Hadoop 2.0安装部署方法

讲师：董西成
博客：dongxicheng.org
微信号：hadoop-123
（二维码见右）

Open　Passion　Value

chinahadoop 中国最大的Hadoop社区

目录

Hadoop 2.0安装方式介绍

- 自动安装部署
  - ✓ **Ambari：**http://ambari.apache.org/
  - ✓ **Minos：**https://github.com/XiaoMi/minos
  - ✓ **Cloudera Manager（收费）**
- 使用**RPM**包安装部署
  - ✓ **Apache Hadoop不提供**
  - ✓ **HDP和CDH提供**
- 使用**JAR**包安装部署
  - ✓ 各版本均提供

Hadoop 2.0安装部署流程

- **步骤1：准备硬件（linux操作系统）**
- **步骤2：准备软件安装包，并安装基础软件（主要是JDK）**
- **步骤3：将Hadoop安装包分发到各个节点的同一个目录下，并解压**
- **步骤4：修改配置文件**
- **步骤5：启动服务**
- **步骤6：验证是否启动成功**

目录

软硬件准备

- ## 硬件准备
  - ✓ 测试（单机）环境：一台Linux（CentOS、Ubuntu等均可以）机器（可以是虚拟机）
  - ✓ 生产（多机）环境：多台Linux机器
- ## 软件准备
  - ✓ JDK1.6或者JDK1.7安装包
  - ✓ Hadoop 2.0安装包

硬件配置要求

- 测试环境
  - ✓ 一台PC机或者服务器
  - ✓ 建议内存不小于4G（2G也行）
- 生产环境
  - ✓ 小于100个节点

建议每台机器配置不小于：

- dual quad-core 2.6 Ghz CPU,
- 24 GB of DDR3 RAM
- dual 1 Gb Ethernet NICs
- a SAS drive controller

✓ 大于100个节点

需提高master节点（ResourceManager/NameNode）硬件配置，建议参考书籍《Hadoop Operations》第四章。

Linux测试环境准备

- 可选用Redhat、Ubuntu、CentOS等
  - 新手可以考虑Ubuntu Linux，下载链接："ubuntu-12.04.1-desktop-i386.iso" http://vdisk.weibo.com/s/jUTBV
- 如果只有装有Windows的PC机
  - 使用Virtual Box或者VMWare搭建linux虚拟机
  - 装双操作系统
- 建议使用非root用户安装hadoop
  - root用户权限过大，误操作容易造成很大损失
  - Hadoop某些功能不允许在root用户下操作
- SSH免密码登陆
  - 目的：启动Hadoop集群方便（可以不设置SSH免密码登陆）
  - 只需设置Master节点到各个Slave节点的免密码登陆

JDK1.6安装

- 每个节点均需要安装
- 如果使用CDH5，需使用JDK 7
- 安装流程：
  - ✓ 步骤1：下载JDK 1.6（注意区分32位和64位）
  - ✓ 步骤2：安装JDK 1.6（以32位为例）

    chmod +x jdk-6u45-linux-i586.bin

    ./jdk-6u45-linux-i586.bin
  - ✓ 步骤3：验证是否安装成功

    建议将JDK安装位置加到PATH环境变量中

目录

1. Hadoop 2.0安装部署流程
2. Hadoop 2.0软硬件准备
3. Hadoop 2.0安装包下载
4. Hadoop 2.0测试环境（单机）搭建方法
5. Hadoop 2.0生产环境（多机）搭建方法
6. 总结

Hadoop 2.0发行版（仅考虑开源免费版本）

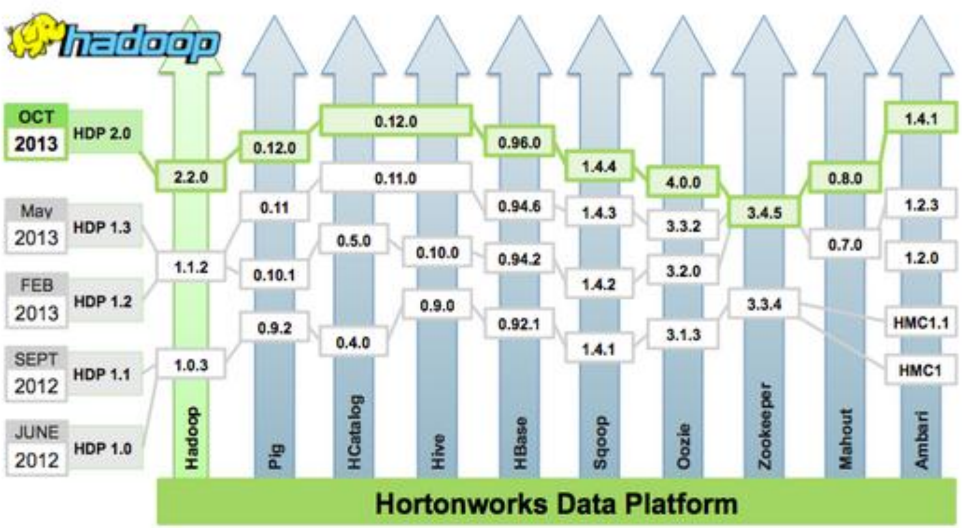
- **Apache Hadoop**
  - 最原始版本，所有其他发行版均基于该发行版实现的
  - 0.23.x：非稳定版
  - 2.x：稳定版
- **HDP**
  - Hortonworks公司的发行版
- **CDH**
  - Cloudera公司的的Hadoop发行版
  - 包含CDH4和CDH5两个版本
  - CDH4：基于Apache Hadoop 0.23.0版本开发
  - CDH5：基于Apache Hadoop 2.2.0版本开发
- 不同发行版兼容性
  - 架构、部署和使用方法一致，不同之处仅在若干内部实现。

Hadoop 2.0发行版选择

- 推荐使用商用公司的开源发行版（生产环境）
  - ✓ HDP或CDH
  - ✓ 好处：

1）Hadoop生态系统中所有软件打包在一起，用户不会面临版本选择问题

2）完善的文档，个别之处独特的优化



Apache Hadoop安装包和源代码下载（1）

hadoop.apache.org

Top Wiki

- About
  - Welcome
  - Releases
  - Mailing Lists
  - Issue Tracking
  - Who We Are?
  - Who Uses Hadoop?
  - Buy Stuff
  - Sponsor Apache
  - Sponsors of Apache
  - Privacy Policy
  - Bylaws
- Documentation
- Related Projects

built with Apache Forrest

Welcome to Apache™ Hadoop®!

- What Is Apache Hadoop?
- Getting Started
- Download Hadoop
- Who Uses Hadoop?
- News
  - 15 October, 2013: release 2.2.0 available
  - 25 August, 2013: release 2.1.0-beta available
  - 27 December, 2011: release 1.0.0 available
  - March 2011 - Apache Hadoop takes top prize at Media Guardian Innovation Awards
  - January 2011 - ZooKeeper Graduates
  - September 2010 - Hive and Pig Graduate
  - May 2010 - Avro and HBase Graduate
  - July 2009 - New Hadoop Subprojects
  - March 2009 - ApacheCon EU
  - November 2008 - ApacheCon US
  - July 2008 - Hadoop Wins Terabyte Sort Benchmark

What Is Apache Hadoop?

The Apache™ Hadoop® project develops open-source software for reliable, scalable, distributed computing.

The Apache Hadoop software library is a framework that allows for the distributed processing of large data sets across clusters of computers using simple programming mo computation and storage. Rather than rely on hardware to deliver high-availability, the library itself is designed to detect and handle failures at the application layer, so del failures.

The project includes these modules:

- **Hadoop Common**: The common utilities that support the other Hadoop modules.
- **Hadoop Distributed File System (HDFS™)**: A distributed file system that provides high-throughput access to application data.
- **Hadoop YARN**: A framework for job scheduling and cluster resource management.
- **Hadoop MapReduce**: A YARN-based system for parallel processing of large data sets.

Other Hadoop-related projects at Apache include:

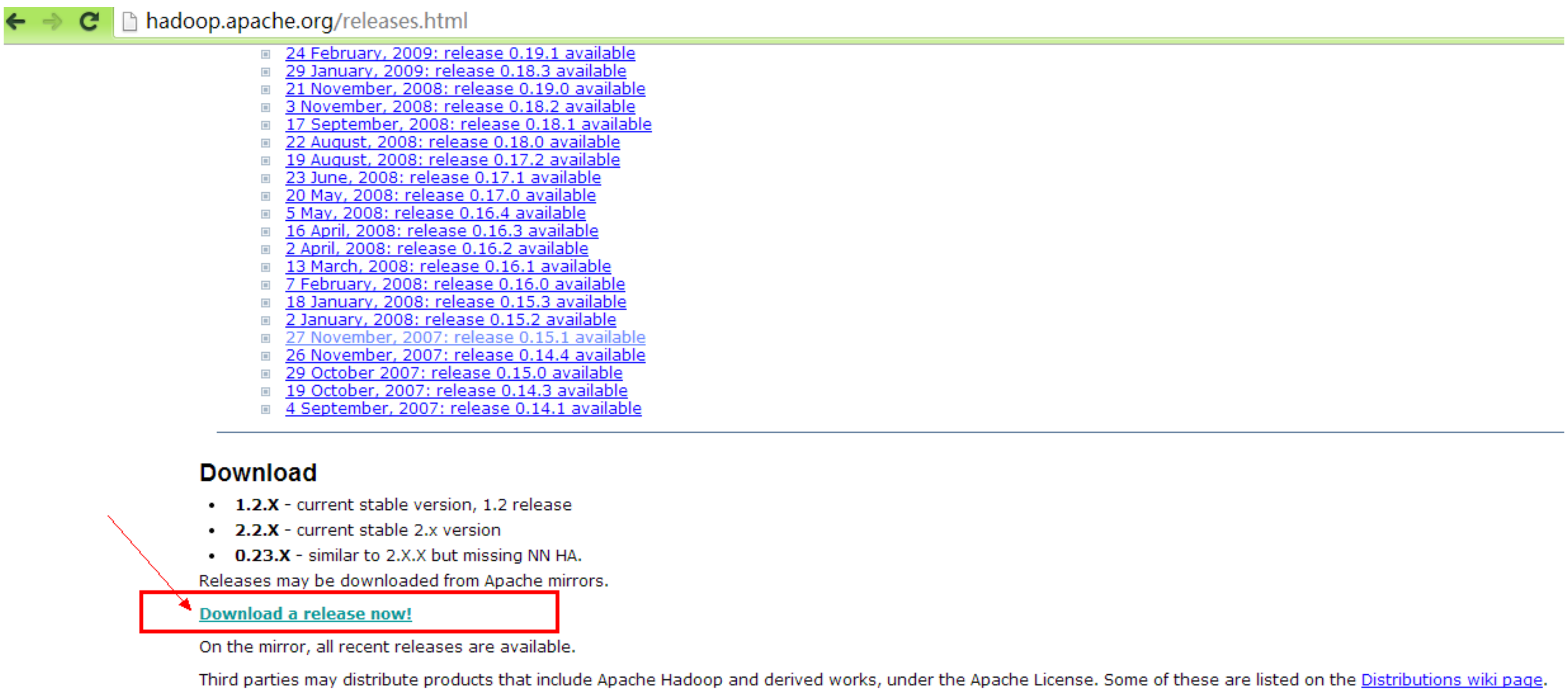
- **Ambari™**: A web-based tool for provisioning, managing, and monitoring Apache Hadoop clusters which includes support for Hadoop HDFS, Hadoop MapReduce, Hive, H cluster health such as heatmaps and ability to view MapReduce, Pig and Hive applications visually alongwith features to diagnose their performance characteristics in a
- **Avro™**: A data serialization system.
- **Cassandra™**: A scalable multi-master database with no single points of failure.
- **Chukwa™**: A data collection system for managing large distributed systems.
- **HBase™**: A scalable, distributed database that supports structured data storage for large tables.
- **Hive™**: A data warehouse infrastructure that provides data summarization and ad hoc querying.
- **Mahout™**: A Scalable machine learning and data mining library.
- **Pig™**: A high-level data-flow language and execution framework for parallel computation.
- **ZooKeeper™**: A high-performance coordination service for distributed applications.

