

企业级Hive实战课程

Hive 预备课程

讲师-梦琪

【声明】 本视频和讲义等为云帆大数据网络课程的教学资料，所有资料只能在课程内使用，不允许在课程以外范围散播，违者将可能被追究法律和经济责任。

课程详情访问云帆大数据培训网站

<http://www.cloudyhadoop.com>

1

Linux环境工具准备

2

Hadoop 2.x分布式部署

3

MySQL数据库安装

4

初识Hive之环境搭建

1

Linux环境工具准备

2

Hadoop 2.x分布式部署

3

MySQL数据库安装

4

初识Hive之环境搭建

◆ 环境操作系统

CentOS 6.4版本64位 VMWare 10虚拟机

◆ 系统用户名与密码

普通用户 cyhp:cyhp 超级管理员用户root:123456

◆ 远程连接工具

SecureCRT、Notepad++、FileZilla

核心：IP地址、用户名、密码、协议SSH（SFTP）

◆ 软件版本

JDK: 7u67-linux-x64

Hadoop: 2.7.1

Hive: 1.2.1

◆ 软件安装目录

/opt/software

/opt/modules

/opt/tools

/opt/datas

1

Linux环境工具准备

2

Hadoop 2.x分布式部署

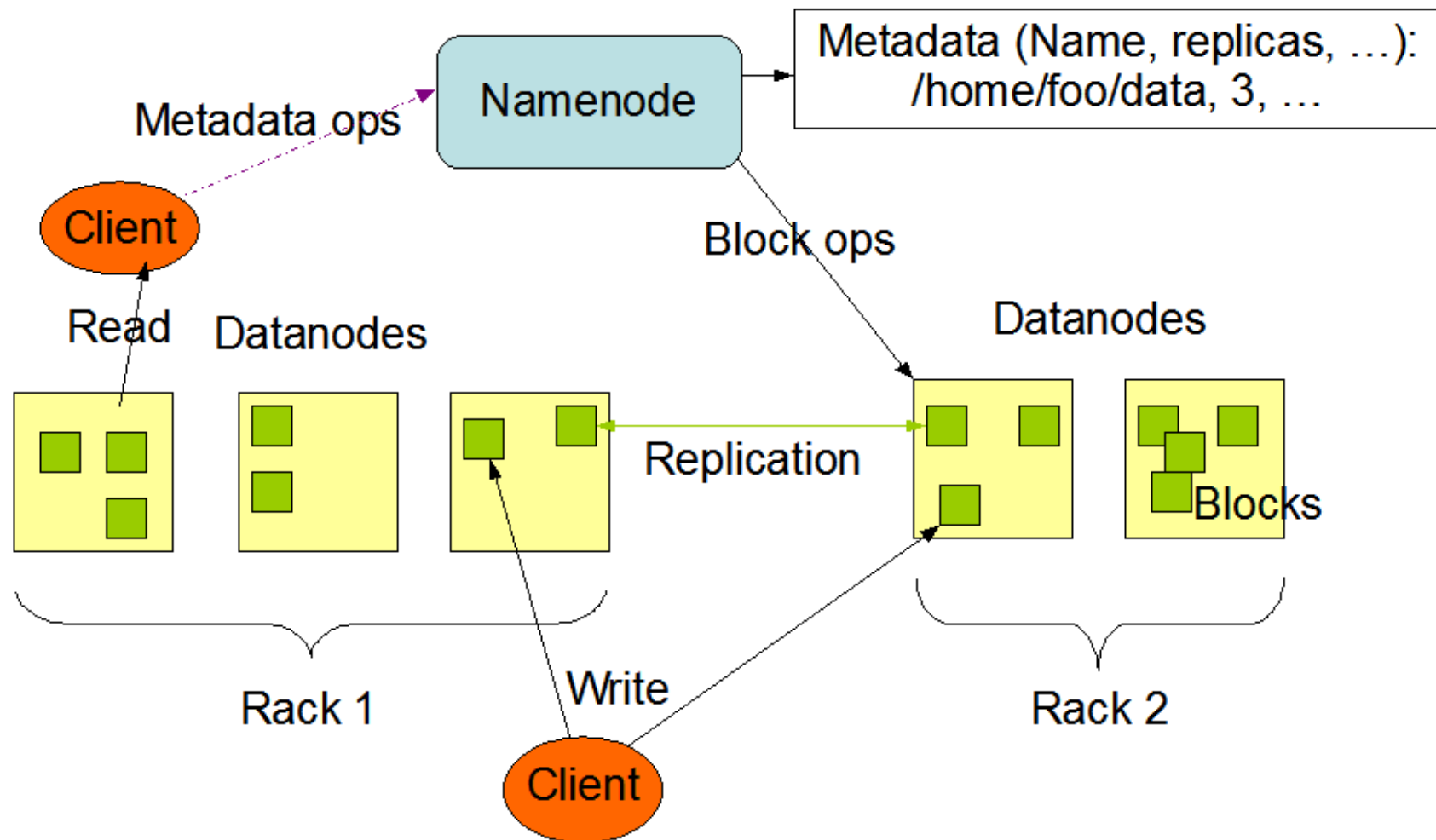
3

MySQL数据库安装

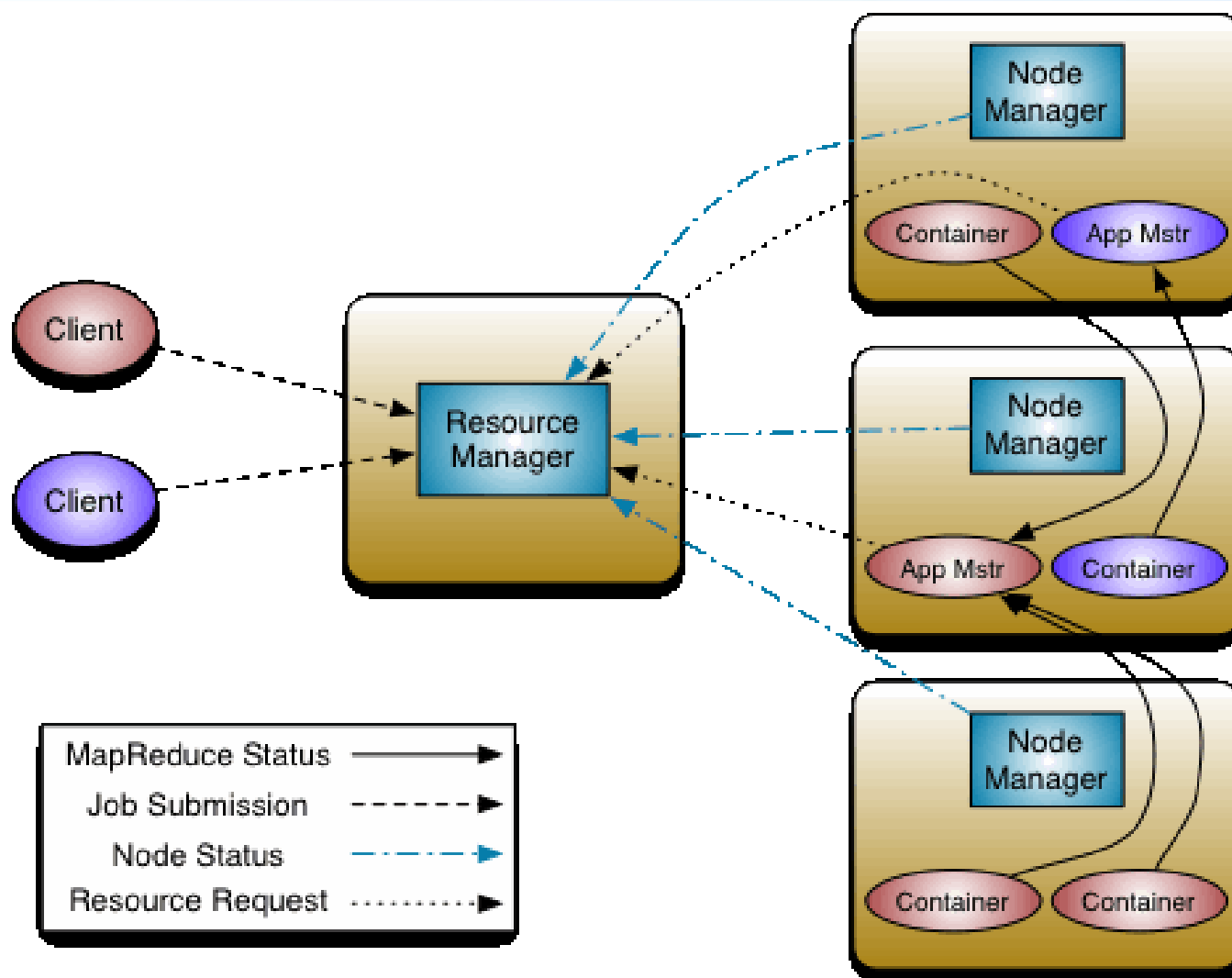
4

初识Hive之环境搭建

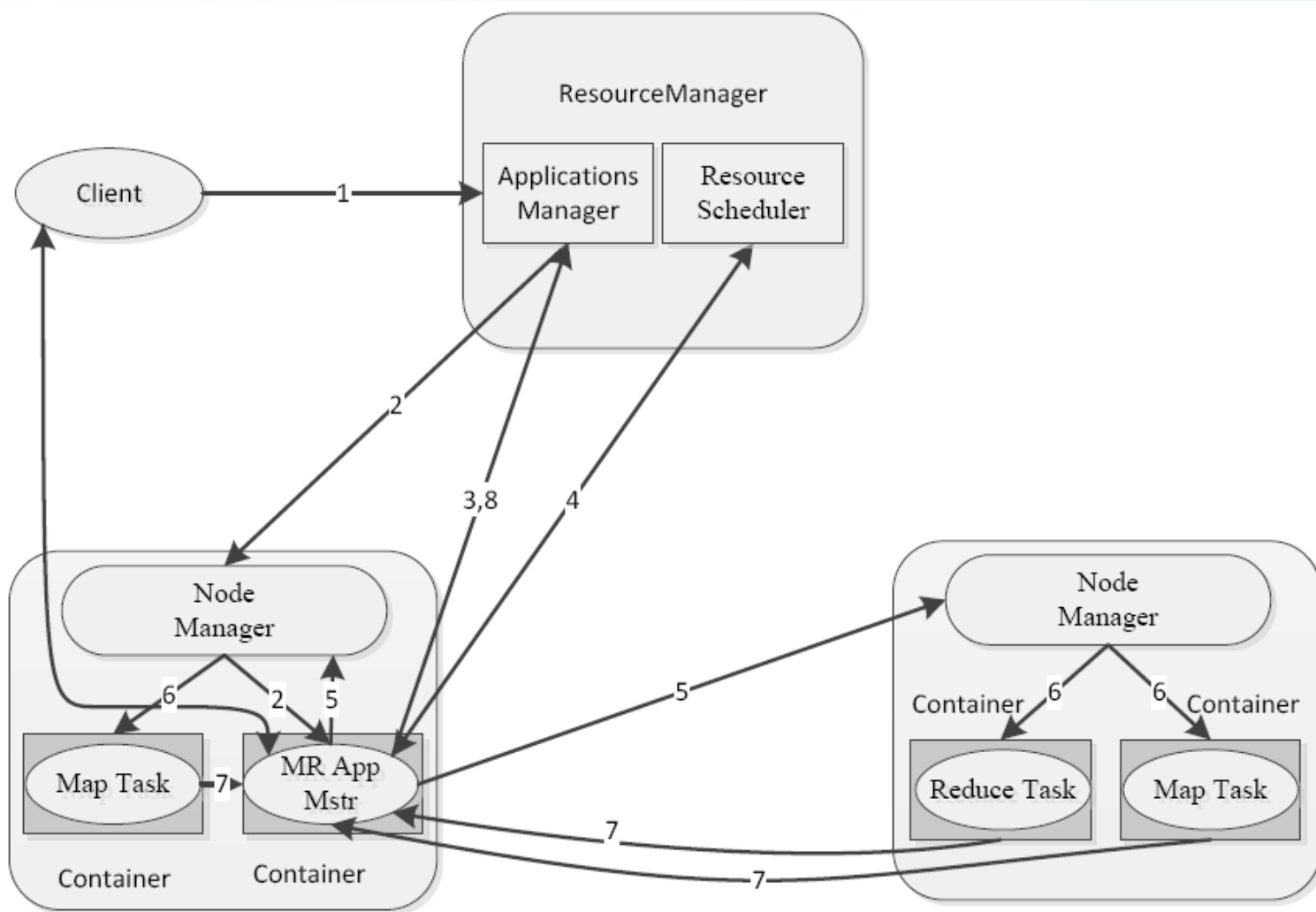
HDFS Architecture



YARN Architecture



MapReduce on YARN





➡ Prepare to Start the Hadoop Cluster

Unpack the downloaded Hadoop distribution. In the distribution, edit the file `etc/hadoop/hadoop-env.sh` to define some parameters as follows:

```
# set to the root of your Java installation
export JAVA_HOME=/usr/java/latest

# Assuming your installation directory is /usr/local/hadoop
export HADOOP_PREFIX=/usr/local/hadoop
```

Try the following command:

```
$ bin/hadoop
```

This will display the usage documentation for the hadoop script.

