Legal Notice
Copyright © 2011-2012 Hortonworks, Inc.

The text and illustrations in this document are licensed by Hortonworks under a Creative Commons Attribution–Share Alike 3.0 Unported license (“CC-BY-SA”). An explanation of CC-BY-SA is available at http://creativecommons.org/licenses/by-sa/3.0/. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Hortonworks, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.
Table of Contents

Getting Ready to Install............................................................................................................ 1
Understand the Basics ............................................................................................................ 1
Meet Minimum System Requirements .................................................................................... 2
  Hardware Recommendations ................................................................................................. 2
  Operating Systems Requirements ......................................................................................... 2
  Graphics Requirements ....................................................................................................... 2
  Software Requirements ....................................................................................................... 2
  Database Requirements ....................................................................................................... 3
Decide on Deployment Type .................................................................................................... 3
Collect Information ............................................................................................................. 3
Prepare the Environment ....................................................................................................... 3
  Check Existing Installs ......................................................................................................... 3
  Set Up Password-less SSH .................................................................................................. 4
  Enable NTP on the Cluster .................................................................................................. 4
  Check DNS ........................................................................................................................... 4
  Create Hostdetail.txt .......................................................................................................... 4
Optional: Configure the Local yum Repository ...................................................................... 5
Running the Installer ............................................................................................................ 6
  Set Up the Bits ....................................................................................................................... 6
    RHEL/CentOS 5.x ................................................................................................................ 6
    RHEL/CentOS 6.x ................................................................................................................ 6
  Start the Hortonworks Management Center Service ......................................................... 6
  Stop iptables .......................................................................................................................... 7
Configuring and Deploying the Cluster ................................................................................ 8
  Log into the Hortonworks Management Center ............................................................... 8
  Step 1: Create Cluster ......................................................................................................... 8
  Step 2: Add Nodes ............................................................................................................... 8
  Step 3: Select Services ........................................................................................................ 9
  Step 4: Assign Hosts .......................................................................................................... 9
  Step 5: Select Mount Points .............................................................................................. 9
  Step 6: Custom Config ....................................................................................................... 9
    Nagios ................................................................................................................................ 10
    Hive/HCatalog .................................................................................................................. 10
    HDFS ............................................................................................................................... 11
    MapReduce ...................................................................................................................... 11
    ZooKeeper ...................................................................................................................... 13
    HBase ............................................................................................................................. 13
    Oozie .............................................................................................................................. 15
    Miscellaneous ............................................................................................................... 15
    Recommended Memory Configurations for the MapReduce Service .......................... 16
  Step 7: Review and Deploy .................................................................................................. 16
Managing Your Cluster ......................................................................................................... 18
  Access the Cluster Management App .................................................................................. 18
  Cluster Summary ................................................................................................................ 18
  Manage Services ............................................................................................................... 18
  Add Nodes .......................................................................................................................... 19
  Uninstall ............................................................................................................................. 19
[Optional] Turn On Password Protection for HMC .............................................................. 19
Getting Ready to Install

This section describes the information and materials you need to get ready to install the Hortonworks Data Platform (HDP) using the Hortonworks Management Center (HMC). HMC is the GUI-based management and monitoring tool provided with HDP.

- Understand the Basics
- Meet Minimum System Requirements
- Decide on Deployment Type
- Collect Information
- Prepare the Environment
- Optional: Configure the Local yum Repository

Understand the Basics

The Hortonworks Data Platform consists of three layers.

- **Core Hadoop**: The basic components of Apache Hadoop.
  - **Hadoop Distributed File System (HDFS)**: A special purpose file system that is designed to work with the MapReduce engine. It provides high-throughput access to data in a highly distributed environment.
  - **MapReduce**: A framework for performing high volume distributed data processing using the MapReduce programming paradigm.

- **Essential Hadoop**: A set of Apache components designed to ease working with Core Hadoop.
  - **Apache Pig**: A platform for creating higher level data flow programs that can be compiled into sequences of MapReduce programs, using Pig Latin, the platform’s native language.
  - **Apache Hive**: A tool for creating higher level SQL-like queries using HiveQL, the tool’s native language, that can be compiled into sequences of MapReduce programs.
  - **Apache HCatalog**: A metadata abstraction layer that insulates users and scripts from how and where data is physically stored.
  - **Apache Templeton**: A component that provides a set of REST-like APIs for HCatalog and related Hadoop components.
  - **Apache HBase**: A distributed, column-oriented database that provides the ability to access and manipulate data randomly in the context of the large blocks that make up HDFS.
  - **Apache ZooKeeper**: A centralized tool for providing services to highly distributed systems. ZooKeeper is necessary for HBase installations.

- **HDP Support**: A set of components that allow you to monitor your Hadoop installation and to connect Hadoop with your larger compute environment.
  - **Apache Oozie**: A server based workflow engine optimized for running workflows that execute Hadoop jobs.
  - **Apache Sqoop**: A component that provides a mechanism for moving data between HDP and external structured data stores. Can be integrated with Oozie workflows.
  - **Ganglia**: An Open Source tool for monitoring high-performance computing systems.
• **Nagios**: An Open Source tool for monitoring systems, services, and networks.

You must always install Core Hadoop, but you can select the components from the other layers based on your needs. For more information on the structure of the HDP, see [Understanding Hadoop Ecosystem](#).

# Meet Minimum System Requirements

To run the Hortonworks Data Platform, your system must meet minimum requirements.

- **Hardware Recommendations**
- **Operating Systems Requirements**
- **Graphics Requirements**
- **Software Requirements**
- **Database Requirements**

## Hardware Recommendations

Although there is no single hardware requirement for installing HDP, there are some basic guidelines. You can see sample setups here: [Suggested Hardware for a Typical Hadoop Cluster](#).

