

本节所讲内容：

- MySQL 集群 概述
- 实战：MySQL 集群 搭建

服务端：xuegod63.cn IP：192.168.1.63

客户端：xuegod64.cn IP：192.168.1.64

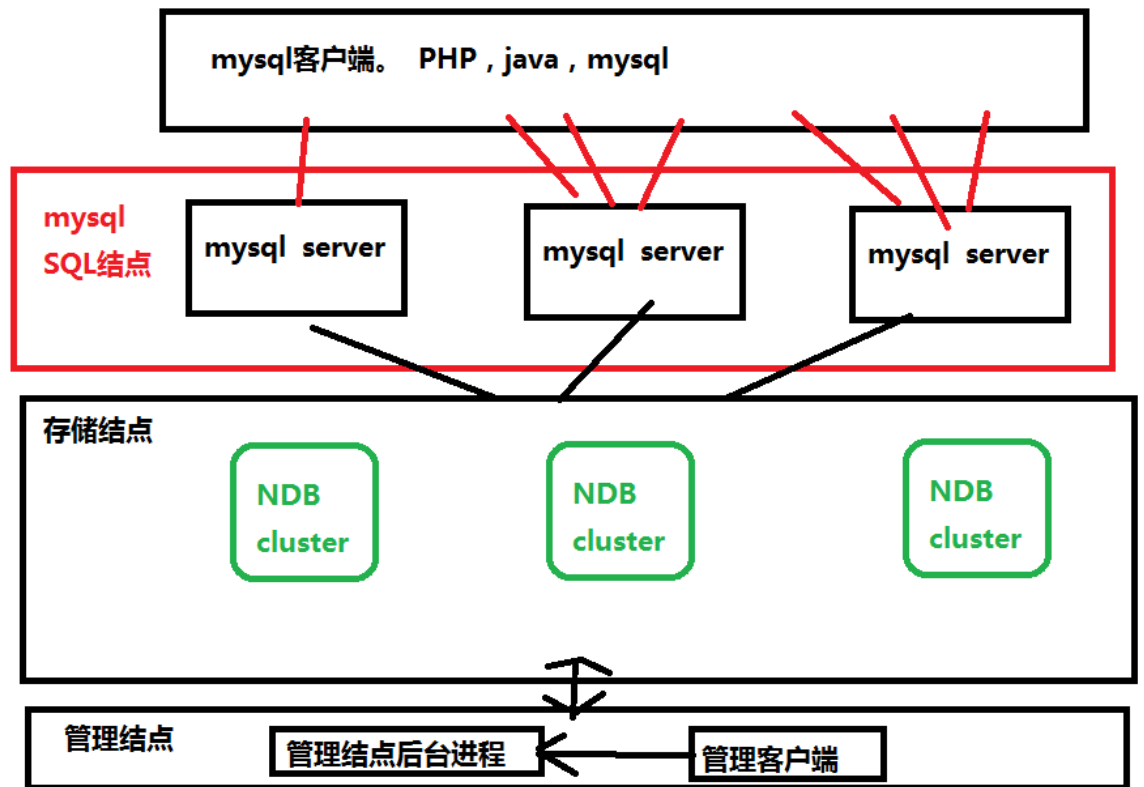
<http://www.mysql.com/products/cluster/features.html>

MySQL Cluster(MySQL 集群) 概述：

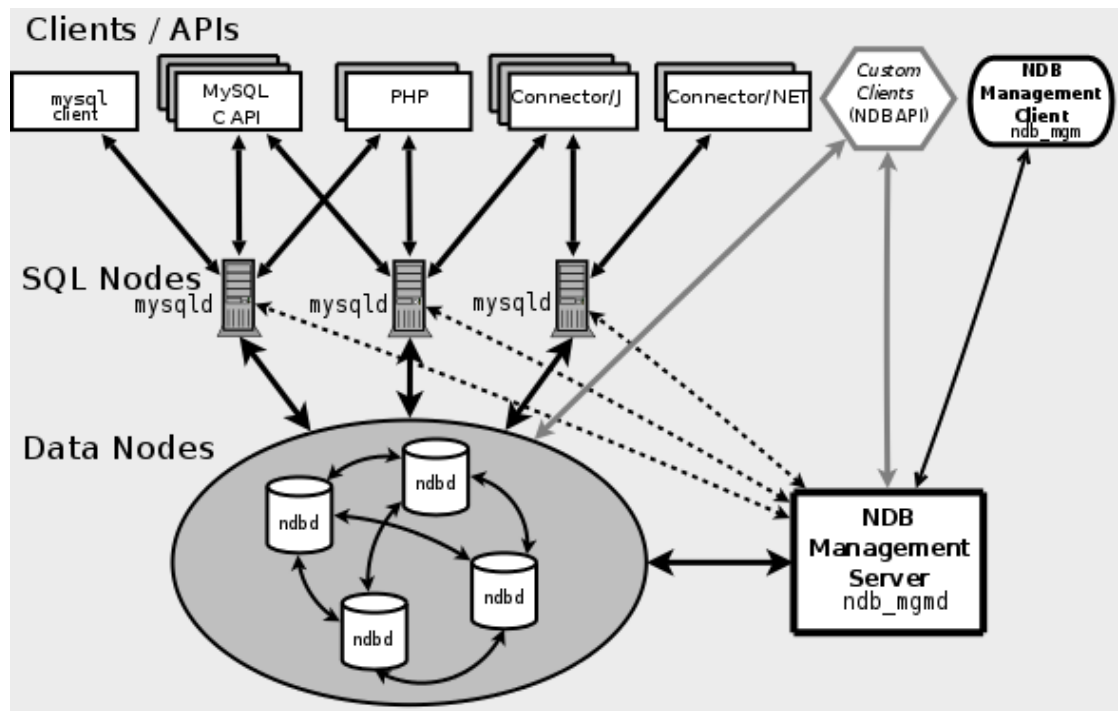
MySQL Cluster 是 MySQL 适合于分布式计算环境的高实用、高冗余版本。它采用了 NDB Cluster 存储引擎，允许在 1 个 Cluster 中运行多个 MySQL 服务器。在 MySQL 5.0 及以上的二进制版本中、以及最新的 Linux 版本兼容的 RPM 中提供了该存储引擎。

