

XCAT 2.x MySQL Setup

Date: 10/26/2009

1.0 Switch to the MySQL database on the Management Node.....	1
1.1 Install MySQL.....	1
1.2 Configure MySQL.....	2
1.3 Migrate xCAT data to MySQL.....	7
1.4 Hierarchy.....	8
1.5 Add ODBC support.....	8
1.6 Removing MySQL database.....	11
1.7 Migrate to new level MySQL.....	11
1.8 References.....	12

1.0 Switch to the MySQL database on the Management Node

The xCAT support for Service Nodes requires a database with remote access capabilities. The MySQL database is provided for this purpose. You must be running the xCAT 2.1 or later code to use MySQL.

1.1 Install MySQL

On AIX:

The xCAT RPM called `xcat-mysql` is provided as a convenience to help simplify the installation of MySQL on an AIX system. This RPM is available at <https://sourceforge.net/projects/xcat/files/>
Download `xcat-mysql-2.*.gz`.

To install the packages use the following commands:

```
gunzip xcat-mysql-2*.gz
tar -xvf xcat-mysql-2*.tar
./instmysql
```

The `xcat-mysql` post processing will automatically unwrap MySQL in the `/usr/local` directory and will create a link for `/usr/local/mysql`. I will also update the `PATH` environment variable in the `/etc/profile` file.

On Linux:

MySQL comes as part of the OS. Insure that the following rpms are installed on your Management Node:

```
perl-DBD-MySQL*  
mysql-server-5.*  
mysql-5.*  
mysql-devel-5.*  
mysql-bench-5.*  
mysql-5.*  
mysql-connector-odbc-*  
mysql-devel-5.*
```

1.2 Configure MySQL

To set up the MySQL database on the Management Node follow these steps. (See <http://dev.mysql.com/doc/refman/5.1/en/installing-binary.html> for additional details.)

This example assumes:

- mn20: hostname of management node
- xcatdb: database name
- mysql: database role (aka user)
- xcatadmin database user id used by xCAT for access
- xcat201: database password for xcatadmin

Substitute your addresses and desired database administration, password and database name as appropriate.

On AIX: The example assumes that **mysql** it was installed in */usr/local* and *commands and the database is under /usr/local/mysql* directory.

On Linux: mysql is already installed in */usr/bin* and the database is in */var/lib/mysql* (tested on Redhat).

Set your paths accordingly.

1. For AIX, add a login user and group for **mysqld** (this is probably already there on Linux):

```
add the group mysql  
add the user mysql
```

```
mkgroup mysql  
mkuser pgrp=mysql mysql
```

2. On AIX, Update permissions on installed mysql

```
cd /usr/local/mysql
chown -R mysql .
chgrp -R mysql .
```
3. Create the MySQL data directory and initialize the grant tables:

```
On AIX: /usr/local/mysql/scripts/mysql_install_db --user=mysql
On Linux: /usr/bin/mysql_install_db --user=mysql
```
4. Edit Server config file to run in ANSI-QUOTES mode (required)
On AIX:

```
cp /usr/local/mysql/support-files/my-large.cnf /etc/my.cnf
```

On Linux and AIX: edit /etc/my.cnf .

under the # The MySQL server [mysqld] section

add the following line: `sql_mode=ANSI_QUOTES`
5. For large systems you may also need to increase max_connections to the database in the my.cnf file. The default is 100. These defaults can be seen with the SHOW VARIABLES statement. (see below). To increase max_connections, add this line to the configuration file:

On Linux and AIX: edit /etc/my.cnf .

under the # The MySQL server [mysqld] section

add the following line: `max_connections=300`

Other system variables might also need to change, this website gives all the MySQL system variables.

http://dev.mysql.com/doc/refman/5.1/en/server-system-variables.html#sysvar_max_connections

6. On AIX:Update permissions for root to own database

```
cd /usr/local/mysql
chown -R root .
chown -R mysql data
```
7. Set the ulimits to unlimited, if not already set and save in the /etc/security/limits file.

