# Operation Document

Online Learning Platform

AUTHOR: ÅSA WEGELIUS, CLOVIS LEBRET, TUDOR STOICA

OWNER: ÅSA WEGELIUS

CLIENT: JARL TUXEN

VERSION: 1.0.5

# 1. Operation Document History

## 1.2 Revision History

| version | Revision | date | Implemented by |
|---|---|---|---|
| 1.0 | Master Slave replica on MySQL | 15-05-16 | Åsa Wegelius |
| 1.1 | Tomcat cluster | 20-05-16 | Åsa Wegelius |

## 1.3 Approvals

| version | Name | title | Date |
|---|---|---|---|

## 1.4 Distribution

| version | Name | title | Date |
|---|---|---|---|

## 1.5 Confidentiality Rating

| Rating | |
|---|---|
| Company Confidential | x |
| Non Confidential | |

## 2. Content Table

# 3. Installation Manual

## 3.1 Master Slave replica on MySQL

### To configure some basic options:

You master need to be set to binary logging and establish a unique server ID. The binary log is the basis for replicating changes from the master to its slaves.

1. Shut down the server. Find the **my.cnf** or **my.ini** file. Find the section that starts with **[mysqld]**. The options you shall set or add are **log-bin** and **server-id**. If they are commented out, just uncomment them and alter the values to your choises, if they are not there you add them. You can see how it looks below with a logfile name prefix of mysql-bin and server-id 1. After it is set you need to restart the server.

   ```
   [mysqld]
   log-bin=mysql-bin
   server-id=1
   ```

2. Your slave or slaves must have a unique id that differs from the master and other slaves. As with the master, find the **my.cnf** or **my.ini** file and the section that starts with **[mysqld]**. Set the server-id to an unique id. You can see how it looks below with a server-id 2. After it is set you need to restart the server.

   ```
   [mysqld]
   server-id=2
   ```

3. It is not necessary to set the log-bin on slaves but if you do you can use the slave's binary log for data backups and crash recovery.

You need to create a slave user for each slave and give them REPLICATION SLAVE privilege on the server. For example, if you want to set up a new user, repl, that can connect for replication from any host within the mydomain.com domain, you can do as below:

```
mysql> CREATE USER 'repl'@'%.mydomain.com' IDENTIFIED BY 'slavepass';
mysql> GRANT REPLICATION SLAVE ON *.* TO 'repl'@'%.mydomain.com';
```

4. You need a fresh starting point on your master before starting to replicate to avoid inconsistency and corrupted data. You need to give the master binary log coordinates to the slave so it starts the replication at the correct point.

Start a session on the master by connecting to it with the command-line client, and flush all tables and block write statements by executing the FLUSH TABLES WITH READ LOCK statement as below:

```
mysql> FLUSH TABLES WITH READ LOCK;
```

5. In a different session on the master, use the SHOW MASTER STATUS statement to determine the current binary log file name and position as below:

```
mysql > SHOW MASTER STATUS;
+------------------+----------+-------------+------------------+
| File             | Position | Binlog_Do_DB | Binlog_Ignore_DB |
+------------------+----------+-------------+------------------+
| mysql-bin.000003 | 73       | test        | manual,mysql     |
+------------------+----------+-------------+------------------+
```

6. You find the name of the log file in the file column and the position in the log file in position column. These are the values you need when you set up the slave since they tell the slave where to start the replica. If the columns are empty the values that you need to use later when specifying the slave's log file and position are the empty string ('') and 4.

You might already have data on your master and need to syncronize your slave. You can use the mysqldump tool to create a dump of all the databases you want to replicate. To dump all databases into a file named **dbdump.db** and include the --master-data that appends the CHANGE MASTER TO statement required on the slave to start the replication process you do as below:

```
shell> mysqldump --all-databases --master-data > dbdump.db
```

To import the data, either copy the dump file to the slave, or access the file from the master when connecting remotely to the slave.

7. The slave need connection information to be able to communicate with the master. To set the master configuration on the slave you execute the below statements after you swap the values with the correct values for your system:

```
mysql> CHANGE MASTER TO
->     MASTER_HOST='master_host_name',
->     MASTER_USER='replication_user_name',
->     MASTER_PASSWORD='replication_password',
->     MASTER_LOG_FILE='recorded_log_file_name',
->     MASTER_LOG_POS=recorded_log_position;b
```

8. If you are setting up a new replica enviroment using data from a different MySQL server, run the dump file from that one on the new master as below. The updates are automatically propagated to the slaves.

```
shell> mysql -h master < fulldb.dump
```

9. If you are setting up replication with existing data there are two ways. If you used **mysqldump** start the slave using the --skip-slave-start option so that replication does not start. Then you import the dump file as below:

```
shell> mysql < fulldb.dump
```

If you created a snapshot using the raw data files you extract the data files into your slave data directory. You can do as below:

```
shell> tar xvf dbdump.tar
```

You may need to set permissions and ownership on the files so that the slave server can access and modify them. Start the slave using the --skip-slave-start option so that replication does not start. Configure the slave with the replication coordinates from the master. This tells the slave the binary log file and position within the file where replication needs to start. Also, configure the slave with the login credentials and host name of the master. Start the slave threads as below:

```
mysql> START SLAVE;
```

After you have performed this procedure, the slave connects to the master and replicates any updates that have occurred on the master since the snapshot was taken.