Getting Started

To get started, begin here:

1. Learn about Hadoop by reading the documentation.
2. Download Hadoop from the release page.
3. Discuss Hadoop on the mailing list.

Apache Hadoop安装包和源代码下载（2）



hadoop.apache.org/releases.html

- 24 February, 2009: release 0.19.1 available
- 29 January, 2009: release 0.18.3 available
- 21 November, 2008: release 0.19.0 available
- 3 November, 2008: release 0.18.2 available
- 17 September, 2008: release 0.18.1 available
- 22 August, 2008: release 0.18.0 available
- 19 August, 2008: release 0.17.2 available
- 23 June, 2008: release 0.17.1 available
- 20 May, 2008: release 0.17.0 available
- 5 May, 2008: release 0.16.4 available
- 16 April, 2008: release 0.16.3 available
- 2 April, 2008: release 0.16.2 available
- 13 March, 2008: release 0.16.1 available
- 7 February, 2008: release 0.16.0 available
- 18 January, 2008: release 0.15.3 available
- 2 January, 2008: release 0.15.2 available
- 27 November, 2007: release 0.15.1 available
- 26 November, 2007: release 0.14.4 available
- 29 October 2007: release 0.15.0 available
- 19 October, 2007: release 0.14.3 available
- 4 September, 2007: release 0.14.1 available

Download

- **1.2.X** - current stable version, 1.2 release
- **2.2.X** - current stable 2.x version
- **0.23.X** - similar to 2.X.X but missing NN HA.

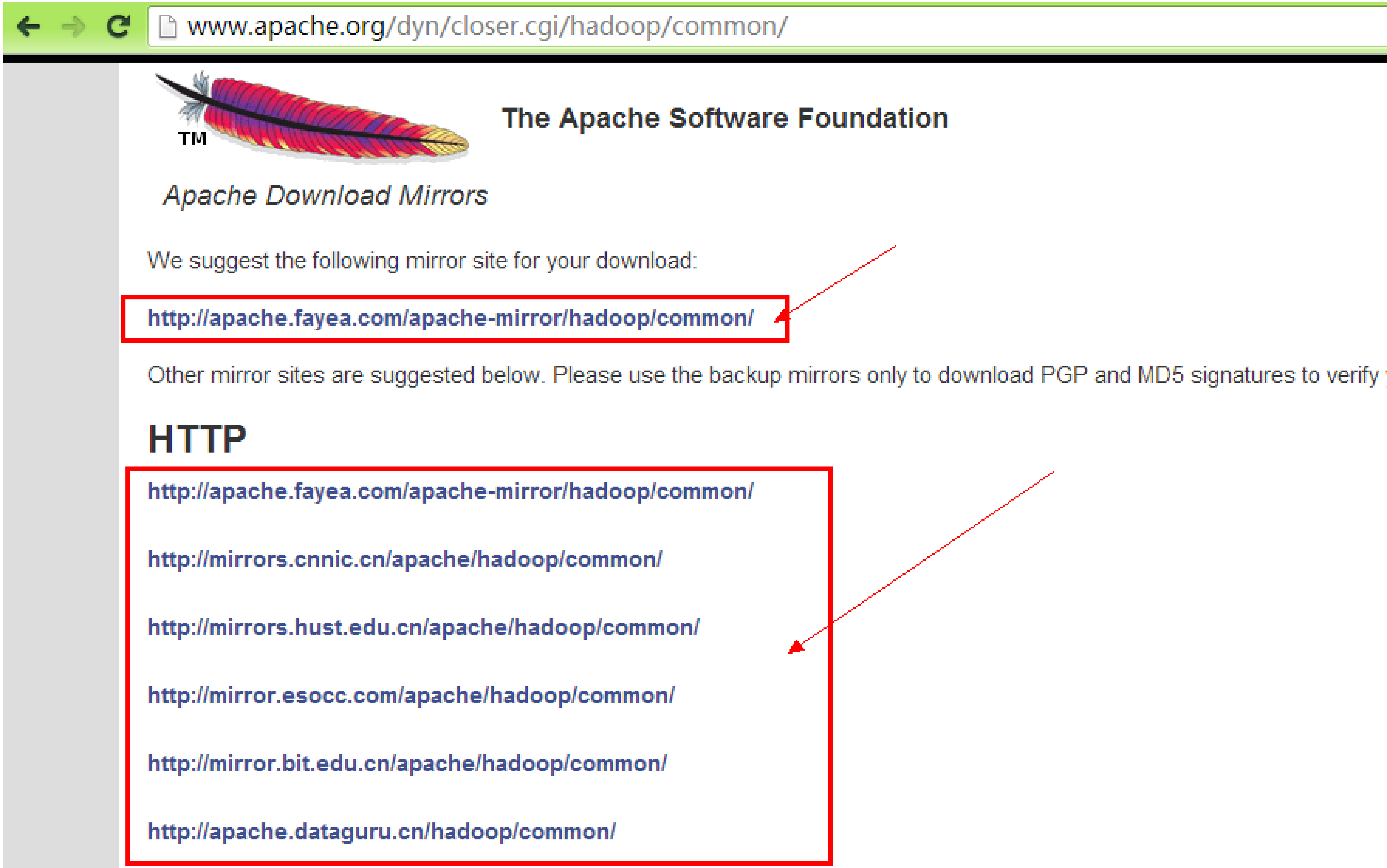
Releases may be downloaded from Apache mirrors.

Download a release now!

On the mirror, all recent releases are available.

Third parties may distribute products that include Apache Hadoop and derived works, under the Apache License. Some of these are listed on the Distributions wiki page.

Apache Hadoop安装包和源代码下载（3）



Apache Hadoop安装包和源代码下载（4）

apache.fayea.com/apache-mirror/hadoop/common/

Hadoop Releases

Please make sure you're downloading from a nearby mirror site, not from www.apache.org.

We suggest downloading the current stable release.

Older releases are available from the archives.

Name	Last modified	Size	Description
Parent Directory		-	
hadoop-0.23.10/	03-Dec-2013 14:07	-	
hadoop-0.23.9/	02-Jul-2013 01:16	-	
hadoop-1.2.1/	23-Jul-2013 06:49	-	
hadoop-2.0.3-alpha/	07-Feb-2013 11:53	-	
hadoop-2.0.4-alpha/	13-Apr-2013 05:53	-	
hadoop-2.0.5-alpha/	05-Jun-2013 05:48	-	
hadoop-2.0.6-alpha/	16-Aug-2013 13:26	-	
hadoop-2.1.0-beta/	16-Aug-2013 05:10	-	
hadoop-2.1.1-beta/	17-Sep-2013 14:36	-	
hadoop-2.2.0/	07-Oct-2013 14:50	-	
stable/	07-Oct-2013 14:50	-	
stable1/	23-Jul-2013 06:49	-	
stable2/	07-Oct-2013 14:50	-	
readme.txt	13-Dec-2012 13:19	187	

apache.fayea.com/apache-mirror/hadoop/common/stable/

Index of /apache-mirror/hadoop/common/sta

Name	Last modified	Size	Description
Parent Directory			
hadoop-2.2.0-src.tar.gz	07-Oct-2013 14:46	19M	
hadoop-2.2.0-src.tar.gz.mds	07-Oct-2013 14:46	1.1K	
hadoop-2.2.0.tar.gz	07-Oct-2013 14:46	104M	
hadoop-2.2.0.tar.gz.mds	07-Oct-2013 14:47	958	

Hadoop 2.0源代码

Hadoop 2.0安装包

CDH4和CDH5安装包和源代码下载

- CDH4安装包下载地址：http://archive.cloudera.com/cdh4/cdh/4
- CDH5安装包下载地址：http://archive.cloudera.com/cdh5/cdh/5/

hadoop-2.0.0-cdh4.3.2.tar.gz	16-Sep-2013 22:38	234M	
hadoop-2.0.0-cdh4.4.0-changes.log	04-Sep-2013 18:18	200K	
hadoop-2.0.0-cdh4.4.0-package-changes.log	04-Sep-2013 18:18	657	
hadoop-2.0.0-cdh4.4.0.CHANGES.txt	04-Sep-2013 18:18	608K	
hadoop-2.0.0-cdh4.4.0.package.CHANGES.txt	04-Sep-2013 18:18	1.2K	
hadoop-2.0.0-cdh4.4.0.releasenotes.html	04-Sep-2013 18:18	133K	CDH 4 Release Notes
hadoop-2.0.0-cdh4.4.0.tar.gz	04-Sep-2013 18:17	235M	
hadoop-2.0.0-cdh4.5.0-changes.log	24-Nov-2013 23:02	206K	
hadoop-2.0.0-cdh4.5.0-package-changes.log	24-Nov-2013 23:02	1.0K	
hadoop-2.0.0-cdh4.5.0.CHANGES.txt	24-Nov-2013 23:02	626K	
hadoop-2.0.0-cdh4.5.0.package.CHANGES.txt	24-Nov-2013 23:02	2.0K	
hadoop-2.0.0-cdh4.5.0.releasenotes.html	24-Nov-2013 23:02	136K	CDH 4 Release Notes
hadoop-2.0.0-cdh4.5.0.tar.gz	24-Nov-2013 23:00	195M	
hadoop-latest.tar.gz	24-Nov-2013 23:00	195M	
hbase-0.92.0+8-changes.log	05-Jun-2012 05:58	841	
hbase-0.92.0+8-package-changes.log	05-Jun-2012 05:58	1.0K	
hbase-0.92.0+8.CHANGES.txt	05-Jun-2012 05:58	1.9K	

HDP2安装包和源代码下载（1）

hortonworks.com/products/hdp-2/?b=1#install

PRODUCTS » HORTONWORKS DATA PLATFORM (HDP)

Hortonworks Data Platform (HDP

VERSION 2.0

Overview | Download & Install | Documentation | Add-ons

Download & Install

Hortonworks provides multiple installation options for the Hortonworks Data Platform. Based on your cluster requirements and environment, choose an option below and follow the associated instructions.

