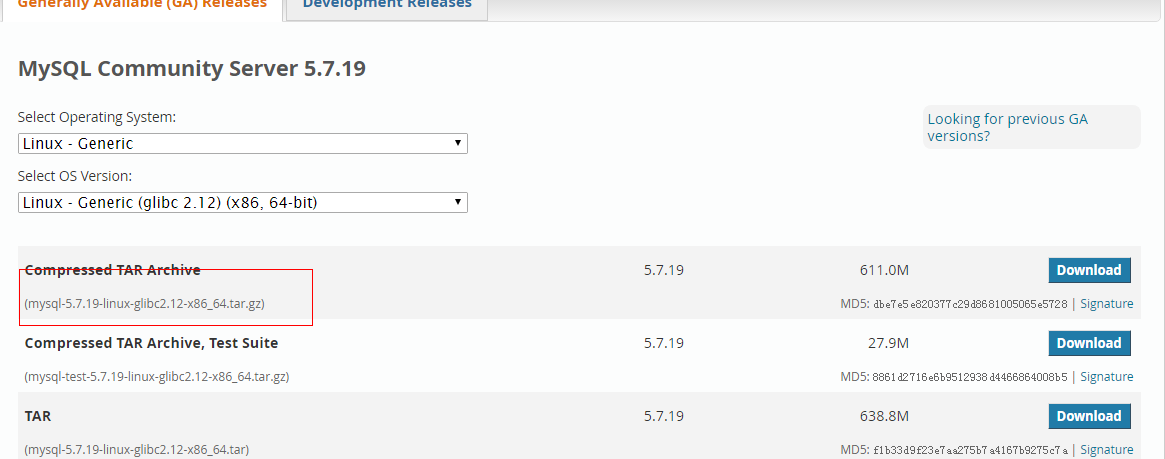
# MySQL环境搭建

## MySQL单机搭建

1. **下载**

下载地址：<https://dev.mysql.com/downloads/mysql/#downloads>



1. **解压**

**tar xvf mysql-5.7.19-linux-glibc2.12-x86\_64.tar**

**mv mysql-5.7.19-linux-glibc2.12-x86\_64 mysql**

1. **添加系统mysql组和mysql用户**

**groupadd mysql  
useradd -r -g mysql -s /bin/false mysql**

1. **进入安装目录且修改拥有者**

**chown -R mysql:mysql ./**

1. **安装**

**./scripts/mysql\_install\_db --user=mysql**

如果mysql的安装目录（解压目录）不是/usr/local/mysql，那么还必须指定目录参数

**./scripts/mysql\_install\_db --user=mysql \**

**--basedir=/opt/mysql/mysql \**

**--datadir=/opt/mysql/mysql/data**



以上错误解决办法: yum install -y perl-Data-Dumper

1. **修改当前目录拥有者为root用户, data目录拥有者为mysql用户**

