

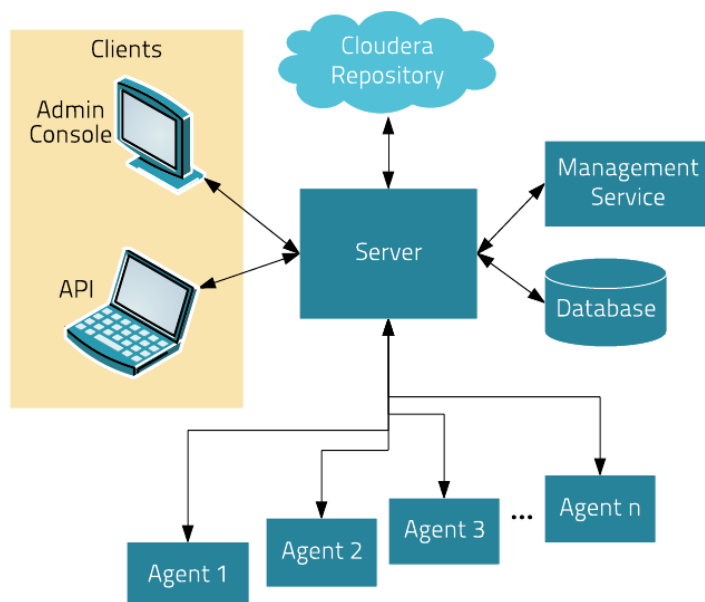
# CM 管理软件使用

## 目录

CM 管理软件使用 .....	1
一 CM 架构原理 .....	1
二 基础环境准备 .....	2
三 CM 软件软件安装 .....	2
3.1 系统要求 .....	2
3.2 操作 .....	3
四 CDH 软件安装 .....	5
五 演示使用 .....	8

## 一 CM 架构原理

**Cloudera Manager** 的核心是管理服务器，该服务器承载管理控制台的 Web 服务器和应用程序逻辑，并负责安装软件，配置，启动和停止服务，以及管理上的服务运行群集。



**Cloudera Manager Server** 由以下几个部分组成：

**Agent：** 安装在每台主机上。该代理负责启动和停止的过程，拆包配置，触发装置和监控主机。

**Management Service：** 由一组执行各种监控，警报和报告功能角色的服务。

**Database：** 存储配置和监视信息。通常情况下，多个逻辑数据库在一个或多个数据库服务器上运行。例如，Cloudera 的

管理服务器和监控角色使用不同的逻辑数据库。

Cloudera Repository: 软件由 Cloudera 管理分布存储库。

Clients: 是用于与服务器进行交互的接口:

Admin Console : 基于 Web 的用户界面与管理员管理集群和 Cloudera 管理。

API : 与开发人员创建自定义的 Cloudera Manager 应用程序的 API。

## 二 基础环境准备

192.168.91.10	hadoop1.aura.com
192.168.91.11	hadoop2.aura.com
192.168.91.12	hadoop3.aura.com

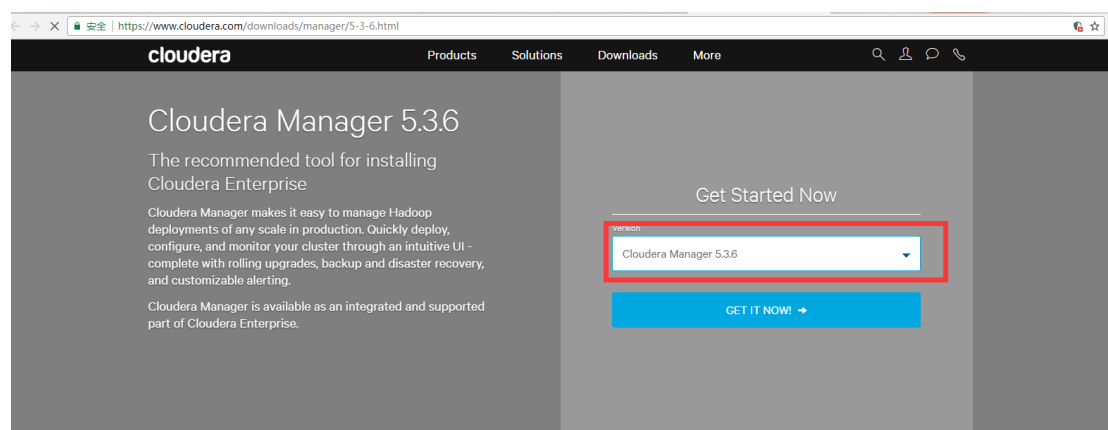
CM 服务分配

	hadoop1.aura.com	Hadoop2.aura.com	Hadoop3.aura.com
Agent	✓	✓	✓
Service	✓		
Database	✓		

## 三 CM 软件软件安装

### 3.1 系统要求

注: 一定要根据官网找到相对应 CM 的版本的系统要求。



比如我需要安装的是 5.3.6(这个版本目前来说有点低)  
往下看:

System Requirements

What's New

Documentation

Supported Operating Systems

Supported JDK Versions

Supported Browsers

Supported Databases

Supported CDH and Managed Service Versions

Resource Requirements

Networking and Security Requirements

### Supported Operating Systems

Cloudera Manager supports the following operating systems:

- RHEL-compatible**
  - Red Hat Enterprise Linux and CentOS
    - 5.7, 64-bit
    - 6.4, 64-bit
    - 6.4 in SE Linux mode
    - 6.5, 64-bit
  - Oracle Enterprise Linux with default kernel and Unbreakable Enterprise Kernel, 64-bit
    - 5.6 (UEK R2)
    - 6.4 (UEK R2)
    - 6.5 (UEK R2, UEK R3)
- SLES - SUSE Linux Enterprise Server 11, 64-bit.** Service Pack 2 or later is required for CDH 5, and Service Pack 1 or later is required for CDH 4. To use the embedded PostgreSQL database that is installed when you follow [Installation Path A - Automated Installation by Cloudera Manager](#), the Updates repository must be active. The [SUSE Linux Enterprise Software Development Kit 11 SP1](#) is required on hosts running the