## Operating Systems Requirements

The following operating systems are supported:

- 64-bit Red Hat Enterprise Linux (RHEL) v5.x or 6.x
- 64-bit CentOS v5.x or 6.x.

## Graphics Requirements

The HMC deployment wizard runs as a browser-based Web app. You must have a machine capable of running a graphical browser to use this tool.

## Software Requirements

On each of your hosts:

- yum
- rpm
- scp
- curl
- wget
- pdsh
- net-snmp
- net-snmp-utils

On the machine from which you will run HMC:

- Firefox v.12+

If you are installing to a group of hosts that do not have Internet access, you must have a copy of the EPEL (Extra Packages for Enterprise Linux) repo available somewhere that those hosts can access. The appropriate repository package for v5.x can be fetched from [http://download.fedora-project.org/pub/epel/5/i386/epel-release-5-4.noarch.rpm](http://download.fedora-project.org/pub/epel/5/i386/epel-release-5-4.noarch.rpm) and for v6.x from [http://mirrors.servercentral.net/fedora/epel/6/i386/epel-release-6-7.noarch.rpm](http://mirrors.servercentral.net/fedora/epel/6/i386/epel-release-6-7.noarch.rpm)
Database Requirements

Hive or HCatalog requires a MySQL database for its use. You can choose to use a current instance or let the HMC deployment wizard create one for you.

Decide on Deployment Type

While it is possible to deploy all of HDP on a single host, this is appropriate only for initial evaluation. In general you should use at least three hosts: one master host and two slaves.

Collect Information

To deploy your HDP installation, you need to collect the following information:

- The fully qualified domain name (FQDN) for each host in your system, and which component(s) you wish to set up on which host. The HMC deployment wizard does not support using IP addresses. You can use `hostname -f` to check for the FQDN if you do not know it.
- The base directories you wish to use as mount points for storing:
  - NameNode data
  - DataNodes data
  - MapReduce data
  - ZooKeeper data, if you install ZooKeeper
  - Various log, pid, and db files, depending on your install type
- The hostname (for an existing instance), database name, username, and password for the MySQL instance, if you install Hive/HCatalog.

NOTE: If you are using an existing instance, the user you create for HDP’s use must have adequate privileges to create a database and tables in that database.

Prepare the Environment

To deploy your HDP instance, you need to prepare your deploy environment:

- Check Existing Installs
- Set Up Password-less SSH
- Enable NTP on the Cluster
- Check DNS
- Create Hostdetail.txt

Check Existing Installs

HMC automatically installs the correct versions of the files that are necessary for HMC and HDP to run. Versions other than the ones that HMC uses can cause problems in running the installer, so remove any existing installs that do not match the following lists.

RHEL/CentOS v5.x

- Ruby 1.8.5-24.el5
- Puppet 2.7.9-2
- Ruby Rack 1.1.0-2.el5
RHEL/CentOS v6.x

• Ruby 1.8.7.*.el6
• Puppet 2.7.9-2.el6
• Ruby Rack 1.1.0-2.el6
• Passenger 3.0.12-1.el6
• Nagios 3.2.3-2.el6
• Nagios plug-ins 1.4.9-1
• Nagios Common 2.12-10.el6
• MySQL v. 5.x
• Ganglia - ganglia-gmond-3.2.0, ganglia-gmetad-3.2.0

Set Up Password-less SSH

You must set up password-less SSH connections between the main installation host and all other machines. The installation host acts as the client and uses the key-pair to access and interact with the other hosts in the cluster. Create public and private keys on the main installation host and copy the public keys to the root account on your target machines. Depending on your version of SSH, you may need to set permissions on your .ssh directory (to 700) and the authorized_keys file in that directory (to 640). You also need a copy of the private key available on the machine from which you run the browser-based deployment wizard.

Enable NTP on the Cluster

The clocks of all the nodes in your cluster must be able to synchronize with each other.

Check DNS

All hosts in your system must be configured for DNS and Reverse DNS.

NOTE: If you are unable to configure DNS and Reverse DNS, you must edit the hosts file on every host in your cluster to contain each of your hosts.

Create Hostdetail.txt

Create a text file listing the newline separated FQDNs of the hosts that make up your cluster. You must use FQDNs. HMC does not support the use of IP addresses. The file should look something like this:

    fully.qualified.hostname1
    fully.qualified.hostname2
    fully.qualified.hostname3
    etc.
INFO: If you are deploying on EC2, use the Internal hostname.

This file must available on the machine from which you run the browser-based deployment wizard.

IMPORTANT: If you are creating a single node installation, you must still create a hostdetail.txt file with a single entry.

Optional: Configure the Local yum Repository

If your cluster does not have access to the Internet, or you are creating a large cluster and you want to conserve bandwidth, you need to provide a local copy of the HDP repository. For more information, see Configuring a Local Mirror.
Running the Installer

This section describes the process for installing the Hortonworks Management Center (HMC) and preparing to deploy the Hortonworks Data Platform.

• Set Up the Bits
• Start the Hortonworks Management Center Service
• Stop iptables

Set Up the Bits

1. Log into the main installation host as root.

RHEL/CentOS 5.x

1. Download the file from the Hortonworks public repo.
   
   ```
   rpm -Uvh http://public-repo-1.hortonworks.com/HDP-1.0.1.14/repos/centos5/hdp-release-1.0.1.14-1.el5.noarch.rpm
   ```

2. Install the epel repository:
   
   ```
   yum install epel-release
   ```

3. Install other components for HMC:
   
   ```
   yum install php-pecl-json
   ```

4. Install the HMC bits using yum:
   
   ```
   yum install hmc
   ```

RHEL/CentOS 6.x

1. Download the file from the Hortonworks public repo.
   
   ```
   ```

2. Install the epel repository:
   
   ```
   yum install epel-release
   ```

3. Install the HMC bits using yum:
   
   ```
   yum install hmc
   ```

NOTE: To access the optional Talend tool set for either version:

   ```
   wget http://public-repo-1.hortonworks.com/HDP-1.0.0.14/tools/HDP-ETL-TOS_BD-V5.1.1.zip
   ```

Start the Hortonworks Management Center Service

The HMC service manages the deployment process.