**chown -R root:root ./**

**chown -R mysql:mysql data**

1. **添加mysql服务开机自启动**

**cp support-files/mysql.server /etc/init.d/mysql**

# 赋予可执行权限

**chmod +x /etc/init.d/mysql**

# 添加服务

**chkconfig --add mysql**

1. **启动MySQL服务**

**service mysql start**

#关闭mysql

**service mysql stop**

#查看运行状态

**service mysql status**

*#加入环境变量，编辑 /etc/profile，这样可以在任何地方用mysql命令了*

***export PATH=$PATH:/usr/local/mysql//bin***

***source /etc/profile***

## MySQL集群搭建

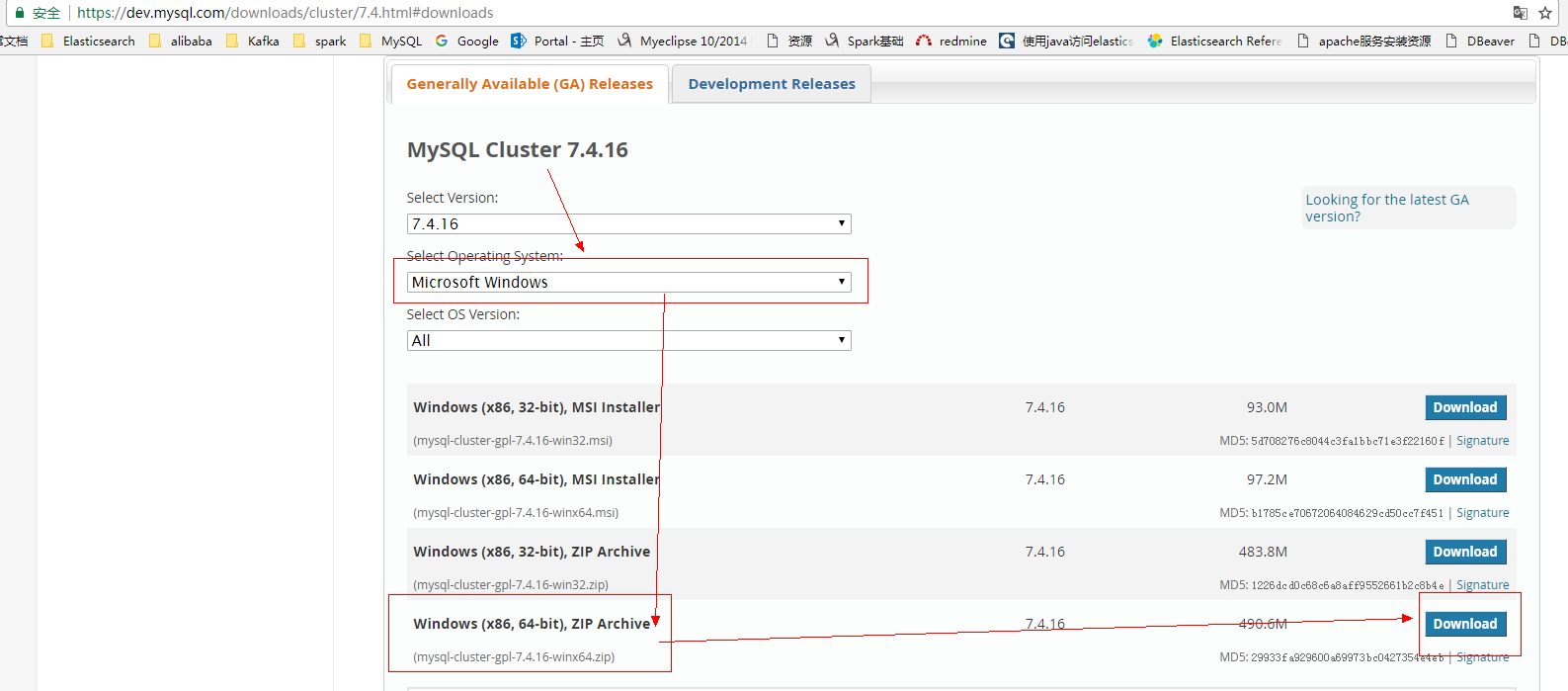
1. **官网下载MySQL-Cluster**

<https://dev.mysql.com/downloads/cluster/>

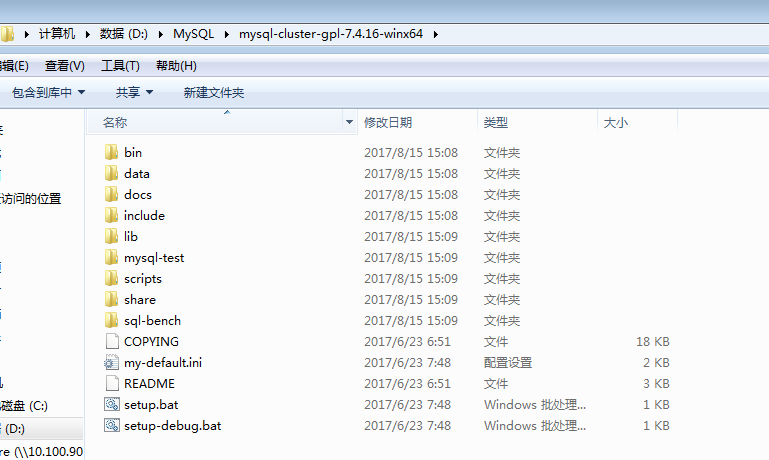
对应版本MySQL-Cluster v7.3.X —> MySQL 5.5.X,MySQL-Cluster v7.4.X —> MySQL 5.6.X,

