update 'cloudera-*'
```

SLES

On a SLES system, use commands such as the following to clean cached repository information and then update only the CDH components:

```
$ sudo zypper clean --all
$ sudo zypper up -r http://internalserver.example.com/path_to_cdh_repo
```

Debian/Ubuntu

Use a command that targets upgrade of your CDH distribution using the custom repository specified in your `apt` configuration files. These files are typically either the `/etc/apt/apt.conf` file or in various files in the `/etc/apt/apt.conf.d/` directory. Information about your custom repository must be included in the repo files. The general form of entries in Debian/Ubuntu is:

```
deb http://server.example.com/directory/ dist-name pool
```

For example, the entry for the default repo is:

```
deb http://us.archive.ubuntu.com/ubuntu/ precise universe
```

On a Debian/Ubuntu system, use commands such as the following to clean cached repository information and then update only the CDH components:

```
$ sudo apt-get clean
$ sudo apt-get upgrade -t your_cdh_repo
```

Step 4. Upgrade your Hive Metastore

1. (Strongly recommended) Make a backup copy of your Hive metastore database.
2. Follow the instructions at Step 4. "Upgrading Your Metastore" in the Hive Installation section in the CDH4 Installation Guide to run the metastore upgrade script.
   - **If you are upgrading to packages**, the upgrade script is at `/usr/lib/hive/scripts/metastore/upgrade/`
   - **If you are upgrading to parcels**, then the upgrade script is located at `/opt/cloudera/parcels/<parcel_name>/lib/hive/scripts/metastore/upgrade/<database>`. `<parcel_name>` should be the name of the parcel to which you have upgraded. `<database>` is the type of database you are running (i.e. mysql, postgres, etc.) For example, if you are installing a CDH 4.2.0 parcel using the default location for the local repository, and using the default database (PostgreSQL) the script will be at: `/opt/cloudera/parcels/CDH-4.2.0-1.cdh4.2.0.p0.10-e16.parcel/lib/hive/scripts/metastore/upgrade/postgres`
   - You must `cd` to the directory the scripts are in.
   - Execute the script in the appropriate DB command shell.

Important:

You must know the password for the database; if you installed Cloudera Manager using the default (embedded PostgreSQL) database, the password was displayed on the Database Setup page during the Cloudera Manager installation wizard. If you do not know the password for your Hive metastore database, you can find it as follows:

- `cat /etc/cloudera-scm-server/db.properties` This shows you Cloudera Manager's internal database credentials.
- Run the following command:

```
psql -p 7432 -U scm scm -c "select s.display_name as hive_service_name,
 s.name as hive_internal_name, c.value as metastore_password from configs
 c, services s where attr='hive_metastore_database_password' and
c.service_id = s.service_id"
```

- Use the password from `com.cloudera.cmf.db.password`. This will output the passwords for the hive service metastore as follows:

```
 hive_service_name | hive_internal_name | metastore_password
-------------------+--------------------+--------------------
 hive1             | hive1              | lF3Cv2zsvI
(1 row)
```

3. If you have multiple instances of Hive, run the upgrade script on each metastore database.

Step 5. (If Upgrading to CDH 4.2) Upgrade the Oozie Sharelib

1. In the Cloudera Manager Admin Console, select **Oozie** from the **Services** tab. The service should already be stopped.
2. From the **Actions** button choose **Install Oozie Sharelib**. The commands to perform this function are run.

Step 6. Start the Services you Stopped

You can now start the services that you stopped in Step 1. Proceed as follows:

1. In the Cloudera Manager Admin Console, click the **Services** tab.
2. Click the top **Actions** button that corresponds to the cluster and choose **Start**. The **Command Details** window shows the progress of starting services. When **All services successfully started** appears, the task is complete and you may close the **Command Details** window.

Repeat this process for all clusters that you previously stopped.

Step 7. Deploy client configurations

1. From the top **Actions** button that corresponds to the cluster and choose **Deploy Client Configuration...**.
2. Click the **Deploy Client Configuration** button in the confirmation pop-up that appears.

Upgrading to the Latest Version of CDH3 in a Cloudera Managed Deployment

Before You Begin

Important:

Before upgrading, be sure to read about the latest Incompatible Changes and Known Issues and Work Arounds in the CDH3 Release Notes.

Note:

If you are upgrading a cluster that is part of a production system, be sure to plan ahead. As with any operational work, be sure to reserve a maintenance window with enough extra time allotted in case of complications. The Hadoop upgrade process is well understood, but it is best to be cautious. For production clusters, Cloudera recommends allocating up to a full day maintenance window to perform the upgrade, depending on the number of hosts, the amount of experience you have with Hadoop and Linux, and the particular hardware you are using.

Upgrading Unmanaged Components

Upgrading unmanaged components is a process that is separate from upgrading managed components. Upgrade the unmanaged components before proceeding to upgrade managed components. For example, if you have unmanaged Flume installed, upgrade that before proceeding to upgrade managed components. Components that you might have installed that are not managed by Cloudera Manager include:

- Flume 0.9.x
- Flume 1.x
- Sqoop
- Pig
- Hive
- Whirr
- Mahout

For information on upgrading these unmanaged components, see CDH3 Installation Guide

Step 1. Stop all the CDH Services on All Hosts

You must stop all Hadoop services before upgrading CDH.

To stop all services

1. In the Cloudera Manager Admin Console, select **Services > All Services**.
2. Click the top **Actions** button that corresponds to the cluster and choose **Stop...**. Click **Stop** in the confirmation screen.

   The **Command Details** window shows the progress of stopping services.

   When **All services successfully stopped** appears, the task is complete and you may close the **Command Details** window.

3. For each Cloudera Management Service entry, click **Actions** and click **Stop...**. Click **Stop** in the confirmation screen.

   The **Command Details** window shows the progress of stopping services.

When **All services successfully stopped** appears, the task is complete and you may close the **Command Details** window.

Repeat this process for all clusters hosting CDH3 machines to be upgraded.

Step 2. Back up the HDFS Metadata on the NameNode

> **Important:**
>
> Do the following when you are sure that all Hadoop services have been shut down. **It is particularly important that the NameNode service is not running so that you can make a consistent backup**.

> **Note:** Cloudera recommends backing up HDFS metadata on a regular basis, as well as before a major upgrade.

1. On the **Services** page of Cloudera Manager, click the HDFS service, then the **Configuration** tab. Navigate to the **NameNode** category and find **NameNode Data Directory**.
2. From the command line on the NameNode machine, back up that directory; for example, if the data directory is `/mnt/hadoop/hdfs/name`, do the following as root:

   ```
   # cd /mnt/hadoop/hdfs/name
   # tar -cvf /root/nn_backup_data.tar .
   ```

   You should see output like this:

   ```
   ./
   ./current/
   ./current/fsimage
   ./current/fstime
   ./current/VERSION
   ./current/edits
   ./image/
   ./image/fsimage
   ```
3. Check the output.

   > **Warning:**
   >
   > If you see a file containing the word *lock*, the NameNode is probably still running. Repeat the preceding steps, starting by shutting down the Hadoop services.

Step 3. Upgrade Managed Components

There are a variety of strategies that you can use to upgrade to the latest version of CDH3.

- You can use your operating system's package management tools to update all packages to the latest version using standard repositories. This approach works well because it minimizes the amount of configuration required and uses the simplest commands. Be aware that this can take a considerable amount of time if you have not upgraded the system recently.
- You can target the `cloudera.com` repository that is added during a typical install, only updating Cloudera components. This limits the scope of updates to be completed, so the process takes less time. This will not work if you created and used a custom repository.
- You can use a custom repository. This process can be more complicated, but enables updating Cloudera components for CDH machines that are not connected to the Internet.

Updating Everything

You can update all components on your system, including Cloudera components. Note that this may take a significant amount of time. To update all packages on your system, use the following command:

- On Red Hat systems:

```
$ sudo yum update
```

- On SLES systems:

```
$ sudo zypper up
```

- On Ubuntu/Debian systems:

```
$ sudo apt-get upgrade
```

Once you complete the process of updating all components, proceed to Start the Services you Stopped.

Updating Cloudera Components Using Default Repositories

To install the new version, you can upgrade from Cloudera's repository by adding an entry to your operating system's package management configuration file. The repository location varies by operating system.

Operating System	Configuration File Repository Entry
Red Hat	http://archive.cloudera.com/redhat/cdh/3/
SLES	http://archive.cloudera.com/sles/11/x86_64/cdh/3/
Debian Squeeze	deb http://archive.cloudera.com/debian/ squeeze-cdh3 contrib
Ubuntu Lucid	deb http://archive.cloudera.com/debian/ lucid-cdh3 contrib
Ubuntu Maverick	deb http://archive.cloudera.com/debian/ maverick-cdh3 contrib

For example, under Red Hat, to upgrade from Cloudera's repository you can run commands such as the following on the CDH host to update only CDH:

```
$ sudo yum clean all
$ sudo yum update 'cloudera-*'
```

Note:

– cloudera-cdh3 is the name of the repository on your system; the name is usually in square brackets on the first line of the repo file, in this example `/etc/yum.repos.d/cloudera-cdh3.repo`:

```
[chris@ca727 yum.repos.d]$ more cloudera-cdh3.repo
[cloudera-cdh3]
...
```

– `yum clean all` cleans up yum's cache directories, ensuring that you download and install the latest versions of the packages. – If your system is not up to date, and any underlying system components need to be upgraded before this yum update can succeed, yum will tell you what those are.

On a SLES system, use commands like this to clean cached repository information and then update only the CDH components. For example:

```
$ sudo zypper clean --all
$ sudo zypper up -r http://archive.cloudera.com/sles/11/x86_64/cdh/
```

The apt configuration files specify repository information. These files are typically either the /etc/apt/apt.conf file or in various files in the /etc/apt/apt.conf.d/ directory. Review the contents of that file to find the Cloudera repository.

On a Debian/Ubuntu system, use commands like this to clean cached repository information and then update only the CDH components. First:

```
$ sudo apt-get clean
```

After cleaning the cache, use one of the following upgrade commands to upgrade CDH.

Maverick:

```
$ sudo apt-get upgrade -t maverick-cdh3
```

Lucid:

```
$ sudo apt-get upgrade -t lucid-cdh3
```

Squeeze:

```
$ sudo apt-get upgrade -t squeeze-cdh3
```

At the end of this process you should have the most recent versions of the CDH packages installed on the host and you can now proceed to Start the Services you Stopped.

Updating Cloudera Components Using Custom Repositories

You can create your own repository, as described in Appendix A - Understanding Custom Installation Solutions. Creating your own repository is necessary if you are upgrading a cluster that does not have access to the Internet.

If you used a custom repository to complete the installation of current files and now you want to update using a custom repository, the details of the steps to complete the process are variable.

In general, begin by updating any existing custom repository that you will use with the installation files you wish to use. This can be completed in a variety of ways. For example, you might use wget to copy the necessary installation files. Once the installation files have been updated, use the custom repository you established for the initial installation to update CDH.

Red Hat

On a Red Hat system ensure you have a custom repo that is configured to use your internal repository. For example, if you could have custom repo file in /etc/yum.conf.d/ called cdh_custom.repo in which you specified a local repository. In such a case, you might use the following commands:

```
$ sudo yum clean all
$ sudo yum update 'cloudera-*'
```

SLES

On a SLES system, use commands such as the following to clean cached repository information and then update only the CDH components:

```
$ sudo zypper clean --all
$ sudo zypper up -r http://internalserver.example.com/path_to_cdh_repo
```

Debian/Ubuntu

Use a command that targets upgrade of your CDH distribution using the custom repository specified in your apt configuration files. These files are typically either the /etc/apt/apt.conf file or in various files in the /etc/apt/apt.conf.d/ directory. Information about your custom repository must be included in the repo files. The general form of entries in Debian/Ubuntu is:

```
deb http://server.example.com/directory/ dist-name pool
```

For example, the entry for the default repo is:

```
deb http://us.archive.ubuntu.com/ubuntu/ precise universe
```

On a Debian/Ubuntu system, use commands such as the following to clean cached repository information and then update only the CDH components:

```
$ sudo apt-get clean
$ sudo apt-get upgrade -t your_cdh_repo
```

Step 4. Start the Services you Stopped

You can now start the services that you stopped in Step 1. Proceed as follows:

1. In the Cloudera Manager Admin Console, click the **Services** tab.
2. Click the top **Actions** button that corresponds to the cluster and choose **Start**.

   The **Command Details** window shows the progress of starting services.

   When **All services successfully started** appears, the task is complete and you may close the **Command Details** window.

Repeat this process for all clusters that you previously stopped.

Specifying the Racks for Hosts

Cloudera Manager includes internal rack awareness scripts, but you must specify the racks where the hosts in your cluster are located. If your cluster contains more than 10 hosts, Cloudera recommends that you specify the rack for each host. HDFS and MapReduce will automatically use the racks you specify.

> **Note:**
>
> Cloudera Manager supports nested rack specifications. For example, you could specify the rack `/rack1`, but you could also specify `/group5/rack3` to indicate the third rack in the fifth group.
>
> Currently, due to a Hadoop limitation, all hosts in a cluster should have the same number of path components in their rack specifications.

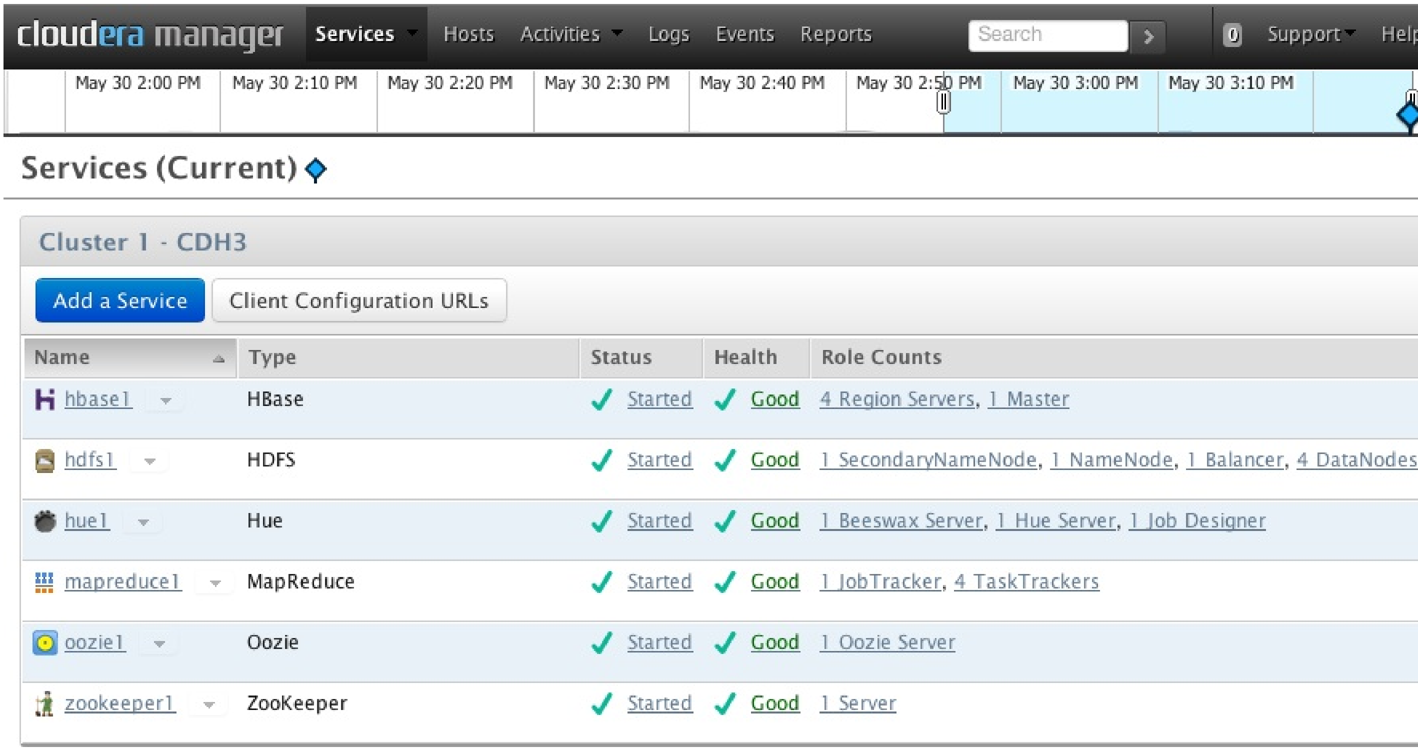
To specify the racks for hosts:

1. Click the **Hosts** tab.
2. Select the host(s) for a particular rack, such as all hosts for `/rack123`. Use shift-click to select multiple hosts at a time.
3. Click **Assign Rack**.
4. Enter a rack name or ID that starts with a slash `/`, such as `/rack123` or `/aisle1/rack123`, and then click **Set Rack**.

After assigning racks, consider restarting affected services as described in Starting, Stopping, and Restarting Services. Rack assignments are not automatically updated for running services.

Testing the Installation

You can check the Cloudera Manager installation on your cluster by logging in to the Cloudera Manager Admin Console and reviewing the contents of the Services tab. The left side of the screen should look something like this:



Services (Current)

Cluster 1 - CDH3

Name	Type	Status	Health	Role Counts
hbase1	HBase	✓ Started	✓ Good	4 Region Servers, 1 Master
hdfs1	HDFS	✓ Started	✓ Good	1 SecondaryNameNode, 1 NameNode, 1 Balancer, 4 DataNodes
hue1	Hue	✓ Started	✓ Good	1 Beeswax Server, 1 Hue Server, 1 Job Designer
mapreduce1	MapReduce	✓ Started	✓ Good	1 JobTracker, 4 TaskTrackers
oozie1	Oozie	✓ Started	✓ Good	1 Oozie Server
zookeeper1	ZooKeeper	✓ Started	✓ Good	1 Server

You can also click on each service to see more detailed information. One way to test the services running in your cluster is to interact with them using Hue. Included among the services you can interact with using Hue are Impala, HDFS, Hive, HBase, Oozie, MapReduce, and YARN. For complete information, see Hue 2 User Guide.

Enabling the Oozie Web Console

You can enable Oozie's web console as a means of working with Cloudera Manager. **To enable the Oozie Web Console**

1. Download ext-2.2. Extract the contents of the file to `/var/lib/oozie/libext`.
2. Connect to the Cloudera Manager Admin Console. Click the **Configuration** tab for the Oozie service you are configuring and enable **Enable Oozie server web console**.
3. Ensure sure the Oozie user has read/write permissions on the directory `/var/lib/oozie/oozie-server/webapps`. For example, you might review the permissions as follows:

```
$ ls -l /var/lib/oozie/oozie-server/
total 12
drwxr-xr-x 3 oozie oozie 4096 Jul 13 05:42 conf
dr--r--r-- 4 oozie oozie 4096 Jul 13 13:15 webapps
drwxr-xr-x 3 oozie oozie 4096 Jul 13 05:42 work
```

4. If the permissions are set appropriately, continue to the next step. If permissions must be modified, you could do so as follows:

```
$ chmod 755 /var/lib/oozie/oozie-server/webapps
$ ls -l /var/lib/oozie/oozie-server/
drwxr-xr-x 3 oozie oozie 4096 Jul 13 05:42 conf
drwxr-xr-x 4 oozie oozie 4096 Jul 13 13:15 webapps
drwxr-xr-x 3 oozie oozie 4096 Jul 13 05:42 work
```

The Oozie web console is enabled the next time the Oozie service is started or restarted.

Using an External Database for Oozie

By default, Cloudera Manager uses Derby for Oozie's database. If necessary, you can configure Oozie to use an external database.

Oozie supports the following databases:

- MySQL 5.0 and 5.5
- PostgreSQL 8.4 and 9.0
- Oracle 11g

To configure Cloudera Manager to use an external database as the database for Oozie:

1. In Cloudera Manager, navigate to the **Services** page and click the Oozie service instance.
2. Click **Configuration** and click **Edit**.
3. Expand **Oozie Server (Base)** and click **Database**.
4. Specify the settings for **Oozie Server database type**, **Oozie Server database name**, **Oozie Server database host**, **Oozie Server database user**, and **Oozie Server database password**.

> **Note:**
>
> If you are using MySQL or Oracle, ensure that `mysql-connector-java.jar` or `ojdbc6-11.2.0.3.0.jar` respectively are placed in `/usr/lib/oozie/libext`.

Using an External Database for Hue

By default, Cloudera Manager uses SQLite for Hue's database. If necessary, you can configure Cloudera Manager to use an external database such as MySQL or PostgreSQL as the database for Hue. The procedure described in this topic illustrates how to migrate the Hue database from the default SQLite installation to another database.

To configure Cloudera Manager to use an external database for Hue

1. Using the Cloudera Manager Admin Console, click the service instance for the Hue database you are reconfiguring. The Hue service instance page in Cloudera Manager Admin Console appears.
2. Click **Actions** and click **Stop**. Confirm you want to stop the service by clicking **Stop**. If the Hue service is already stopped, skip this step.
3. Click **Actions** for the Hue service, and click **Dump Database**. Confirm you want to dump the database by clicking **Dump Database**.
4. Open the database dump file (by default `/tmp/hue_database_dump.json`) and remove all JSON objects with 'useradmin.userprofile' in the 'model' field. (You can verify the location of the Database Dump File by searching for Database Dump File in the Hue Configuration settings.)

Continue with instructions for configuring Hue to use MySQL (MyISAM or InnoDB) or PostgreSQL. When you complete the instructions, start the Hue server.

Configuring the Hue Server to Store Data in MySQL

To configure the Hue Server to store data in MySQL:

1. Create a new database and grant privileges to a Hue user to manage this database. For example:

```
mysql> create database hue;
Query OK, 1 row affected (0.01 sec)
mysql> grant all on hue.* to 'hue'@'localhost' identified by 'secretpassword';
Query OK, 0 rows affected (0.00 sec)
```

2. Using the Cloudera Manager Admin Console, click the service instance for the Hue database you are reconfiguring. The Hue service instance page in Cloudera Manager Admin Console appears.
3. Click **Configuration > View and Edit**. In the **Category** pane, click the instance of **Database** under **Service-Wide**.
4. Specify the settings for **Hue's Database Type**, **Hue's Database Hostname**, **Hue's Database Port**, **Hue's Database Username**, **Hue's Database Password**, and **Hue's Database Name**. For example, for a MySQL database on the local host, you might use the following values:

```
Hue's Database Type = mysql
Hue's Database Hostname = localhost
Hue's Database Port = 3306
Hue's Database Username = hue
Hue's Database Password = secretpassword
Hue's Database Name = hue
```

5. The following steps are for restoring the Hue data to the new database. If you would like Hue to start from a fresh state, you can start your Hue service now.
6. Click **Actions** and click **Synchronize Database**.
7. Determine the foreign key ID.