Automated (Ambari)

The fastest way to get started. Apache Ambari simplifies the provisioning, management and monitoring of your cluster.

RHEL/CentOS/SLES (64-bit)

Ambari Docs »

Manual (RPMs)

Roll up your sleeves and use this option to install and configure your cluster manually with the RPM packages.

Ubuntu/RHEL/CentOS/SLES (64-bit)

RPM Docs »

Tarball Links

Use the project tarballs (which follow the Apache structure) for the most custom and flexible installation.

Ubuntu/RHEL/CentOS/SLES (64-bit)

Tarballs »

HDP2安装包和源代码下载（2）

docs.hortonworks.com/HDPDocuments/HDP2/HDP-2.0.8.0/bk_installing_manually_book/content/rpm-chap13.html

Chapter 16. Manual Install Appendix: Tarballs

Chapter 16. Manual Install Appendix: Tarballs

Individual links to the Apache structured tarball files for the projects included with Hortonworks Data Platform are li

- **RHEL 5 and CentOS 5:**

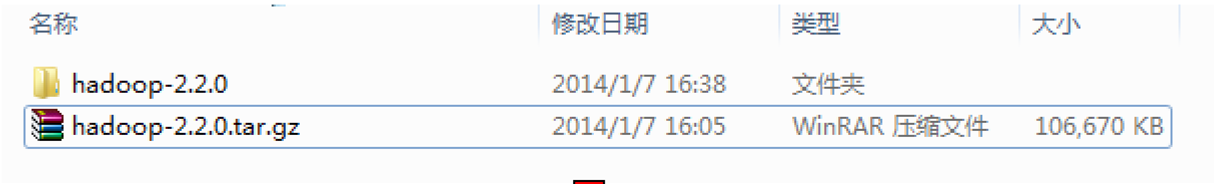
源代码与安装包

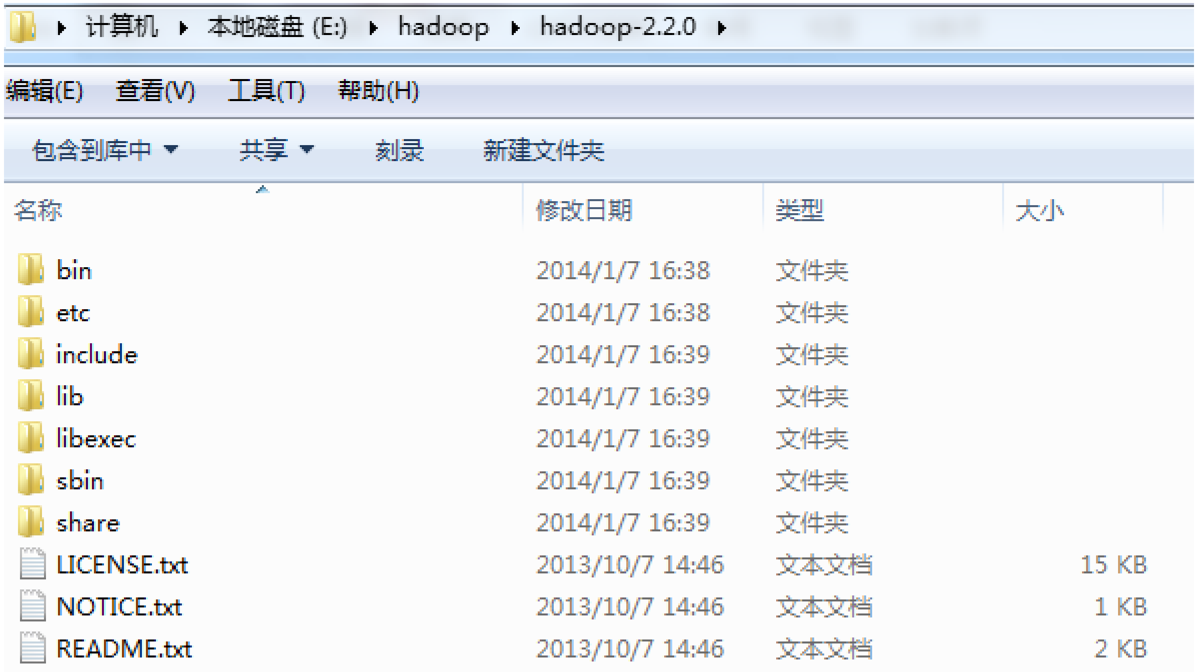
Table 16.1. RHEL/CentOS 5

Project	Download
Hadoop	hadoop-2.2.0.2.0.6.0-76.tar.gz
Pig	pig-0.12.0.2.0.6.0-76.tar.gz
Hive and HCatalog	hive-0.12.0.2.0.6.0-76.tar.gz hcatalog-0.12.0.2.0.6.0-76.tar.gz
HBase and ZooKeeper	hbase-0.96.0.2.0.6.0-76-hadoop2-bin.tar.gz zookeeper-3.4.5.2.0.6.0-76.tar.gz
Oozie	oozie-4.0.0.2.0.6.0-76-distro.tar.gz
Sqoop	sqoop-1.4.4.2.0.6.0-76.bin__hadoop-2.2.0.2.0.6.0-76.tar.gz
Flume	apache-flume-1.4.0.2.0.6.0-76-bin.tar.gz
Mahout	mahout-distribution-0.8.0.2.0.6.0-76.tar.gz

Hadoop 2.0安装包目录结构分析

➢各个发行版均一样





Hadoop 2.0安装包目录结构分析

- bin：Hadoop最基本的管理脚本和使用脚本所在目录，这些脚本是sbin目录下管理脚本的基础实现，用户可以直接使用这些脚本管理和使用Hadoop。
- etc：Hadoop配置文件所在的目录，包括core-site.xml、hdfs-site.xml、mapred-site.xml等从Hadoop 1.0继承而来的配置文件和yarn-site.xml等Hadoop 2.0新增的配置文件。
- include：对外提供的编程库头文件（具体动态库和静态库在lib目录中），这些头文件均是用C++定义的，通常用于C++程序访问HDFS或者编写MapReduce程序。
- lib：该目录包含了Hadoop对外提供的编程动态库和静态库，与include目录中的头文件结合使用。
- libexec：各个服务对应的shell配置文件所在目录，可用于配置日志输出目录、启动参数（比如JVM参数）等基本信息。
- sbin：Hadoop管理脚本所在目录，主要包含HDFS和YARN中各类服务的启动/关闭脚本。
- share：Hadoop各个模块编译后的jar包所在目录。

本课程采用的Hadoop版本

- 采用**Apache Hadoop 2.2.0**
- 如果采用**HDP**和**CDH**版本，可仿照操作，安装部署方式一模一样。

目录

Hadoop 2.0测试环境安装部署

步骤1：将安装包hadoop-2.2.0.tar.gz存放到某一目录下，并解压

步骤2：修改解压后的目录中的文件夹etc/hadoop下的xml配置文件（如果文件不存在，则自己创建）

- ✓ **hadoop-env.sh修改以下配置：**

  export JAVA_HOME=/home/dongxicheng/hadoop/jdk1.6.0_45

- ✓ **Slaves文件修改为以下配置：**

  YARN001

注：以下四个XML配置文件，需在标签<configuration>和</configuration>之间增加配置项。

- ✓ **mapred-site.xml：**

```
<property>
   <name>mapreduce.framework.name</name>
   <value>yarn</value>
</property>
```

Hadoop 2.0测试环境安装部署

- ✓ **core-site.xml（其中“YARN001”是在/etc/hosts中设置的host，如果未设置，则换为localhost）：**

```
<property>
    <name>fs.default.name</name>
    <value>hdfs://YARN001:8020</value>
</property>
```

- ✓ **yarn-site.xml：**

```
<property>
     <name>yarn.nodemanager.aux-services</name>
     <value>mapreduce_shuffle</value>
</property>
```

- ✓ **core-site.xml：**

```
<property>
    <name>dfs.replication</name>
    <value>1</value>
</property>
```