Now you are ready to start your Hadoop cluster in one of the three supported modes:

- Local (Standalone) Mode
- Pseudo-Distributed Mode
- Fully-Distributed Mode

◆ core-site.xml

```
<configuration>
  <property>
    <name>hadoop.tmp.dir</name>
    <value>/opt/modules/hadoop-2.7.1/data/tmp</value>
  </property>
  <property>
    <name>fs.defaultFS</name>
    <value>hdfs://bigdata-wh01.cloudyhadoop.com:8020</value>
  </property>
</configuration>
```

◆ hdfs-site.xml

```
<configuration>
  <property>
    <name>dfs.namenode.secondary.http-address</name>
    <value>bigdata-wh01.cloudyhadoop.com:50090</value>
  </property>
  <property>
    <name>dfs.replication</name>
    <value>1</value>
  </property>
</configuration>
```

◆ yarn-site.xml

```
<configuration>
  <property>
    <name>yarn.resourcemanager.hostname</name>
    <value>bigdata-wh01.cloudyhadoop.com</value>
  </property>
  <property>
    <name>yarn.nodemanager.aux-services</name>
    <value>mapreduce_shuffle</value>
  </property>
  <property>
    <name>yarn.log-aggregation-enable</name>
    <value>true</value>
  </property>
  <property>
    <name>yarn.log-aggregation.retain-seconds</name>
    <value>604800</value>
  </property>
</configuration>
```

◆ mapred-site.xml

```
<configuration>
  <property>
    <name>mapreduce.framework.name</name>
    <value>yarn</value>
  </property>
  <property>
    <name>mapreduce.jobhistory.address</name>
    <value>bigdata-wh01.cloudyhadoop.com:10020</value>
  </property>
  <property>
    <name>mapreduce.jobhistory.webapp.address</name>
    <value>bigdata-wh01.cloudyhadoop.com:19888</value>
  </property>
</configuration>
```

◆ slaves

```
bigdata-wh01.cloudyhadoop.com
```

1

Linux环境工具准备

2

Hadoop 2.x分布式部署

3

MySQL数据库安装

4

初识Hive之环境搭建



采用yum安装方式安装

```
1. yum install mysql-server
```

判断MySQL是否已经安装好：

```
1. chkconfig --list|grep mysql
```

启动mysql服务：

```
1. service mysqld start
```

或者

```
1. /etc/init.d/mysqld start
```

检查是否启动mysql服务：

```
1. /etc/init.d/mysqld status
```

设置MySQL开机启动：

```
1. chkconfig mysqld on
```

检查设置MySQL开机启动是否配置成功：显示2 3 4 5为on

```
1. chkconfig --list|grep mysql
```

创建root管理员：

```
1. mysqladmin -uroot password root
```

登录

```
1. mysql -uroot -proot
```



```
$ sudo cat /root/.mysql_secret
```

修改 root 用户密码为 123456

```
> SET PASSWORD = PASSWORD('123456');
```

查询用户信息

```
mysql> select User,Host>Password from user;
```

更新用户信息

```
mysql> update user set Host='%' where User = 'root' and Host='localhost';
```

删除用户信息

```
mysql> delete from mysql.user where user='' and host='127.0.0.1';
```

刷新信息

```
mysql> flush privileges;
```

> show variables like "%char%";↵

show variables like "%char%";

> show variables like "max_connections";↵

show variables like "max_connections";

```
$ sudo cp /usr/share/mysql/my-default.cnf /etc/my.cnf↵
```