里面都是 CM 对系统的要求

包括：

支持的操作系统

包括 jdk 的版本

包括浏览器

包括数据库

包括 CDH

包括资源要求

包括网络和安全要求

查看后，经总结一共有如下要求：

操作系统：

CentOS

JDK :JDK 1.7.0\_67

DataBase:MySQL - 5.5 and 5.6

server 节点 内存至少 8G

agent 节点 内存 2G

## 3.2 操作

3.2.1 修改 IP 地址，主机名，IP 映射关系，并配置好免密码操作

192.168.91.10	hadoop1.aura.com
192.168.91.11	hadoop2.aura.com
192.168.91.12	hadoop3.aura.com

并在 Windows 的/etc/hosts 上也添加上如上表格内容

3.2.2 关闭防火墙

# service iptables stop

# chkconfig iptables off

在所有主机上执行如下两条命令

### 3.2.3 禁用 IPV 6

```
echo "                " >> /etc/modprobe.d/dist.conf
echo "alias net-pf-10 off" >> /etc/modprobe.d/dist.conf
echo "alias ipv6 off" >> /etc/modprobe.d/dist.conf
```

在所有主机上执行如上命令，并重启生效。

### 3.2.4 关闭 selinux

```
vim /etc/sysconfig/selinux
```

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
#     disabled - No SELinux policy is loaded.
SELINUX=disabled
# SELINUXTYPE= can take one of these two values:
#     targeted - Targeted processes are protected,
#     mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

在所有主机上执行如上操作

### 3.2.5 安装 jdk

要根据官方的要求去安装相对应的 JDK。

先到官网去查相对应的版本。

[http://archive.cloudera.com/cm5/redhat/5/x86\\_64/cm/5.3.6/RPMS/x86\\_64/](http://archive.cloudera.com/cm5/redhat/5/x86_64/cm/5.3.6/RPMS/x86_64/)

然后查看当前操作系统的 JDK 版本，如果不符合，则删除 jdk。

```
rpm -qa | grep java  查找
rpm -e --nodeps xxx 如果有则执行该命令，删除 jdk
根据我们这次的版本，执行如下命令：
rpm -ivh jdk-7u67-linux-x64.rpm
```

```
并配置环境变量 /etc/profile
export JAVA_HOME=/usr/java/jdk1.7.0_67
export PATH=.:$PATH:$JAVA_HOME/bin
```

### 3.2.6 ntp 系统时间同步

所有主机之间时间同步

必须要使用 ntpd 服务。进行节点之间的时间同步

### 3.2.7 设置用户最大能打开文件数目、进程数和内存（所有节点）

```
/etc/security/limits.conf
```

通过拷贝修改

*	soft	nofile	32728
*	hard	nofile	1029345

*	soft	nproc	65536
*	hard	nproc	unlimited
*	soft	memlock	unlimited
*	hard	memlock	unlimited

修改后 reboot 重启系统

### 3.2.8 安装 MySQL 数据

```
yum list | grep mysql
```

添加用户组: groupadd mysql

添加用户名: useradd -r -g mysql mysql

在 MySQL 里执行如下命令:

```
CREATE USER 'temp'@'localhost' IDENTIFIED BY 'temp';
```

```
grant all privileges on *.* to 'temp'@'%' identified by 'temp' with grant option;
```

```
grant all privileges on *.* to 'temp'@'hadoop1.aura.com' identified by 'temp' with grant option;
```

```
flush privileges;
```

## 四 CDH 软件安装

### 4.1 先创建安装目录(所有节点)

创建目录 /opt/cloudera-manager

```
mkdir -p /opt/cloudera-manager
```

### 4.2 到官网下载相对应的安装包:

[http://archive.cloudera.com/cm5/cm/5/cloudera-manager-el6-cm5.3.6\\_x86\\_64.tar.gz](http://archive.cloudera.com/cm5/cm/5/cloudera-manager-el6-cm5.3.6_x86_64.tar.gz)

把 cloudera-manager-el6-cm5.3.6\_x86\_64.tar.gz 解压到/opt/cloudera-manager (所有节点)

修改所有主机/opt/cloudera-manager/cm-5.3.6/etc/cloudera-scm-agent 下的 config.ini, 只需要修改这一行

**server\_host=hadoop1.aura.com**

### 4.3 创建 CM 用户

在 CM 的 server 节点上, 找到/opt/cloudera-manager/cm-5.3.6/share/cmf/schema 目录下的脚本

```
scm_prepare_database.sh mysql -h hadoop1.aura.com -utemp -ptemp --scm-host hadoop1.aura.com scm scm scm
```