MySQL Cluster 是一种技术，该技术允许在无共享的系统中部署“内存中”数据库的 Cluster。通过无共享体系结构，系统能够使用廉价的硬件，而且对软硬件无特殊要求。此外，由于**每个组件有自己的内存和磁盘，不存在单点故障。**

MySQL Cluster 由一组计算机构成，每台计算机上均运行着多种进程，包括 MySQL 服务器，NDB Cluster 的数据节点，管理服务器，以及(可能)专门的数据访问程序。关于 Cluster 中这些组件的关系，请参见下图：



详细运行机制图：



SQL 节点： 给上层应用层提供 sql 访问。

NDB 引擎： 是一种 “内存中” 的存储引擎，它具有可用性高和数据一致性好的特点
存储/数据节点： 保存 cluster 中的数据。 数据节点，可以提供副本。实现数据冗余。
管理节点 (MGM)： 管理整个集群。 启动，关闭集群。 通过 `ndb_mgmd` 启动集群
工作原理：

所有的这些节点构成一个完成的 MySQL 集群体系。数据保存在 “NDB 存储服务器” 的存储引擎中，表 (结构) 则保存在 SQL 节点即 “MySQL 服务器” 中。应用程序通过 “MySQL 服务器” 访问这些数据表，集群管理服务器通过管理工具(`ndb_mgmd`)来管理 “NDB 存储服务器”。

二、NDB 概述：

“NDB” 是一种 “内存中” 的存储引擎，它具有可用性高和数据一致性好的特点。

NDB 引擎

MySQL Cluster 使用了一个专用的基于内存的存储引擎——NDB 引擎，这样做的好处是速度快，没有磁盘 I/O 的瓶颈，但是由于是基于内存的，所以数据库的规模受系统总内存的限制，如果运行 NDB 的 MySQL 服务器一定要内存够大，比如 4G，8G，甚至 16G。NDB 引擎是分布式的，它可以配置在多台服务器上来实现数据的可靠性和扩展性，理论上通过配置 2 台 NDB 的存储节点就能实现整个数据库集群的冗余性和解决单点故障问题。

缺陷

- 基于内存，数据库的规模受集群总内存的大小限制
- 基于内存，断电后数据可能会有数据丢失，这点还需要通过测试验证。
- 多个节点通过网络实现通讯和数据同步、查询等操作，因此整体性受网络速度影响，因此速度也比较慢

2.2 优点

- 多个节点之间可以分布在不同的地理位置，因此也是一个实现分布式数据库的方案。
- 扩展性很好，增加节点即可实现数据库集群的扩展。
- 冗余性很好，多个节点上都有完整的数据库数据，因此任何一个节点宕机都不会造成服务中断。

MySQL Cluster 能够使用多种故障切换和负载均衡选项配置 NDB 存储引擎，但在 Cluster 级别上的存储引擎上做这个最简单。MySQL Cluster 的 NDB 存储引擎包含完整的数据集，仅取决于 Cluster 本身内的其他数据。

目前，MySQL Cluster 的 Cluster 部分可独立于 MySQL 服务器进行配置。在 MySQL Cluster 中，Cluster 的每个部分被视为 1 个节点。

- **管理(MGM)节点**：这类节点的作用是管理 MySQL Cluster 内的其他节点，如提供配置数据、启动并停止节点、运行备份等。由于这类节点负责管理其他节点的配置，应在启动其他节点之前首先启动这类节点。MGM 节点是用命令“ndb_mgmd”启动的。
- **数据节点** 这类节点用于保存 Cluster 的数据。数据节点的数目与副本的数目相关，是片段的倍数。例如，对于两个副本，每个副本有两个片段，那么就有 4 个数据节点。不过没有必要设置多个副本。数据节点是用命令“ndbd”启动的。
- **SQL 节点**：这是用来访问 Cluster 数据的节点。对于 MySQL Cluster，客户端节点是使用 NDB Cluster 存储引擎的传统 MySQL 服务器。通常，SQL 节点是使用命令“mysqld --ndbcluster”启动的，或将“ndbcluster”添加到“my.cnf”后使用“mysqld”启动。

下载地址：

<http://dev.mysql.com/downloads/cluster/>
切换成 RHEL 后，等待显示 Linux 对应的软件包

• [Generally Available \(GA\) Releases](#)

MySQL Cluster 7.4.6

Select Platform:

Red Hat Enterprise Linux / Oracle Linux

[Looking for previous versions?](#)