编辑 my.cnf↵

```
$ sudo vi /etc/my.cnf↵
```

在[mysqld]上面↵

添加[client]，配置 **default-character-set** 值为 utf8，↵

在[mysqld]底部↵

default-character-set=utf8

配置 **character_set_server** 值为 utf8，↵

character_set_client 值为 utf8，↵

max_connections 值为 1000（默认 151）↵

character_set_server=utf8

character_set_client=utf8

max_connections=1000

```
[client]
default-character-set=utf8
```

```
[mysqld]
```

```
# Remove leading # and set to the amount of RAM for the most important data
# cache in MySQL. Start at 70% of total RAM for dedicated server, else 10%.
# innodb_buffer_pool_size = 128M

# Remove leading # to turn on a very important data integrity option: logging
# changes to the binary log between backups.
# log_bin

# These are commonly set, remove the # and set as required.
# basedir = .....
# datadir = .....
# port = .....
# server_id = .....
# socket = .....

# Remove leading # to set options mainly useful for reporting servers.
# The server defaults are faster for transactions and fast SELECTs.
# Adjust sizes as needed, experiment to find the optimal values.
# join_buffer_size = 128M
# sort_buffer_size = 2M
# read_rnd_buffer_size = 2M

sql_mode=NO_ENGINE_SUBSTITUTION,STRICT_TRANS_TABLES
character_set_server=utf8
character_set_client=utf8
max_connections=1000
```

1

Linux环境工具准备

2

Hadoop 2.x分布式部署

3

MySQL数据库安装

4

初识Hive之环境搭建

Requirements

- Java 1.7

Note: Hive versions 1.2 onward require Java 1.7 or newer. Hive versions 0.14 to 1.1 work with Java 1.6 as well. Users are strongly advised to start moving to Java 1.8 (see [HIVE-8607](#)).

- Hadoop 2.x (preferred), 1.x.

Hive versions up to 0.13 also supported Hadoop 0.20.x, 0.23.x.

- Hive is commonly used in production Linux and Windows environment. Mac is a commonly used development environment. The instructions in this document are applicable to Linux and Mac. Using it on Windows would require slightly different steps.

Installing Hive from a Stable Release

Start by downloading the most recent stable release of Hive from one of the Apache download mirrors (see [Hive Releases](#)).

Next you need to unpack the tarball. This will result in the creation of a subdirectory named `hive-x.y.z` (where `x.y.z` is the release number):

```
$ tar -xzvf hive-x.y.z.tar.gz
```

Set the environment variable `HIVE_HOME` to point to the installation directory:

```
$ cd hive-x.y.z  
$ export HIVE_HOME={ {pwd} }
```

Finally, add `$HIVE_HOME/bin` to your `PATH`:

```
$ export PATH=$HIVE_HOME/bin:$PATH
```



Hive uses Hadoop, so:

- you must have Hadoop in your path OR
- `export HADOOP_HOME=<hadoop-install-dir>`

In addition, you must create `/tmp` and `/user/hive/warehouse` (aka `hive.metastore.warehouse.dir`) and set them `chmod g+w` in HDFS before you can create a table in Hive.

Commands to perform this setup:

```
$ $HADOOP_HOME/bin/hadoop fs -mkdir      /tmp
$ $HADOOP_HOME/bin/hadoop fs -mkdir      /user/hive/warehouse
$ $HADOOP_HOME/bin/hadoop fs -chmod g+w  /tmp
$ $HADOOP_HOME/bin/hadoop fs -chmod g+w  /user/hive/warehouse
```

You may find it useful, though it's not necessary, to set `HIVE_HOME`:

```
$ export HIVE_HOME=<hive-install-dir>
```

Running Hive CLI

To use the Hive [command line interface](#) (CLI) from the shell:

```
$ $HIVE_HOME/bin/hive
```


Simple Example Use Cases

create table user

create table user(id int,name string) ROW FORMAT DELIMITED FIELDS TERMINATED BY '\t' ;

load data

load data local inpath '/opt/datas/user.tsv' overwrite into table user ;

query data

select * from user ;

user.tsv

1 Zhangsan

2 Lisi

3 wangwu

- ◆ 云帆大数据是国内首家坚持实时在线授课、提供高端开发课程网络培训机构。采用新兴的互联网教育模式，坚持实时在线授课模式，既继承传统教育的学习交流特点，又发挥互联网的无处不在的时空特性，将天南地北有志向的人才组织在一起学习交流，使原先孤立的个体学习，组合成有组织的学习探讨，并且把原先的学费用降低到十分之一左右，使更多的人能学习到最新的高端课程技术。云帆大数据同时是一个平台，如果你是一个学员，可以尽情的学习和交流；如果你是一个有梦想有才华的人，可以联系我们，给你提供一片驰骋的原野。
- ◆ 云帆大数据已推出国内首家《企业级Hadoop 2.x 应用开发课程》更多其他详情，请登录我们的培训网站<http://www.cloudyhadoop.com>。



实时在线授课，专业课程辅导