Hadoop 2.0测试环境安装部署

步骤3：启动服务：

✓ 格式化HDFS：

bin/hadoop namenode -format

✓ 启动HDFS：

sbin/ start-dfs.sh

✓ 启动YARN：

sbin/start-yarn.sh

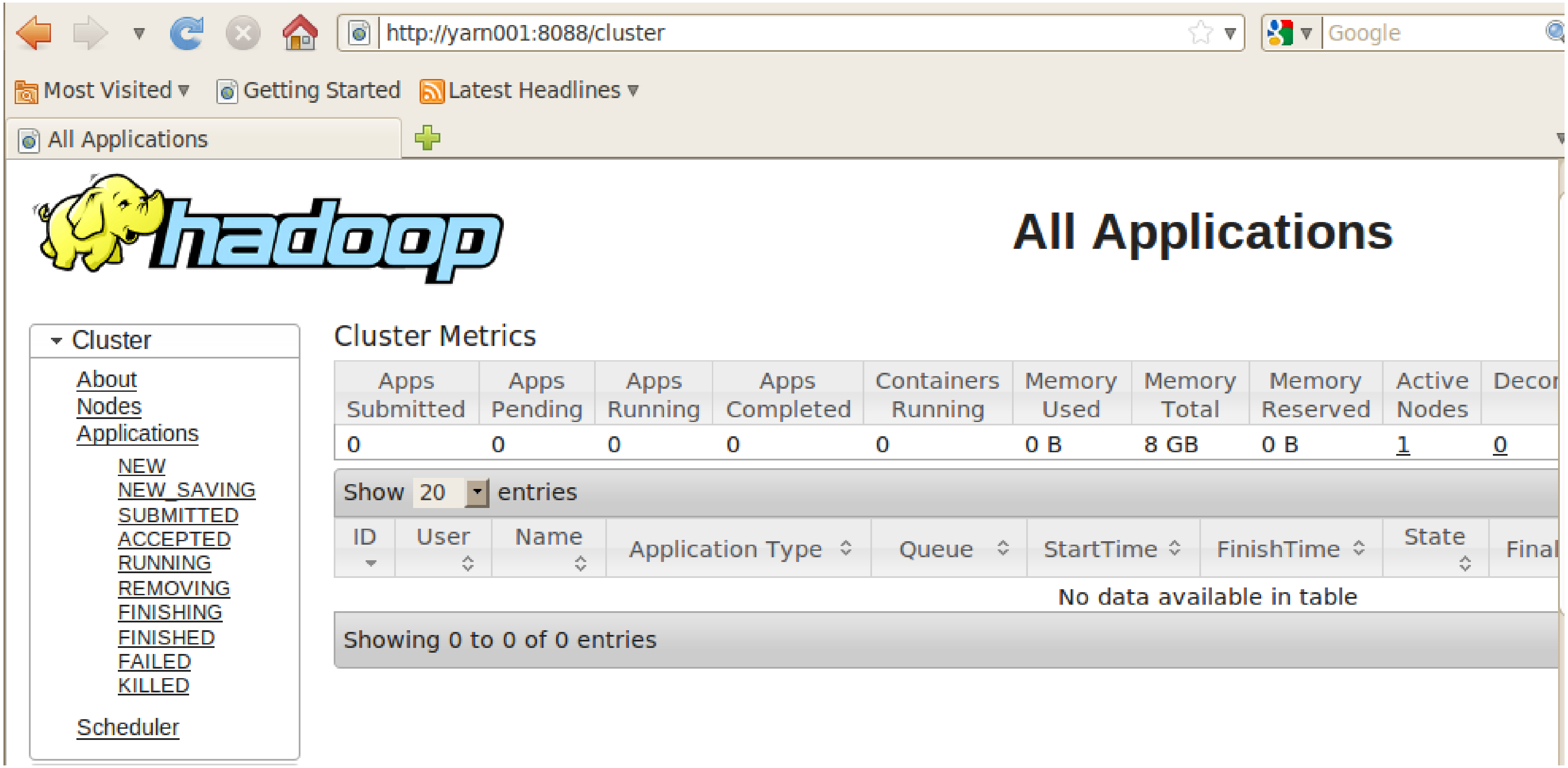
步骤4：验证是否启动成功：

```
dongxicheng@dongxicheng-laptop:~/hadoop/hadoop-2.2.0$ /home/dongxicheng/hadoop/jdk1.6.0_45/bin/jps

3902 Jps
3085 NameNode
3225 DataNode
3832 NodeManager
3603 ResourceManager
3442 SecondaryNameNode
```

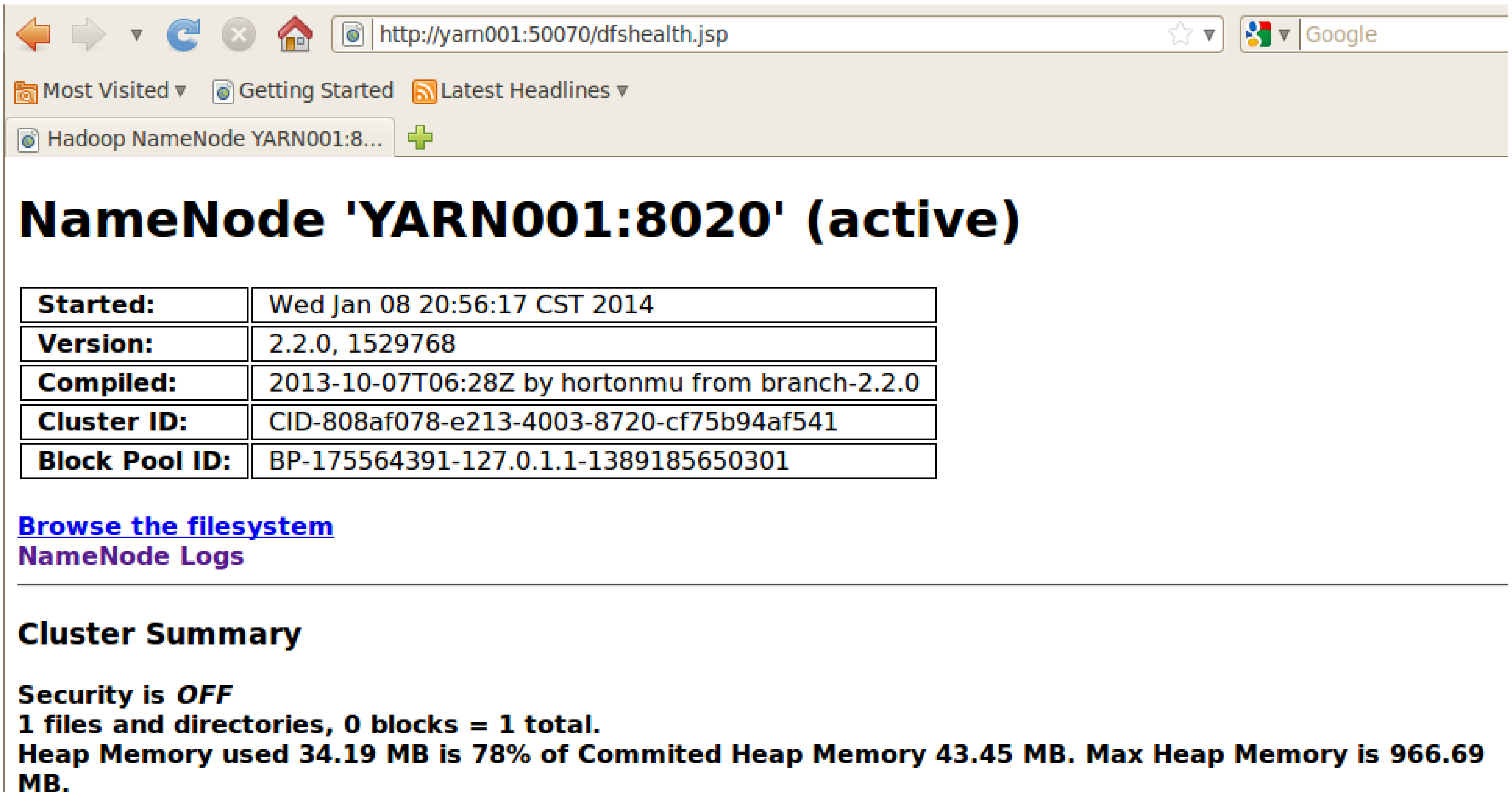
YARN Web界面

访问URL地址：http://yarn001:8088



HDFS Web界面

访问URL地址：http://yarn001:50070



http://yarn001:50070/dfshealth.jsp

Most Visited ▾ Getting Started Latest Headlines ▾

Hadoop NameNode YARN001:8...

NameNode 'YARN001:8020' (active)

Started:	Wed Jan 08 20:56:17 CST 2014
Version:	2.2.0, 1529768
Compiled:	2013-10-07T06:28Z by hortonmu from branch-2.2.0
Cluster ID:	CID-808af078-e213-4003-8720-cf75b94af541
Block Pool ID:	BP-175564391-127.0.1.1-1389185650301

Browse the filesystem
NameNode Logs

Cluster Summary

Security is *OFF*
1 files and directories, 0 blocks = 1 total.
Heap Memory used 34.19 MB is 78% of Commited Heap Memory 43.45 MB. Max Heap Memory is 966.69 MB.

常见问题

问题1：采用虚拟机搭建Hadoop环境，每次虚拟机重启后，Hadoop无法启动成功。

解决方案：

- ✓ **在core-site.xml中增加以下两个配置：**

```
<property>
   <name>dfs.namenode.name.dir</name>
   <value>/home/dongxicheng/hadoop/dfs/name</value>
 </property>
<property>
   <name>dfs.datanode.data.dir</name>
   <value>/home/dongxicheng/hadoop/dfs/data</value>
 </property>
```

其中，各个目录一定是非/tmp下的目录

目录

1. Hadoop 2.0安装部署流程
2. Hadoop 2.0软硬件准备
3. Hadoop 2.0安装包下载
4. Hadoop 2.0测试环境（单机）搭建方法
5. Hadoop 2.0生产环境（多机）搭建方法
6. 总结

Hadoop 2.0生产环境安装部署流程概述

步骤1：将安装包hadoop-2.2.0.tar.gz存放到某一目录下，并解压；

步骤2：修改解压后的目录中的文件夹etc/hadoop下的xml配置文件（如果文件不存在，则自己创建），包括hadoop-env.sh、mapred-site.xml、core-site.xml、hdfs-site.xml和yarn-site.xml；

步骤3：格式化并启动HDFS；

步骤4：启动YARN。

以上整个过程与实验环境基本一致，不同的是步骤2中配置文件设置内容以及步骤3的详细过程。

HDFS 2.0的HA配置方法（主备NameNode）

注意事项：

（1）主备NameNode有多种配置方法，本课程使用Journal Node方式。为此，需要至少准备3个节点作为Journal Node，这三个节点可与其他服务，比如NodeManager共用节点

（2）主备两个NameNode应位于不同机器上，这两台机器不要再部署其他服务，即它们分别独享一台机器。（注：HDFS 2.0中无需再部署和配置Secondary Name，备NameNode已经代替它完成相应的功能）

（3）主备NameNode之间有两种切换方式：手动切换和自动切换，其中，自动切换是借助Zookeeper实现的，因此，需单独部署一个Zookeeper集群（通常为奇数个节点，至少3个）。本课程使用手动切换方式。