1. From a shell on the main installation host, enter:
   
   ```
   service hmc start
   ```

2. Agree to the Oracle JDK license when asked. You must accept this license to be able to download the necessary JDK from Oracle. The JDK is installed during the deploy phase.

NOTE: If you already have a local copy of the Oracle JDK v 1.6 update 31 32 and 64-bit binaries accessible from the install host, you can skip this and the next step. Use the Miscell-
neous tab on the Customize Settings page in the deployment wizard to provide the path to your binaries.
3. Agree to let the installer download the JDK binaries.

Stop iptables
The deployment will not succeed with iptables running:

/etc/init.d/iptables stop
Configuring and Deploying the Cluster

This section describes using the HMC deployment wizard in your browser to complete your installation, configuration and deployment of HDP.

- Log into the Hortonworks Management Center
- Step 1: Create Cluster
- Step 2: Add Nodes
- Step 3: Select Services
- Step 4: Assign Hosts
- Step 5: Select Mount Points
- Step 6: Custom Config
- Step 7: Review and Deploy

Log into the Hortonworks Management Center

Once you have started the HMC service, you can access the HMC deployment wizard through your browser.

2. At the Welcome page, click the Get started button.

NOTE: Out of the box the HMC GUI is not password-protected. Once you have finished initial configuration and deployment, you can configure the GUI to require a username/password. See [Optional] Turn on Password Protection for information on doing this.

Step 1: Create Cluster

The first step of the deployment process creates the cluster name.

1. Type in a name for the cluster in the text box. No whitespaces or special characters can be used in the name.
2. Click the Next button.

Step 2: Add Nodes

In order to build up the cluster, the deployment wizard needs to access the private key file you created in Set Up Password-less SSH and the hostdetail.txt file you created in Create Hostdetail.txt. It uses these to locate all the hosts in the system and to access and interact with them securely.

1. Use the Choose File button to find the private key file that matches the public key you installed on all your hosts.
   
   NOTE: The related public key must already exist on all the hosts.

2. Use the Choose File button to find the host detail file.
3. If you are using a local repository (if your installation does not have access to the Internet), check Use local yum mirror instead of downloading packages from the Internet. Add the path to the local mirror in the popup text box.
4. Click the **Add Nodes** button.
5. A popup appears and displays the status of the installation host’s attempt to find, access, and bootstrap all the nodes in the cluster.
6. When all the steps are complete, click the **Proceed to Select Services** button.

**Step 3: Select Services**

Hortonworks Data Platform is made up of a number of components. You must always install HDFS and MapReduce, but you can decide which of the other services you want to install. See **Understand the Basics** for more information on your options.

---

**INFO:** The two monitoring tools, Nagios and Ganglia, are also automatically deployed if you use the deployment wizard.

1. Uncheck the boxes next to the services you do not want to deploy. Dependencies are automatically selected: selecting HBase selects ZooKeeper, and so forth.
2. Click the **Select Services** button.

**Step 4: Assign Hosts**

The HMC deployment wizard attempts to assign the various services you have selected on appropriate hosts in your cluster. Each host/node displays the memory and CPU cores available on that host.

1. If you wish to change locations, click the dropdown list next to the service and select the appropriate host.
2. When you are satisfied with the assignments, click the **Next** button.

**Step 5: Select Mount Points**

The deployment wizard needs to determine a base directory for storing various files, like data files, logs, and pids, on your cluster.

1. To change the suggested mount points, uncheck the given options.
2. Enter a comma-separated list of your preferred mount points into the text box.
3. If you wish to see exactly which directories will be used with your choices, click **Preview directories to be used**. A popup with the information appears. Click the **OK** button to close.
4. Click the **Next** button.

**Step 6: Custom Config**

The Customize Settings screen presents you with a set of tabs that let you manage configuration settings for HDP components. The deployment wizard sets reasonable defaults for each of the options here, but you can use this set of tabs to tweak those settings. Hover your mouse over each of the parameters to see a brief description of what it does. There are eight groups of configuration parameters:

- Nagios
- Hive/HCatalog
• HDFS
• MapReduce
• ZooKeeper
• HBase
• Oozie
• Miscellaneous

Nagios
The first tab covers access settings for the Nagios monitoring tool.

<table>
<thead>
<tr>
<th>Name</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nagios Admin User</td>
<td>Nagios Web UI Admin username [default:nagiosadmin]</td>
</tr>
<tr>
<td>Nagios Admin Password</td>
<td>Nagios Web UI Admin password. This is a required parameter.</td>
</tr>
<tr>
<td>Retype Nagios Admin Password</td>
<td></td>
</tr>
<tr>
<td>Hadoop Admin email</td>
<td>The email address to which Nagios should send alerts. This is a required parameter.</td>
</tr>
</tbody>
</table>

Table 1: Nagios Settings

Hive/HCatalog
The second tab covers Hive/HCatalog settings for the MySQL instance:

<table>
<thead>
<tr>
<th>Name</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>MySQL host</td>
<td>MySQL host on which the Hive/HCatalog Metastore is hosted. Specify this path only if you wish to use an existing MySQL instance. Leave blank if you wish to have HMC create a new instance for you.</td>
</tr>
<tr>
<td>MySQL Database Name</td>
<td>MySQL Database name for the Hive/HCatalog Metastore. Any legal name you like</td>
</tr>
<tr>
<td>MySQL user</td>
<td>The username for accessing the Hive/HCatalog Metastore</td>
</tr>
<tr>
<td>MySQL Password</td>
<td>The password for accessing the Hive/HCatalog Metastore. This is a required parameter.</td>
</tr>
<tr>
<td>Retype MySQL Password</td>
<td></td>
</tr>
</tbody>
</table>

Table 2: Hive/HCatalog Settings
HDFS
The third tab covers HDFS settings.