MySQL-Cluster v7.5.X —> MySQL 5.7.X,

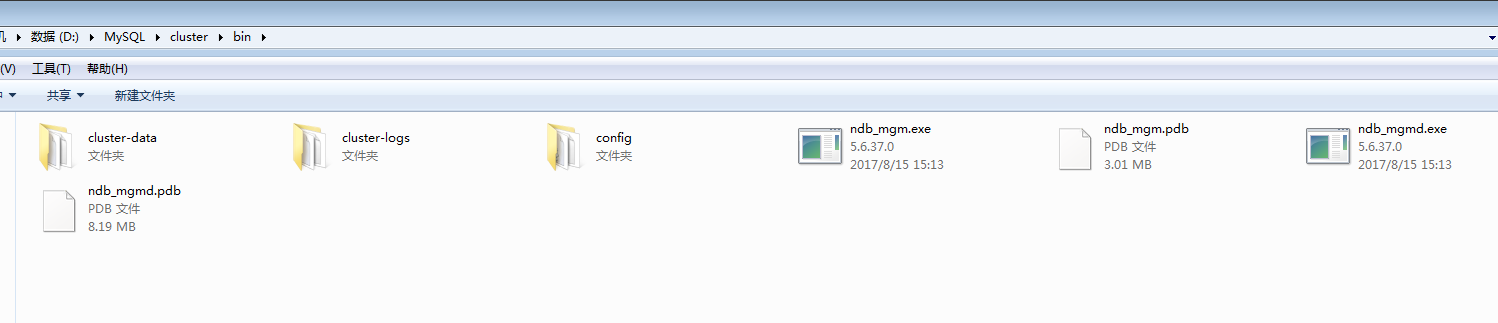
以MySQL Cluster 7.4.16(对应MySQL 5.6.37)为例



1. **解压文件**



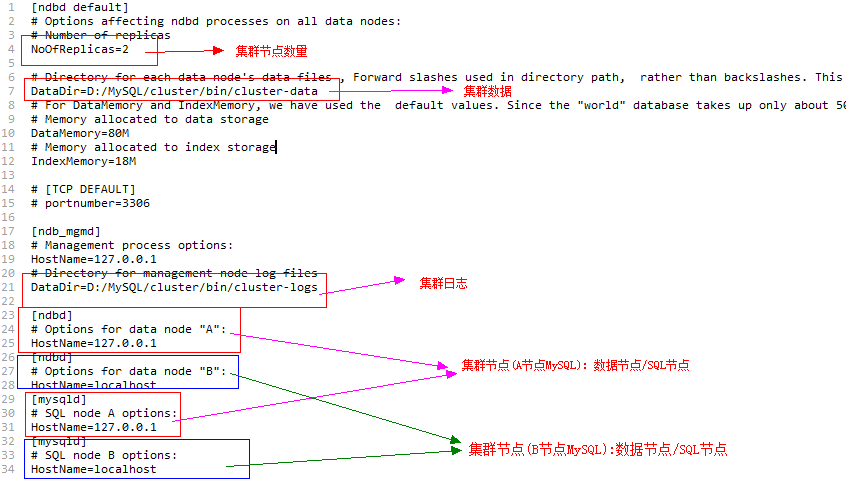
1. **配置管理点**



建立集群管理点，创建bin目标并在其下建立config、cluster-data、cluster-logs目录

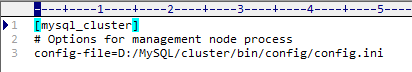
将ndb\_mgm.exe、ndb\_mgmd.exe移至bin目录，并在config建立配置文件config.ini/my.ini

1. **config.ini**



|  |
| --- |
| [ndbd default]  # Options affecting ndbd processes on all data nodes:  # Number of replicas  NoOfReplicas=2  # Directory for each data node's data files , Forward slashes used in directory path, rather than backslashes. This is correct; see Important note in text  DataDir=D:/MySQL/cluster/bin/cluster-data  # For DataMemory and IndexMemory, we have used the default values. Since the "world" database takes up only about 500KB, this should be more than enough for this example Cluster setup.  # Memory allocated to data storage  DataMemory=80M  # Memory allocated to index storage  IndexMemory=18M  # [TCP DEFAULT]  # portnumber=3306  [ndb\_mgmd]  # Management process options:  HostName=127.0.0.1  # Directory for management node log files  DataDir=D:/MySQL/cluster/bin/cluster-logs  [ndbd]  # Options for data node "A":  HostName=127.0.0.1  [ndbd]  # Options for data node "B":  HostName=localhost  [mysqld]  # SQL node A options:  HostName=127.0.0.1  [mysqld]  # SQL node B options:  HostName=localhost |