Run the command ulimit -a and check the settings:

```
ulimit -a  
time(seconds) unlimited  
file(blocks) 2097151  
data(kbytes) 131072  
stack(kbytes) 32768  
memory(kbytes) 32768  
coredump(blocks) 2097151  
nofiles(descriptors) 2000  
threads(per process) unlimited
```

Change the ulimit setting to unlimited.

```
ulimit -m unlimited  
ulimit -d unlimited  
ulimit -f unlimited  
ulimit -s unlimited  
ulimit -t unlimited
```

Change the ulimit setting to unlimited, permanently:

Edit /etc/security/limits and make sure these limits will stay unlimited though reboot or MySQL may not come up after reboot. Once running with MySQL, the xcatd daemon will get errors starting, if the database is not running.

8. Start the MySQL server (running as root must use the --user option):

```
On AIX:  
/usr/local/mysql/bin/mysqld_safe --user=mysql &  
On Linux:  
/usr/bin/mysqld_safe --user=mysql & or  
service mysqld start
```

9. Setup MySQL to make sure it is started automatically on reboot of the Management Node

On AIX:

In the /etc/inittab, add the following line before the line for xcatd to make sure it is started before the restart of xcatd.

```
mysql:2:once:/usr/local/mysql/bin/mysqld_safe --user=mysql &
```

On Linux:

chkconfig mysqld on

10. Change Admin password

ON AIX:

```
/usr/local/mysql/bin/mysqladmin -u root password  
'new-password'
```

On Linux:

```
/usr/bin/mysqladmin -u root password 'new-password'
```

11. If command fails,

On AIX: check the /usr/local/mysql/data/mn20.err file.

On Linux: check /var/log/mysqld.log

12. If you need to stop MySQL server run the following command:

On AIX: /usr/local/mysql/bin/mysqladmin -u root -p shutdown

On Linux: /usr/bin/mysqladmin -u root -p shutdown
or service mysqld stop

13. Create the *xcatdb* database.

On AIX: /usr/local/mysql/bin/mysql -u root -p

On Linux: /usr/bin/mysql -u root -p

Enter password:

mysql > CREATE DATABASE xcatdb;

14. Create the xcat admin id with password xcat201 and grant permissions to hosts to access the database.

On AIX: /usr/local/mysql/bin/mysql -u root -p

On Linux: /usr/bin/mysql -u root -p

Enter password:

```
mysql > CREATE USER xcatadmin IDENTIFIED BY 'xcat201';
```

15. Grant permissions to hosts to access the database, first add you Management Node where the database is running and then all other nodes that need access like servicenodes and possibly compute nodes. A good name to use for your Management node is the name in the site table “master” attribute. The important thing is the name, is a resolvable hostname for the Management Node.

```
mysql > GRANT ALL on xcatdb.* TO xcatadmin@mn20  
IDENTIFIED BY 'xcat201';
```

If using long hostnames:

```
mysql > GRANT ALL on xcatdb.* TO  
xcatadmin@mn20.<domainname> IDENTIFIED BY 'xcat201';
```

```
mysql > GRANT ALL on xcatdb.* TO xcatadmin@<servicenode  
ip(s)> IDENTIFIED BY 'xcat201';
```

Substitute your own admin id name and password and management node and service node ip address or name.

Note: you want to do a GRANT ALL to every ipaddress or nodename that will need to access the database. You can use wildcards as follows:

```
mysql > GRANT ALL on xcatdb.* TO xcatadmin@'%.example.com'  
IDENTIFIED BY 'xcat201';
```

```
mysql > GRANT ALL on xcatdb.* TO xcatadmin@'8.113.33.%'  
IDENTIFIED BY 'xcat201';
```

16. Check the user table.

```
mysql > SELECT host, user FROM mysql.user;
```

17. Check system variables

```
mysql > SHOW VARIABLES;
```

18. Check the defined databases.

```
mysql > SHOW DATABASES;
```

19. To run commands against the *xcatdb* database you can enter MySQL commands as follows:

```
mysql > use xcatdb;
```

```
mysql > SHOW TABLES;
```

```
mysql > DESCRIBE <table name>;
```

To exit:

```
mysql > quit;
```

1.3 Migrate xCAT data to MySQL

1. Back up your xCAT data. (This is required even if you have not added anything to your xCAT database yet. Required default entries were created when the xCAT RPMs were installed on the management node and they must be migrated to the new MySQL database.)

```
mkdir -p ~/xcat-dbback  
dumpxCATdb -p ~/xcat-dbback
```

2. Creating the `/etc/xcat/cfgloc` tells the xcatd daemon what database to use. If the file does not exist, it uses by default SQLite which is what it sets up during install. My MySQL it should contain the following line, substituting your specific info.