第一个: hadoop1.aura.com: MySQL 的安装位置

第二个: hadoop1.aura.com: CM server 服务的位置

```
[root@hadoop1 schema]# scm_prepare_database.sh mysql -h hadoop1.aura.com -utemp -ptemp --scm-host hadoop1.aura.com scm scm scm
```

```
JAVA_HOME=/usr/java/jdk1.7.0_67
```

```
Verifying that we can write to /opt/cloudera-manager/cm-5.3.6/etc/cloudera-scm-server
log4j:ERROR Could not find value for key log4j.appender.A
log4j:ERROR Could not instantiate appender named "A".
[2017-08-20      00:04:26,111]ERROR                                0[main]      -
com.cloudera.enterprise.dbutil.DbProvisioner.executeSql(DbProvisioner.java:272) - Unable to find
the MySQL JDBC driver. Please make sure that you have installed it as per instruction in the
installation guide.
[2017-08-20      00:04:26,115]ERROR                                4[main]      -
com.cloudera.enterprise.dbutil.DbProvisioner.main(DbProvisioner.java:121) - Stack Trace:
java.lang.ClassNotFoundException: com.mysql.jdbc.Driver
    at java.net.URLClassLoader$1.run(URLClassLoader.java:366)
    at java.net.URLClassLoader$1.run(URLClassLoader.java:355)
    at java.security.AccessController.doPrivileged(Native Method)
    at java.net.URLClassLoader.findClass(URLClassLoader.java:354)
    at java.lang.ClassLoader.loadClass(ClassLoader.java:425)
    at sun.misc.Launcher$AppClassLoader.loadClass(Launcher.java:308)
    at java.lang.ClassLoader.loadClass(ClassLoader.java:358)
    at java.lang.Class.forName0(Native Method)
    at java.lang.Class.forName(Class.java:190)
    at com.cloudera.enterprise.dbutil.DbProvisioner.executeSql(DbProvisioner.java:270)
    at com.cloudera.enterprise.dbutil.DbProvisioner.doMain(DbProvisioner.java:95)
    at com.cloudera.enterprise.dbutil.DbProvisioner.main(DbProvisioner.java:110)
--> Error 1, giving up (use --force if you wish to ignore the error)
```

需要把 MySQL 的驱动包发送到 /usr/share/java 目录下 (server)  
注意要把 jar 包名称重命名为 mysql-connector-java.jar

#### 4.4 配置 CHD 源

找到相对应的源:

<http://archive.cloudera.com/cdh5/parcels/5.3.6/>

← → ↻ [archive.cloudera.com/cdh5/parcel/5.3.6/](http://archive.cloudera.com/cdh5/parcel/5.3.6/)

## Index of /cdh5/parcels/5.3.6

	<a href="#">Name</a>	<a href="#">Last modified</a>	<a href="#">Size</a>	<a href="#">Description</a>
🔗	<a href="#">Parent Directory</a>		-	
🔍	<a href="#">CDH-5.3.6-1.cdh5.3.6.p0.11-el5.parcel</a>	2015-07-31 21:47	1.4G	
🔍	<a href="#">CDH-5.3.6-1.cdh5.3.6.p0.11-el5.parcel.sha1</a>	2015-07-31 21:47	41	
🔍	<a href="#">CDH-5.3.6-1.cdh5.3.6.p0.11-el6.parcel</a>	2015-07-31 21:47	1.4G	
🔍	<a href="#">CDH-5.3.6-1.cdh5.3.6.p0.11-el6.parcel.sha1</a>	2015-07-31 21:47	41	
🔍	<a href="#">CDH-5.3.6-1.cdh5.3.6.p0.11-precise.parcel</a>	2015-07-31 21:46	1.5G	
🔍	<a href="#">CDH-5.3.6-1.cdh5.3.6.p0.11-precise.parcel.sha1</a>	2015-07-31 21:46	41	
🔍	<a href="#">CDH-5.3.6-1.cdh5.3.6.p0.11-sles11.parcel</a>	2015-07-31 21:47	1.4G	
🔍	<a href="#">CDH-5.3.6-1.cdh5.3.6.p0.11-sles11.parcel.sha1</a>	2015-07-31 21:47	41	
🔍	<a href="#">CDH-5.3.6-1.cdh5.3.6.p0.11-trusty.parcel</a>	2015-07-31 21:47	1.5G	
🔍	<a href="#">CDH-5.3.6-1.cdh5.3.6.p0.11-trusty.parcel.sha1</a>	2015-07-31 21:47	41	
🔍	<a href="#">CDH-5.3.6-1.cdh5.3.6.p0.11-wheezy.parcel</a>	2015-07-31 21:47	1.5G	
🔍	<a href="#">CDH-5.3.6-1.cdh5.3.6.p0.11-wheezy.parcel.sha1</a>	2015-07-31 21:47	41	
🔍	<a href="#">manifest.json</a>	2015-07-31 21:47	42K	

cdh 源：比如 hadoop、zookeeper 等组件的安装文件

把这两个放置到服务器上：

cdh 源放置目录：

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

```
mkdir -p /opt/cloudera/parcels/
```

**Server 服务器：** `/opt/cloudera/parcel-repo` 把上面的两个源统一放到这个目录下

**Agent 服务器：** `/opt/cloudera/parcels/`

注意：

**a:** 我们一开始只需要把源放到 server 端的 `/opt/cloudera/parcel-repo` 目录下。

然后安装的时候 server 会把相对应的源文件放到 agent 的 `/opt/cloudera/parcels/` 目录下

**b:** 注意修改 `CDH-5.3.6-1.cdh5.3.6.p0.11-el6.parcel.sha1` 的文件名，把末尾 1 去掉

#### 4.5 CDH 内部用户

执行以下命令创建 cloudera-scm 用户 [所有节点]上都要创建

```
useradd --system --home=/opt/cloudera-manager/cm-5.3.6/run/cloudera-scm-server --no-create-home --shell=/bin/false --comment "Cloudera SCM User" cloudera-scm
```

修改

server: `/opt/cloudera/parcel-repo`

agent: `/opt/cloudera/parcels/`

的所属者和所属组为 cloudera-scm 用户

所有 agent 节点执行: `chown -R cloudera-scm:cloudera-scm /opt/cloudera/`

## 4.6 启动 CM

**a.** 首先启动 server (在 server 服务器上)

注意: MySQL 一定要保证启动

到/opt/cloudera-manager/cm-5.3.6/etc/init.d 下执行 **./cloudera-scm-server start**

查看启动结果: 需要通过查看 server 的启动日志文件

/opt/cloudera-manager/cm-5.3.6/log/cloudera-scm-server

tail -f cloudera-scm-server.log

WebServerImpl:org.mortbay.log: jetty-6.1.26.cloudera.4

2016-06-19 15:34:12,184 INFO WebServerImpl:org.mortbay.log: Started

SelectChannelConnector@0.0.0.0:7180

2016-06-19 15:34:12,184 INFO

WebServerImpl:com.cloudera.server.cmf.WebServerImpl: Started Jetty server.