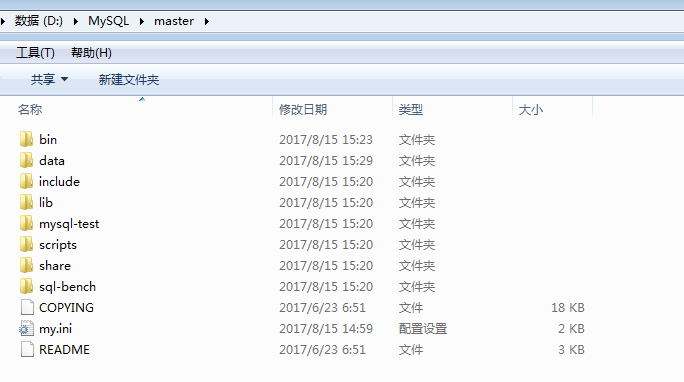
1. **my.ini**

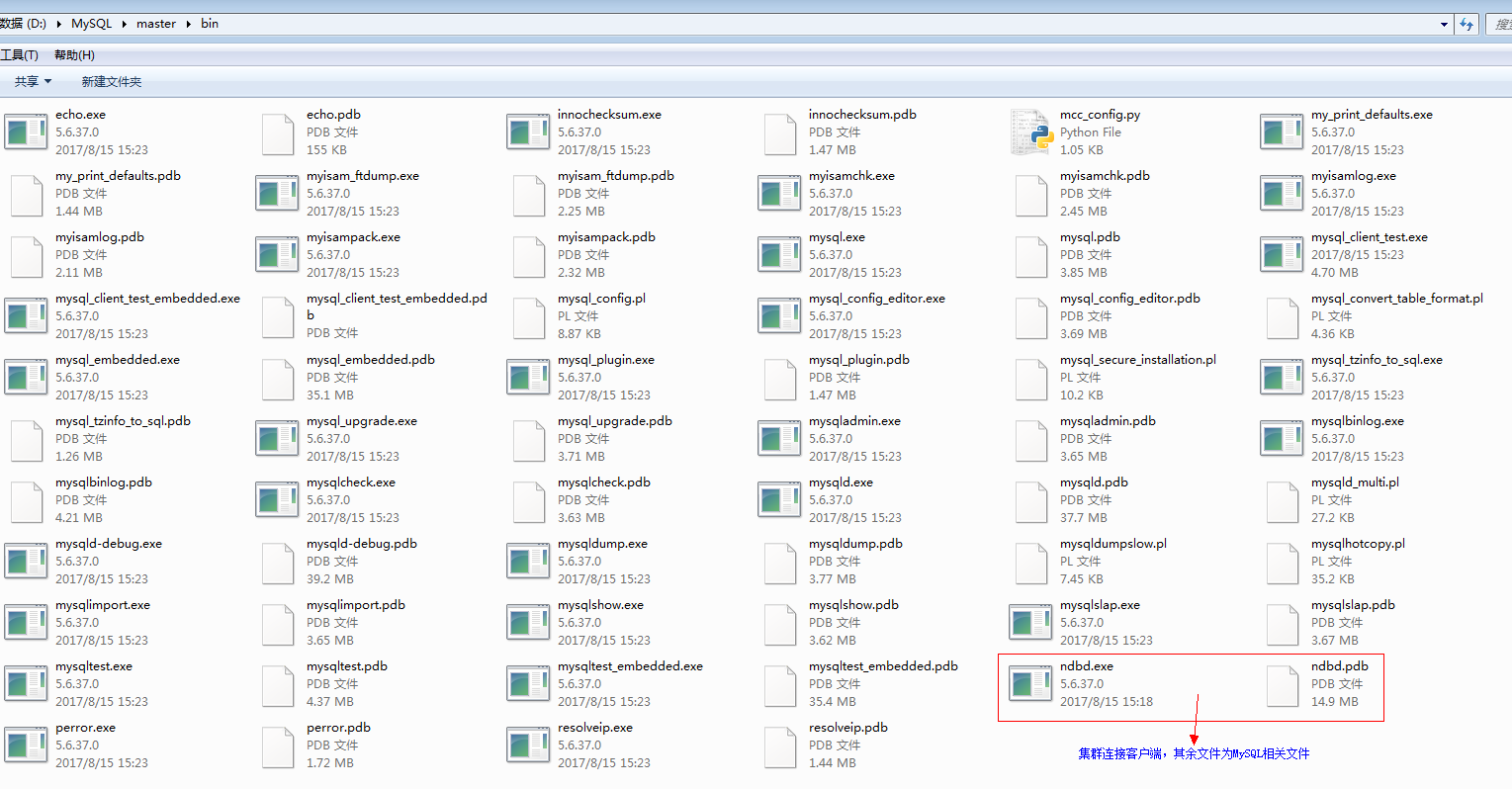


|  |
| --- |
| [mysql\_cluster]  # Options for management node process  config-file=D:/MySQL/cluster/bin/config/config.ini |

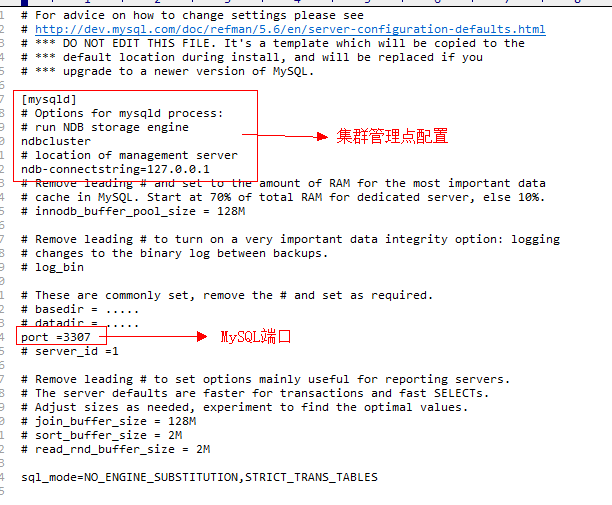
1. **配置数据点**

MySQL的A/B节点可以单独安装部署也可以使用解压包中的MySQL相关文件：



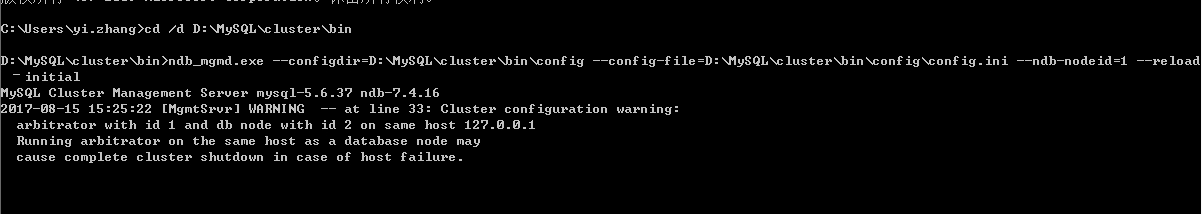


1. 复制解压包中的ndbd.exe集群客户端至MySQL安装目录bin下以便数据节点连接
2. 配置MySQL的my.ini文件(分别配置A/B节点)



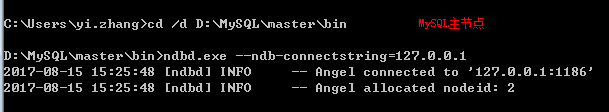
|  |
| --- |
| [mysqld]  # Options for mysqld process:  # run NDB storage engine  ndbcluster  # location of management server  ndb-connectstring=127.0.0.1 |

1. **启动集群**
2. **启动管理点**



|  |
| --- |
| cd /d D:\MySQL\cluster\bin  ndb\_mgmd.exe --configdir=D:\MySQL\cluster\bin\config --config-file=D:\MySQL\cluster\bin\config\config.ini --ndb-nodeid=1 --reload –initial |