<table>
<thead>
<tr>
<th>Name</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>NameNode directories</td>
<td>NameNode directories for HDFS to store the file system image. Based on the mount points and host settings you chose previously.</td>
</tr>
<tr>
<td>DataNode directories</td>
<td>DataNode directories for HDFS to store the data blocks. Based on the mount points and host settings you chose previously.</td>
</tr>
<tr>
<td>SecondaryNameNode Checkpoint directory</td>
<td>Directory on the local filesystem where the Secondary NameNode should store the temporary images to merge. Based on the mount points and host settings you chose previously.</td>
</tr>
<tr>
<td>WebHDFS enabled</td>
<td>Check the box to enable WebHDFS</td>
</tr>
<tr>
<td>Hadoop maximum Java heap size</td>
<td>Maximum Java heap size for daemons such as Balancer (Java option -Xmx)</td>
</tr>
<tr>
<td>NameNode Java heap size</td>
<td>Initial and maximum Java heap size for NameNode (Java options -Xms and -Xmx)</td>
</tr>
<tr>
<td>NameNode new generation size</td>
<td>Default size of Java new generation for NameNode (Java option -XX:NewSize)</td>
</tr>
<tr>
<td>Reserved Space for HDFS</td>
<td>Reserved space in GB per volume</td>
</tr>
<tr>
<td>DataNode maximum Java heap size</td>
<td>Maximum Java heap size for DataNode (Java option -Xmx)</td>
</tr>
<tr>
<td>DataNode volumes failure toleration</td>
<td>The number of volumes that are allowed to fail before a DataNode stops offering services.</td>
</tr>
<tr>
<td>HDFS Maximum Checkpoint Delay</td>
<td>Maximum delay between two consecutive checkpoints for HDFS</td>
</tr>
<tr>
<td>HDFS Maximum Edit Log Size for Checkpointing</td>
<td>Maximum size of the edits log file that forces an urgent checkpoint even if the maximum checkpoint delay is not reached</td>
</tr>
</tbody>
</table>

Table 3: HDFS Settings

MapReduce
The fourth tab covers MapReduce settings:
<table>
<thead>
<tr>
<th>Name</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>MapReduce local directories</td>
<td>Directories for MapReduce to store intermediate data files. Based on the mount points and host settings you chose previously</td>
</tr>
<tr>
<td>JobTracker new generation size</td>
<td>Default size of Java new generation size for JobTracker (Java option -XX:NewSize)</td>
</tr>
<tr>
<td>JobTracker maximum new generation size</td>
<td>Maximum size of Java new generation for JobTracker (Java option -XX:MaxNewSize)</td>
</tr>
<tr>
<td>JobTracker maximum Java heap size</td>
<td>Maximum Java heap size for JobTracker in MB (Java option -Xmx)</td>
</tr>
<tr>
<td>Number of Map slots per node</td>
<td>Number of slots that Map tasks that run simultaneously can occupy on a TaskTracker</td>
</tr>
<tr>
<td>Number of Reduce slots per node</td>
<td>Number of slots that Reduce tasks that run simultaneously can occupy on a TaskTracker.</td>
</tr>
<tr>
<td>Cluster's Map slot size (virtual memory)</td>
<td>The virtual memory size of a single Map slot in the MapReduce framework. Use -1 for no limit</td>
</tr>
<tr>
<td>Cluster's Reduce slot size (virtual memory)</td>
<td>The virtual memory size of a single Reduce slot in the MapReduce framework. Use -1 for no limit.</td>
</tr>
<tr>
<td>Upper limit on virtual memory for single Map task</td>
<td>Upper limit on virtual memory for single Map task. Use -1 for no limit.</td>
</tr>
<tr>
<td>Upper limit on virtual memory for single Reduce task</td>
<td>Upper limit on virtual memory for single Reduce task. Use -1 for no limit.</td>
</tr>
<tr>
<td>Default virtual memory for a job's map-task</td>
<td>Virtual memory for single Map task. Use -1 for no limit.</td>
</tr>
<tr>
<td>Default virtual memory for a job's reduce-task</td>
<td>Virtual memory for single Reduce task. Use -1 for no limit.</td>
</tr>
<tr>
<td>Java options for MapReduce tasks</td>
<td>Java options for the TaskTracker child processes.</td>
</tr>
<tr>
<td><strong>IMPORTANT:</strong> For more information on sizing, see <a href="#">Recommended Memory Configurations for the MapReduce Service.</a></td>
<td></td>
</tr>
<tr>
<td>Map-side sort buffer memory</td>
<td>The total amount of Map-side buffer memory to use while sorting files (Expert-only configuration)</td>
</tr>
</tbody>
</table>

**Table 4:** MapReduce Settings
ZooKeeper

The fifth tab covers ZooKeeper settings:

<table>
<thead>
<tr>
<th>Name</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Limit on buffer</td>
<td>Percentage of sort buffer used for record collection (Expert-only configuration)</td>
</tr>
<tr>
<td>Job Log Retention (hours)</td>
<td>The maximum time, in hours, for which the user-logs are to be retained after the job completion.</td>
</tr>
<tr>
<td>Maximum number tasks for a Job</td>
<td>Maximum number of tasks for a single Job. Use -1 for no limit.</td>
</tr>
<tr>
<td>LZO compression</td>
<td>Check if you wish to enable LZO compression in addition to Snappy</td>
</tr>
</tbody>
</table>

Table 5: Zookeeper Settings

HBase

The sixth tab covers HBase settings:

Table 4: MapReduce Settings

<table>
<thead>
<tr>
<th>Name</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ZooKeeper directory</td>
<td>Data directory for ZooKeeper. Based on the mount points and host settings you chose previously.</td>
</tr>
<tr>
<td>Length of Single Tick</td>
<td>The length of a single tick in milliseconds, which is the basic time unit used by ZooKeeper</td>
</tr>
<tr>
<td>Ticks to allow for sync at Init</td>
<td>Amount of time in ticks to allow followers to connect and sync to a leader</td>
</tr>
<tr>
<td>Ticks to allow for sync at Runtime</td>
<td>Amount of time, in ticks to allow followers to connect an</td>
</tr>
<tr>
<td>Port for Running ZK Server</td>
<td>Port for running ZK server</td>
</tr>
<tr>
<td>Name</td>
<td>Notes</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>HBase Region Servers maximum Java heap size</td>
<td>Maximum Java heap size for HBase Region Servers (Java option -Xmx)</td>
</tr>
<tr>
<td><strong>IMPORTANT:</strong> If you are co-deploying HBase RegionServers and the MapReduce service on the same node, check Recommended Memory Configurations for the MapReduce Service for recommended sizing.</td>
<td></td>
</tr>
<tr>
<td>HBase Master Maximum Java heap size</td>
<td>Maximum Java heap size for HBase master (Java option -Xmx)</td>
</tr>
<tr>
<td>HBase HStore compaction threshold</td>
<td>When HStoreFiles in any one HStore are greater than this number, a compaction is run to rewrite all HStoreFiles files as one.</td>
</tr>
<tr>
<td>HFile block cache size</td>
<td>Percentage of maximum heap (-Xmx setting) to allocate to block cache used by HFile/StoreFile. You can set this to 0 to disable but this is not recommended.</td>
</tr>
<tr>
<td>Maximum HStoreFile Size</td>
<td>If any one of a column families' HStoreFiles has grown to exceed this value, the hosting HRegion is split in two.</td>
</tr>
<tr>
<td>HBase Region Server Handler</td>
<td>Count of RPC Listener instances spun up on RegionServers</td>
</tr>
<tr>
<td>HBase Region Major Compaction</td>
<td>The time between major compactions of all HStoreFiles in a region. Set to 0 to disable automated major compactions.</td>
</tr>
<tr>
<td>HBase Region Block Multiplier</td>
<td>Block updates if memstore reaches &quot;Multiplier * HBase Region Memstore Flush Size&quot; bytes. Useful preventing runaway memstore size during spikes in update traffic</td>
</tr>
<tr>
<td>HBase Region Memstore Flush Size</td>
<td>Memstore will be flushed to disk if size of the memstore exceeds this number of bytes.</td>
</tr>
<tr>
<td>HBase Client Scanner Caching</td>
<td>Number of rows that will be fetched when calling next on a scanner if it is not served from (local, client) memory. Do not set this value such that the time between invocations is greater than the scanner timeout</td>
</tr>
<tr>
<td>Zookeeper timeout for HBase Session</td>
<td>HBase passes this to the zk quorum as suggested maximum time for a session</td>
</tr>
</tbody>
</table>

**Table 6: HBase Settings**
The seventh tab covers Oozie settings:

<table>
<thead>
<tr>
<th>Name</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>HBase Client Maximum key-value</td>
<td>Specifies the combined maximum allowed size of a KeyValue instance. It should be set to a fraction of the maximum region size</td>
</tr>
<tr>
<td>Size</td>
<td></td>
</tr>
</tbody>
</table>

Table 6: HBase Settings

Oozie
The seventh tab covers Oozie settings:

<table>
<thead>
<tr>
<th>Name</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oozie DB directory</td>
<td>Data Directory in which the Oozie DB exists. Based on the mount points and host settings you chose previously.</td>
</tr>
</tbody>
</table>

Table 7: Oozie Settings

Miscellaneous
The last tab covers miscellaneous settings:

<table>
<thead>
<tr>
<th>Name</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>URL to Download 32/64 bit</td>
<td>Use this if you wish to have HMC set up the JDKs with the install. There is a default, HMC-provided URL, but you can also substitute a URL for your own source.</td>
</tr>
<tr>
<td>Java 32 bit Home</td>
<td>Use this to specify your Java home if you wish to have HDP use an existing JDK that is not set up by HMC. You must have installed this separately and exported it. This value must be the same for all hosts in the cluster. If you use this option, HMC does no further checks to make sure that your Java has been properly set up.</td>
</tr>
<tr>
<td>Java 64 bit Home</td>
<td>Use this to specify your Java home if you wish to have HDP use an existing JDK that is not set up by HMC. You must have installed this separately and exported it. This value must be the same for all hosts in the cluster. If you use this option, HMC does no further checks to make sure that your Java has been properly set up.</td>
</tr>
</tbody>
</table>

Table 8: Miscellaneous Settings

When you have made all your changes, click the **Finished customizing all components** button.
NOTE: Some available configuration parameters are not surfaced in the HMC GUI. These can be accessed by editing the property template files directly. This option is for advanced users only. The files can be found in `<master-install-machine-for-HDP>/etc/puppet/master/modules/hdp-hadoop/templates`.