看到如下内容, 那么说明启动成功。

**b.** 启动 agent


到/opt/cloudera-manager/cm-5.3.6/etc/init.d 执行: **./cloudera-scm-agent start**

## 五 演示使用

http://hadoop1.aura.com:7180/cmf/login

登录页面

用户 密码 admin admin



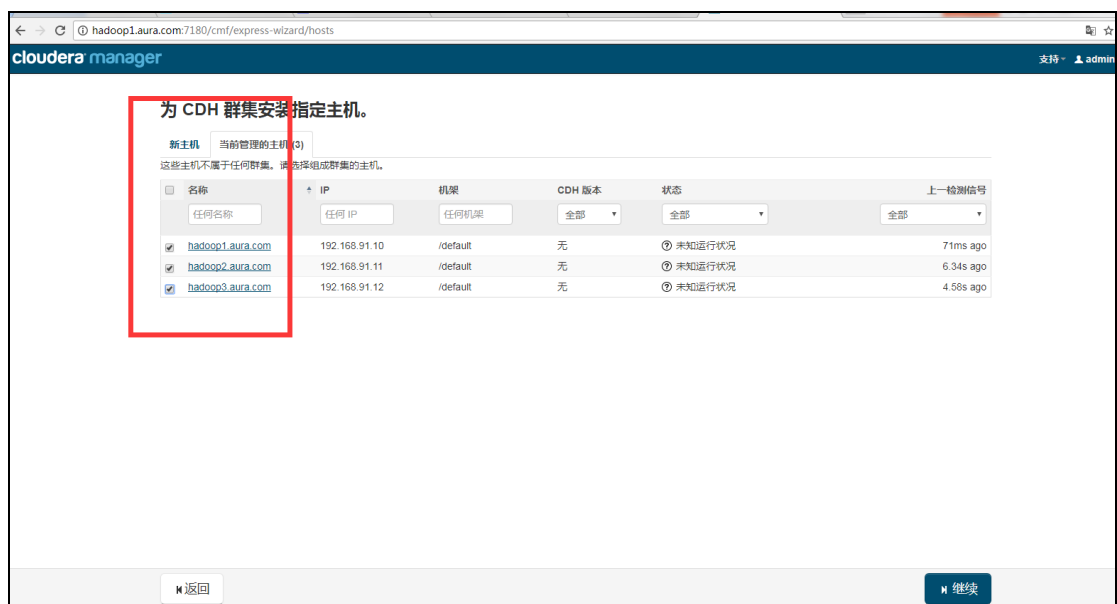
The image shows the Cloudera Manager web interface. At the top, it says '欢迎使用 Cloudera Manager. 您想要部署哪一个版本?' (Welcome to Cloudera Manager. Which version do you want to deploy?). Below this, there's a table comparing Cloudera Express and Cloudera Enterprise. Cloudera Express is marked as '免费' (Free). Cloudera Enterprise has a '60 天' (60 days) trial period. A '下载许可证' (Download License) button is visible. At the bottom right, there is a '继续' (Continue) button.

	Cloudera Express	Cloudera Enterprise 数据集成器试用版	Cloudera Enterprise
许可证	免费	60 天 在试用期之后, 该产品将继续作为 Cloudera Express 运行。您的群集和数据将会保持不受影响。	年度订阅 <a href="#">下载许可证</a>
节点限制	无限制	无限制	无限制
CDH	✓	✓	✓
Cloudera Manager 核心功能	✓	✓	✓
Cloudera Manager 高级功能		✓	✓
Cloudera Navigator		✓	✓
Cloudera 支持			✓

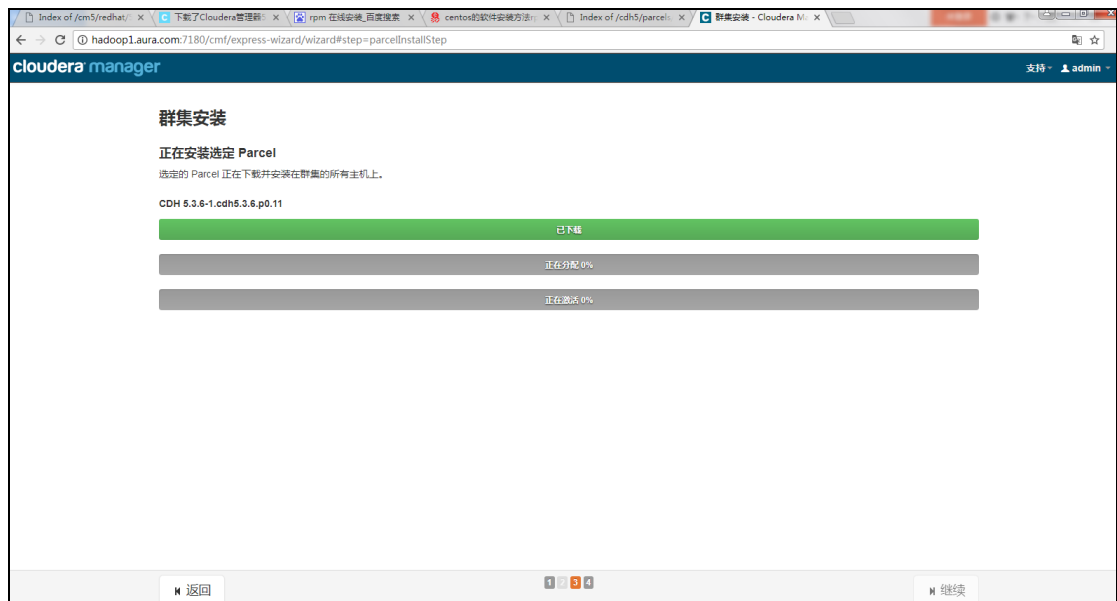
如需获得 Cloudera Express 和 Cloudera Enterprise 中提供的功能的完整列表, [单击此处](#)。