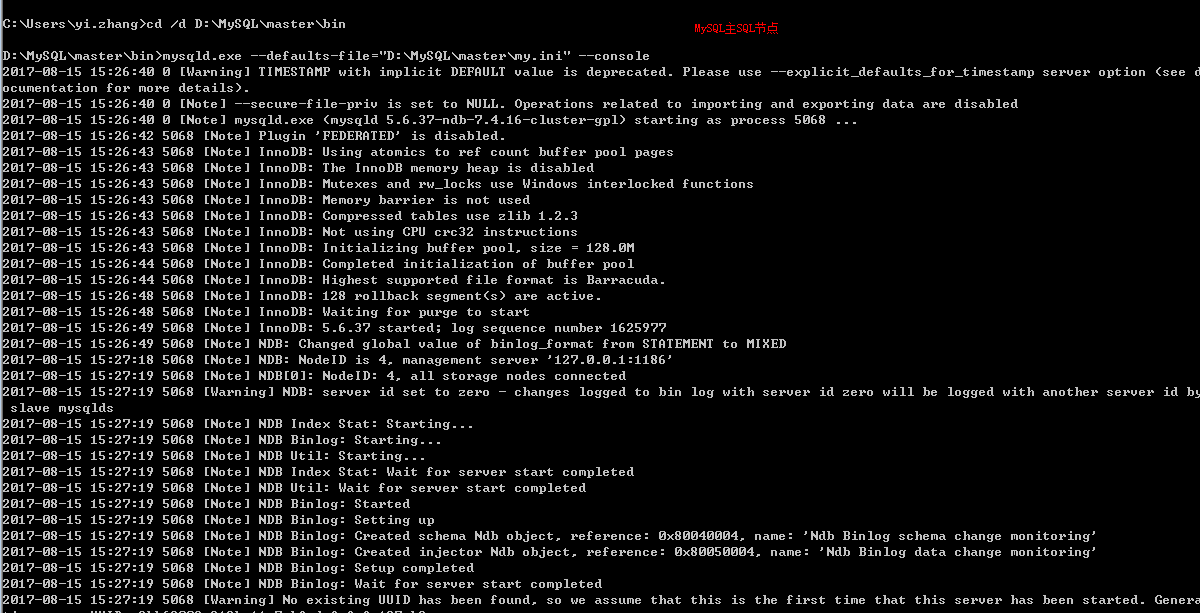
1. **启动A/B数据节点**(IP地址为管理点地址)