To modify the HDFS service, use the following files:
- `hdfs-site.xml.erb`
- `core-site.xml.erb`

To modify the MapReduce service, use the following file:
- `mapred-site.xml.erb`

Recommended Memory Configurations for the MapReduce Service

- Make sure that there is enough memory for all the processes. Remember that system processes take around 10% of the memory available.
- For co-deploying HBase RegionServers and MapReduce service on the same node, reduce the RegionServer's heap size (use the HBase Region Servers maximum Java heap size property to modify the RegionServer heap size).
- For co-deploying HBase RegionServers and the MapReduce service on the same node, or for memory intensive MapReduce applications, modify the map and reduce slots as suggested in the following examples:

EXAMPLE: For co-deploying RegionServers and the MapReduce service on a machine with 16GB of available memory, the following would be a recommended configuration:

- 2 GB: system processes
- 8 GB: MapReduce slots - 6 Map + 2 Reduce slots per 1 GB task
- 4 GB: HBase RegionServers
- 1 GB: TaskTrackers
- 1 GB: DataNodes

To change the number of Map and Reduce slots based on the memory requirements of your application, use the following properties:

- Number of Map slots per node
- Number of Reduce slots per node

Step 7: Review and Deploy

The host assignments and service settings are displayed. Check to make sure everything is as you wish. If you need to make changes, click a previous tab and return to the appropriate screen.

When you are satisfied with your choices, click the Deploy button. A popup window appears that tracks the deployment status of each of the components, including a related test run per component. This process can take up to 40 minutes, depending on your configuration, so please be patient. If an error occurs, you will get an error message.

When the process completes, you can click click here to start managing your cluster to check on the status of your cluster. For more information, see Managing Your Cluster.

NOTE: If you are deploying the Nagios or Ganglia instance on the HMC installation host, you need to restart the HMC service. From a console prompt on that machine:

```
    service hmc restart
```
Once the HMC service is restarted, you can access the Cluster Information page at **http://<hostname for the main installation host>/hmc/html/index.php**
Managing Your Cluster

This section describes using the Cluster Management app to manage individual services and to add additional slave nodes to your cluster. The page also has a link to the separate web application which provides the Monitoring Dashboard.

- Access the Cluster Management App
- Cluster Summary
- Open the Monitoring Dashboard

Access the Cluster Management App

The Cluster Management app can be reached by pointing your browser to http://<hostname for the main installation host>/hmc/html/index.php. On this page there are four tabs where you can perform basic management functions.

- Cluster Summary
- Manage Services
- Add Nodes
- Uninstall

You can always return to this page by clicking the Cluster Management button at the top of the screen.

Cluster Summary

The Cluster Management app opens on the Cluster Summary tab. You can check the status of your cluster, and see the host locations of your services on this tab.

Manage Services

To check on your services, or to stop, start, or reconfigure your services, click the Manage Services tab. The Manage Services page opens. The services you have installed and their status are displayed.

Stop a Service/Start a Service

To stop a service, click the red rectangle button next to the name of the service you wish to stop. A confirmation pop-up appears. Confirm your choice.

To start a service, click the green arrow next to the name of the service you wish to start.

NOTE: If you stop a service, like HDFS, on which other services are dependent, those services are also automatically shut down.

Reconfigure a Service

To change the configuration of a service, click the button with the gear on the right. The relevant configuration parameters appear in a pop-up screen. Make your changes. When you are finished, click the Apply Changes button.

NOTE: Reconfiguring a service automatically stops and restarts the service and any services that depend on it. For example, reconfiguring HDFS stops and starts all services, because all services depend on HDFS.
Add Nodes

To increase the number of slave nodes in your cluster, click the Add nodes tab. The Add Nodes page appears. You must have set up the keys for SSH and the hostdetails.txt file as you did for the initial install. See Prepare the Environment for more information.

1. Use the Choose File button to find the private key for root.

   **NOTE:** You must have copied the related public key to every host you are adding.

2. Use the Choose File button to find the hosts text file.

3. Click the Add Nodes button.

4. A popup appears and displays the status of the installation host’s attempt to find, access, and bootstrap all the nodes in the cluster.

5. When all the steps on the popup have succeeded, click Proceed to Select Services

The Select Services page appears.

1. Check the services you wish to deploy on the added nodes.

2. Click the Deploy Nodes button.

A popup window appears that displays the deployment status of the services on the nodes.

Uninstall

If you wish to uninstall the cluster, or uninstall and wipe all your data, click the Uninstall tab on the Cluster Management page. The Uninstall page appears.

1. Check the SSH username. It should be root.

2. Put the path to the private key file in the text box or use the Browse button to find it.

3. If you wish to delete all of your data, check Perform wipe out (delete all data).

4. Click the Uninstall Cluster button.

5. When the popup window appears, click the Proceed with Uninstall button.

A popup window appears that displays the status of the uninstall.

[Optional] Turn On Password Protection for HMC.

Out of the box the Cluster Management app section of HMC is not password-protected. If you wish to require a username/password to access the app, you must edit a configuration file on the HMC host.

1. Open /etc/httpd/conf.d/hmc.conf on the HMC host with a text editor.

2. Make whatever edits are necessary so that your file looks like this:

   ```
   <Directory "/usr/share/hmc">
   #  SSLRequireSSL
   Options None
   AllowOverride None
   Order allow,deny
   Allow from all
   #  Order deny,allow
   #  Deny from all
   #  Allow from 127.0.0.1
   AuthName "HMC Access"
   AuthType Basic
   AuthUserFile /etc/hmc/htpasswd.users
   ```
Require valid-user
</Directory>

3. Save the file and run service hmc restart for the changes to take effect.
The username/password is now set to hmcadmin:hmcadmin.