The information is from the setup you put in MySQL in Configure MySQL. The dbname is the database you created, the host must match the Management Node host entry, where the MySQL database is running, that you gave database access when you executed the step to Grant permissions to hosts to access the database command above. You can find valid names by Check the user table. The admin and password must match user ids you put in the step: Create the xcat admin id.

```
mysql:dbname=xcatdb;host=mn20|xcatadmin|xcat201
```

Note: may need to use long hostname or ip address

```
mysql:dbname=xcatdb;host=mn20.cluster.net|xcatadmin|xcat201  
Check the user table.
```

```
chmod 0600 /etc/xcat/cfgloc - this protects the password
```

3. Restore your database to MySQL (bypass mode runs the command without xcatd):

```
XCATBYPASS=1 restorexCATdb -p ~/xcat-dbback
```

Note: if you get any SQL error, check to make sure you updated the my.cnf file as indicated in step: Edit Server config file to run in ASCII-QUOTES mode (required). If this needs to be done, restart MySQL server after editing my.cnf, and run the restore command again.

4. If you get errors restoring the database still, you can go back to using the SQLite database, simply by moving the `/etc/xcat/cfgloc` file in `/etc/xcat/cfgloc.mysql`. With no `cfgloc` file in `/etc/xcat` the daemon will start using SQLite, until problem determination can be done on MySQL.

5. Unset XCATBYPASS
6. With the restore successful, start the xcatd daemon using the MySQL database
 - On AIX: `xcatstart`
 - On Linux: `service xcatd restart`

1.4 Hierarchy

If running xCAT with Service Nodes, hierarchical mode, you will also using MySQL on the Service Nodes.

For AIX, read the [xCAT2onAIXServiceNodes](#) document.

For Linux, read the [xCAT2 Advanced Cookbook](#), section on Postgresql and just substitute MySQL as your database.

1.5 Add ODBC support

Note: You only need to follow the steps in this section on adding ODBC support, if you plan to develop C, C++ database applications on the database or run such applications (like LoadLeveler). Otherwise skip to the next section.

1. Install ODBC package and MySQL connector.

On AIX:

You'll need `unixODBC` and `mysql-connector-odbc` packages. `unixODBC` package is included in the `dep-aix-xxxx.tar.gz` file and `mysql-connector-odbc` package is included in `xcat-mysql-xxxx.tar.gz`. Both `.gz` files were downloaded when setting up the xCAT management node.

To install the packages use the following commands:

```
rpm -i unixODBC-* ( this is in the xcat AIX dependency package and  
will be on the system if you have installed the latest deps package).  
rpm -i mysql-connector-odbc-*
```

On Linux:

These packages comes as part of the OS. Please make sure the following packages are installed on your management node. Please refer to chapter 2.2.18 of the [xCAT Top Doc](#) for how to get packages from the distro.

For RedHat and Fedora:

```
unixODBC-*  
mysql-connector-odbc-*
```

For SLES:

```
unixODBC-*  
mysql-client-*  
libmysqlclient*  
MyODBC-unixODBC-*
```

Please note that MyODBC-unixODBC rpm can be found in SDK CD 1 for SLES 11.

2. Configure ODBC

You need to make changes to *odbc.ini* and *odbcinst.ini* files so that ODBC works with xCAT database. **These files come with unixODBC package and are located under /etc directory. (On SLES they are located under /etc/unixODBC directory).**

First find the ODBC driver. For AIX, RH and Fedora:

```
rpm -ql mysql-connector-odbc
```

For SLES:

```
rpm -ql MyODBC-unixODBC
```

It will list the file names come for the MySQL connector. Pick the one that is called *libmyodbc#.so* as the driver and put it into *odbcinst.ini* file.

```
vi /etc/odbcinst.ini or vi /etc/unixODBC/odbcinst.ini (SLES)  
[MySQL]  
Description = ODBC for MySQL  
Driver      = /usr/lib64/libmyodbc3.so
```

Next, configure the DSN for ODBC, note the SERVER and DATABASE should match what was put in Creating the */etc/xcat/cfgloc* file.

```
vi /etc/odbc.ini or vi /etc/unixODBC/odbc.ini (SLES)  
[xCATDB]  
Driver    = MySQL  
SERVER    = mn20  
PORT      = 3306  
DATABASE  = xcatdb
```