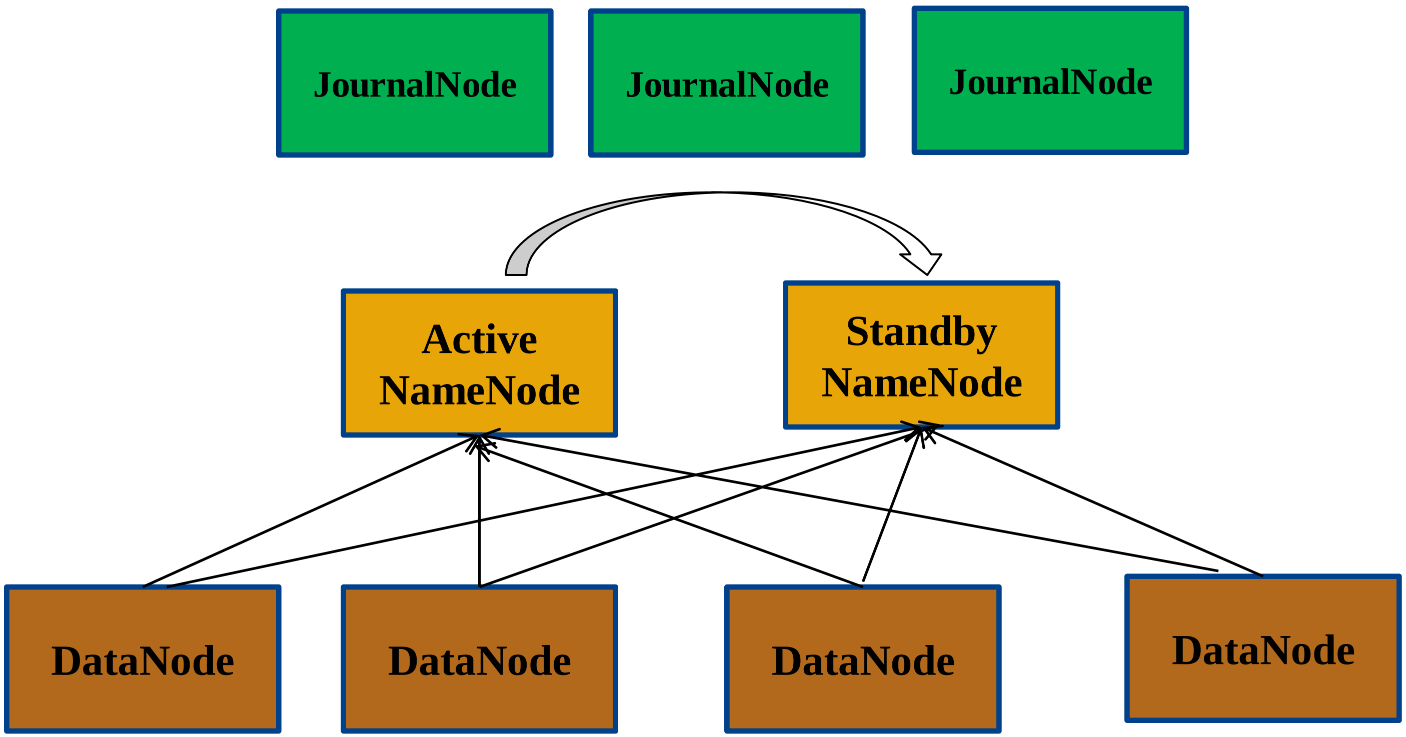
接下来介绍

1. **HDFS HA部署方法**

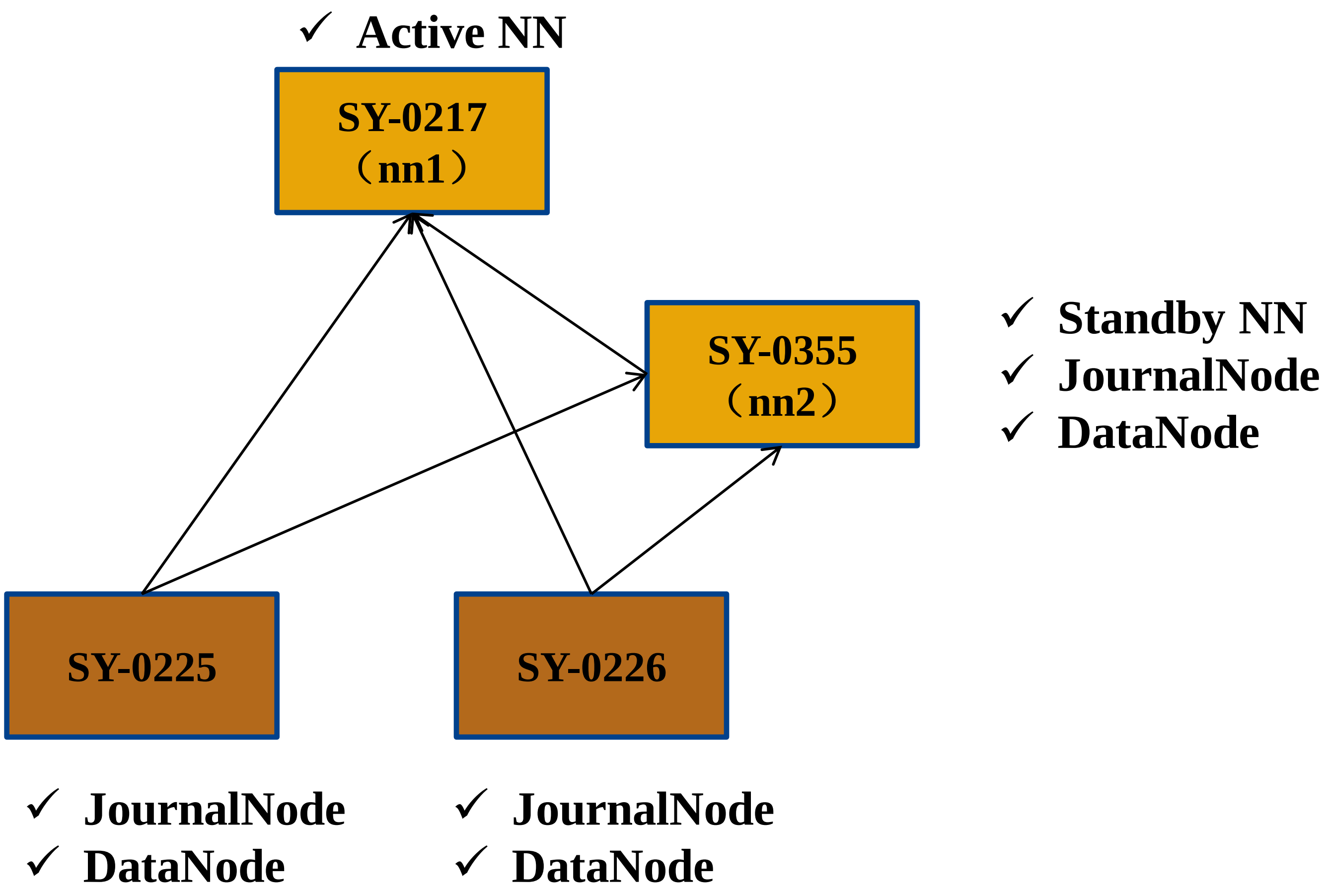
2. **HDFS HA+Federation的部署方法**

3. **YARN部署方法**

HDFS HA部署架构图



HDFS HA部署架构图



HDFS HA部署流程—hdfs-site.xml配置

- **dfs.nameservices**

集群中命名服务列表（自定义）

- **dfs.ha.namenodes.${ns}**

命名服务中的namenode逻辑名称（自定义）

- **dfs.namenode.rpc-address.${ns}.${nn}**

命名服务中逻辑名称对应的RPC地址

- **dfs.namenode.http-address..${ns}.${nn}**

命名服务中逻辑名称对应的HTTP地址

HDFS HA部署流程—hdfs-site.xml配置

- **dfs.namenode.name.dir**

NameNode fsiamge存放目录

- **dfs.namenode.shared.edits.dir**

主备NameNode同步元信息的共享存储系统

- **dfs.journalnode.edits.dir**

Journal Node数据存放目录

HDFS HA部署流程—hdfs-site.xml配置实例

```xml
<configuration>

<property>
  <name>dfs.nameservices</name>
  <value>hadoop-test</value>
</property>

<property>
  <name>dfs.ha.namenodes.hadoop-test</name>
  <value>nn1,nn2</value>
</property>

<property>
  <name>dfs.namenode.rpc-address.hadoop-test.nn1</name>
  <value>SY-0217:8020</value>
</property>

<property>
  <name>dfs.namenode.rpc-address.hadoop-test.nn2</name>
  <value>SY-0355:8020</value>
</property>

<property>
  <name>dfs.namenode.http-address.hadoop-test.nn1</name>
  <value>SY-0217:50070</value>
</property>

<property>
  <name>dfs.namenode.http-address.hadoop-test.nn2</name>
  <value>SY-0355:50070</value>
</property>
```

HDFS HA部署流程—hdfs-site.xml配置实例

```
<property>
  <name>dfs.namenode.name.dir</name>
  <value>file:///home/dongxicheng/hadoop/hdfs/name</value>
</property>

<property>
  <name>dfs.namenode.shared.edits.dir</name>
  <value>qjournal://SY-0355:8485;SY-0225:8485;SY-0226:8485/hadoop-test</value>
</property>

<property>
  <name>dfs.datanode.data.dir</name>
  <value>file:///home/dongxicheng/hadoop/hdfs/data</value>
</property>

<property>
  <name>dfs.ha.automatic-failover.enabled</name>
  <value>false</value>
</property>

<property>
  <name>dfs.journalnode.edits.dir</name>
  <value>/home/dongxicheng/hadoop/hdfs/journal/</value>
</property>

</configuration>
```

slaves

```
SY-0226
SY-0225
SY-0355
```

core-site.xml配置文件

```
<configuration>

<property>
  <name>fs.defaultFS</name>
  <value>hdfs://SY-0217:8020</value>
</property>

</configuration>
```

HDFS HA部署流程—启动/关闭HDFS

（注意：所有操作均在Hadoop安装目录下进行。）

步骤1：在各个JournalNode节点上，输入以下命令启动journalnode服务：

sbin/hadoop-daemon.sh start journalnode

步骤2：在[nn1]上，对其进行格式化，并启动：

bin/hdfs namenode -format

sbin/hadoop-daemon.sh start namenode

步骤3：在[nn2]上，同步nn1的元数据信息：

bin/hdfs namenode -bootstrapStandby

步骤4：在[nn2]，启动NameNode：

sbin/hadoop-daemon.sh start namenode

（经过以上四步操作，nn1和nn2均处理standby状态）

步骤5：在[nn1]上，将NameNode切换为Active

bin/hdfs haadmin -transitionToActive nn1

步骤6：在[nn1]上，启动所有datanode

sbin/hadoop-daemons.sh start datanode

关闭Hadoop集群：

在[nn1]上，输入以下命令

sbin/stop-dfs.sh

HDFS HA WEB界面—Active Namenode

10.10.64.226:50070/dfshealth.jsp

NameNode 'SY-0217:8020' (active)

Started:	Thu Jan 09 15:44:58 CST 2014
Version:	2.2.0, 1529768
Compiled:	2013-10-07T06:28Z by hortonmu from branch-2.2.0
Cluster ID:	CID-7e701e39-aaea-4ff1-86fd-a97727535046
Block Pool ID:	BP-661436583-10.10.64.226-1389253491160

Browse the filesystem
NameNode Logs

Cluster Summary

Security is *OFF*
1 files and directories, 0 blocks = 1 total.
Heap Memory used 83.45 MB is 8% of Commited Heap Memory 958.38 MB. Max Heap Memory is 958.38 MB.
Non Heap Memory used 31.16 MB is 95% of Commited Non Heap Memory 32.56 MB. Max Non Heap Memory is 130 MB.