Open the Monitoring Dashboard

You can access the Monitoring Dashboard from the Cluster Management page either by clicking Monitoring in the top menu bar or by entering http://<host name for the Nagios server>/hdp/dashboard/ui/home.html. Use the username/password you specified during configuration to log in. For more information on using the Monitoring Dashboard, see the Monitoring HDP section of the main Hortonworks documentation page, at Understanding Monitoring for Hadoop.
Configuring a Local Mirror

The standard install mode requires access to the Internet to fetch some HDP software components. In some cases this access is not possible or desirable. In this situation you must set up a version of the software repository that your machines can access locally. This section describes setting up such a repository.

- About the yum Package Management Tool
- Get the yum Client Configuration File
- Set Up the Mirror
- Modify the HDP yum Client Configuration File for All Cluster Nodes

About the yum Package Management Tool

The HMC installer for HDP on RHEL and CentOS uses the 'yum' package management tool. This tool fetches packages from configured repositories and installs those packages using automatic dependency resolution.

Get the yum Client Configuration File

First you must get the Hortonworks yum client configuration file.

1. Select a local server
   This server must be able to access both the Internet and the servers on which you wish to install HDP. This server will act as your local mirror server. It must run either CentOS v.5.0/6.0 or RHEL v.5.0/6.0. You need 446MB disk space to host the HDP repo.

2. Log into that server

3. Ensure that the mirror server has yum installed

4. Add the yum-utils and createrepo packages
   These packages are necessary for creating a local repository. From a shell window, type:
   ```bash
   yum install yum-utils createrepo
   ```

5. Download the appropriate Hortonworks yum client configuration file
   For RHEL/Centos 5.0
   From the yum.repos.d directory on the mirror server, type:
   ```bash
   ```
   For RHEL/Centos 6.0
   From the yum.repos.d directory on the mirror server, type:
   ```bash
   ```

Set Up the Mirror

Once you have the client configuration in place you can set up your local mirror:

1. Create an HTTP server
On the mirror server, install and activate an HTTP server. You can get the Apache HTTP Server [here](http://httpd.apache.org) or use whatever server you wish. Using the default configuration should work. Ensure that firewall settings (if any) allow inbound HTTP access for your mirror server. If you are using EC2, make sure that selinux is disabled.

2. Create a directory for the web server
   For example, from a shell window, type:
   ```bash
   mkdir -p /var/www/html/rpms
   ```
   **NOTE:** If you are using a symlink, make sure that `followsymlinks` is enabled on your web server.

3. Put the files in an appropriate directory
   Put all the HDP RPM files in the directory served by your web server. Using the previous example, from a shell window, type:
   ```bash
   cd /var/www/html/rpms
   reposync -r HDP-1.0.1.14
   reposync -r HDP-UTILS-1.0.1.14
   ```
   You should now see both an `HDP-1.0.1.14` directory and an `HDP-UTILS-1.0.1.14` directory, with two sub-directories ‘`distro`’ and ‘`extras`’.

4. Generate the appropriate metadata
   This step defines each directory as a yum repository. From a shell window, type:
   ```bash
   createrepo /var/www/html/rpms/HDP-1.0.1.14
   createrepo /var/www/html/rpms/HDP-UTILS-1.0.1.14
   ```
   You should see a new folder called ‘`repodata`’ inside both HDP directories.

5. Verify your installation
   You should now be able to access the directory through your web browser. To test this out, browse to the following location:
   ```text
   http://<yourwebserver>/rpms/HDP-1.0.1.14/rpms
   ```

6. [Conditional] If you have multiple repositories configured in your environment, install and configure the yum priorities plugin.
   ```bash
   yum install yum-priorities
   ```
   Open the `priorities.conf` file in a text editor (the file is in `/etc/yum/plugin-conf.d`) and make sure that the file contains the following lines:
   ```conf
   [main]
   enabled=1
   ```
   Save the file and exit.

You have now completed the creation and configuration of the local yum HDP mirror repository.

**Modify the HDP yum Client Configuration File for All Cluster Nodes**

1. On the mirror server, copy the HDP yum client configuration file you downloaded in [Get the yum Client Configuration File](#) to a temporary directory. From a shell window, type:
   ```bash
   cp /etc/yum.repos.d/hdp.repo ~/hdp.repo
   ```

2. Edit the `hdp.repo` file, changing the value of the `baseurl` property to the local mirror URL (for example, `http://<yourwebserver>/rpms/HDP-1.0.1.14/rpms`).
3. Use `scp` or `pdsh` to copy the edited client yum configuration file to the `/etc/yum.repos.d/` directory on every node in the cluster.

4. If you are installing to a group of hosts that do not have Internet access, you must also have a copy of the EPEL (Extra Packages for Enterprise Linux) repository available somewhere that those hosts can access. The appropriate repository package for v5.x can be fetched from [http://download.fedoraproject.org/pub/epel/5/i386/epel-release-5-4.noarch.rpm](http://download.fedoraproject.org/pub/epel/5/i386/epel-release-5-4.noarch.rpm) and for v6.x from [http://mirrors.servercentral.net/fedora/epel/6/i386/epel-release-6-7.noarch.rpm](http://mirrors.servercentral.net/fedora/epel/6/i386/epel-release-6-7.noarch.rpm)
Troubleshooting HMC Deployments

The following information can help you troubleshoot issues you may run into with your HMC-based installation.

- Getting the Logs
- Quick Checks
- Specific Issues

IMPORTANT: Use the following information to troubleshoot a failed installation. Fix any issues, uninstall, and prepare to run the installer again. Uninstall instructions can be found at Uninstall the Cluster. Do not attempt to recover files manually.