```
$ mysql -uhue -psecretpassword
mysql > SHOW CREATE TABLE auth_permission;
```

8. (**InnoDB only**) Drop the foreign key that you retrieved in the previous step.

```
mysql > ALTER TABLE auth_permission DROP FOREIGN KEY content_type_id_refs_id_XXXXXX;
```

9. Delete the rows in the django_content_type table.

```
mysql > DELETE FROM hue.django_content_type;
```

10. In Hue service instance page, click **Actions**, and click **Load Database**. Confirm you want to load the database by clicking **Load Database**.
11. (**InnoDB only**) Add back the foreign key.

```
mysql > ALTER TABLE auth_permission ADD FOREIGN KEY (`content_type_id`) REFERENCES
 `django_content_type` (`id`);
```

Configuring the Hue Server to Store Data in PostgreSQL

To configure the Hue Server to store data in PostgreSQL:

1. Install required packages.

   To install on RHEL systems:

   ```
   $ sudo yum install postgresql-devel gcc python-devel
   ```

   To install on SLES systems:

   ```
   $ sudo zypper install postgresql-devel gcc python-devel
   ```

   To install on Ubuntu or Debian systems:

   ```
   $ sudo apt-get install postgresql-devel gcc python-devel
   ```

2. Install the module that provides the connector to PostgreSQL.

   ```
   sudo -u hue /usr/share/hue/build/env/bin/pip install setuptools
   sudo -u hue /usr/share/hue/build/env/bin/pip install psycopg2
   ```

3. Install the PostgreSQL server. **To install PostgreSQL on a RHEL system:**

   ```
   $ sudo yum install postgresql-server
   ```

   To install PostgreSQL on SLES systems:

   ```
   $ sudo zypper install postgresql-server
   ```

   To install PostgreSQL on Ubuntu or Debian systems:

   ```
   $ sudo apt-get install postgresql
   ```

4. Initialize the data directories.

   ```
   $ service postgresql initdb
   ```

5. Configure client authentication.

   a. Edit `/var/lib/pgsql/data/pg_hba.conf`.

b. Set the authentication methods for local to `trust` and for host to `password` and add the following line at the end.

```
host hue hue 0.0.0.0/0 md5
```

6. Start the PostgreSQL server.

```
$ su - postgres
# /usr/bin/postgres -D /var/lib/pgsql/data > logfile 2>&1 &
```

7. Configure PostgreSQL to listen on all network interfaces.

a. Edit `/var/lib/pgsql/data/postgresql.conf` and set list_addresses.

```
listen_addresses = ‘0.0.0.0’     # Listen on all addresses
```

8. Create the hue database and grant privileges to a hue user to manage the database.

```
# psql -U postgres
postgres=# create database hue;
postgres=# \c hue;
You are now connected to database 'hue'.
postgres=# create user hue with password 'secretpassword';
postgres=# grant all privileges on database hue to hue;
postgres=# \q
```

9. Restart the PostgreSQL server.

```
$ sudo service postgresql restart
```

10. Verify connectivity.

```
psql –h localhost –U hue –d hue
Password for user hue: secretpassword
```

11. Configure the PostgreSQL server to start at boot.'

On RHEL systems:

```
$ sudo /sbin/chkconfig postgresql on
$ sudo /sbin/chkconfig --list postgresql
postgresql          0:off   1:off   2:on    3:on    4:on    5:on    6:off
```

On SLES systems:

```
$ sudo chkconfig --add postgresql
```

On Ubuntu or Debian systems:

```
$ sudo chkconfig postgresql on
```

12. Using the Cloudera Manager Admin Console, click the service instance for the Hue database you are reconfiguring. The Hue service instance page in Cloudera Manager Admin Console appears.
13. Click **Configuration > View and Edit**. In the **Category** pane, click **Advanced** under **Service-Wide**.

14. Specify the settings for **Hue Service Configuration Safety Valve**:

    ```
    [desktop]
    [[database]]
    host=localhost
    port=5432
    engine=postgresql_psycopg2
    user=hue
    password=secretpassword
    name=hue
    ```

15. Click **Save Changes**.
16. The following steps are for restoring the Hue data to the new database. If you would like Hue to start from a fresh state, you can start your Hue service now.
17. Click **Actions** and click **Synchronize Database**.
18. Determine the foreign key ID.

    ```
    bash# su - postgres
    $ psql -h localhost -U hue -d hue
    postgres=# \d auth_permission;
    ```

19. Drop the foreign key that you retrieved in the previous step.

    ```
    postgres=# ALTER TABLE auth_permission DROP CONSTRAINT
    content_type_id_refs_id_XXXXXX;
    ```

20. Delete the rows in the django_content_type table.

    ```
    postgres=# TRUNCATE django_content_type CASCADE;
    ```

21. In Hue service instance page, click **Actions**, and click **Load Database**. Confirm you want to load the database by clicking **Load Database**.
22. Add back the foreign key you dropped.

    ```
    bash# su - postgres
    $ psql -h localhost -U hue -d hue
    postgres=# ALTER TABLE auth_permission ADD CONSTRAINT content_type_id_refs_id_XXXXXX
     FOREIGN KEY (content_type_id) REFERENCES django_content_type(id) DEFERRABLE
    INITIALLY DEFERRED;
    ```

Using Custom Java Home Locations

Java, which Cloudera services require, may be installed at a custom location. In a such a case, Cloudera services may be unable to find this resource. If the JDK cannot be found, services such as MapReduce or HDFS may not start. If you installed the JDK to a custom location, you may need to modify the system configuration to ensure the JDK can be found.

For more information on installing the JDK, see "Java Development Kit Installation for CDH3" in the CDH3 Installation Guide or "Java Development Kit Installation for CDH4" in the CDH4 Installation Guide. If Java is installed at a custom location, update system settings so the custom location is used, and then restart Cloudera Manager Agent on the host where the failing service is assigned. Note that you must know the custom java location that was established during the JDK installation process. Cloudera provides two ways to ensure Cloudera services can find your JDK installation.

Modifying CMF_AGENT_JAVA_HOME

In many cases, modifying the `CMF_AGENT_JAVA_HOME` environment variable is an effective solution for updating the configuration to accommodate a custom `JAVA_HOME`. Modifying the `CMF_AGENT_JAVA_HOME` environment variable enables all services on the host to find the JDK. **To modify the CMF_AGENT_JAVA_HOME environment variable**

1. Open `/etc/default/cloudera-scm-agent`.
2. Set the `CMF_AGENT_JAVA_HOME` environment variable to the java home in your environment. For example, you might modify the file to include the following line:

```
export CMF_AGENT_JAVA_HOME=/usr/custom_java
```

3. Save and close the `cloudera-scm-agent` file.
4. Restart the Cloudera Manager Agent using the following command:

```
sudo service cloudera-scm-agent restart
```

Modifying Service Settings

You can modify service settings to use your custom `JAVA_HOME`. This is done as an alternative to modifying the `CMF_AGENT_JAVA_HOME` environment variable. Modifying service settings to use a custom `JAVA_HOME` applies to all nodes in the cluster, but you must repeat this process of updating `JAVA_HOME` for all services. **To modify service settings to use your custom JAVA_HOME**

1. Open the Cloudera Manager Admin Console, click any service that fails to start because the JDK location is misconfigured, and click **Configuration**.
2. Under **Service-Wide**, click **Advanced**.
3. Click the **Value** cell for the **Service Environment Safety Valve** and add your custom java home to the property here. For example, you might enter the value `JAVA_HOME=/opt/java/jdk6`.
4. Click **Save Changes**.
5. If your deployment includes Cloudera Management services, add your custom `JAVA_HOME` value to `/usr/share/cmf/bin/cmf-server`.
6. In the Cloudera Manager Admin Console, for the service you are configuring, click **Actions** and click **Restart**.
7. Repeat this process for all services that need the updated JAVA_HOME value.

To see how Cloudera Manager chooses a default JDK, review the contents of `/usr/lib64/cmf/service/common/cloudera-config.sh`.

Using the LZO Parcel

This section describes how to install and use the LZO parcel.

The Repository

Add the appropriate repository to Cloudera Manager's list of parcel repositories. The HADOOP_LZO parcel will then become available on the parcel management screen. If required, the repository can be mirrored in the same way as the CDH repo. Public customer repository: http://archive.cloudera.com/gplextras/parcels/latest.

Activation

The HADOOP_LZO parcel can be downloaded/distributed/activated in the same way as the CDH parcel. Once activated, it will be necessary to reconfigure and restart services that intend to use lzo functionality.

MapReduce

The HADOOP_LZO parcel can be downloaded/distributed/activated in the same way as the CDH parcel. Once activated, it will be necessary to reconfigure and restart services that intend to use lzo functionality.

1. Add the following entries to the MapReduce environment safety valve:
   - `HADOOP_CLASSPATH=/opt/cloudera/parcels/HADOOP_LZO/lib/hadoop/lib/*`
   - `JAVA_LIBRARY_PATH=/opt/cloudera/parcels/HADOOP_LZO/lib/hadoop/lib/native`
2. Add the following entries to the MapReduce Client environment safety valve:
   - `HADOOP_CLASSPATH=$HADOOP_CLASSPATH:/opt/cloudera/parcels/HADOOP_LZO/lib/hadoop/lib/*`
   - `JAVA_LIBRARY_PATH=$JAVA_LIBRARY_PATH:/opt/cloudera/parcels/HADOOP_LZO/lib/hadoop/lib/native`
3. Restart MapReduce
4. Redeploy MapReduce Client Configuration

Oozie

1. Go to `/var/lib/oozie` on each Oozie server and symlink the hadoop lzo jar.
   - `/opt/cloudera/parcels/HADOOP_LZO/lib/hadoop/lib/hadooplzocdh40.4.15gplextras. jar`
2. Restart Oozie

HBase

1. Add the following entries to the HBase environment safety valve:
   - `HBASE_CLASSPATH=/opt/cloudera/parcels/HADOOP_LZO/lib/hadoop/lib/*`
   - `JAVA_LIBRARY_PATH=/opt/cloudera/parcels/HADOOP_LZO/lib/hadoop/lib/native`
2. Restart HBase

Impala (1.0 or later)

This only works with Impala 1.0 or later.

1. Add the following entry to the Impala environment safety valve:
   - `LD_LIBRARY_PATH=/opt/cloudera/parcels/HADOOP_LZO/lib/impala/lib`
2. Restart Impala

Notes

- Any service that does not require the use of LZO need not be configured. For example, if you are not using HBase, you do not need to do anything to the safety valve.
- The Oozie step is required, with or without parcels. The only difference is where you find the LZO jar to copy/replace. The LZO jar may already be present in /var/lib/oozie. Replacing any existing jar with the parcel jar (as described abvoe) is strongly recommended.

Deploying Clients

Client configuration files are generated automatically by Cloudera Manager based on the services you install.

Cloudera Manager deploys these configurations automatically at the end of the installation workflow. You can also download the client configuration files to deploy them manually.

If you modify the configuration of your cluster, you may need to redeploy the client configuration files. For example, a service whose status is "Running with outdated configuration" indicates you may need to redeploy those files.

See Deploying Client Configuration Files in the *Cloudera Manager User Guide* for information on downloading client configuration files, or redeploying them through Cloudera Manager.

Uninstalling Cloudera Manager

If necessary, you can use the following instructions to uninstall the Cloudera Manager Server and Agents.

> **Note:**
>
> **Recovering from a failed install**
>
> If you have come to this page because your installation did not complete (for example, if it was interrupted by a virtual machine timeout), and you want to proceed with the installation, do the following:
>
> 1. Remove files and directories:
>
> ```
> $ sudo rm -Rf /usr/share/cmf /var/lib/cloudera* /var/cache/yum/cloudera*
> ```
>
> 2. Run the installer again.

Uninstalling Cloudera Manager Server and Agents

> **Note:**
>
> After uninstalling Cloudera Manager, you may want to keep or remove the Hadoop data on your cluster. The following instructions do not remove the Hadoop data. To find out where the Hadoop data directories are located, you can navigate to the **Configuration** tab for the HDFS and MapReduce services in the Cloudera Manager Admin Console, and search for the **Data Directory** property setting. For additional information about uninstalling CDH, including clean-up of CDH files, see the entry on Uninstalling CDH Components in the Maintenance Tasks and Notes topic in the CDH4 Installation Guide.

Step 1: Stop all services

You must stop all Hadoop services before uninstalling the Cloudera Manager Server and Agents.

> **Note:**
>
> As necessary, repeat the process of stopping services for each cluster.

To stop all services

1. In the Cloudera Manager Admin Console, select **Services > All Services**.
2. Click the top **Actions** button that corresponds to the cluster and choose **Stop...**. Click **Stop** in the confirmation screen.

   The **Command Details** window shows the progress of stopping services.

   When **All services successfully stopped** appears, the task is complete and you may close the **Command Details** window.

3. For each Cloudera Management Service entry, click **Actions** and click **Stop...**. Click **Stop** in the confirmation screen.

   The **Command Details** window shows the progress of stopping services.

When **All services successfully stopped** appears, the task is complete and you may close the **Command Details** window.

Step 2: Deactivate and remove Parcels, if used for CDH installation.

If you installed CDH using parcels, you should remove them from the Parcels page in the Cloudera Manager Admin Console.

> *If you installed using packages, you can skip this step. Go to Step 3.*

1. Click the parcel indicator in the top navigation bar. This takes you to the **Administration** page, **Parcels** tab. By default, any parcels available for download or distribution are shown, as well as the activated parcel.
2. Click the arrow on the **Actions** button and select **Deactivate**.

   When this action has completed, the parcel button will change to say **Activate**.

3. Click the arrow on the **Activate** button and select **Remove from Hosts**.

   When this action has completed, the parcel button will change to say **Distribute**.

4. Click the arrow on the **Distribute** button and select **Delete**. This will remove the parcel from the local parcel repository.

Note that there can only be one active parcel; however, there may be multiple parcels that have been downloaded and distributed, but that are not active. If this is the case, you should also remove those parcels from any hosts onto which they have been distributed, and delete the parcels from the local repository.

Step 3: Uninstall the Cloudera Manager Server.

The commands for uninstalling the Cloudera Manager Server depend on the method you used to install the Cloudera Manager Server. Refer to section below that corresponds to the method you used to install the Cloudera Manager Server. This process described below also removes the embedded PostgreSQL database, if you installed that option. If you did not use the PostgreSQL database, omit the cloudera-manager-server-db steps.

If you used the cloudera-manager-installer.bin file:

If you installed the Cloudera Manager Server using `cloudera-manager-installer.bin`, run the following command on the Cloudera Manager Server host:

```
$ sudo /usr/share/cmf/uninstall-cloudera-manager.sh
```

> **Note:**
>
> If the `uninstall-cloudera-manager.sh` is not installed on your cluster, use the following instructions to uninstall the Cloudera Manager Server.

If you did not use the cloudera-manager-installer.bin file:

If you installed the Cloudera Manager Server using a different installation method such as Puppet, run the following commands on the Cloudera Manager Server host.

1. Stop the Cloudera Manager Server and its database:

   ```
   sudo service cloudera-scm-server stop
   sudo service cloudera-scm-server-db stop
   ```

2. Uninstall the Cloudera Manager Server and its database:

On Red Hat systems:

```
sudo yum remove cloudera-manager-server
sudo yum remove cloudera-manager-server-db
```

On SLES systems:

```
sudo zypper -n rm --force-resolution cloudera-manager-server
sudo zypper -n rm --force-resolution cloudera-manager-server-db
```

On Debian/Ubuntu systems:

```
sudo apt-get remove cloudera-manager-server
sudo apt-get remove cloudera-manager-server-db
```

Step 4: On all Agent hosts, uninstall CDH and the Cloudera Manager Agents.

If you installed using packages:

On all Agent hosts, run the following commands as root to uninstall the Cloudera Manager Agent and CDH on your cluster machines.

1. Stop the Cloudera Manager agent.

   On Red Hat/SLES systems:

   ```
   $ sudo service cloudera-scm-agent hard_stop
   ```

   On Debian/Ubuntu systems:

   ```
   $ sudo /usr/sbin/service cloudera-scm-agent hard_stop
   ```

2. Uninstall the CDH and Cloudera Manager packages:

   On Red Hat systems: If you are running CDH3:

   ```
   $ sudo yum remove 'cloudera-manager-*' hadoop-0.20 hue-common hadoop-pig
   oozie-client hive hadoop-hbase hadoop-zookeeper bigtop-utils
   ```

   If you are running CDH4 (including Impala):

   ```
   $ sudo yum remove 'cloudera-manager-*' hadoop hue-common 'bigtop-*'
   ```

   On Red SLES systems: If you are running CDH3:

   ```
   $ sudo zypper remove  'cloudera-manager-*' hadoop-0.20 hue-common hadoop-pig
   oozie-client hive hadoop-hbase hadoop-zookeeper bigtop-utils
   ```

   If you are running CDH4:

   ```
   $ sudo zypper remove 'cloudera-manager-*' hadoop hue-common 'bigtop-*'
   ```

   On Debian/Ubuntu systems: If you are running CDH3:

   ```
   $ sudo apt-get purge 'cloudera-manager-*' hadoop-0.20 hue-common hadoop-pig
   oozie-client hive hadoop-hbase hadoop-zookeeper bigtop-utils
   ```

   If you are running CDH4:

   ```
   $ sudo apt-get purge 'cloudera-manager-*' hadoop hue-common 'bigtop-*'
   ```

3. Run this command:

   On Red Hat systems:

   ```
   $ sudo yum clean all
   ```

   On SLES systems:

   ```
   $ sudo zypper clean
   ```

   On Debian/Ubuntu systems:

   ```
   $ sudo apt-get clean
   ```

If you installed using parcels:

On all Agent hosts, run the following commands as root to uninstall the Cloudera Manager Agent on your cluster machines. You should already have removed CDH by deactivating and deleting the parcel.

1. Stop the Cloudera Manager agent.

   On Red Hat/SLES systems:

   ```
   $ sudo service cloudera-scm-agent hard_stop
   ```

   On Debian/Ubuntu systems:

   ```
   $ sudo /usr/sbin/service cloudera-scm-agent hard_stop
   ```

2. Uninstall the Cloudera Manager packages:

   On Red Hat systems:

   ```
   $ sudo yum remove 'cloudera-manager-*'
   ```

   On SLES systems:

   ```
   $ sudo zypper remove 'cloudera-manager-*'
   ```

   On Debian/Ubuntu systems:

   ```
   $ sudo apt-get purge 'cloudera-manager-*'
   ```

3. Run this command:

   On Red Hat systems:

   ```
   $ sudo yum clean all
   ```

   On SLES systems:

   ```
   $ sudo zypper clean
   ```

   On Debian/Ubuntu systems:

   ```
   $ sudo apt-get clean
   ```

Step 5: On all Agent hosts, remove all Cloudera Manager data.

This step permanently removes Cloudera Manager data. If you want to be able to access any of this data in the future, you must back it up before removing it. If you used an embedded PostgreSQL database, that data is stored in /var/lib/cloudera-scm-server-db. To permanently remove all Cloudera Manager data, run the following command as root:

```
$ sudo rm -Rf /usr/share/cmf /var/lib/cloudera* /var/cache/yum/cloudera*
```

Step 6: On all Agent hosts, kill any running Cloudera Manager and Hadoop processes.

On all Agent hosts, kill any running Cloudera Manager and Hadoop processes:

```
$ for u in hdfs mapred cloudera-scm hbase hue zookeeper oozie hive impala flume; do
sudo kill $(ps -u $u -o pid=); done
```

Note:

This step should not be necessary if you stopped all the services and the Cloudera Manager agent correctly.

Step 7: Remove the Cloudera Manager lock file.

On all Agent hosts, run this command to remove the Cloudera Manager lock file:

```
$ sudo rm /tmp/.scm_prepare_node.lock
```

Troubleshooting Installation and Upgrade Problems

Use the information in this section to troubleshoot installation problems. For information on known issues, see Known Issues and Work Arounds in Cloudera Manager Enterprise Edition 4.

Symptom	Problem	What to Do
"Failed to start server" reported by cloudera-manager-installer.bin. `/var/log/cloudera-scm-server/cloudera-scm-server.log` contains a message beginning `Caused by: java.lang.ClassNotFoundException: com.mysql.jdbc.Driver...`	You may have SELinux enabled.	You can disable SELinux by running `sudo setenforce 0` on the Cloudera Manager Server host. To disable it permanently, edit `/etc/selinux/config`.
Installation interrupted and installer won't restart.	You need to do some manual cleanup.	See Uninstalling Cloudera Manager.
Cloudera Manager Server fails to start. Server is configured to use a MySQL database to store information about service configuration.	Tables may be configured with the ISAM engine.	Make sure that the InnoDB engine is configured, not the MyISAM engine. To check what engine your tables are using, run the following command from the MySQL shell: `mysql> show table status;`
Agents fail to connect to server. Error 113 ('No route to host') in `cloudera-scm-agent.log`.	You may have SELinux or `iptables` enabled.	Check `/var/log/cloudera-scm-server/cloudera-scm-server.log` on the Server system and `/var/log/cloudera-scm-agent/cloudera-scm-agent.log` on the Agent system(s). Disable SELinux and `iptables`.
Some cluster hosts do not appear when you click **Find Hosts** in install or update wizard.	You may have network connectivity problems.	▪ Make sure all cluster hosts have SSH port 22 open. ▪ Check other common causes of loss of connectivity such as firewalls and interference from SELinux.
"Access denied" in install or update wizard during database configuration for Activity Monitor, Report Manager, or Service Monitor.	Hostname mapping or permissions are incorrectly set up.	▪ For hostname configuration, see information on Configuring Network Names in the CDH3 Deployment on a Cluster topic in CDH3 Installation Guide. ▪ For permissions, make sure the values you enter into the wizard match those you used when you configured the databases. For more information, see Checking Database Hostnames.
Activity Monitor, Report Manager, or Service Monitor databases fail to start.	MySQL `binlog` format problem.	Set `binlog_format=mixed` in `/etc/my.cnf`. For more information, see this MySQL bug report. See also

Symptom	Problem	What to Do
		Installing and Configuring Databases.
You have upgraded the Cloudera Manager Server to 4.5, but now cannot start services.	You may have mismatched versions of the Cloudera Manager Server and Agents.	Make sure you have upgraded the Cloudera Manager Agents on all host machines to 4.5. (The previous version of the Agents will heartbeat with the new version of the Server, but you can't start HDFS and MapReduce with this combination.)
Cloudera services fail to start.	Java may not be installed or may be installed at a custom location.	See Using Custom Java Home Locations for more information on resolving this issue.
The Service Monitor, Activity Monitor, or Host Monitor display a status of **BAD** in the Cloudera Manager Admin Console. The log file contains the following message: `ERROR 1436 (HY000): Thread stack overrun: 7808 bytes used of a 131072 byte stack, and 128000 bytes needed. Use 'mysqld -O thread_stack=#' to specify a bigger stack.`	The MySQL thread stack is too small.	Update the `thread_stack` value in `my.cnf` to `256KB`. The `my.cnf` file is normally located in `/etc` or `/etc/mysql`. Restart the mysql service: `$ sudo service mysql restart` Restart the failed service using the Cloudera Manager Admin Console.
The Service Monitor or Activity Monitor agents fail to start. Logs contain the error `read-committed isolation not safe for the statement binlog format.`	The `binlog_format` is not set to mixed.	Modify the mysql.cnf file to include the entry for `binlog format` as specified in Installing and Configuring a MySQL Database.
Attempts to reinstall older versions of CDH or Cloudera Manager using Yum fails.	It is possible to install, uninstall, and reinstall CDH and Cloudera Manager. In certain cases, this does not complete as expected. If you install Cloudera Manager 4 and CDH 4, then uninstall Cloudera Manager and CDH, and then attempt to install CDH 3.7 and Cloudera Manager 3.7, incorrect cached information may result in the installation of an incompatible version of the Oracle JDK.	To resolve this issue, you must clear information in the yum cache. Clear cache information as follows: Connect to the CDH host. Execute either of the following commands: $ yum `--enablerepo='*'clean all` or $ `rm -rf /var/cache/yum/cloudera*` After clearing cache information, proceed with installing CDH 3.7 and Cloudera Manager 3.7.

Checking Database Hostnames

The value you enter into the wizard as the database hostname **must** match the value you entered for the hostname (if any) when you configured the database.

For example, if you entered the following for the Activity Monitor database

```
grant all on activity_monitor.* TO 'amon_user'@'localhost' IDENTIFIED BY 'amon_password';
```

the value you enter here for the database hostname must be `localhost`. On the other hand, if you had entered the following when you created the database

```
grant all on activity_monitor.* TO 'amon_user'@'myhost1.myco.com' IDENTIFIED BY
'amon_password';
```

the value you enter here for the database hostname must be myhost1.myco.com. If you did not specify a host, or used a wildcard to allow access from any host, you can enter either the fully-qualified domain name (FQDN) here, or `localhost`. For example, if you entered

```
grant all on activity_monitor.* TO 'amon_user'@'%' IDENTIFIED BY 'amon_password';
```

the value you enter here for the database hostname can be either the FQDN or `localhost`. Similarly, if you entered

```
grant all on activity_monitor.* TO 'amon_user' IDENTIFIED BY 'amon_password';
```

the value you enter here for the database hostname can be either the FQDN or `localhost`.

Recovering from Cloudera Manager Host Failures

Cloudera Manager uses databases to store information about the Cloudera Manager system and jobs. If the machine hosting Cloudera Manager fails, it is possible to re-establish the installation if the database information is still available. Database information is typically available for either of the following reasons:

- You backed up the database.
- The database and Cloudera Manager are on separate servers and the database server is still available.

Before beginning this process, find the failed machine's name IP address and hostname. It is not absolutely necessary to have the old Cloudera Manager server name and IP address, but it simplifies the process. You could use a new IP address and hostnames, but this would require updating the configuration of every agent to use this new information. Because it is easier to use the old server name and address in most cases, using a new hostname and IP address is not described.

To restore a Cloudera Manager when the database server is available

1. Identify a new server on which to install Cloudera Manager. Assign the failed Cloudera Manager server's IP address and hostname to the new server.

   Note:

   If the agents were configured with the server's hostname, you do not need to assign the old machine's IP address to the new host. Simply assigning the hostname will suffice.

2. Install Cloudera Manager on a new server, using the method described under Step 3: Install the Cloudera Manager Server. Do not install the other components, such as CDH and databases, as those should still exist in your environment
3. Update `/etc/cloudera-scm-server/db.properties` with the necessary information so Cloudera Manager server connects to the restored database. This information is typically the database name, database instance name, user name, and password.
4. Start the Cloudera Manager server.

To restore a Cloudera Manager deployment from database backups when the database server is not available

1. Identify a new server on which to install Cloudera Manager. Assign the failed Cloudera Manager server's IP address and hostname to the new server.

   Note:

   If the agents were configured with the server's hostname, you do not need to assign the old machine's IP address to the new host. Simply assigning the hostname will suffice.

2. Install Cloudera Manager on a new server, using whatever method you used before, as described in Step 3: Install the Cloudera Manager Server.
3. Install the database packages on the machines that will host the restored database. This could be the same server on which you have just installed Cloudera Manager or it could be a different server. The details of which package to install varies based on which database was initially installed on your system. If you used an external MySQL, PostgreSQL, or Oracle database, reinstall that now. If you used the embedded PostgreSQL database, you will need to install the `cloudera-manager-server-db` package as described in Installing an Embedded PostgreSQL Database. After installing that package, you must initialize and start the database as described in Configuring Your Systems to Support PostgreSQL.
4. Restore the backed up databases to the new database installations.
5. Update `/etc/cloudera-scm-server/db.properties` with the necessary information so Cloudera Manager server connects to the restored database. This information is typically the database name, database instance name, user name, and password.
6. Start the Cloudera Manager server.

At this point, Cloudera Manager should resume functioning as it did before the failure. Because you restored the database from the backup, the server should accept the running state of the agents, meaning it will not terminate any running Hadoop processes.

This process is similar with secure clusters, though additional files in `/etc/cloudera-scm-server` must be restored in addition to the database.

Changing Embedded PostgreSQL Database Passwords

When Cloudera Manager installs and configures embedded PostgreSQL databases, it creates user accounts and passwords. You may wish to change passwords associated with the embedded PostgreSQL database accounts. To change these passwords, you must know what the original password was, but since the accounts were automatically created, this information is often unknown.

To achieve the goal of changing the password, you can retrieve the user name or password, as well as other database information.

- The Cloudera Manager service connects to the database using the `scm` account. Information about this account is stored in the `db.properties` file.
- The root account for the database is the `cloudera-scm` account. Information about this account is stored in the `generated_password.txt` file.

To find information about the PostgreSQL database user account that the SCM service uses, read the `/etc/cloudera-scm-server/db.properties` file:

```
# cat /etc/cloudera-scm-server/db.properties

Auto-generated by scm_prepare_database.sh
#
Sat Oct 1 12:19:15 PDT 201
#
com.cloudera.cmf.db.type=postgresql
com.cloudera.cmf.db.host=localhost:7432
com.cloudera.cmf.db.name=scm
com.cloudera.cmf.db.user=scm
com.cloudera.cmf.db.password=TXqEESuhj5
```

To find information about the root account for the database, read the `/var/lib/cloudera-scm-server-db/data/generated_password.txt` file:

```
# cat /var/lib/cloudera-scm-server-db/data/generated_password.txt

MnPwGeWaip

The password above was generated by /usr/share/cmf/bin/initialize_embedded_db.sh (part
 of the cloudera-scm-server-db package)
and is the password for the user 'cloudera-scm' for the database in the current
directory.