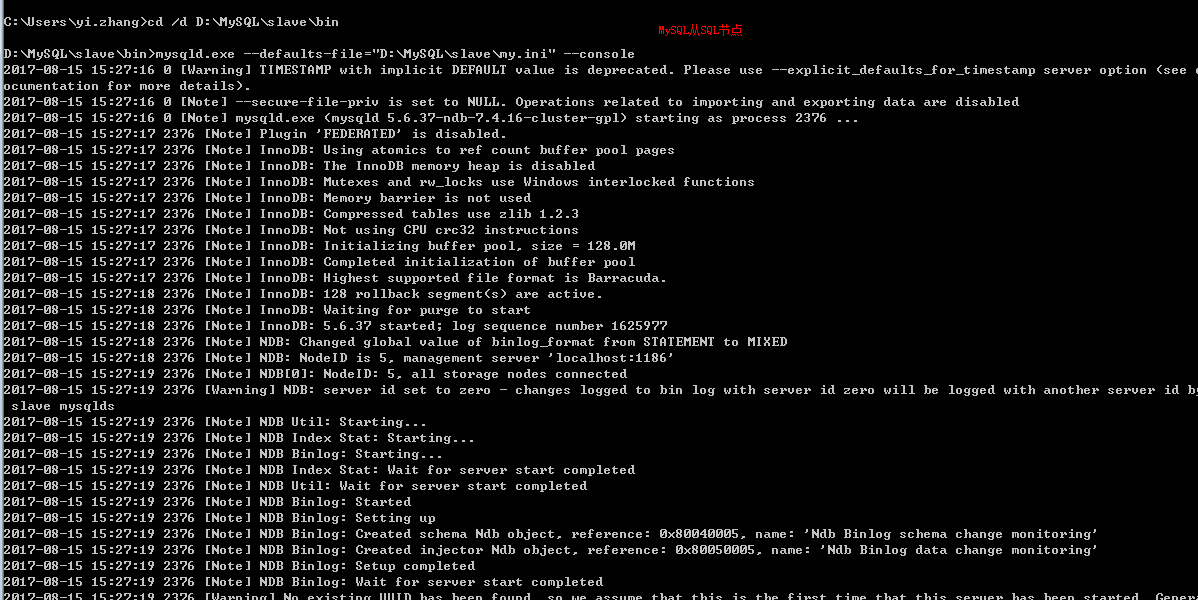




|  |
| --- |
| 主节点：  cd /d D:\MySQL\master\bin  ndbd.exe --ndb-connectstring=127.0.0.1  从节点：  cd /d D:\MySQL\slave\bin  ndbd.exe --ndb-connectstring=127.0.0.1 |