Windows (x86, 32-bit), MSI Installer (mysql-cluster-gpl-7.4.6-win32.msi)	7.4.6	109.0M	Download
MD5: 0ad9963c5eec4714130645Ee043ec65 Signature			
Windows (x86, 64-bit), MSI Installer (mysql-cluster-gpl-7.4.6-winx64.msi)	7.4.6	114.1M	Download
MD5: bf8c00640998bfe0b9e3ba09e99db7c Signature			
Windows (x86, 32-bit), ZIP Archive (mysql-cluster-gpl-7.4.6-win32.zip)	7.4.6	506.6M	Download
MD5: 1b7f931858641b2659da89095670f1b4 Signature			
Windows (x86, 64-bit), ZIP Archive (mysql-cluster-gpl-7.4.6-winx64.zip)	7.4.6	514.8M	Download
MD5: 419a8f3c321488d34b44d6920ea52ced Signature			

xuegod63 和 xuegod64 是服务器

将 MySQL-Cluster-gpl-7.3.7-1.el6.x86_64.rpm-bundle.tar 上传到 xuegod63 和 xuegod64 上。

mysql cluster 集群各机器角色：

mysql 管理结点：xuegod63.cn IP：192.168.1.63

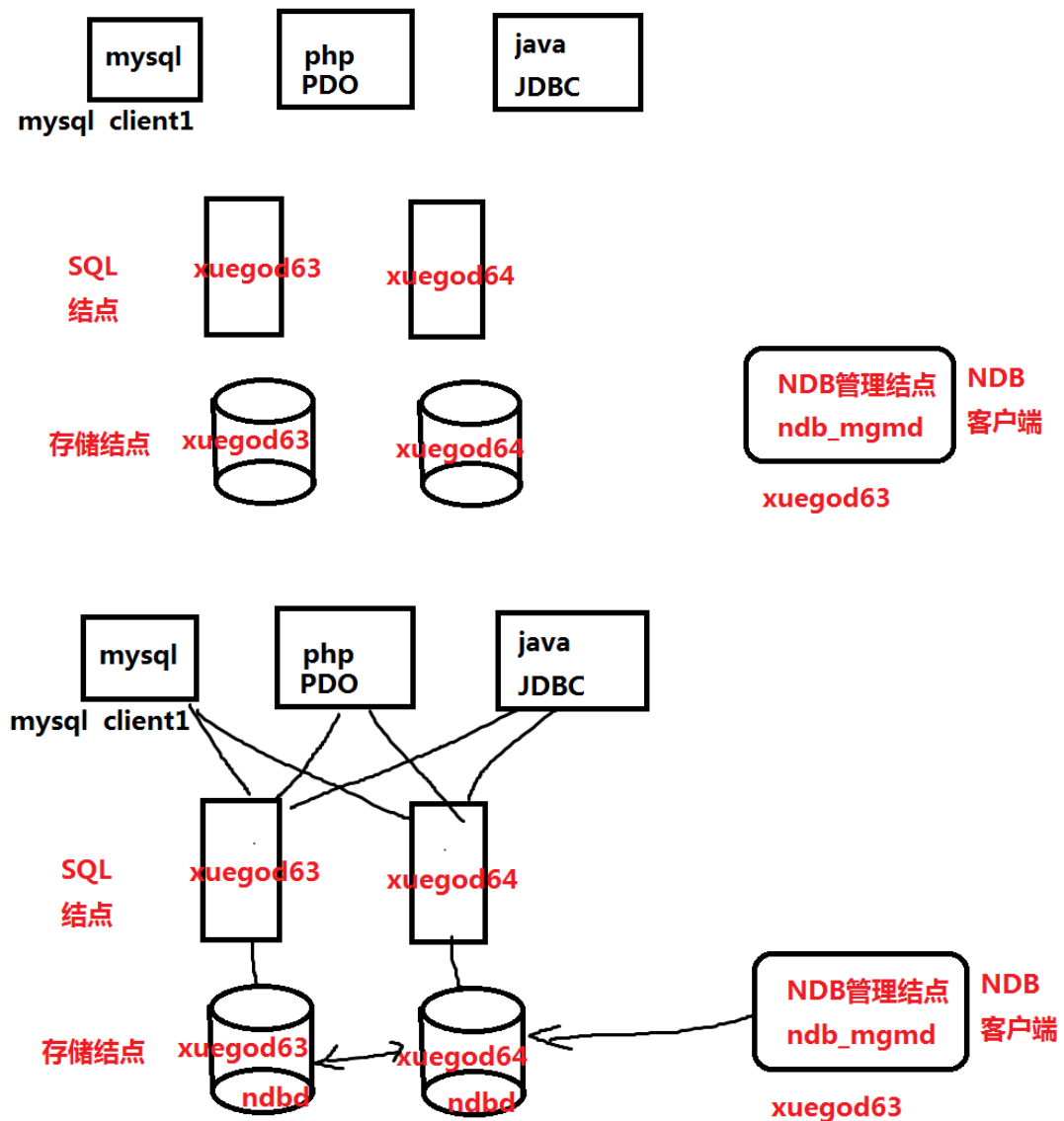
mysql 数据结点：xuegod63.cn IP：192.168.1.63

mysql 数据结点：xuegod64.cn IP：192.168.1.64

mysql SQL 结点：xuegod63.cn IP：192.168.1.63

mysql SQL 结点：xuegod64.cn IP：192.168.1.64

如图：



实战 1：MySQL 集群 搭建

二、环境清理以及安装：

xuegod63 上执行：

1. mysql 旧版本清除：

首先使用如下命令来清理之前操作系统自带的 mysql 安装：`yum -y remove mysql`

然后使用如下命令：

`rpm -qa | grep mysql*`

对于找到的 2 个剩余 mysql 包，请按照如下的命令格式予以删除：

`rpm -e --nodeps mysql-libs-5.1.71-1.el6.x86_64`

2. mysql cluster 版本安装准备：

解压，得到 mysql-Cluster 对应的软件包：

```
[root@xuegod63 ~]# tar -xvf MySQL-Cluster-gpl-7.3.7-1.el6.x86_64.rpm-bundle.tar
```