Configured Capacity	:	5.27 TB				
DFS Used	:	72 KB				
Non DFS Used	:	432.13 GB				
DFS Remaining	:	4.85 TB				
DFS Used%	:	0.00%				
DFS Remaining%	:	92.00%				
Block Pool Used	:	72 KB				
Block Pool Used%	:	0.00%				
DataNodes usages	:	Min %	Median %	Max %	stdev %	
		0.00%	0.00%	0.00%	0.00%	
Live Nodes	:	3 (Decommissioned: 0)				
Dead Nodes	:	0 (Decommissioned: 0)				
Decommissioning Nodes	:	0				
Number of Under-Replicated Blocks	:	0				

HDFS HA WEB界面—Standby Namenode

10.10.65.78:50070/dfshealth.jsp

NameNode 'SY-0355:8020' (standby)

Started:	Thu Jan 09 15:48:11 CST 2014
Version:	2.2.0, 1529768
Compiled:	2013-10-07T06:28Z by hortonmu from branch-2.2.0
Cluster ID:	CID-7e701e39-aaea-4ff1-86fd-a97727535046
Block Pool ID:	BP-661436583-10.10.64.226-1389253491160

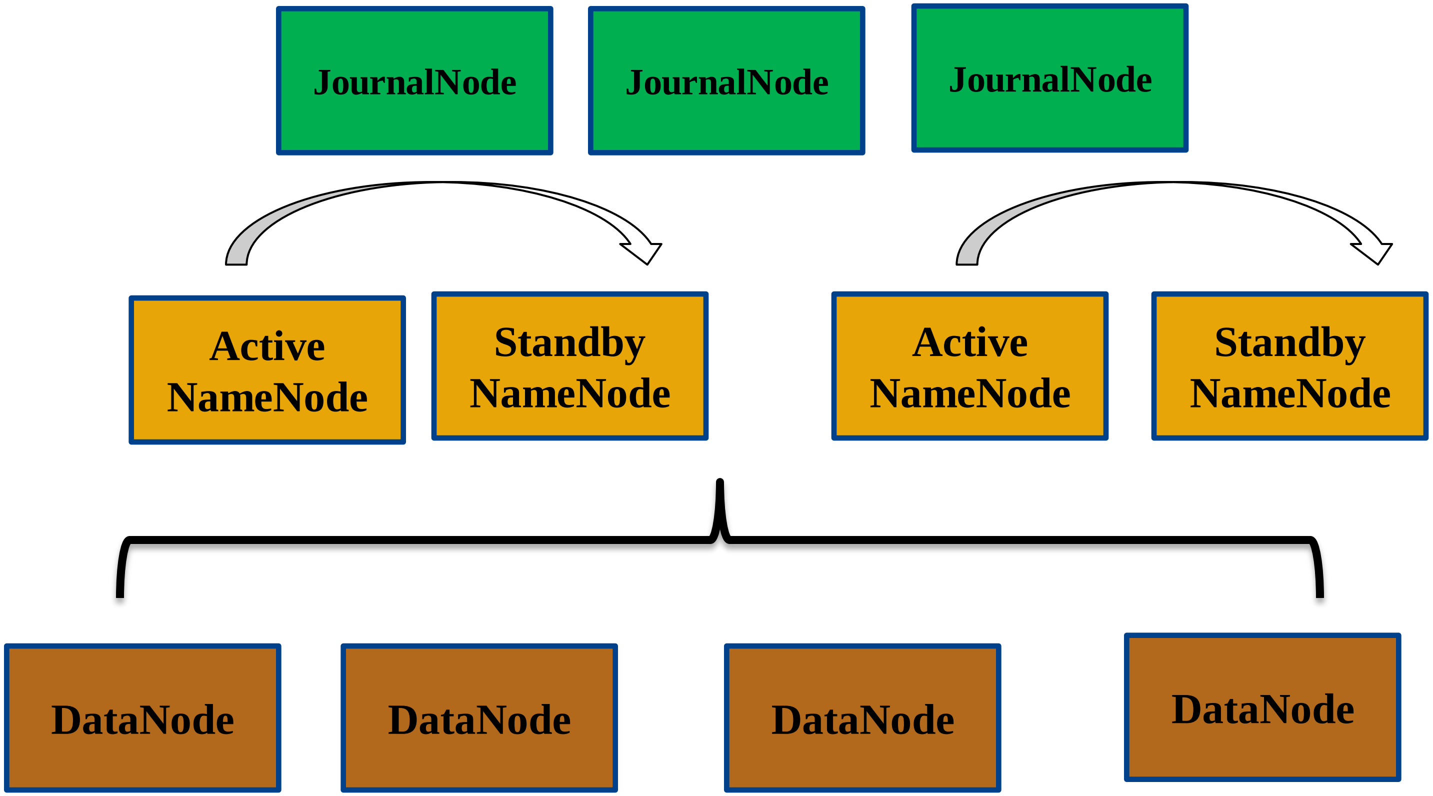
NameNode Logs

Cluster Summary

Security is *OFF*
1 files and directories, 0 blocks = 1 total.
Heap Memory used 27.95 MB is 2% of Commited Heap Memory 958.38 MB. Max Heap Memory is 958.38 MB.
Non Heap Memory used 30.22 MB is 96% of Commited Non Heap Memory 31.44 MB. Max Non Heap Memory is 130 MB.

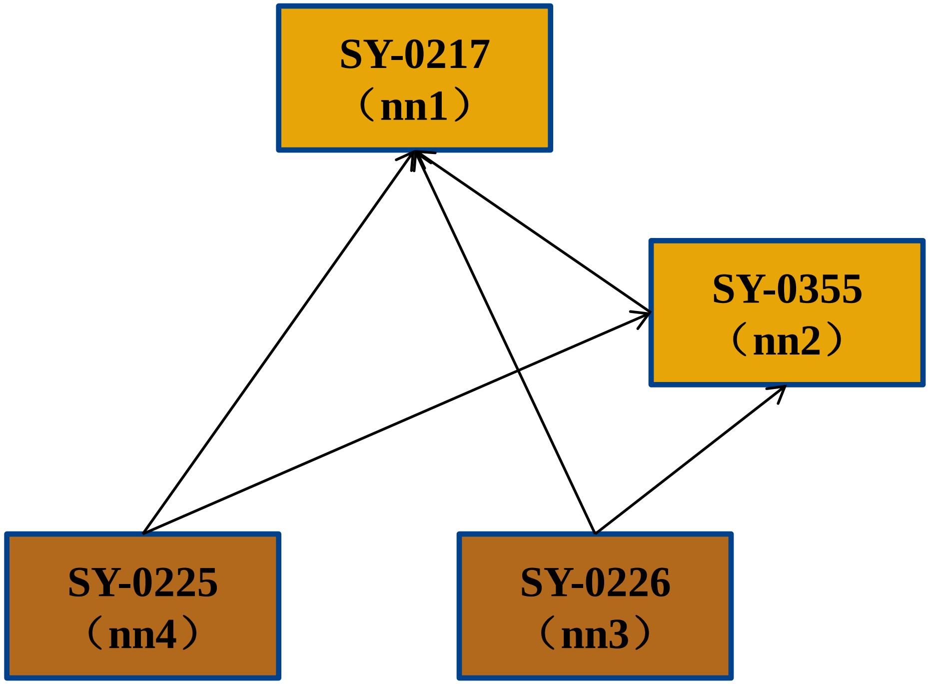
Configured Capacity	:	5.27 TB			
DFS Used	:	72 KB			
Non DFS Used	:	432.13 GB			
DFS Remaining	:	4.85 TB			
DFS Used%	:	0.00%			
DFS Remaining%	:	92.00%			
Block Pool Used	:	72 KB			
Block Pool Used%	:	0.00%			
DataNodes usages	:	Min %	Median %	Max %	stdev %
		0.00%	0.00%	0.00%	0.00%
Live Nodes	:	3 (Decommissioned: 0)			
Dead Nodes	:	0 (Decommissioned: 0)			
Decommissioning Nodes	:	0			

HDFS HA+Federation部署架构图



HDFS HA+Federation部署架构图

- ✓ Active NN



- ✓ Standby NN
- ✓ JournalNode
- ✓ DataNode

- ✓ Active NN
- ✓ JournalNode
- ✓ DataNode

- ✓ Standby NN
- ✓ JournalNode
- ✓ DataNode

HDFS HA+Federation部署流程—hdfs-site.xml配置实例

```
<configuration>

<property>
  <name>dfs.nameservices</name>
  <value>hadoop-cluster1,hadoop-cluster2</value>
</property>

<!--
   hadoop cluster1
-->
<property>
  <name>dfs.ha.namenodes.hadoop-cluster1</name>
  <value>nn1,nn2</value>
</property>

<property>
  <name>dfs.namenode.rpc-address.hadoop-cluster1.nn1</name>
  <value>SY-0217:8020</value>
</property>

<property>
  <name>dfs.namenode.rpc-address.hadoop-cluster1.nn2</name>
  <value>SY-0355:8020</value>
</property>

<property>
  <name>dfs.namenode.http-address.hadoop-cluster1.nn1</name>
  <value>SY-0217:50070</value>
</property>

<property>
  <name>dfs.namenode.http-address.hadoop-cluster1.nn2</name>
  <value>SY-0355:50070</value>
</property>
```

HDFS HA+Federation部署流程—hdfs-site.xml配置实例

```xml
<!--
    hadoop cluster2
-->
<property>
  <name>dfs.ha.namenodes.hadoop-cluster2</name>
  <value>nn3,nn4</value>
</property>

<property>
  <name>dfs.namenode.rpc-address.hadoop-cluster2.nn3</name>
  <value>SY-0226:8020</value>
</property>

<property>
  <name>dfs.namenode.rpc-address.hadoop-cluster2.nn4</name>
  <value>SY-0225:8020</value>
</property>

<property>
  <name>dfs.namenode.http-address.hadoop-cluster2.nn3</name>
  <value>SY-0226:50070</value>
</property>

<property>
  <name>dfs.namenode.http-address.hadoop-cluster2.nn4</name>
  <value>SY-0225:50070</value>
</property>
```