选择免费版, 一路 【继续】





选择 CDH 版本时，选择我们之前已经上传上去的版本



花费时间较长



根据提示修改 `echo 0 > /proc/sys/vm/swappiness`

选择自定义，需要什么就安装什么

HDFS、YARN (包括 MapReduce 2)、ZooKeeper、Oozie、Hive、Hue、Sqoop、HBase、Impala、Solr、Spark 和键/值 Store Indexer

● 自定义服务

选择您自己的服务。将自动包含所选服务需要的服务。只有在设置了初始群集之后才能添加 Flume。

服务类型	说明
<input type="checkbox"/> HBase	Apache HBase 提供对大型数据库的随机、实时的读/写访问权限（需要 HDFS 和 ZooKeeper）。
<input checked="" type="checkbox"/> HDFS	Apache Hadoop 分布式文件系统 (HDFS) 是 Hadoop 应用程序使用的主要存储系统。HDFS 创建多个数据块副本并将它们分布在整个集群的计算主机上，以启用可靠且极其快速的计算功能。
<input checked="" type="checkbox"/> Hive	Hive 是一种数据仓库系统，提供名为 HiveQL 的 SQL 类语言。
<input checked="" type="checkbox"/> Hue	Hue 是与包括 Apache Hadoop 的 Cloudera Distribution 一起配合使用的图形用户界面（需要 HDFS、MapReduce 和 Hive）。
<input checked="" type="checkbox"/> Impala	Impala 为存储在 HDFS 和 HBase 中的数据提供了一个实时 SQL 查询接口。Impala 需要 Hive 服务，并与 Hue 共享 Hive Metastore。
<input type="checkbox"/> Isilon	EMC Isilon is a distributed filesystem.
<input type="checkbox"/> Key-Value Store Indexer	键/值 Store Indexer 侦听 HBase 中所含表内的数据变化，并使用 Solr 为其创建索引。
<input type="checkbox"/> MapReduce	Apache Hadoop MapReduce 支持对整个集群中的大型数据库进行分布式计算（需要 HDFS）。 <b>建议改用 YARN（包括 MapReduce 2）。包括 MapReduce 用于向后兼容性。</b>
<input type="checkbox"/> Oozie	Oozie 是群集中管理数据处理作业的工作流协调服务。
<input type="checkbox"/> Solr	Solr 是一个分布式服务，用于编制存储在 HDFS 中的数据的索引并搜索这些数据。
<input checked="" type="checkbox"/> Spark	Apache Spark is an open source cluster computing system. This service runs Spark as an application on YARN.
<input type="checkbox"/> Sqoop 2	Sqoop 是一个设计用于在 Apache Hadoop 和结构化数据存储（如关系数据库）之间高效地传输大批量数据的工具。Cloudera Manager 支持的版本为 <b>Sqoop 2</b> 。
<input checked="" type="checkbox"/> YARN (MR2 Included)	Apache Hadoop MapReduce 2.0 (MRv2) 或 YARN 是支持 MapReduce 应用程序的数据计算框架（需要 HDFS）。
<input checked="" type="checkbox"/> ZooKeeper	Apache ZooKeeper 是用于维护和同步配置数据的集中服务。

本向导还将安装 Cloudera Management Service。有一系列组件可启用监控、报告、事件和警报；这些组件需要数据库存储信息，这将在下一页面上配置。

**返回**

1 2 3 4 5 6

▶ 继续

➤ [hadoop1.aura.com:7180/cm/cluster/1/express-add-services/index#step=commandDetailsStep](http://hadoop1.aura.com:7180/cm/cluster/1/express-add-services/index#step=commandDetailsStep)