Generated at Fri Jun 29 16:25:43 PDT 2012.
```

Once you have gathered passwords, you can change the passwords for users, if desired.

Getting Help and Support

Cloudera Support

Cloudera can help you install, configure, optimize, tune, and run Hadoop for large scale data processing and analysis. Cloudera supports Hadoop whether you run our distribution on servers in your own data center, or on hosted infrastructure services such as Amazon EC2, Rackspace, SoftLayer, or VMware's vCloud.

If you are a Cloudera customer, you can:

- Create a Cloudera Support Ticket.
- Visit the Cloudera Knowledge Base.
- Learn how to register for an account to create a support ticket.

Community Support

Register for the Cloudera Manager Users group.

Register for the CDH Users group.

Report Issues

Cloudera tracks software and documentation bugs and enhancement requests for CDH on issues.cloudera.org. Your input is appreciated, but before filing a request, please search the Cloudera issue tracker for existing issues and send a message to the CDH user's list, cdh-user@cloudera.org, or the CDH developer's list, cdh-dev@cloudera.org.

Get Announcements about New CDH and Cloudera Manager Releases

Cloudera provides the following public mailing lists that send announcements about new CDH and Cloudera Manager product releases and updates:

- To receive CDH release announcements, subscribe to the CDH-announce list.
- To receive Cloudera Manager release announcements, subscribe to the CM-announce list.

Appendix A - Understanding Custom Installation Solutions

Cloudera hosts two types of software repositories that you can use to install products such as Cloudera Manager or CDH — repositories of RPM (RHEL and SLES) and Debian/Ubuntu packages, and parcel repositories, newly available with Cloudera Manager 4.5.

With parcels you can download, distribute and activate a new CDH version all from within Cloudera Manager. Further, only the Cloudera Manager server needs Internet access for downloading the desired parcel to a local repository on the Cloudera Manager server. Distribution of parcels to the remaining cluster members does not require internet access. Parcels are available for CDH 4.1.3 and onwards. Cloudera Manager continues to work with RPM (RHEL and SLES) and Debian/Ubuntu packages.

These repositories are effective solutions in most cases, but custom installation solutions are sometimes required. Using the software repositories requires client access over the Internet and results in the installation of the latest version of products.

An alternate solution is required if:

- You need to install older product versions. For example, in a CDH cluster, all hosts must run the same CDH version. After completing an initial installation, you may want to add nodes. This could be to increase the size of your cluster to handle larger tasks or to replace older hardware.
- The hosts on which you want to install Cloudera products are not connected to the Internet, so they are unable to reach the Cloudera repository. (Note that for a parcel installation, only the Cloudera manager server needs Internet access, but for a package installation, all cluster members need access to the Cloudera repository). Some organizations choose to partition parts of their network from outside access. Isolating segments of a network can provide greater assurance that valuable data is not compromised by individuals out of maliciousness or for personal gain. In such a case, the isolated computers are unable to access Cloudera's software repositories for new installations or upgrades.

In both of these cases, using a custom repository solution allows you to meet the needs of your organization, whether that means installing older versions of Cloudera software or installing any version of Cloudera software on machines that are disconnected from the Internet.

Understanding Parcels

Parcel is a new packaging format that facilitates upgrading CDH from within the Cloudera Manager Admin console. You download, distribute, and activate a parcel from within the **Parcels** page, found under the **Hosts** tab in the Admin console.

Cloudera Manager downloads a parcel to a local repository, by default at `/opt/cloudera/parcel-repo`. (The location is configurable — see "Parcel Configuration Settings" in Managing Parcels.) Once the parcel is downloaded to the CM server, an internet connection is no longer needed to deploy the parcel. Once you click "Distribute", every Cloudera Manager agent will start to download the parcel from the Cloudera Manager server.

If your Cloudera Manager server does not have Internet access, you can obtain the required parcel file(s) and put them into the local repository. Once you copy the .parcel file into that directory and create the associated .sha file, CM will automatically pick it up and show it in the parcel page.

See Creating a Local Parcel Repository for instructions.

Understanding Package Management

Before getting into the details of how to configure a custom package management solution in your environment, it can be useful to have more information about:

- How package management tools work

- Which tools come with which operating systems
- Each tool's configuration files

How Do Packaging and Package Management Tools Interact?

Packages (`rpm` or `deb` files) help ensure that installations complete successfully by encoding each package's dependencies. That means that if you request the installation of a solution, all required elements can be installed at the same time. For example, `hadoop-0.20-hive` depends on `hadoop-0.20`. Package management tools, such as yum (RedHat), zypper (SUSE), or apt-get (Debian/Ubuntu) are tools that can find and install any required packages. For example, for RedHat, you might enter `yum install hadoop-0.20-hive`. Yum would inform you that the hive package requires `hadoop-0.20` and offers to complete that installation for you. Zypper and apt-get provide similar functionality.

How Do Package Management Tools Find all Available Packages?

Package management tools rely on a list of repositories. Information about the tool's repository is stored in configuration files, the location of which varies according to the particular package management tool.

- Yum on RedHat/CentOS: `/etc/yum.repos.d`
- Zypper on SUSE: `/etc/zypp/zypper.conf`
- Apt-get on Debian/Ubuntu: `/etc/apt/apt.conf` (Additional repositories are specified using `*.list` files in the `/etc/apt/sources.list.d/` directory.)

For example, on a typical CentOS system, you might find:

```
[user@localhost ~]$ ls -l /etc/yum.repos.d/
total 24
-rw-r--r-- 1 root root 2245 Apr 25  2010 CentOS-Base.repo
-rw-r--r-- 1 root root  626 Apr 25  2010 CentOS-Media.repo
```

Inside those `.repo` files are pointers to one or many repositories. There are similar pointers inside configuration files for zypper and apt-get. In the following snippet from `CentOS-Base.repo`, there are two repositories defined: one named `Base` and one named `Updates`. The `mirrorlist` parameter points to a website which has a list of places where this repository can be downloaded.

```
# ...
[base]
name=CentOS-$releasever - Base
mirrorlist=http://mirrorlist.centos.org/?release=$releasever&arch=$basearch&repo=os
#baseurl=http://mirror.centos.org/centos/$releasever/os/$basearch/
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-5

#released updates
[updates]
name=CentOS-$releasever - Updates
mirrorlist=http://mirrorlist.centos.org/?release=$releasever&arch=$basearch&repo=updates
#baseurl=http://mirror.centos.org/centos/$releasever/updates/$basearch/
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-5
# ...
```

You can list the repositories you have enabled. The command varies according to operating system:

- RedHat/CentOS: `yum repolist`
- SUSE: `zypper repos`
- Debian/Ubuntu: `Apt-get` does not include a command to display sources, but you can determine sources by reviewing the contents of `/etc/apt/sources.list` and any files contained in `/etc/apt/sources.list.d/`.

The following shows an example of what you might find on a CentOS system in repolist:

```
[root@localhost yum.repos.d]$ yum repolist
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
 * addons: mirror.san.fastserv.com
 * base: centos.eecs.wsu.edu
 * extras: mirrors.ecvps.com
 * updates: mirror.5ninesolutions.com
repo id                         repo name                                 status
addons                          CentOS-5 - Addons                         enabled:
 0
base                           CentOS-5 - Base                          enabled: 3,434
extras                          CentOS-5 - Extras                         enabled:
296
updates                        CentOS-5 - Updates                       enabled: 1,137
repolist: 4,867
```

Creating and Using your own Repository

Custom repositories leverage package management solutions, which were designed to provide a convenient, efficient means to install software on many computers. This topic describes how to create a local package repository and then how to direct machines in your environment to use that repository.

To create a repository, you simply put the RPMs or DEBs you want to host in one directory. Then complete tasks with createrepo or reprepro, and then publish the resulting repository on a website.

Step 1: Download Installation Files

Creating a custom repository requires RPM or DEB files. If you already have these files, proceed to the next step. Otherwise, download the appropriate versions of Cloudera Manager. Available versions can be found at Cloudera Manager Version and Download Information.

Step 2: Prepare the RPM or DEB Files

Move the RPMs or DEBs to a directory that you will use for your repository. Suppose you have downloaded the Oracle JDK and want to host it internally to ease installation on various machines. After you have unpacked the RPMs or DEBs, you would have a collection of those files in your directory. For example, for a CentOS system, you might have:

```
$ls -l
total 79392
-rw-r--r-- 1 user group 70530937 Mar 13 14:42 jdk-6u24-linux-amd64.rpm
-rw-r--r-- 1 user group   499375 Mar 13 14:42 sun-javadb-client-10.6.2-1.1.i386.rpm
-rw-r--r-- 1 user group    14627 Mar 13 14:42 sun-javadb-common-10.6.2-1.1.i386.rpm
-rw-r--r-- 1 user group  4080625 Mar 13 14:42 sun-javadb-core-10.6.2-1.1.i386.rpm
-rw-r--r-- 1 user group   969861 Mar 13 14:42 sun-javadb-demo-10.6.2-1.1.i386.rpm
-rw-r--r-- 1 user group  4865183 Mar 13 14:42 sun-javadb-docs-10.6.2-1.1.i386.rpm
-rw-r--r-- 1 user group   201273 Mar 13 14:42 sun-javadb-javadoc-10.6.2-1.1.i386.rpm
```

Step 3: Create a Repository

You can using createrepo or reprepro to create a repository. If you don't have createrepo or reprepro installed, install it using the following command:

- RedHat/CentOS: yum install createrepo
- SLES: zypper install createrepo
- Debian/Ubuntu: apt-get install reprepro

Creating a CentOS/RHEL/Oracle/SLES Repository

Having expanded the RPMs to a directory and ensured `createrepo` is installed, you can now create a repository. When you run `createrepo` in the directory that contains the files you will use to create a repo, the program creates an extra directory with some XML files that describing the repository. For example:

```
$ createrepo .
7/7 - sun-javadb-javadoc-10.6.2-1.1.i386.rpm
Saving Primary metadata
Saving file lists metadata
Saving other metadata
$ ls -l
total 79400
-rw-r--r-- 1 user group 70530937 Mar 13 14:42 jdk-6u24-linux-amd64.rpm
drwxr-xr-x 2 user group     4096 Mar 13 14:45 repodata/
-rw-r--r-- 1 user group   499375 Mar 13 14:42 sun-javadb-client-10.6.2-1.1.i386.rpm
-rw-r--r-- 1 user group    14627 Mar 13 14:42 sun-javadb-common-10.6.2-1.1.i386.rpm
-rw-r--r-- 1 user group  4080625 Mar 13 14:42 sun-javadb-core-10.6.2-1.1.i386.rpm
-rw-r--r-- 1 user group   969861 Mar 13 14:42 sun-javadb-demo-10.6.2-1.1.i386.rpm
-rw-r--r-- 1 user group  4865183 Mar 13 14:42 sun-javadb-docs-10.6.2-1.1.i386.rpm
-rw-r--r-- 1 user group   201273 Mar 13 14:42 sun-javadb-javadoc-10.6.2-1.1.i386.rpm
$ ls -l repodata/
total 64
-rw-r--r-- 1 user group 32997 Mar 13 14:45 filelists.xml.gz
-rw-r--r-- 1 user group   482 Mar 13 14:45 other.xml.gz
-rw-r--r-- 1 user group  2429 Mar 13 14:45 primary.xml.gz
-rw-r--r-- 1 user group   951 Mar 13 14:45 repomd.xml
```

After this command completes, the specified RPMs have been added to your private repository.

> **Note:**
>
> Over time, you may need to install multiple versions Cloudera software. To achieve this, you may need to update repository information. Do not create a situation where there are multiple repo files that package management tools could use to install Cloudera software. Either overwrite existing repo files with the new information or create a new repo file and delete the old one. If your machines have multiple repo files, installations may not complete as expected.

Creating a Debian/Ubuntu Repository

Having expanded the DEBs to a directory and ensured `reprepro` is installed, you can now create a repository.

To create a Debian/Ubuntu repository:

Create a directory that will host the repository and creating a configuration directory and file. The repo directory can have any location and name. The configuration directory must be named `conf` and be at the root level of the repository directory you create. For example:

```
$ mkdir /tmp/repo
$ mkdir /tmp/repo/conf
```

Create a configuration file that contains basic information about the repository. For example, after creating such a file, you could check its contents as follows:

```
$ cat /tmp/repo/conf/distributions
Origin: Cloudera
Label: Cloudera
Suite: stable
Codename: cloudera
Version: 0.1
Architectures: i386 amd64 source
Components: contrib
Description: Cloudera
```

Run `reprepro` on DEB files. If you had expanded the DEBs to the `/tmp/repo/` directory, you might use the following command:

```
$ find /tmp/repo -name \*.deb -exec reprepro -Vb repo includedeb cloudera {} \;
```

After this command completes, the specified DEBs have been added to your private repository.

Step 4: Install a Web Server

The repository is typically hosted using HTTP on a machine inside your network. If you already have a web server in your organization, you can move the repository directory, which will include both the RPMs and the `repodata/` subdirectory, to some a location hosted by the web server. If you are able to use an existing web server, then note the URL and skip to Modifying Clients to Find Repos.

An easy web server to install is the Apache HTTPD.

To install Apache HTTPD:

Install Apache HTTPD. You may need to respond to some prompts to confirm you want to complete the installation. **For RedHat/CentOS:**

```
[root@localhost yum.repos.d]$ yum install httpd
```

For SLES:

```
[root@localhost zypp]$ zypper install httpd
```

For Debian/Ubuntu:

```
[root@localhost apt]$ apt-get install httpd
```

Start Apache HTTPD: **For RedHat**

```
[root@localhost tmp]$  service httpd start
Starting httpd:                                            [  OK  ]
```

For SLES

```
[root@localhost tmp]$ service apache2 start
Starting httpd:                                            [  OK  ]
```

For Debian/Ubuntu

```
[root@localhost tmp]$ service apache2 start
Starting httpd:                                            [  OK  ]
```

Step 5: Publish Repository Files

Move your files to the web server directory and modify file permissions. For example, you might use the following commands:

```
[root@localhost tmp]$ mv /tmp/repo /var/www/html
[root@localhost tmp]$ chmod -R ugo+rX /var/www/html/repo
```

After moving files and changing permissions, visit `http://<hostname>:80/repo` to verify that you see an index of files. Note that Apache may have been configured to not show indexes, which is also acceptable.

Modifying Clients to Find Repos

Having established the repository, modify the clients so they find the repository.

For RedHat/CentOS systems: Create files on client systems with the following information and format, where `hostname` is the name of the web server you created in the previous step:

```
[myrepo]
name=myrepo
baseurl=http://hostname/repo
enabled=1
gpgcheck=0
```

See `man yum.conf` for more details. Put that file into `/etc/yum.repos.d/myrepo.repo` on all of your host machines to enable them to find the packages that you are hosting.

For SLES systems: Use the zypper utility to update client system repo information by issuing the following command:

```
$ zypper addrepo http://hostname/repo alias
```

For Debian/Ubuntu systems: Add a new `list` file to `/etc/apt/sources.list.d/` on client systems. For example, you might create the file `/etc/apt/sources.list.d/my-private-cloudera-repo.list`. In that file, create an entry to your newly created repository. For example:

```
$ cat /etc/apt/sources.list.d/my-private-cloudera-repo.list
deb http://hostname/repo cloudera
```

After adding your `.list` file, ensure `apt-get` uses the latest information by issuing the following command:

```
$ sudo apt-get update
```

After completing these steps, you have established the environment necessary to install a previous version of Cloudera Manager or install Cloudera Manager to machines that are not connected to the Internet. Proceed with the installation process, being sure to target the newly created repository with your package management tool.

Installing Older Versions of Cloudera Manager 4

The Cloudera Manager installation solutions, such as the installer downloadable from the Cloudera Downloads website, install the most recent version of Cloudera Manager. This ensures that you install the latest features and bug fixes. While having the latest version of Cloudera Manager is valuable, in some cases it may be necessary to install previous versions.

> **Note:**
>
> These instructions apply to installing a previous version of Cloudera Manager 4.x. You cannot use these instructions to install a version of Cloudera Manager 3.7.

The most common reason to install a previous version is when you want to expand an existing cluster. In this case the **Add Hosts** wizard allows you to specify that the version of the Cloudera Manager agent installed on your new hosts should match the version of Cloudera Manager running on your cluster (this is the default).

You may also want to install a previous version of the Cloudera Manager server on a new cluster if, for example, you have validated a specific version and want to deploy that version on additional clusters.

You can also add a cluster to be managed by the same instance of Cloudera Manager – you do this using the **Add Cluster** feature from the Services page in the Cloudera Manager Admin Console. When you do this, the default choice will be to use the same Cloudera manager version as used on the other clusters under the same Cloudera Manager.

Adding Hosts to cluster running a previous version of Cloudera Manager

In the Cloudera Manager Admin Console, go the the **Hosts** tab.

Click **Add Hosts**.

When you reach the page where you choose the software versions to install, select "Matched repository for this Cloudera Manager server" as your choice for Cloudera Manager. This will ensure that the version of Cloudera Manager used for your new host matches the version installed on the rest of your cluster. See Adding a Host to the Cluster for further information.

If you are using a local mirror, you can specify the mirror using the "custom repository" option. You must ensure that you have mirrored the same correct older version.

Adding a new cluster to an existing Cloudera Manager deployment

In the Cloudera Manager Admin Console, go the the **Services** tab.

Click **Add Cluster...**.

When you reach the page where you choose the software versions to install, select "Matched repository for this Cloudera Manager server" as your choice for Cloudera Manager. This will ensure that the version of Cloudera Manager used for your new host matches the version installed on the rest of your cluster. See Adding a Cluster for further information.

If you are using a local mirror, you can specify the mirror using the "custom repository" option. You must ensure that you have mirrored the same correct older version.

Installing a Previous Version of the Cloudera Manager Server

Installing an older version of Cloudera Manager requires several manual steps to install and configure the database and the correct version of the Cloudera Manager server. Once these are done, you can run the Express wizard to complete the installation of Cloudera Manager and CDH.

- For Red Hat/CentOS/SLES/Oracle, download and edit the Cloudera Manager repo file to point to the version of Cloudera Manager you want to use. For Debian or Ubuntu, download and edit the `cloudera.list` file.
- Install the Oracle JDK
- Install a Database for the Cloudera Manager Server and Services
- Install the Cloudera Manager Server
- Configure the Database for the Cloudera Manager Server
- Start the Cloudera Manager Server
- Log in to Cloudera Manager to run the Express Wizard and proceed with the rest of the installation.

Step 1: Download and edit the cloudera-manager.repo or cloudera.list file

Edit the cloudera-manager.repo file for Red Hat/CentOS or SLES

Download the Cloudera Manager repo file (`cloudera-manager.repo`) for your OS version using the links provided in the Cloudera Manager Version and Download Information page. For example, for Red Hat/CentOS 6, this is found at `http://archive.cloudera.com/cm4/redhat/6/x86_64/cm/cloudera-manager.repo`

Edit the file to change the baseurl to point to the specific version of Cloudera Manager you want to download. For example, if you want to install Cloudera Manager version 4.0.4, change:
`baseurl=http://archive.cloudera.com/cm4/redhat/6/x86_64/cm/4/` to
`baseurl=http://archive.cloudera.com/cm4/redhat/6/x86_64/cm/4.0.4/`

Save the edited file: For Red Hat/CentOS, save it in the directory `/etc/yum.repos.d/` For SLES, save it in `/etc/zypp/repos.d`

Edit the cloudera.list file for Debian or Apt

Download the Cloudera Manager list file (`cloudera.list`) using the links provided at Cloudera Manager Version and Download Information. For example, for for Ubuntu 10.04 (lucid), this is found at `http://archive.cloudera.com/cm4/ubuntu/lucid/amd64/cm/cloudera.list`

Edit the file to change the second-to-last element to specify the version of Cloudera Manager youw ant to install: For example, with Ubuntu lucid, for if you want to install Cloudera Manager version 4.0.4, change: `deb http://archive.cloudera.com/cm4/ubuntu/lucid/amd64/cm lucid-cm4 contrib` to `deb http://archive.cloudera.com/cm4/ubuntu/lucid/amd64/cm lucid-cm4.0.4 contrib`.

Save the edited file in the directory `/etc/apt/sources.list.d/`

Step 2: Install the Oracle JDK

The JDK is included in the Cloudera Manager 4.x repos. Once you have the repo or list file in the corrct place, you can install the JDK as follows: **On a Red Hat/CentOS system:**

```
$ sudo yum install jdk
```

On a SUSE system:

```
$ sudo zypper install jdk
```

On a Debian/Ubuntu system:

```
$ sudo apt-get install jdk
```

Step 3: Install a Database for the Cloudera Manager Server and Services

Cloudera Manager supports various database solutions including the PostgreSQL embedded database, PostgreSQL external databases, MySQL databases, or Oracle databases. These instructions will discuss installing either the embedded PostgreSQL database, or an externla MySQL database. For other options, see Installing and Configuring Databases in the Cloudera Manager Installation Guide.

Installing the Embedded PostGreSQL Database

On a Red Hat/CentOS system:

```
$ sudo yum install cloudera-manager-server-db
```

On a SUSE system:

```
$ sudo zypper install cloudera-manager-server-db
```

On a Debian/Ubuntu system:

```
$ sudo apt-get install cloudera-manager-server-db
```

Installing the MySQL Database for the Cloudera Manager Server

Follow the instructions in Installing and Configuring a MySQL Database. These instructions lead you through installing and configuring MySQL and setting up the databases that will be used by the Cloudera Management Services (Activity Monitor, Service Monitor, Host Monitor and Report Manager).

Step 4: Install the Cloudera Manager Server

On the Cloudera Manager Server machine, type the following commands to install the Cloudera Manager packages.

On a Red Hat/CentOS system:

```
$ sudo yum install cloudera-manager-daemons
$ sudo yum install cloudera-manager-server
```

On a SUSE system:

```
$ sudo zypper install cloudera-manager-daemons cloudera-manager-server
```

On a Debian/Ubuntu system:

```
$ sudo apt-get install cloudera-manager-daemons cloudera-manager-server
```

Step 5: Configure the Database for the Cloudera Manager Server

Depending on which database you have chosen, do one of the two procedures to initialize the database for use by the Cloudera Manager server.

Initialize and start the embedded PostgreSQL database

Prepare the embedded PostgreSQL database for use with the Cloudera Manager Server by running this command:

```
$ sudo service cloudera-scm-server-db initdb
```

Start the embedded PostgreSQL database by running this command:

```
$ sudo service cloudera-scm-server-db start
```

Prepare the MySQL database for the Cloudera Manager Server

Cloudera Manager configuration can be completed using the `scm_prepare_database.sh script`, which is installed in the `/usr/share/cmf/schema` directory on the host where the Cloudera Manager Server package is installed. You must run the script on the Cloudera Manager Server host.

This script enables Cloudera Manager Server to connect to an external database in MySQL, PostgreSQL, or Oracle. The script prepares the database by:

- Creating the Cloudera Manager Server database configuration file.
- Creating a database for the Cloudera Manager Server to use. This is optional and is only completed if options are specified.
- Setting up a user account for the Cloudera Manager Server. This is optional and is only completed if options are specified.

Script syntax

```
scm_prepare_database.sh database-type [options] database-name username password
```

Required Parameter	Description
`database-type`	To connect to a MySQL database, specify `mysql` as the database type. To connect to an Oracle database,

Required Parameter	Description
	specify `oracle`. To connect to an external PostgreSQL database, specify `postgresql`.
`database-name`	The name of the Cloudera Manager Server database you want to create.
`username`	The username for the Cloudera Manager Server database you want to create.
`password`	The password for the Cloudera Manager Server database you want to create. If you don't specify the password on the command line, the script will prompt you to enter it.

Option	Description
-h or --host	The IP address or hostname of the host where MySQL or Oracle is installed. The default is to use the local host.
-P or --port	The port number to use to connect to MySQL or Oracle. The default port is 3306. This option is used for a remote connection only.
-u or --user	The username for the MySQL or Oracle application. The default is `root`.
-p or --password	The password for the MySQL or Oracle application. The default is no password.
--scm-host	The hostname where the Cloudera Manager Server is installed. Omit if the Cloudera Manager server and MySQL or Oracle are installed on the same host.
--config-path	The path to the Cloudera Manager Server configuration files. The default is `/etc/cloudera-scm-server`.
--schema-path	The path to the Cloudera Manager schema files. The default is `/usr/share/cmf/schema` (the location of the script).
-f	The script will not stop if an error is encountered.
-? or --help	Display help.

Note:

You can also run `scm_prepare_database.sh` without options to see the syntax.

Example: Running the script when MySQL is installed on another host

This example explains how to run the script on the Cloudera Manager Server machine (*myhost2*) and create and use a temporary MySQL user account to connect to MySQL remotely on the MySQL machine (*myhost1*).

At *myhost1*'s MySQL prompt, create a temporary user who can connect from *myhost2*:

```
mysql> grant all on *.* to 'temp'@'%' identified by 'temp' with grant option;
Query OK, 0 rows affected (0.00 sec)
```

On the Cloudera Manager Server host (*myhost2*), run the script:

```
$ sudo /usr/share/cmf/schema/scm_prepare_database.sh mysql -h myhost1.sf.cloudera.com
 -u temp -ptemp --scm-host myhost2.sf.cloudera.com scm scm scm
```

After the script has completed, on *myhost1*, delete the temporary user:

```
mysql> drop user 'temp'@'%';
Query OK, 0 rows affected (0.00 sec)
```

For more details, or information on running this script with other databases see Step 3: Configure a Database for the Cloudera Manager Server in the Cloudera Manager Installation Guide.

Step 6: Start the Cloudera Manager Server

Start the Cloudera Manager Server by running this command:

```
$ sudo service cloudera-scm-server start
```

If you have problems starting the Server, such as database permissions problems, you can use the Server's log `/var/log/cloudera-scm-server/cloudera-scm-server.log` to troubleshoot the problem.

Step 7: Start the Cloudera Manager installation wizard

Enter the URL and port for the Cloudera Manager server: `http://<Server host>:<port>`

Log into Cloudera Manager. The default credentials are: Username: admin Password: admin

You are now at the first step of the Express Installation Wizard. Follow the steps as described Installation Path A - Automated Installation by Cloudera Manager for installing CDH and any other packages that may be available.

Creating a Local Parcel Repository

If the Cloudera Manager server does not have Internet access, you can access the Cloudera parcels directories (at `http://archive.cloudera.com/cdh4/parcels/` or `http://archive.cloudera.com/impala/parcels/`) from another location, and then drop the `.parcel` file into your local parcel-repo directory. You will also need to create a `.sha` file from the information found in the `manifest.json` file in the parcels directory for the parcel version you want to use.

To make a parcel available for distribution on your cluster:

Verify the location of the local parcel repository on your Cloudera Manager server: Go to the **Administration** page, **Properties** tab, **Parcels** category. You can change the local repository path in the **Local Parcel Repository Path** property. By default it is `/opt/cloudera/parcel-repo`.

Go to Cloudera's parcel repository at `http://archive.cloudera.com/cdh4/parcels/` or `http://archive.cloudera.com/impala/parcels/`.

Go to the directory for the software version you want to make available to your cluster.

Copy the `.parcel` file for your operating system: (`el5` or `el6` for Red Hat 5 or 6, `lucid` or `precise` for Ubuntu and so on) and place it into the local parcel repository on your Cloudera Manager server.

Open the `manifest.json` file in the same directory as the `.parcel` file you just copied.

Find the section of the manifest that corresponds to the parcel you downloaded: For example, if you are running RHEL 6 and copied the parcel file `CDH-4.2.0-1.cdh4.2.0.p0.10-el6.parcel`, then you would look for the section:

```
 {
      "parcelName": "CDH-4.2.0-1.cdh4.2.0.p0.10-el6.parcel",
      "components": [
        { "name":     "flume-ng",
          "version":  "1.3.0-cdh4.2.0",
          "pkg_version":   "1.3.0+86"
        }
        ,{ "name":     "mr1",
          "version":  "2.0.0-mr1-cdh4.2.0",
          "pkg_version":   "0.20.2+1341"
        }
        ,{ "name":     "hadoop-hdfs",
          "version":  "2.0.0-cdh4.2.0",
          "pkg_version":   "2.0.0+922"
        }
          . . . .  <snip> . . .

        ,{ "name":     "whirr",
          "version":  "0.8.0-cdh4.2.0",
          "pkg_version":   "0.8.0+21"
        }
        ,{ "name":     "zookeeper",
          "version":  "3.4.5-cdh4.2.0",
          "pkg_version":   "3.4.5+14"
        }
      ],
      "hash": "f1a08b5f7aeef6335d577c5f6fad0bca55f0c2d9"
    },
```

Create a text file whose name is `<parcel file name> .sha` (e.g. `CDH-4.2.0-1.cdh4.2.0.p0.10-el6.parcel.sha`) and copy the hash code into it: e.g.

```
# cat > CDH-4.2.0-1.cdh4.2.0.p0.10-el6.parcel.sha
f1a08b5f7aeef6335d577c5f6fad0bca55f0c2d9
^C
```

Place this file into your local parcel repository.

Once these files are in place, Cloudera Manager will pick up the parcel and it will appear on the **Hosts** > **Parcels** page. Note that how quickly this occurs depends on the **Parcel Update Frequency** setting, set by default to 1 hour. You can change this on the **Administration** page, **Properties** tab, **Parcels** category.

Using Whirr to Launch Cloudera Manager

Cloudera Manager provides an installation wizard that installs Cloudera Manager, CDH and Impala on a cluster of Amazon Web Services (AWS) EC2 instances. See Installing Cloudera Manager and CDH on EC2. Alternatively, you can install Cloudera Manager using Whirr following the instructions here. Follow these instructions to start a cluster on Amazon Elastic Compute Cloud (EC2) running Cloudera Manager. Cloudera Manager allows you to install, run, and manage a Hadoop cluster.

> **Note:**
>
> At present you can launch and run only an MRv1 cluster; YARN is not supported.

This method uses Whirr to start a cluster with:

- one node running the Cloudera Manager Admin Console, and
- a user-selectable number of nodes for the Hadoop cluster itself.

Once Whirr has started the cluster, you use Cloudera Manager in the usual way.

Step 1: Set your AWS credentials as environment variables

Run the following commands from your local machine:

```
$ export AWS_ACCESS_KEY_ID=...
$ export AWS_SECRET_ACCESS_KEY=...
```

Step 2: Install Whirr

Install CDH repositories; for example for CDH4, see the CDH4 Installation Guide.

Install the whirr package; for example for CDH4, see the Installing Whirr heading in Whirr Installation topic in the CDH4 Installation Guide.

Create environment variables:

```
$ export WHIRR_HOME=/usr/lib/whirr
$ export PATH=$WHIRR_HOME/bin:$PATH
```

Step 3: Create a password-less SSH Key Pair

Create a password-less SSH Key Pair for Whirr to use:

```
ssh-keygen -t rsa -P '' -f ~/.ssh/id_rsa_cm
```

Step 4: Get your Whirr-Cloudera-Manager Configuration

You can download a sample Whirr EC2 Cloudera Manager configuration as follows:

```
$ curl -O https://raw.github.com/cloudera/whirr-cm/master/cm-ec2.properties
```

To upload a Cloudera Manager License as part of the installation (Cloudera can provide this if you do not have one), place the license in a file `cm-license.txt` on the Whirr classpath (for example in `$WHIRR_HOME/conf`), using a command such as the following:

```
$ mv -v eval_acme_20120925_cloudera_enterprise_license.txt
$WHIRR_HOME/conf/cm-license.txt
```

To upload a Cloudera Manager configuration as part of the installation, place the configuration in a file called `cm-config.json` on the Whirr classpath (for example in `$WHIRR_HOME/conf`). The format of this file should match the JSON as downloaded from the Cloudera Manager UI. For example:

```
$ curl -O https://raw.github.com/cloudera/whirr-cm/master/cm-config.json
$ mv -v cm-config.json $WHIRR_HOME/conf/cm-config.json
```

Step 5: Launch a Cloudera Manager Cluster

The following command starts a cluster with five Hadoop nodes:

```
$ whirr launch-cluster --config cm-ec2.properties
```

Note:

- To change the number of nodes edit the `whirr.instance-templates` line in the `cm-ec2.properties` file. For example, to launch a cluster with 20 nodes: `whirr.instance-templates=1 cmserver,20 cmagent`
- To add a no-op node to use as gateway node: `whirr.instance-templates=1 cmserver,20 cmagent,1 noop`

Whirr reports progress to the console as it runs. The command exits when the cluster is ready to be used.

Using the Cluster

Once the Hadoop cluster is up and running you can run jobs from any Cloudera Manager Agent machine, or from a Cloudera Manager gateway node.

Using a Gateway Node (Optional)

In most cases, you will not a need a gateway node, but you may want to consider using one if you want to run jobs on a machine that is not also running CDH TaskTracker and DataNode processes. In that case, edit `whirr.instance-templates` to use the noop option shown in the previous section, launch the cluster, and then follow Cloudera Manager instructions to add a gateway role on the no-op node, which you can find in the documentation for your version of Cloudera Manager, for example at Adding Role Instances.

Then SSH to the gateway machine. Now you can interact with the cluster; for example, to list files in HDFS:

```
hadoop fs -ls /tmp
```

Shutting Down the Cluster

When you want to shut down the cluster, run the following command.

Important:

All data and state stored on the cluster will be lost.

```
whirr destroy-cluster --config cm-ec2.properties
```