HDFS Federation部署流程—注意！！！！

- **hadoop-cluster1中的nn1和nn2和hadoop-cluste2中的nn3和nn4可以公用同样的journal node，但dfs.namenode.shared.edits.dir配置不能相同。**
- **hadoop-cluster1中的nn1和nn2配置如下：**

```
<property>
  <name>dfs.namenode.shared.edits.dir</name>
  <value>qjournal://SY-0355:8485;SY-0225:8485;SY-0226:8485/hadoop-cluster1</value>
  <description>A directory on shared storage between the multiple namenodes
  in an HA cluster. This directory will be written by the active and read
  by the standby in order to keep the namespaces synchronized. This directory
  does not need to be listed in dfs.namenode.edits.dir above. It should be
  left empty in a non-HA cluster.
  </description>
</property>
```

- **hadoop-cluster2中的nn3和nn4配置如下：**

```
<property>
  <name>dfs.namenode.shared.edits.dir</name>
  <value>qjournal://SY-0355:8485;SY-0225:8485;SY-0226:8485/hadoop-cluster2</value>
  <description>A directory on shared storage between the multiple namenodes
  in an HA cluster. This directory will be written by the active and read
  by the standby in order to keep the namespaces synchronized. This directory
  does not need to be listed in dfs.namenode.edits.dir above. It should be
  left empty in a non-HA cluster.
  </description>
</property>
```

HDFS HA+Federation部署流程—启动/关闭HDFS

在nn1和nn2两个节点上进行如下操作：

步骤1：在各个JournalNode节点上，输入以下命令启动journalnode服务：

sbin/hadoop-daemon.sh start journalnode

步骤2：在[nn1]上，对其进行格式化，并启动：

bin/hdfs namenode –format -clusterId hadoop-cluster

sbin/hadoop-daemon.sh start namenode

步骤3：在[nn2]上，同步nn1的元数据信息：

bin/hdfs namenode -bootstrapStandby

步骤4：在[nn2]，启动NameNode：

sbin/hadoop-daemon.sh start namenode

（经过以上四步操作，nn1和nn2均处理standby状态）

步骤5：在[nn1]上，将NameNode切换为Active

bin/hdfs haadmin -ns hadoop-cluster1 -transitionToActive nn1

HDFS HA+Federation部署流程—启动/关闭HDFS

在nn3和nn4两个节点上进行如下操作：

步骤1：在各个JournalNode节点上，输入以下命令启动journalnode服务：

sbin/hadoop-daemon.sh start journalnode

步骤2：在[nn3]上，对其进行格式化，并启动：

bin/hdfs namenode –format -clusterId hadoop-cluster

sbin/hadoop-daemon.sh start namenode

步骤3：在[nn4]上，同步nn1的元数据信息：

bin/hdfs namenode -bootstrapStandby

步骤4：在[nn4]，启动NameNode：

sbin/hadoop-daemon.sh start namenode

（经过以上四步操作，nn3和nn4均处理standby状态）

步骤5：在[nn3]上，将NameNode切换为Active

bin/hdfs haadmin -ns hadoop-cluster2 -transitionToActive nn3

HDFS HA+Federation部署流程—启动/关闭HDFS

最后：在[nn1]上，启动所有datanode

sbin/hadoop-daemons.sh start datanode

HDFS HA WEB界面—Hadoop-cluster1/nn1

10.10.64.226:50070/dfshealth.jsp

NameNode 'SY-0217:8020' (active)

Started:	Fri Jan 10 09:48:34 CST 2014
Version:	2.2.0, 1529768
Compiled:	2013-10-07T06:28Z by hortonmu from branch-2.2.0
Cluster ID:	hadoop-cluster
Block Pool ID:	BP-1752168733-10.10.64.226-1389318505635

Browse the filesystem
NameNode Logs

Cluster Summary

Security is *OFF*
3 files and directories, 0 blocks = 3 total.
Heap Memory used 114.42 MB is 11% of Commited Heap Memory 958.38 MB. Max Heap Memory is 958.38 MB.
Non Heap Memory used 32.75 MB is 96% of Commited Non Heap Memory 33.88 MB. Max Non Heap Memory is 130 MB.

Configured Capacity	:	5.27 TB			
DFS Used	:	144 KB			
Non DFS Used	:	436.63 GB			
DFS Remaining	:	4.85 TB			
DFS Used%	:	0.00%			
DFS Remaining%	:	91.91%			
Block Pool Used	:	72 KB			
Block Pool Used%	:	0.00%			
DataNodes usages	:	Min %	Median %	Max %	stdev %
		0.00%	0.00%	0.00%	0.00%
Live Nodes	:	3 (Decommissioned: 0)			
Dead Nodes	:	0 (Decommissioned: 0)			
Decommissioning Nodes	:	0			
Number of Under-Replicated Blocks	:	0			

HDFS HA WEB界面—Hadoop-cluster1/nn2

10.10.65.78:50070/dfshealth.jsp

NameNode 'SY-0355:8020' (standby)

Started:	Fri Jan 10 09:52:20 CST 2014
Version:	2.2.0, 1529768
Compiled:	2013-10-07T06:28Z by hortonmu from branch-2.2.0
Cluster ID:	hadoop-cluster
Block Pool ID:	BP-1752168733-10.10.64.226-1389318505635

NameNode Logs

Cluster Summary

Security is *OFF*
1 files and directories, 0 blocks = 1 total.
Heap Memory used 228.68 MB is 23% of Commited Heap Memory 958.38 MB. Max Heap Memory is 958.38 MB.
Non Heap Memory used 31.94 MB is 96% of Commited Non Heap Memory 33.13 MB. Max Non Heap Memory is 130 MB.

Configured Capacity	:	5.27 TB				
DFS Used	:	144 KB				
Non DFS Used	:	436.62 GB				
DFS Remaining	:	4.85 TB				
DFS Used%	:	0.00%				
DFS Remaining%	:	91.91%				
Block Pool Used	:	72 KB				
Block Pool Used%	:	0.00%				
DataNodes usages	:	Min %	Median %	Max %	stdev %	
		0.00%	0.00%	0.00%	0.00%	
Live Nodes	:	3 (Decommissioned: 0)				
Dead Nodes	:	0 (Decommissioned: 0)				
Decommissioning Nodes	:	0				

HDFS HA WEB界面—Hadoop-cluster2/nn3

10.10.64.235:50070/dfshealth.jsp

NameNode 'SY-0226:8020' (active)

Started:	Fri Jan 10 09:30:05 CST 2014
Version:	2.2.0, 1529768
Compiled:	2013-10-07T06:28Z by hortonmu from branch-2.2.0
Cluster ID:	hadoop-cluster
Block Pool ID:	BP-1754480490-10.10.64.235-1389317394267

Browse the filesystem
NameNode Logs

Cluster Summary

Security is *OFF*
1 files and directories, 0 blocks = 1 total.
Heap Memory used 166.21 MB is 17% of Commited Heap Memory 958.38 MB. Max Heap Memory is 958.38 MB.
Non Heap Memory used 33.23 MB is 97% of Commited Non Heap Memory 34.25 MB. Max Non Heap Memory is 130 MB.

Configured Capacity	:	5.27 TB			
DFS Used	:	148.06 KB			
Non DFS Used	:	436.65 GB			
DFS Remaining	:	4.85 TB			
DFS Used%	:	0.00%			
DFS Remaining%	:	91.91%			
Block Pool Used	:	72 KB			
Block Pool Used%	:	0.00%			
DataNodes usages	:	Min %	Median %	Max %	stdev %
		0.00%	0.00%	0.00%	0.00%
Live Nodes	:	3 (Decommissioned: 0)			
Dead Nodes	:	0 (Decommissioned: 0)			
Decommissioning Nodes	:	0			
Number of Under-Replicated Blocks	:	0			

HDFS HA WEB界面—Hadoop-cluster2/nn4

10.10.64.234:50070/dfshealth.jsp

NameNode 'SY-0225:8020' (standby)

Started:	Fri Jan 10 09:37:10 CST 2014
Version:	2.2.0, 1529768
Compiled:	2013-10-07T06:28Z by hortonmu from branch-2.2.0
Cluster ID:	hadoop-cluster
Block Pool ID:	BP-1754480490-10.10.64.235-1389317394267