得到如下文件清单：

```
MySQL-Cluster-client-gpl-7.3.7-1.el6.x86_64.rpm
MySQL-Cluster-devel-gpl-7.3.7-1.el6.x86_64.rpm
MySQL-Cluster-embedded-gpl-7.3.7-1.el6.x86_64.rpm
MySQL-Cluster-gpl-7.3.7-1.el6.x86_64.rpm-bundle.tar
MySQL-Cluster-server-gpl-7.3.7-1.el6.x86_64.rpm
MySQL-Cluster-shared-compat-gpl-7.3.7-1.el6.x86_64.rpm
MySQL-Cluster-shared-gpl-7.3.7-1.el6.x86_64.rpm
MySQL-Cluster-test-gpl-7.3.7-1.el6.x86_64.rpm
```

3. mysql cluster 版本安装：

需要创建的文件件：

创建文件夹（分如下 3 个类来创建对应的文件夹）

数据结点存放数据，：mkdir /var/lib/mysql

管理节点：mkdir /var/lib/mysql-cluster SQL 节点：可不用 文件夹授权

进程 DIR: mkdir /var/run/mysqld

```
[root@xuegod63 ~]# rpm -ivh MySQL-Cluster-server-gpl-7.3.7-1.el6.x86_64.rpm
```

注：mysql 集群中，每个节点上都需要安装这个包，不同的角色是通过配置来区别。

特别注意，当安装完毕 MySQL-Cluster-server-gpl 包后，将出现如下提示信息，提醒我们整个 cluster 安装后的初次超级账户密码存在 /root/.mysql_secret 这个文件当中。

```
-----
-----
A RANDOM PASSWORD HAS BEEN SET FOR THE MySQL root USER !
You will find that password in '/root/.mysql_secret'.
You must change that password on your first connect,
no other statement but 'SET PASSWORD' will be accepted.
See the manual for the semantics of the 'password expired' flag.
Also, the account for the anonymous user has been removed.
In addition, you can run:
  /usr/bin/mysql_secure_installation
which will also give you the option of removing the test database.
This is strongly recommended for production servers.
-----
-----
```

安装集群客户端：

```
[root@xuegod63 ~]# rpm -ivh MySQL-Cluster-client-gpl-7.3.7-1.el6.x86_64.rpm
[root@xuegod63 ~]# rpm -qpl MySQL-Cluster-client-gpl-7.3.7-1.el6.x86_64.rpm | more
#查看生成的文件，发现 MySQL-Cluster-client 和 /mnt/Packages/mysql.xx.rpm 作用差不多
/usr/bin/memclient
```

```
/usr/bin/msql2mysql
/usr/bin/mysql
/usr/bin/mysql_config_editor
/usr/bin/mysql_find_rows
/usr/bin/mysql_waitpid
/usr/bin/mysqlaccess
/usr/bin/mysqlaccess.conf
/usr/bin/mysqladmin
/usr/bin/mysqlbinlog
/usr/bin/mysqlcheck
/usr/bin/mysqldump
...
```

使用如下的命令来变更权限保证可写入：

```
[root@xuegod63 ~]# id mysql
uid=496(mysql) gid=493(mysql) 组=493(mysql)
[root@xuegod63 ~]# chown mysql:mysql -R /var/lib/mysql
[root@xuegod63 ~]# chown mysql:mysql -R /var/lib/mysql-cluster/
[root@xuegod63 ~]# chown mysql:mysql -R /var/run/mysqld/
```

xuegod64 清除旧版本 mysql：

1. mysql 旧版本清除：

首先使用如下命令来清理之前操作系统自带的 mysql 安装：

```
[root@xuegod64 ~]# yum remove mysql -y
[root@xuegod63 ~]# rpm -qa | grep mysql*
mysql-libs-5.1.52-1.el6_0.1.x86_64
```

对于找到的 1 个剩余 mysql 包，请按照如下的命令格式予以删除：

```
[root@xuegod63 ~]# rpm -e --nodeps mysql-libs
[root@xuegod63 ~]# rpm -qa | grep mysql*
```

2. 复制软件包 MySQL-Cluster-server 和 MySQL-Cluster-client 两个包到 xuegod64 上：

```
[root@xuegod63 ~]# scp MySQL-Cluster-server-gpl-7.3.7-1.el6.x86_64.rpm
MySQL-Cluster-client-gpl-7.3.7-1.el6.x86_64.rpm 192.168.1.64:/root/
```

切换到 64 服务器上去再做下面的操作

xuegod64 安装 mysql 集群相关的软件包：

```
[root@xuegod64 ~]# rpm -ivh MySQL-Cluster-server-gpl-7.3.7-1.el6.x86_64.rpm
```

注：

数据结点存放数据：/var/lib/mysql 会自动创建。

创建文件夹：

存放进程 PID 文件: `mkdir /var/run/mysqld`

```
[root@xuegod64 ~]# id mysql
```

```
uid=496(mysql) gid=493(mysql) 组=493(mysql)
```

```
[root@xuegod64 ~]# mkdir /var/run/mysqld （已经创建）
```

```
[root@xuegod64 ~]# chown mysql:mysql -R /var/run/mysqld
```

注意，当安装完毕 MySQL-Cluster-server-gpl 包后，将出现如下提示信息，提醒我们整个 cluster 安装后的初次超级账户密码存在 `/root/.mysql_secret` 这个文件当中。

A RANDOM PASSWORD HAS BEEN SET FOR THE MySQL root USER !

You will find that password in '/root/.mysql_secret'.

You must change that password on your first connect,

no other statement but 'SET PASSWORD' will be accepted.

See the manual for the semantics of the 'password expired' flag.

Also, the account for the anonymous user has been removed.