Getting the Logs

The first thing to do if you run into trouble is to find the logs. The installer logs are on the install host at /var/log/hmc/hmc.log. Logs for the various services can be found in /var/log/<service-name> on their respective hosts.

Quick Checks

- Make sure the directories to which HMC needs to write information are writable.
- Make sure all the appropriate services are running. If you have access to the HMC Cluster Management web application, use the Manage Services tab to check the status of each component. If you do not have access to Manage Services, you must start and stop the services manually. For information on how to do this, see Manage Services Manually.
- If the first HDFS put command fails to replicate the block, the clocks in the nodes may not be synchronized. Make sure that Network Time Protocol (NTP) is enabled for your cluster.
- If HBase does not start, check if its slaves are running on 64-bit JVMs. The ZooKeeper service must run on a 64-bit host machine.
- Make sure the hosts specified in the hostdetails.txt are listed as FQDN, not IP addresses.
- Make sure umask is set to 0022.
- Make sure the HCatalog host can access the MySQL server. From a shell try:
  
  ```
  mysqlld -h $<FQDN_for_MySQL_server> -u $<FQDN_for_HCatalog_Server> -p
  ```
  
  You will need to provide the password you set up for Hive/HCatalog during the installation process.
- Make sure MySQL is running. By default, MySQL server does not start automatically on reboot.
  - To set auto-start on boot, from a shell, type:
    ```
    chkconfig --level 35 mysql on
    ```
  - To then start the service manually from a shell, type:
    ```
    service mysqld start
    ```
Specific Issues

The following are common issues you might encounter.

Problem: “Unable to create new native thread” listed in exceptions in the HDFS datanode logs or those of any system daemon

If your nproc limit is incorrectly configured, the smoke tests fail and you see an error similar to this in the DataNode logs:

```
INFO org.apache.hadoop.hdfs.DFSClient: Exception
   incrementBlockOutputStream java.io.EOFException
INFO org.apache.hadoop.hdfs.DFSClient: Abandoning block
   blk_-6935524980745310745_139190
```

Solution:

In certain recent Linux distributions (like RHEL v6.x/CentOS v6.x), the default value of `nproc` is lower than the value required if you are deploying the HBase service. To change this value:

1. Using a text editor, open `/etc/security/limits.d/90-nproc.conf` and change the `nproc` limit to approximately 32000. For more information, see `ulimit` and `nproc` recommendations for HBase servers.
2. Restart the HBase server.

Problem: The “yum install hmc” Command Fails

You are unable to get the initial install command to run.

Solution:

You may have incompatible versions of some software components in your environment. Check the list in Check Existing Installs and make any necessary changes. Also make sure the operating system is RHEL v.5.x/6.x or CentOS v.5.x/6.x

Problem: Data Node Smoke Test Fails

If your DataNodes are incorrectly configured, the smoke tests fail and you get this error message in the DataNode logs:

```
DisallowedException
org.apache.hadoop.hdfs.server.protocol.
DisallowedException
```

Solution:

1. Make sure that reverse DNS look-up is properly configured for all nodes in your cluster.
2. Make sure you have the correct FQDNs in your `hostdetails.txt` file. Do not use IP addresses - they are not supported.
3. Restart the installation process.

Problem: The HCatalog Daemon Metastore Smoke Test Fails

If the HCatalog smoke test fails, this is displayed in your console:

```
Metastore startup failed, see /var/log/hcatalog/hcat.err
```

Solution:
1. Log into the HCatalog node in your cluster
2. Open /var/log/hcatalog/hcat.err or /var/log/hive/hive.log (one of the two will exist depending on the installation) with a text editor
3. In the file, see if there is a **MySQL Unknown Host Exception** like this:
   ```java
   at java.lang.reflect.Method.invoke (Method.java:597)
   at org.apache.hadoop.util.Runjar.main (runjar.java:156)
   Caused by: java.net.UnknownHostException:mysql.host.com
   at java.net.InetAddress.getAllByName(InetAddress.java:1157)
   ```
   This exception can be thrown if you are using a previously existing MySQL instance and you have incorrectly identified the hostname during the installation process. When you do the reinstall, make sure this name is correct.
4. In the file, see if there is an **ERROR Failed initializing database** entry like this:
   ```
   11/12/29 20:52:04 ERROR DataNucleus.Plugin: Bundle org.eclipse.jdt.core required
   ```
   This exception can be thrown if you are using a previously existing MySQL instance and you have incorrectly identified the username/password during the installation process. It can also occur when the user you specify does not have adequate privileges on the database. When you do the reinstall, make sure this username/password is correct and that the user has adequate privilege.
5. Restart the installation process.

**Problem: Hadoop Streaming Jobs Don’t Work with Templeton**

A required .jar file, hadoop-streaming.jar, very occasionally fails to load properly during the installation process.

**Solution:**

Check to see if the .jar file is present on HDFS. From a shell, type:
```bash
su - templeton
hadoop dfs -ls /user/templeton/hadoop-streaming.jar
```
If this command fails, add the file. From a shell, type:
```bash
su - templeton
/usr/bin/hadoop --config ${hadoopconfdir} fs -copyFromLocal
/usr/share/hadoop/contrib/streaming/hadoop-streaming*.jar
/user/templeton/hadoop-streaming.jar
```

**Problem: Inconsistent State is Shown on Monitoring Dashboard**

The Cluster Summary section of the Dashboard indicates that a service (HDFS/MapReduce/HBase) is down while the Services table above it shows the service as running.

**Solution:**

This can be the transient result of a heavy load on the service master node or of a slow network. The http call to the service master can time out. Unless the service master is really not accessible, refreshing the browser page should eliminate this inconsistency. In general, there is about a one minute latency before the Services table reflects a service down state on the Dashboard.