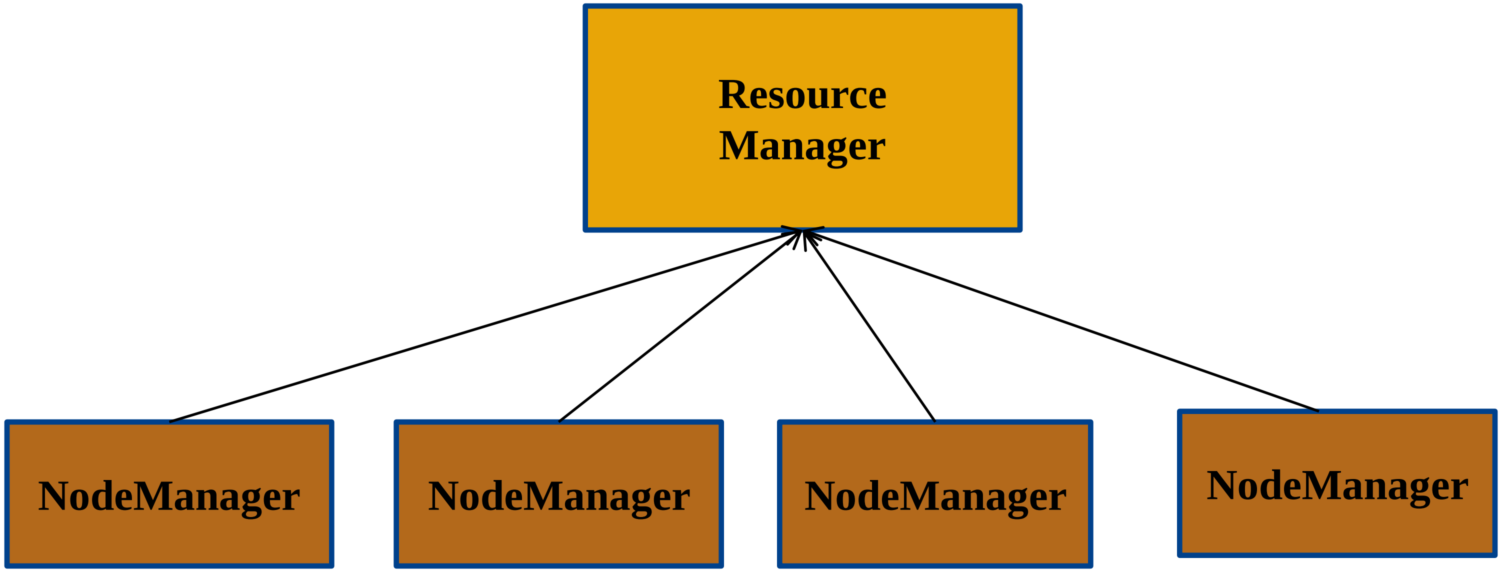
NameNode Logs

Cluster Summary

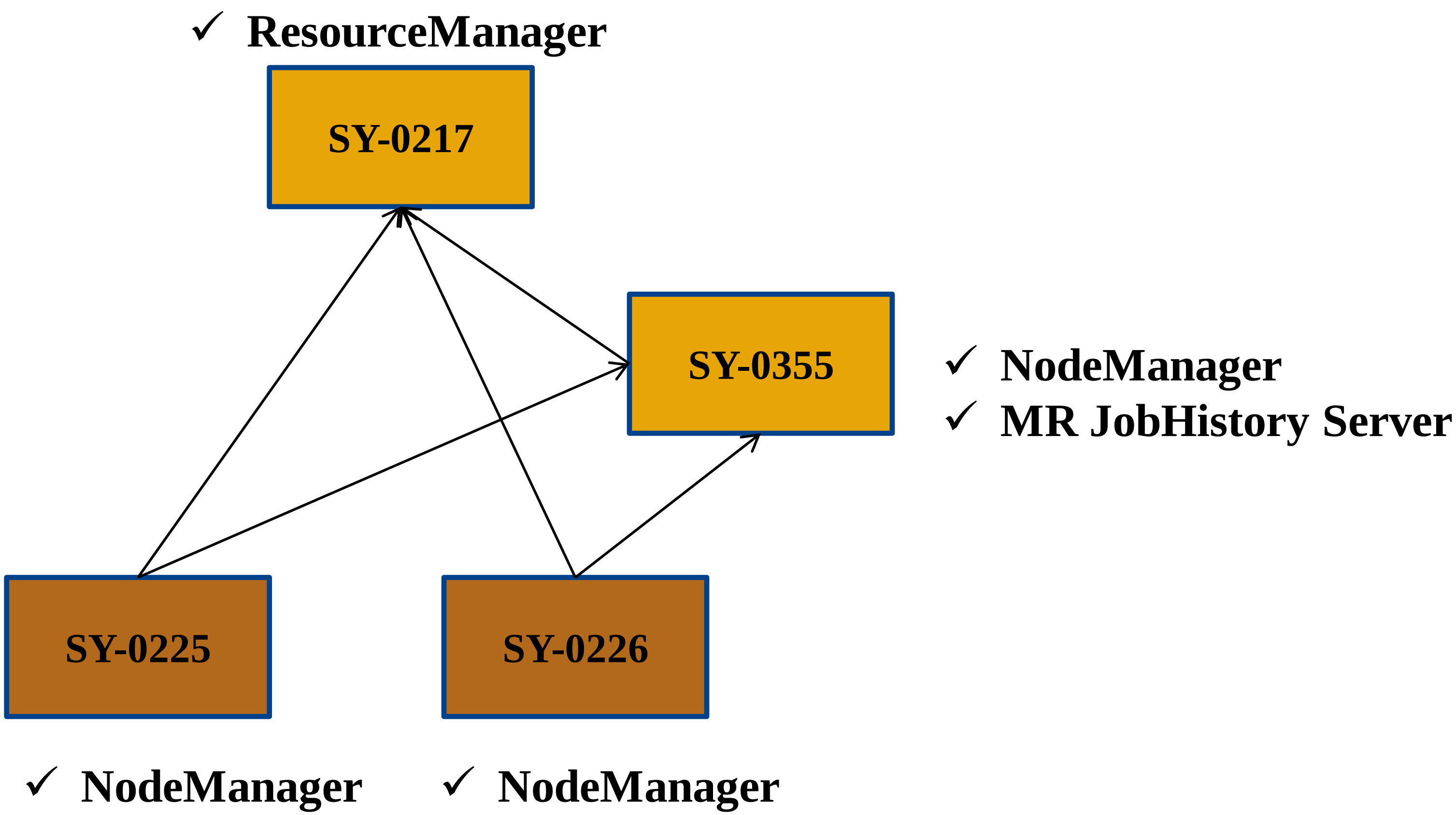
Security is *OFF*
1 files and directories, 0 blocks = 1 total.
Heap Memory used 43.60 MB is 4% of Commited Heap Memory 958.38 MB. Max Heap Memory is 958.38 MB.
Non Heap Memory used 32.02 MB is 96% of Commited Non Heap Memory 33.13 MB. Max Non Heap Memory is 130 MB.

Configured Capacity	:	5.27 TB			
DFS Used	:	144 KB			
Non DFS Used	:	436.61 GB			
DFS Remaining	:	4.85 TB			
DFS Used%	:	0.00%			
DFS Remaining%	:	91.91%			
Block Pool Used	:	72 KB			
Block Pool Used%	:	0.00%			
DataNodes usages	:	Min %	Median %	Max %	stdev %
		0.00%	0.00%	0.00%	0.00%
Live Nodes	:	3 (Decommissioned: 0)			
Dead Nodes	:	0 (Decommissioned: 0)			
Decommissioning Nodes	:	0			

YARN部署架构图


Resource Manager
NodeManager
NodeManager
NodeManager
NodeManager

YARN部署架构图



✓ ResourceManager
SY-0217
SY-0355
✓ NodeManager
✓ MR JobHistory Server
SY-0225
SY-0226
✓ NodeManager
✓ NodeManager

yarn-site.xml配置实例

```xml
<configuration>

  <property>
    <name>yarn.resourcemanager.hostname</name>
    <value>SY-0217</value>
  </property>

  <property>
    <name>yarn.resourcemanager.address</name>
    <value>${yarn.resourcemanager.hostname}:8032</value>
  </property>

  <property>
    <name>yarn.resourcemanager.scheduler.address</name>
    <value>${yarn.resourcemanager.hostname}:8030</value>
  </property>

  <property>
    <name>yarn.resourcemanager.webapp.address</name>
    <value>${yarn.resourcemanager.hostname}:8088</value>
  </property>

  <property>
    <name>yarn.resourcemanager.webapp.https.address</name>
    <value>${yarn.resourcemanager.hostname}:8090</value>
  </property>

  <property>
    <name>yarn.resourcemanager.resource-tracker.address</name>
    <value>${yarn.resourcemanager.hostname}:8031</value>
  </property>
```

yarn-site.xml配置实例

```xml
<property>
  <name>yarn.resourcemanager.admin.address</name>
  <value>${yarn.resourcemanager.hostname}:8033</value>
</property>

<property>
 <name>yarn.resourcemanager.scheduler.class</name>
  <value>org.apache.hadoop.yarn.server.resourcemanager.scheduler.fair.FairScheduler</value>
</property>

<property>
  <name>yarn.scheduler.fair.allocation.file</name>
  <value>${yarn.home.dir}/etc/hadoop/fairscheduler.xml</value>
</property>

<property>
  <name>yarn.nodemanager.local-dirs</name>
  <value>/home/dongxicheng/hadoop/yarn/local</value>
</property>

<property>
  <name>yarn.log-aggregation-enable</name>
  <value>true</value>
</property>

<property>
  <name>yarn.nodemanager.remote-app-log-dir</name>
  <value>/tmp/logs</value>
</property>
```

yarn-site.xml配置实例

```
<property>
 <name>yarn.nodemanager.resource.memory-mb</name>
  <value>30720</value>
</property>

<property>
 <name>yarn.nodemanager.resource.cpu-vcores</name>
  <value>12</value>
</property>

<property>
  <name>yarn.nodemanager.aux-services</name>
  <value>mapreduce_shuffle</value>
</property>

</configuration>
```

fairscheduler.xml配置实例

```xml
<?xml version="1.0"?>
<allocations>

  <queue name="infrastructure">
    <minResources>102400 mb, 50 vcores </minResources>
    <maxResources>153600 mb, 100 vcores </maxResources>
    <maxRunningApps>200</maxRunningApps>
    <minSharePreemptionTimeout>300</minSharePreemptionTimeout>
    <weight>1.0</weight>
    <aclSubmitApps>root,yarn,search,hdfs</aclSubmitApps>
  </queue>

   <queue name="tool">
      <minResources>102400 mb, 30 vcores</minResources>
      <maxResources>153600 mb, 50 vcores</maxResources>
   </queue>

   <queue name="sentiment">
      <minResources>102400 mb, 30 vcores</minResources>
      <maxResources>153600 mb, 50 vcores</maxResources>
   </queue>

</allocations>
```

mapred-site.xml配置实例

```
<configuration>

<property>
  <name>mapreduce.framework.name</name>
  <value>yarn</value>
  <description>The runtime framework for executing MapReduce jobs.
  Can be one of local, classic or yarn.
  </description>
</property>

<!-- jobhistory properties -->
<property>
  <name>mapreduce.jobhistory.address</name>
  <value>SY-0355:10020</value>
  <description>MapReduce JobHistory Server IPC host:port</description>
</property>

<property>
  <name>mapreduce.jobhistory.webapp.address</name>
  <value>SY-0355:19888</value>
  <description>MapReduce JobHistory Server Web UI host:port</description>
</property>

</configuration>
```

YARN启动/停止步骤

在SY-0217上执行以下命令：

✓启动YARN：

sbin/start-yarn.sh

✓停止YARN：

sbin/stop-yarn.sh

在SY-0355上执行以下命令：

✓ 启动MR JobHistory Server：

sbin/mr-jobhistory-daemon.sh start historyserver

目录

1. Hadoop 2.0安装部署流程
2. Hadoop 2.0软硬件准备
3. Hadoop 2.0安装包下载
4. Hadoop 2.0测试环境（单机）搭建方法
5. Hadoop 2.0生产环境（多机）搭建方法
6. 总结

总结

1. Hadoop 2.0安装部署流程
2. Hadoop 2.0软硬件准备
3. Hadoop 2.0安装包下载
4. Hadoop 2.0测试环境（单机）搭建方法
5. Hadoop 2.0生产环境（多机）搭建方法