Then, put the admin user id and password for xcatdb database on the root's home directory so that user root will not have to specify them when accessing the database through ODBC, the SERVER, DATABASE, USER, PASSWORD should match what was put in Creating the */etc/xcat/cfgloc* file.

```
vi $roothome/.odbc.ini  
[xCATDB]  
SERVER = mn20  
DATABASE = xcatdb  
USER    = xcatadmin
```

```
PASSWORD = xcat201  
chmod 0600 .odbc.ini
```

3. Configure the Service Node

[Skip this step if there are no service nodes in the cluster. That is you are not using xCAT hierarchical support.]

If there are service nodes in the cluster you need to install unixODBC and MySQL connector on them and modify the ODBC configuration files just as we did in step 1 and 2. xCAT has utilities to install additional software on the nodes. To install ODBC and MySQL on to the service nodes, refer to the following documents for details:

Linux: [xCAT2-updatenode.pdf](#)

AIX: [xCAT2onAIXUpdates.pdf](#)

Then sync the odbcinst.ini and odbc.ini files to the service nodes:

For AIX:

```
xhcp service -v /etc/odbcinst.ini /etc/odbcinst.ini  
xhcp service -v /etc/odbc.ini /etc/odbc.ini  
xhcp service -v /.odbc.ini /.odbc.ini
```

For RH and Fedora:

```
xhcp service -v /etc/odbcinst.ini /etc/odbcinst.ini  
xhcp service -v /etc/odbc.ini /etc/odbc.ini  
xhcp service -v $roothome/.odbc.ini $roothome/.odbc.ini
```

For SLES

```
xhcp service -v /etc/unixODBC/odbcinst.ini /etc/unixODBC/odbcinst.ini  
xhcp service -v /etc/unixODBC/odbc.ini /etc/unixODBC/odbc.ini  
xhcp service -v $roothome/.odbc.ini $roothome/.odbc.ini
```

where service is the node group name for all the service nodes.

4. Test ODBC

Please note that the path for isql command on AIX is /usr/local/bin.

As user root:

```
isql -v xcatdb
```

As a non-root user:

```
isql -v xcatdb xcatadmin xcat201
```

Then

```
mysql > SHOW TABLES;
```

To exit:

```
mysql > quit;
```

1.6 Removing MySQL database

To remove the database, first backup you copy:

```
mkdir -p ~/xcat-dbback  
dumpxCATdb -p ~/xcat-dbback
```

Now remove the database:

On AIX: */usr/local/mysql/bin/mysql -u root -p*

On Linux: */usr/bin/mysql -u root -p*

Enter password:

To drop the database, enter:

```
mysql> drop database xcatdb;
```

1.7 Migrate to new level MySQL

If you want to migrate to a new xCAT level of MySQL, note the entire setup will have to be redone. It is best to stay on your current level, even though a new one has been made available. In the future, we will be changing the install of MySQL to be more automated so this will not be the case.

If you need to update to the new xCAT MySQL, then do the following:

- 1) Backup your database as you did in Migrate xCAT data to MySQL.
- 2) Stop the MySQL server as you did in stop MySQL server.
- 3) Stop xcatd daemon

On AIX: *xcatstop*

On Linux: *service xcatd stop*

- 4) Unlink the previous version of MySQL

```
cd /usr/local
```

```
rm mysql
```

- 5) Download the latest MySQL as indicated in Install MySQL.

- 6) Follow the entire install process outlined in Install MySQL and Configure MySQL. You will not have to create the mysql id or group on AIX, since they already exist. You will not have to update the /etc/my.cnf file, since it was previously setup.

- 7) Restore your database and start xcatd as you did in Migrate xCAT data to MySQL.
- 8) You will now be running on the new database level. You can remove the old level by going to /usr/local and removing the mysql-5.0.67-aix5.3-powerpc-64bit directory. It takes up a lot of space under /usr/local. Be sure your new level is running and your database is restored, because the old directory does contain the mysql data.

1.8 References

- <http://www.pantz.org/software/mysql/mysqlcommands.html>
- <http://dev.mysql.com/doc/refman/5.0/en/tutorial.html>