In addition, you can run:

```
/usr/bin/mysql_secure_installation
```

which will also give you the option of removing the test database.

This is strongly recommended for production servers.

查看 mysql root 用户密码：

```
[root@xuegod64 ~]# cat /root/.mysql_secret
```

```
# The random password set for the root user at Thu Aug 11 08:24:53 2016 (local time):
```

```
YRzWkQNK0j2lrlmT （密码生成）
```

上面已经安装好软件包，下面开始配置：

配置 xuegod63 成为管理结点：

实际上只需要两个命令文件：`ndb_mgmd` 和 `ndb_mgm`

```
[root@xuegod63 ~]# which ndb_mgm
```

```
/usr/bin/ndb_mgm
```

```
[root@xuegod63 ~]# rpm -qf /usr/bin/ndb_mgm
```

```
MySQL-Cluster-server-gpl-7.3.7-1.el6.x86_64
```

```
[root@xuegod63 ~]# which ndb_mgmd
```

```
/usr/sbin/ndb_mgmd
```

```
[root@xuegod63 ~]# rpm -qf /usr/sbin/ndb_mgmd
```

```
MySQL-Cluster-server-gpl-7.3.7-1.el6.x86_64
```

还会生成用户：`mysql`

```
[root@xuegod63 ~]# id mysql
```

```
uid=496(mysql) gid=493(mysql) 组=493(mysql)
```

xuegod63：

创建管理结点上配置文件：

管理节点目录：

```
[root@xuegod63 ~]# ls /var/lib/mysql-cluster #创建此目录,为了存放管理结点的配置文件,
只需要在管理结点创建。
```

```
[root@xuegod63 ~]# cd /var/lib/mysql-cluster
```

```
[root@xuegod63 ~]# vi config.ini #手创建文件: config.ini,默认是没有的。并写入以下内容
```

```
[ndbd default]
```

```
NoOfReplicas=2
```

```
DataMemory=200M
```

```
IndexMemory=100M
```

```
[ndb_mgmd]
```

```
id=1
```

```
datadir=/var/lib/mysql-cluster
```

```
HostName=192.168.1.63
```

```
##### data node options:
```

```
[ndbd]
```

```
HostName=192.168.1.63
```

```
DataDir=/var/lib/mysql
```

```
id=2
```

```
[ndbd]
```

```
HostName=192.168.1.64
```

```
DataDir=/var/lib/mysql
```

```
id=3
```

```
# SQL node options:
```

```
[mysqld]
```

```
HostName=192.168.1.63
```

```
id=4
```

```
[mysqld]
```

```
HostName=192.168.1.64
```

```
id=5
```

参数说明：

在这个文件里,我们分别给五个节点分配了 ID,这有利于更好的管理和区分各个节点。当然,要是不指定,MySQL 也会动态分配一个

```
cat /var/lib/mysql-cluster/config.ini
```

[ndbd default]

NoOfReplicas=2 #数据写入数量。2 表示两份，这样当其中一个存储节点坏了，另一个还可以正常使用。

DataMemory=200M #配置数据存储可以使用的内存

IndexMemory=100M #配置缓存索引可以使用的内存

[ndb_mgmd]

id=1

datadir=/var/lib/mysql-cluster # 指定存放管理节点的日志

HostName=192.168.1.63 #管理节点的 IP 地址。本机 IP

data node options: #存储结点

[ndbd]

HostName=192.168.1.63

DataDir=/var/lib/mysql #mysql 数据存储路径

id=2

[ndbd]

HostName=192.168.1.64

DataDir=/var/lib/mysql

id=3

SQL node options: #关于 SQL 结点

[mysqld]

HostName=192.168.1.63 # 指定可以连接 ndbd 引擎的 sql 结点。

id=4

[mysqld]

HostName=192.168.1.64

id=5

到此，管理结点配置结束。

5. xuegod63 数据节+SQL 节点配置文档：

xuegod63：

[root@xuegod63 ~]# vim /etc/my.cnf #手动创建数据节+SQL 节点配置文档/etc/my.cnf,写入以下内容。

[mysqld]

datadir=/var/lib/mysql

ndbcluster

ndb-connectstring=192.168.1.63

```
[mysqld_safe]
log-error=/var/log/mysqld.log
pid-file=/var/run/mysqld/mysqld.pid
```

```
[mysql_cluster]
ndb-connectstring=192.168.1.63
```

注释：

```
[root@xuegod63 /]# vim /etc/my.cnf #写入以下内容
```

```
[mysqld]
datadir=/var/lib/mysql      #mysql 数据存储路径
ndbcluster      #启动 ndb 引擎
ndb-connectstring=192.168.1.63  # 管理节点 IP 地址
```

```
[mysqld_safe]
log-error=/var/log/mysqld.log
pid-file=/var/run/mysqld/mysqld.pid
```

```
[mysql_cluster]
ndb-connectstring=192.168.1.63  #管理节点 IP 地址
```

-----注：如果 xuegod63 只做 SQL 节点，则配置文档为：-----

```
[root@xuegod63 /]# cat !$
cat /etc/my.cnf
[mysqld]
ndbcluster      #启动 ndb 引擎
ndb-connectstring=192.168.1.63  # 管理节点 IP 地址
```

```
[mysqld_safe]
log-error=/var/log/mysqld.log
pid-file=/var/run/mysqld/mysqld.pid
```

```
[mysql_cluster]
ndb-connectstring=192.168.1.63  #管理节点 IP 地址
```