## 命令进度

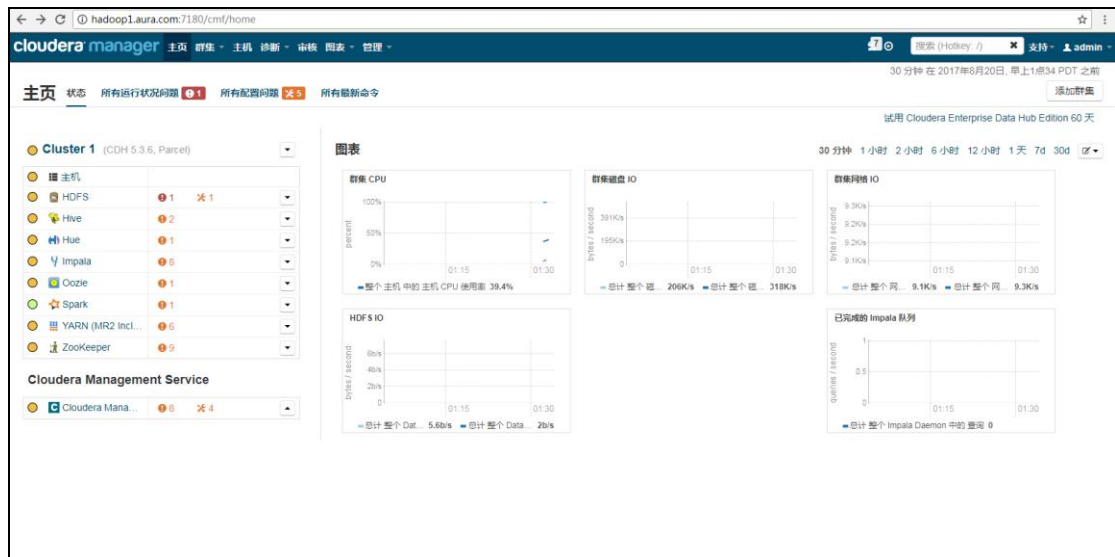
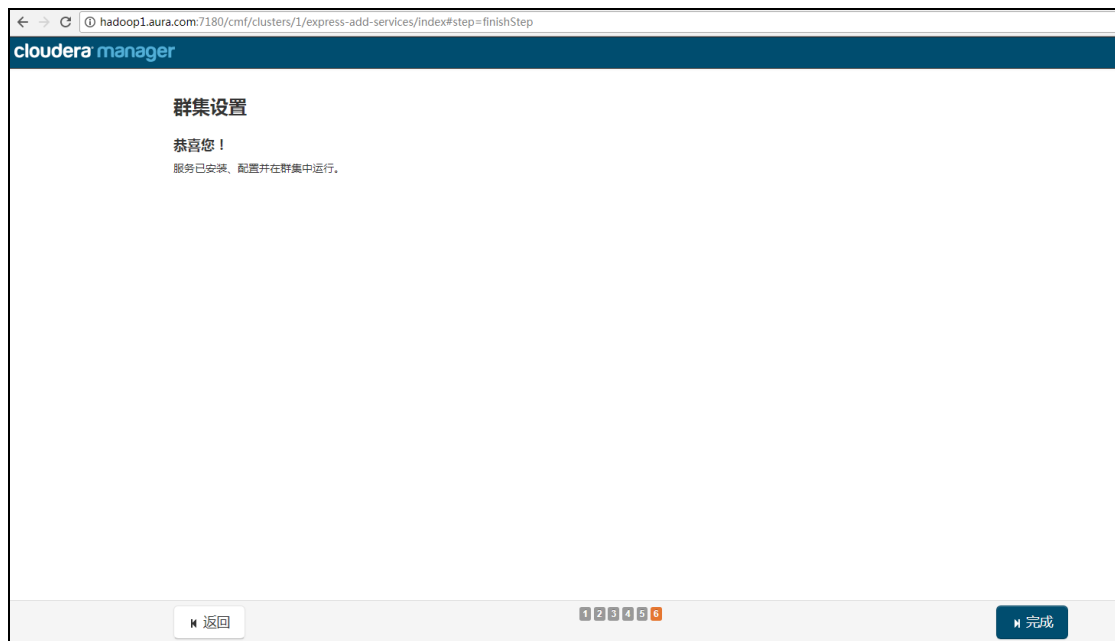
已完成 24 个步骤 (共 24 个)。

- 正在等待 ZooKeeper 服务初始化  
已成功完成 3 个步骤。
- 正在启动 ZooKeeper 服务  
Completed 3 steps successfully.  
[详细信息](#)
- 正在检查 NameNode 的名称目录是否为空。仅在为空时格式化 HDFS。  
Successfully formatted NameNode.  
[详细信息](#)
- 正在启动 HDFS 服务  
Successfully started HDFS service  
[详细信息](#)
- 正在创建 HDFS /tmp 目录  
Successfully created HDFS directory /tmp.  
[详细信息](#)
- 正在创建 MR2 作业历史记录目录  
Successfully created HDFS directory.  
[详细信息](#)
- 正在创建 NodeManager 远程应用程序日志目录  
Successfully created HDFS directory.  
[详细信息](#)
- 正在启动 YARN (MR2 Included) 服务  
Successfully started service.  
[详细信息](#)
- 正在创建 Hive Metastore 数据库表  
Created Hive Metastore Database Tables successfully.  
[详细信息](#)
- 正在创建 Hive 元数据目录

[⏮ 返回](#)

1 2 3 4 5 6

▶ 继续



HDFS 实现 HA



### 启用 HDFS 的 High Availability

#### 入门

此向导引导您添加备用 NameNode, 重启 HDFS 服务和任何依赖服务, 然后重新部署客户端配置。

Nameservice 名称:

启用 High Availability 将创建新的 nameservice。请接受默认名称 nameservice1 或在 Nameservice 名称中提供另一名称。

[返回](#) [继续](#)

### 启用 HDFS 的 High Availability

#### 分配角色

NameNode 主机:

JournalNode 主机:

我们建议在主机上运行 JournalNode 时使用与 NameNode 相同的硬件规格。通常, NameNode 主机和 JobTracker 都是不错的选择。您必须要有至少三个以上的奇数 JournalNode。

[返回](#) [继续](#)

hadoop1.aura.com:7180/cm/services/18/enableHA/index?roleId=68#step=reviewStep

cloudera manager

支持admin

### 启用 HDFS 的 High Availability

#### 审核更改

为新角色设置下列配置值。必填值以 \* 标出。

参数	组	值	说明
服务 HDFS			
NameNode 数据目录* dfs.namenode.name.dir	hadoop1	/dfs/nm	确定 NameNode 在本地文件系统中存储名称表 (tsimage) 的位置。对于冗余的副本，需输入目录的逗号分隔列表以复制所有目录中的名称表。典型值为 /data/Ndfs/nn，其中 N = 1..3。
	hadoop2	/dfs/nm	继承自：NameNode Default Group
JournalNode 编辑目录* dfs.journalnode.edits.dir	hadoop1	/dfs/journalnode	写入 NameNode 编辑的本地文件系统上的目录。
	hadoop2	/dfs/journalnode	重置为空默认值
	hadoop3	/dfs/journalnode	重置为空默认值