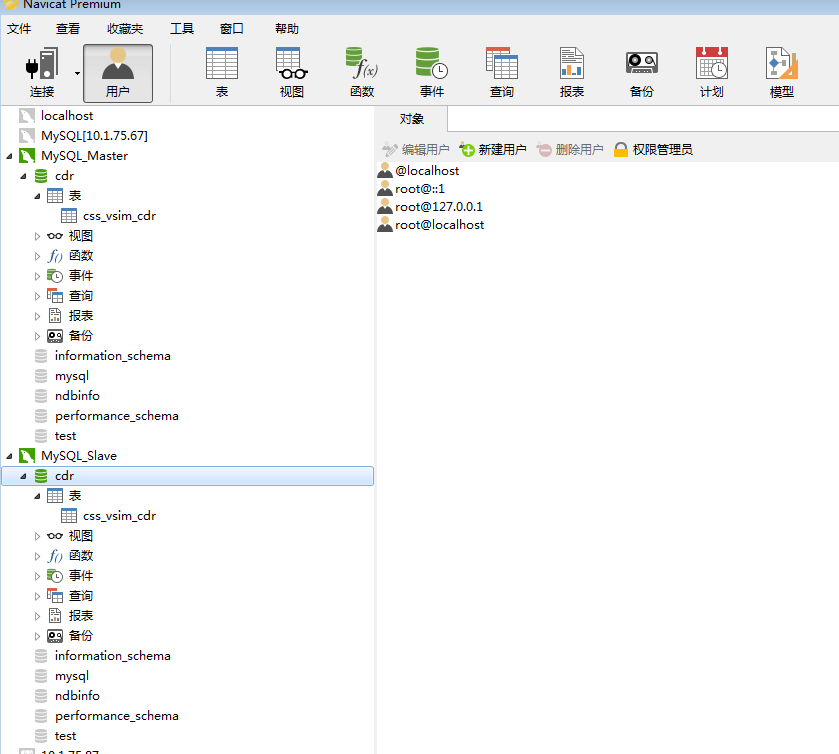
1. **启动A/B SQL节点**



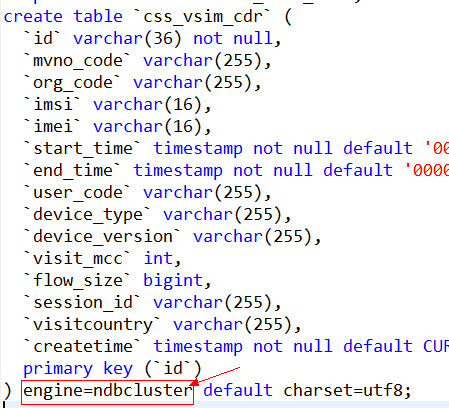


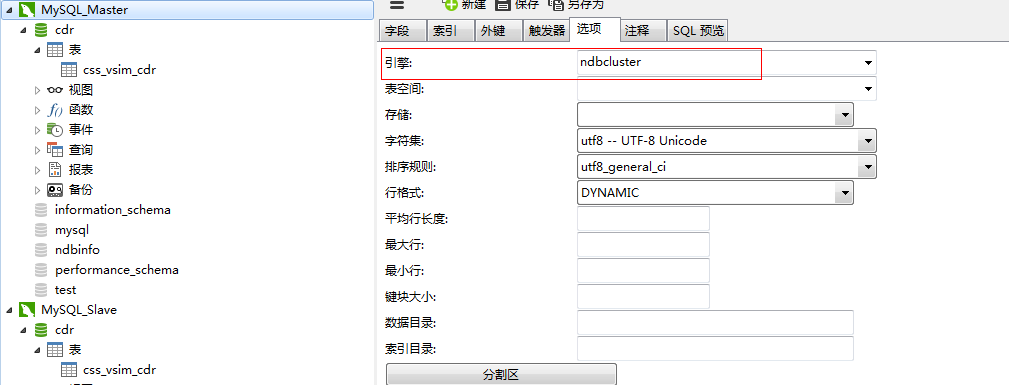
|  |
| --- |
| 主节点：  mysqld.exe --defaults-file="D:\MySQL\master\my.ini" –console  从节点：  mysqld.exe --defaults-file="D:\MySQL\slave\my.ini" --console |

1. **客户端连接操作**
2. 修改root密码(默认密码为空)

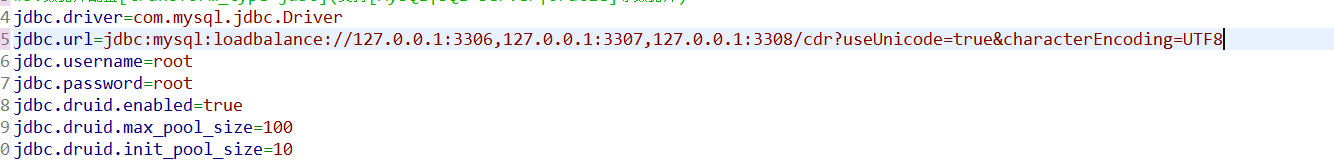


1. 创建数据库
2. 创建数据表(表引擎必须为: engine=ndbcluster)





1. JDBC连接



|  |
| --- |
| jdbc:mysql:loadbalance://127.0.0.1:3306,127.0.0.1:3307,127.0.0.1:3308/cdr?useUnicode=true&characterEncoding=UTF8 |

## MySQL主从搭建

主服务器IP：192.168.1.11

从服务器IP：192.168.1.12

192.168.1.11主操作：

1、关闭防火墙、selinux

Serviceiptables  stop

Setenforce  0

2、vim /etc/my.cnf

在[mysqld]下添加：

server\_id = 1

log-bin=mysql-bin

binlog\_format=mixed

log-slave-updates=true

3、连接到数据库，创建一个有复制权限的slave用户test进行远程连接slave

grant  replication slave on \*.\* to‘test’@’192.168.1.%’ identified by ‘123’;

4、刷新授权

flush  privileges;

5、重启mysql服务

Service  mysqld restart

6、查看master的状态信息

mysql> showmaster status;

+------------------+----------+--------------+------------------+-------------------+

| File             | Position | Binlog\_Do\_DB |Binlog\_Ignore\_DB |

+------------------+----------+--------------+------------------+-------------------+

| mysql-bin.000001 |      120 |              |                  |                   |

+------------------+----------+--------------+------------------+-------------------+

1 row in set (0.00sec)

192.168.1.12从操作：

1、关闭防火墙、selinux

Serviceiptables  stop

Setenforce  0

2、vim /etc/my.cnf

在[mysqld]下添加：

server-id  =  2

relay-log=relay-log-bin

relay-log-index=slave-relay-bin.index

3、删除/usr/local/mysql/data/auto.cnf的uuid

[auto]

server-uuid=fe4179ff-ca1c-11e7-be22-000c29766e1e

4、重启mysql服务

Service  mysqld restart

[root@rooo ~]#service mysqld restart

Shutting downMySQL.                                      [  OK  ]

StartingMySQL.                                                  [  OK  ]

5、配置slave连接master

change mastertomaster\_host='192.168.1.11',master\_user='test',master\_password='123..com',master\_log\_file='mysql-bin-000001',master\_log\_pos=log\_pos=120;

6、开启slave

Resetslave

Startslave

7、查看slave状态

Show   slave status;

 Slave\_IO\_Running: Yes

 Slave\_SQL\_Running: Yes