说明：数据节点和 SQL 节点配置文件区别，就多一行
数据节点有：datadir=/var/lib/mysql #mysql 数据存储路径。
SQL 节点上没有。
到此，xuegod64 上管理，存储，SQL 三个节点都配置完成。

xuegod64 安装 mysql 集群相关的软件包：

```
[root@xuegod64 ~]# rpm -ivh MySQL-Cluster-server-gpl-7.3.7-1.el6.x86_64.rpm
```

还会生成用户：mysql

```
[root@xuegod64 ~]# id mysql
uid=496(mysql) gid=493(mysql) 组=493(mysql)
```

安装 mysql 集群客户端命令：

```
[root@xuegod64 ~]# rpm -ivh MySQL-Cluster-client-gpl-7.3.7-1.el6.x86_64.rpm
warning: waiting for transaction lock on /var/lib/rpm/.rpm.lock #需要等服务器安装完，再安装客户端软件。
```

```
[root@xuegod64 ~]# rpm -ivh MySQL-Cluster-client-gpl-7.3.7-1.el6.x86_64.rpm
```

把 xuegod64 配置成：数据结点和 SQL 结点：

生成数据结点和 SQL 结点的配置文件：

```
[root@xuegod63 mysql-cluster]# scp /etc/my.cnf 192.168.1.64:/etc/
root@192.168.1.64's password:
my.cnf                                100% 215    0.2KB/s  00:00
```

```
[root@xuegod64 /]# vim /etc/my.cnf #写入以下内容。默认没有 my.cnf 文件
```

```
[mysqld]
datadir=/var/lib/mysql      #mysql 数据存储路径
ndbcluster                  #启动 ndb 引擎
ndb-connectstring=192.168.1.63 # 管理节点 IP 地址
```

```
[mysqld_safe]
log-error=/var/log/mysqld.log
pid-file=/var/run/mysqld/mysqld.pid
```

```
[mysql_cluster]
ndb-connectstring=192.168.1.63 #管理节点 IP 地址
```

三、SQL Cluster 初次启动命令以及用户密码更改调整：（请严格按照次序启动）

启动 mysql cluster

先启动顺序：管理结点服务-》需要手动启动 ndbd 数据结点服务-》sql 结点服务

关闭顺序：关闭管理结点服务-》关闭管理结点服务后，nbdb 数据结点服务会自动关闭-》手动把 sql 结点服务关了。

执行初次启动前请先确认 将两台机器的防火墙关闭（service iptables stop 或者 设定 防火墙端口可通，两个端口即：通讯端口 1186、数据端口 3306）

启动管理结点命令：

```
[root@xuegod63 ~]# ndb_mgmd -f /var/lib/mysql-cluster/config.ini # mysql cluster 后台运行进程
```

MySQL Cluster Management Server mysql-5.6.21 ndb-7.3.7

```
2015-04-01 20:23:39 [MgmtSrvr] WARNING -- at line 7: [MGM] id is deprecated, use NodeId instead
```

```
2015-04-01 20:23:39 [MgmtSrvr] WARNING -- at line 15: [DB] id is deprecated, use NodeId instead
```

```
2015-04-01 20:23:39 [MgmtSrvr] WARNING -- at line 20: [DB] id is deprecated, use NodeId instead
```

```
2015-04-01 20:23:39 [MgmtSrvr] WARNING -- at line 27: [API] id is deprecated, use NodeId instead
```

```
2015-04-01 20:23:39 [MgmtSrvr] WARNING -- at line 31: [API] id is deprecated, use NodeId instead
```

```
2015-04-01 20:23:39 [MgmtSrvr] WARNING -- at line 32: Cluster configuration warning:
    arbitrator with id 1 and db node with id 2 on same host 192.168.1.63
    Running arbitrator on the same host as a database node may
    cause complete cluster shutdown in case of host failure.
```

```
2015-04-01 20:23:40 [MgmtSrvr] WARNING -- at line 7: [MGM] id is deprecated, use NodeId instead
```

```
2015-04-01 20:23:40 [MgmtSrvr] WARNING -- at line 15: [DB] id is deprecated, use NodeId instead
```

```
2015-04-01 20:23:40 [MgmtSrvr] WARNING -- at line 20: [DB] id is deprecated, use NodeId instead
```

```
2015-04-01 20:23:40 [MgmtSrvr] WARNING -- at line 27: [API] id is deprecated, use NodeId instead
```

```
2015-04-01 20:23:40 [MgmtSrvr] WARNING -- at line 31: [API] id is deprecated, use NodeId instead
```

```
2015-04-01 20:23:40 [MgmtSrvr] WARNING -- at line 32: Cluster configuration warning:
    arbitrator with id 1 and db node with id 2 on same host 192.168.1.63
    Running arbitrator on the same host as a database node may
    cause complete cluster shutdown in case of host failure.
```

#说明：仲裁员 id 为 1 和 DB 节点上同一主机 192.168.1.63 ID2

在同一台主机作为数据库节点上运行仲裁人
造成完整的集群关机的情况下主机发生故障。

虽然报了一些警告，但是，我们已经执行成功了。

查看端口号：

```
[root@xuegod63 ~]# netstat -antup | grep 1186
```

tcp	0	0	0.0.0.0:1186	0.0.0.0:*	LISTEN
7057/ndb_mgmd					
tcp	0	0	127.0.0.1:1186	127.0.0.1:60324	ESTABLISHED
7057/ndb_mgmd					
tcp	0	0	127.0.0.1:60324	127.0.0.1:1186	ESTABLISHED
7057/ndb_mgmd					

查看日志存储位置：

```
[root@xuegod63 ~]# ls /var/lib/mysql-cluster/  
config.ini ndb_1_cluster.log ndb_1_out.log ndb_1.pid
```