#### 附加选项

☒ 强制初始化用于自动故障转移的 Zookeeper ZNode。将覆盖以前用于此 nameservice 的所有 ZNode。

☒ 清除备用 NameNode 的名称目录中的任何现有数据。

12345

返回继续

hadoop1.aura.com:7180/cm/services/18/enableHA/index?roleId=68#step=commandDetailsStep

cloudera manager

支持admin

对当前 NameNode 的名称目录进行格式化。如果名称目录不为空，此操作将失败。

命令 (149) 已失败

详细信息

正在初始化 NameNode 的共享编辑目录。

Successfully initialized shared edits directory of the NameNode

详细信息

正在启动将转换为活动模式 namenode (hadoop1) 的 NameNode。

Supervisor returned RUNNING

详细信息

正在等待启动活动的 NameNode。

NameNode started responding to RPCs successfully.

详细信息

正在通过初始化其名称目录来引导备用 NameNode。

Successfully bootstrapped Standby NameNode namenode (hadoop2).

详细信息

正在启动备用 NameNode

Supervisor returned RUNNING

详细信息

启动活动 NameNode 主机上的 Failover Controller。

Supervisor returned RUNNING

详细信息

启动备用 NameNode 主机上的 Failover Controller。

Supervisor returned RUNNING

详细信息

正在等待备用 NameNode 启动。

NameNode started responding to RPCs successfully.

详细信息

正在创建 HDFS/tmp 目录 (如尚未创建)。

HDFS directory /tmp already exists.

详细信息

启动 hdfs 及其依赖服务

12345

返回重试继续

hadoop1.aura.com:7180/cm/services/18/enableHA/index?roleId=68#step=finishStep

cloudera manager

支持admin

### 启用 HDFS 的 High Availability

#### 恭喜您！

已成功启用 High Availability。

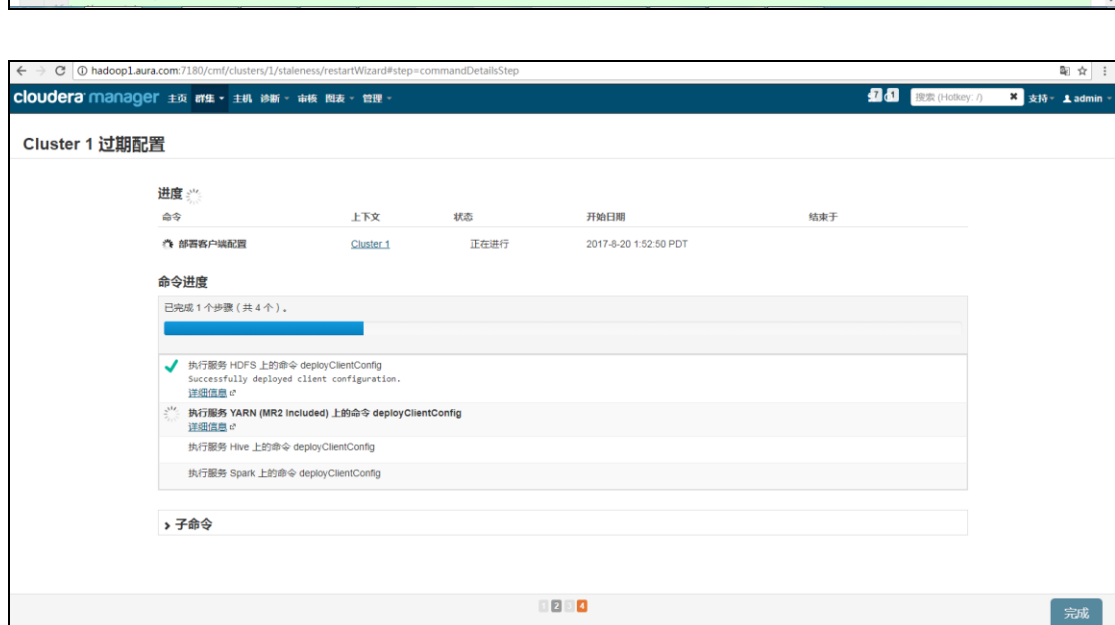
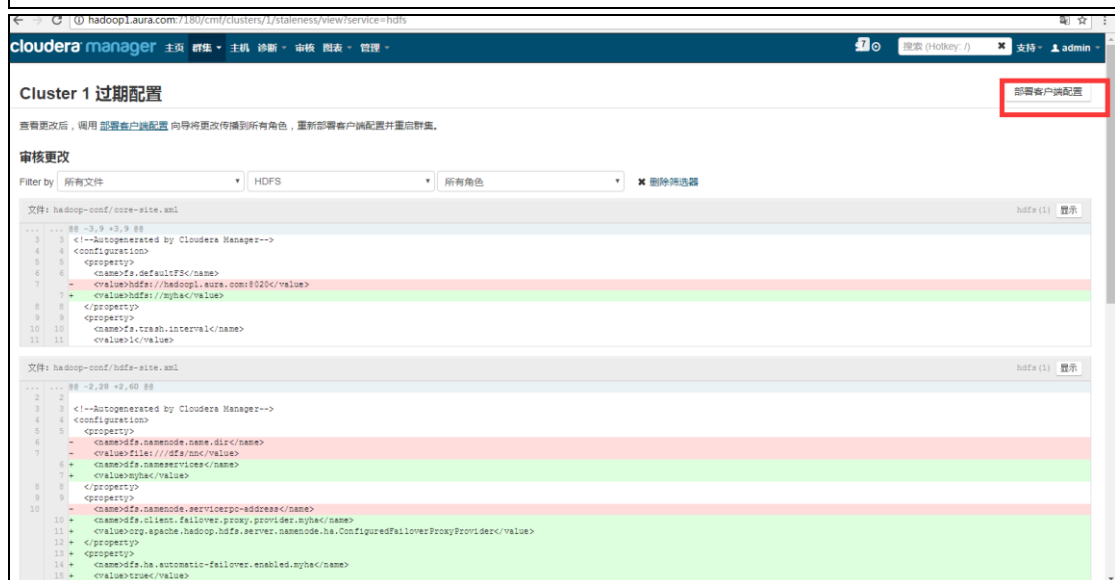
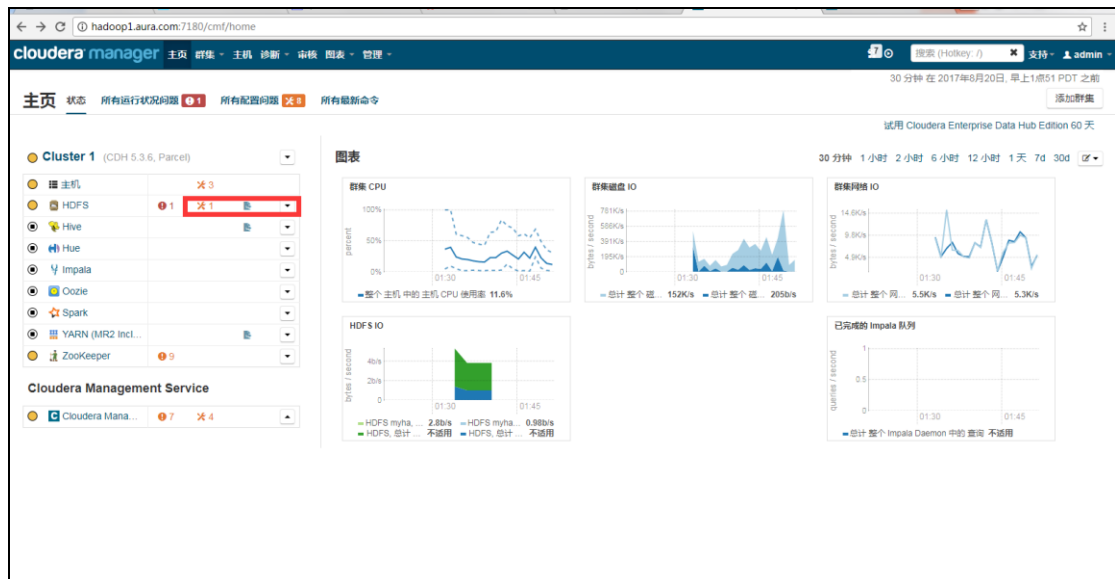
完成本向导后必须手动执行下列步骤：

- 对于每个 Hive 服务 Hive，停止 Hive 服务，将 Hive Metastore 数据库备份到永久性存储中，运行服务命令“更新 Hive Metastore NameNodes”，然后重启 Hive 服务。

12345

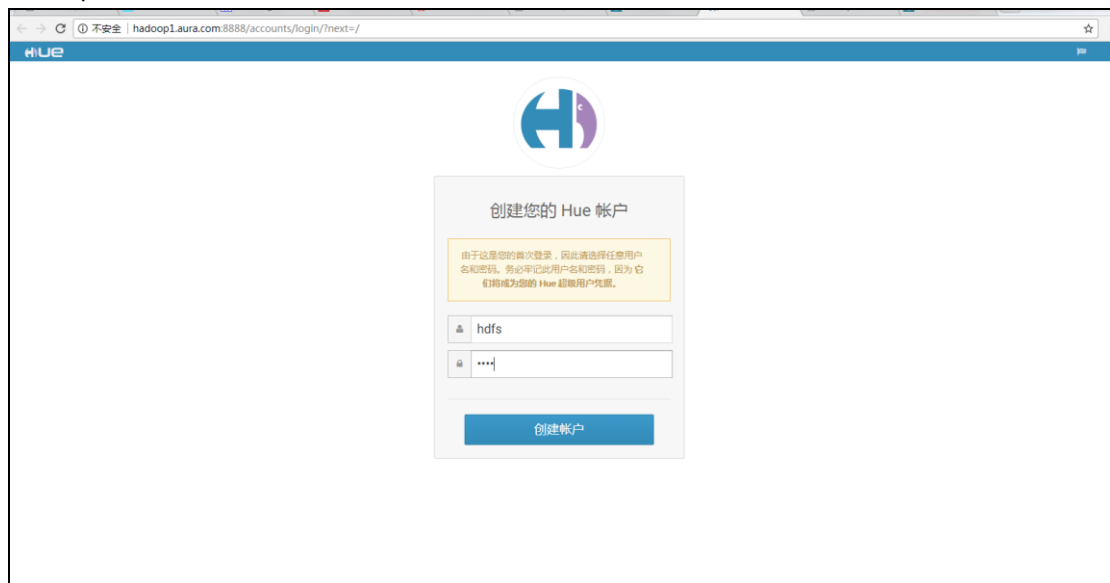
返回完成

## 同步 HA 的配置文件



## Hue 的使用

hadoop1.aura.com:8888



Hue:impla

invalidate metadata 与 Hive 同步元数据

Hue:hive

重启:

1: 先启动 hadoop1.aura.com 的 MySQL

service mysqld start

2: 启动相对应的 server 服务和 agent 服务

/opt/cloudera-manager/cm-5.3.6/etc/init.d/cloudera-scm-server start

/opt/cloudera-manager/cm-5.3.6/etc/init.d/cloudera-scm-agent start

3:

http://hadoop1.aura.com:7180/cmf/login