启动存储结点服务：

```
[root@xuegod63 /]# ndbd --initial  
2014-12-02 22:24:03 [ndbd] INFO      -- Angel connected to '192.168.1.63:1186'#可以看到  
ndbd 去访问管理结点 "192.168.1.63" 的 1186 端口  
2014-12-02 22:24:03 [ndbd] INFO      -- Angel allocated nodeid: 2
```

```
[root@xuegod64 ~]# ndbd --initial  
2014-12-02 22:24:31 [ndbd] INFO      -- Angel connected to '192.168.1.63:1186'  
2014-12-02 22:24:31 [ndbd] INFO      -- Angel allocated nodeid: 3
```

```
[root@xuegod63 ~]# vim /var/lib/mysql-cluster/config.ini  #查看数据结点对应的 ID  
[ndbd]  
HostName=192.168.1.63  
DataDir=/var/lib/mysql  
id=2
```

```
[ndbd]  
HostName=192.168.1.64  
DataDir=/var/lib/mysql  
id=3
```

xuegod63 和 xuegod64 启动 SQL 结点服务：

```
[root@xuegod63 ~]# mysqld_safe --defaults-file=/etc/my.cnf &  
[1] 7617  
150710 21:52:29 mysqld_safe Logging to '/var/log/mysqld.log'.  
150710 21:52:29 mysqld_safe Starting mysqld daemon with databases from /var/lib/mysql
```

```
[root@xuegod64 ~]# mysqld_safe --defaults-file=/etc/my.cnf &  
[1] 5576  
150710 21:53:05 mysqld_safe Logging to '/var/log/mysqld.log'.  
150710 21:53:06 mysqld_safe Starting mysqld daemon with databases from /var/lib/mysql
```

测试：

查看 mysql 集群状态：

```
[root@xuegod63 /]# ndb_mgm  
-- NDB Cluster -- Management Client --  
ndb_mgm> show  
Connected to Management Server at: 192.168.1.63:1186
```

Cluster Configuration

[ndbd(NDB)] 2 node(s)

id=2 @192.168.1.63 (mysql-5.6.21 ndb-7.3.7, Nodegroup: 0, *)

id=3 @192.168.1.64 (mysql-5.6.21 ndb-7.3.7, Nodegroup: 0)

[ndb_mgmd(MGM)] 1 node(s)

id=1 @192.168.1.63 (mysql-5.6.21 ndb-7.3.7)

[mysqld(API)] 2 node(s)

id=4 @192.168.1.63 (mysql-5.6.21 ndb-7.3.7)

id=5 @192.168.1.64 (mysql-5.6.21 ndb-7.3.7)

测试集群，数据同步：

xuegod63

[root@xuegod63 /]# cat /root/.mysql_secret

The random password set for the root user at Tue Dec 2 21:25:59 2014 (local time):

ENEHUAUV3Gv9ylu7Q (密码使用你自己的密码，不要直接从文档复制)

[root@xuegod63 /]# mysql -uroot -prvmW3nVHnqh_FmOx2

mysql> show databases; #第一使用数据库时，必须修改密码

ERROR 1820 (HY000): You must SET PASSWORD before executing this statement

mysql> set password=password('123456');

Query OK, 0 rows affected (0.72 sec)

mysql> show databases;

```
+-----+
| Database          |
+-----+
| information_schema |
| mysql              |
| ndbinfo            |
| performance_schema |
| test               |
+-----+
```

mysql> exit;

Bye

[root@xuegod63 /]# mysql -uroot -p123456

mysql> show databases;

```
+-----+
| Database          |
+-----+
| information_schema |
| mysql              |
```

```
| ndbinfo          |
| performance_schema |
| test             |
+-----+
```

xuegod64 :

```
[root@xuegod64 ~]# cat /root/.mysql_secret (查看密码)
# The random password set for the root user at Tue Dec 2 21:31:11 2014 (local time):
vWyreU1PFj5yRfq5
```

此处如果报错 : ps -aux |grep ndbd

```
Warning: bad syntax. perhaps a bogus '-'? See /usr/share/doc/procps-3.2.8/FAQ
root      3246  0.0  0.1  27080  2556 ?        Ss   Aug25   0:00 ndbd --initial
root      3247  1.3 38.0 1338348 781236 ?        Sl   Aug25   0:46 ndbd --initial
root      3890  0.0  0.0  103260   856 pts/1    S+   00:40   0:00 grep ndbd
[root@xuegod64 ~]# kill -9 3246
[root@xuegod64 ~]# kill -9 3247
[root@xuegod64 ~]# ndbd --initial
```

```
[root@xuegod64 ~]# mysql -uroot -p344XSmsiC2LOBoTY
```

```
mysql> set password=password('123456');
```

```
mysql> exit;
```

Bye

```
[root@xuegod64 ~]# mysql -uroot -p123456
```

```
mysql> show databases;
```

```
+-----+
| Database          |
+-----+
| information_schema |
| mysql              |
| ndb_3_fs           |
| ndbinfo            |
| performance_schema |
| test               |
+-----+
6 rows in set (0.14 sec)
```

插入数据 :

```
mysql -u root -p123456
```

xuegod63 :

```
mysql> create database tree;
```

xuegod64 :

```
mysql> show databases;
```

```
+-----+
```



```
| Database          |
+-----+
| information_schema|
| tree              |
| mysql             |
| ndb_3_fs          |
| ndbinfo           |
| performance_schema|
| test              |
```

关闭 mysql 集群顺序：关闭管理节点服务-》关闭管理节点时，数据结点服务自动关闭 -》需要手动关闭 SQL 结点服务

```
[root@xuegod63 /]# ndb_mgm
-- NDB Cluster -- Management Client --
ndb_mgm> shutdown
Node 2: Cluster shutdown initiated
Node 3: Cluster shutdown initiated
3 NDB Cluster node(s) have shutdown.
Disconnecting to allow management server to shutdown.
Node 2: Node shutdown completed.
ndb_mgm> exit
ps -axu | grep ndbd    #查看不到，说明数据节点已经被关
```

手动关闭 SQL 结点服务

xuegod64 上，手动关闭 SQL 结点服务

```
[root@xuegod63 ~]# ps -axu | grep mysql
Warning: bad syntax, perhaps a bogus '-'? See /usr/share/doc/procps-3.2.8/FAQ
root      7617  0.0  0.1 106280 1424 pts/0    S   21:52   0:00 /bin/sh
/usr/bin/mysqld_safe --defaults-file=/etc/my.cnf
mysql     7743  0.2 38.9 1342572 453680 pts/0    Sl  21:52   0:03 /usr/sbin/mysqld
--defaults-file=/etc/my.cnf      --basedir=/usr      --datadir=/var/lib/mysql
--plugin-dir=/usr/lib64/mysql/plugin  --user=mysql      --log-error=/var/log/mysqld.log
--pid-file=/var/run/mysqld/mysqld.pid
root      7889  0.0  0.2 125868 2364 pts/3    S+  22:11   0:00 mysql -uroot -px xxxx
root      7916  0.0  0.0 103300 852 pts/1     S+  22:17   0:00 grep mysql
[root@xuegod63 ~]# kill -9 7617
[root@xuegod63 ~]# kill -9 7743
[root@xuegod63 ~]# ps -axu | grep mysql
Warning: bad syntax, perhaps a bogus '-'? See /usr/share/doc/procps-3.2.8/FAQ
root      7889  0.0  0.2 125868 2364 pts/3    S+  22:11   0:00 mysql -uroot -px xxxx
root      7920  0.0  0.0 103300 852 pts/1     S+  22:18   0:00 grep mysql
```

xuegod64 上, 手动关闭 SQL 结点服务

```
[root@xuegod64 ~]# ps -axu | grep mysql
```

Warning: bad syntax, perhaps a bogus '-'? See /usr/share/doc/procps-3.2.8/FAQ

```
root      5576   0.0   0.0 106284    644 pts/0      S      21:53    0:00 /bin/sh
```

```
/usr/bin/mysqld_safe --defaults-file=/etc/my.cnf
```

```
mysql     5701   0.8 37.6 1352816 437908 pts/0    Sl     21:53    0:10 /usr/sbin/mysqld
```

```
--defaults-file=/etc/my.cnf          --basedir=/usr          --datadir=/var/lib/mysql
```

```
--plugin-dir=/usr/lib64/mysql/plugin  --user=mysql          --log-error=/var/log/mysqld.log
```

```
--pid-file=/var/run/mysqld/mysqld.pid
```

```
root      5801   0.0   0.2 125860    2740 pts/0    S+     22:10    0:00 mysql -uroot -px xxxx
```

```
root      5826   0.0   0.0 103300     844 pts/1    S+     22:15    0:00 grep mysql
```

```
[root@xuegod64 ~]# kill -9 5576
```

```
[root@xuegod64 ~]# kill -9 5701
```

```
[root@xuegod64 ~]# ps -axu | grep mysql
```

Warning: bad syntax, perhaps a bogus '-'? See /usr/share/doc/procps-3.2.8/FAQ

```
root      5801   0.0   0.2 125860    2740 pts/0    S+     22:10    0:00 mysql -uroot -px xxxx
```

```
root      5836   0.0   0.0 103300     844 pts/1    S+     22:16    0:00 grep mysql
```

```
[root@xuegod64 ~]#
```

总结：

再次启动，mysql 集群启动：

```
[root@xuegod63 /]# ndb_mgmd -f /var/lib/mysql-cluster/config.ini
```

MySQL Cluster Management Server mysql-5.6.21 ndb-7.3.7

```
[root@xuegod63 /]# ndbd
```

```
2014-12-02 22:41:59 [ndbd] INFO      -- Angel connected to '192.168.1.63:1186'
```

```
2014-12-02 22:41:59 [ndbd] INFO      -- Angel allocated nodeid: 2
```

```
[root@xuegod64 /]# ndbd
```

```
2014-12-02 22:41:59 [ndbd] INFO      -- Angel connected to '192.168.1.63:1186'
```

```
2014-12-02 22:41:59 [ndbd] INFO      -- Angel allocated nodeid: 2
```

```
[root@xuegod63 /]# mysqld_safe --defaults-file=/etc/my.cnf &
```

```
[root@xuegod64 /]# mysqld_safe --defaults-file=/etc/my.cnf &
```

测试：

查看 mysql 集群状态：

```
[root@xuegod63 /]# ndb_mgm
```

```
-- NDB Cluster -- Management Client --
```

```
ndb_mgm> show
```

```
Connected to Management Server at: 192.168.1.63:1186
```

```
Cluster Configuration
```

[ndbd(NDB)] 2 node(s)

id=2 @192.168.1.63 (mysql-5.6.21 ndb-7.3.7, Nodegroup: 0, *)

id=3 @192.168.1.64 (mysql-5.6.21 ndb-7.3.7, Nodegroup: 0)

[ndb_mgmd(MGM)] 1 node(s)

id=1 @192.168.1.63 (mysql-5.6.21 ndb-7.3.7)

[mysqld(API)] 2 node(s)

id=4 @192.168.1.63 (mysql-5.6.21 ndb-7.3.7)

id=5 @192.168.1.64 (mysql-5.6.21 ndb-7.3.7)