## 3.2 Tomcat cluster

### Install Apache HTTP Server on Windows

The Apache HTTP Server Project itself does not provide binary releases, only source code. But there are binary distributions available on the Internet. You can find one at Apache Haus. #####Steps after downloading the binary: - Ensure you have Visual C++ 2008 Redistributable Package. If not, you need to install it. You can find it at Microsoft. - Unzip the Apache24 folder in the package zip file to the root directory on any drive. Example: c:24. - Open a command prompt window and cd to the 24folder on the drive you unzipped the zip file to. To Start Apache in the command prompt type: *httpd.exe* - This distribution comes pre-configured for localhost. You can now test your installation by opening up your Web Browser and typing in the address: *http://localhost*. If everything is working properly you should see the Apache Haus's test page. - You can shut down Apache by pressing Ctrl+C (It may take a few seconds)

#### *Change port for Apache HTTP Server*

Open httpd.conf in the folder c:24. Find *Listen 80* and change to *Listen 8089*. Next find ServerName localhost:80 and change to *localhost:8089*

#### *Install Apache HTTP Server as service*

In most cases you will want to run Apache as a Windows Service. To do so you install Apache as a service by typing at the command prompt [1];

*httpd -k install* You can then start Apache by typing

*httpd -k start* Apache will then start and eventually release the command prompt window.

### *Adding mod_jk*

You need a mod_jk. It is an Apache module used to connect the Tomcat servlet container with other web servers. Find and download the binary release you need from Apache. Unzip and place the mod_jk.so in the *c:* folder.

### *Configure mod_jk in C:24.conf*

Open httpd.conf and add this:

```
# START configs for load balancing
LoadModule jk_module modules\mod_jk.so
#the worker configuration file
JkWorkersFile C:\Apache24\conf\workers.properties
#for logging and memory usage
JkShmFile  C:\Apache24\logs\mod_jk.shm
JkLogFile C:\Apache24\logs\mod_jk.log
JkLogLevel info
# END configs for load balancing
```

### *Configure C:24.properties file*

Create a workers.properties file in C:24 and add the code below

```
# define virtual worker's list
worker.list=jkstatus, LoadBalancer

# Enable virtual workers earlier
worker.jkstatus.type=status
worker.LoadBalancer.type=lb

# Add Tomcat instances as workers, three workers in our case
worker.worker1.type=ajp13
worker.worker1.host=192.168.1.144
worker.worker1.port=8009

worker.worker2.type=ajp13
worker.worker2.host=localhost
worker.worker2.port=8009

worker.worker3.type=ajp13
worker.worker3.host=localhost
worker.worker3.port=8009
```

```
# Provide workers list to the load balancer
worker.LoadBalancer.balance_workers=worker1,worker2,worker3
```

### Configuring Tomcat instances for the cluster

Edit the server.xml in Tomcat's conf folder. Add attribute jvmRoute with the name of the worker to the existing element as below (here worker1 as in the workers.properties file):

```
    <Engine name="Catalina" defaultHost="localhost" jvmRoute="worker1">
```

Uncomment the Cluster element, add channelSendOptions="8" attribute to set clustering communication to asynchronous. All other clustering elements are nested within the Cluster element:

```
       <Cluster className="org.apache.catalina.ha.tcp.SimpleTcpCluster"
                 channelSendOptions="8">

          <Manager
className="org.apache.catalina.ha.session.DeltaManager"
                   expireSessionsOnShutdown="false"
                   notifyListenersOnReplication="true"/>

          <Channel
className="org.apache.catalina.tribes.group.GroupChannel">
            <Membership
className="org.apache.catalina.tribes.membership.McastService"
                        address="228.0.0.4"
                        port="45564"
                        frequency="500"
                        dropTime="3000"/>
            <Receiver
className="org.apache.catalina.tribes.transport.nio.NioReceiver"
                      address="auto"
                      port="4000"
                      autoBind="100"
                      selectorTimeout="5000"
                      maxThreads="6"/>

            <Sender
className="org.apache.catalina.tribes.transport.ReplicationTransmitter"
>
              <Transport
className="org.apache.catalina.tribes.transport.nio.PooledParallelSende
r"/>
            </Sender>
            <Interceptor
```

```
className="org.apache.catalina.tribes.group.interceptors.TcpFailureDete
ctor"/>
            <Interceptor
className="org.apache.catalina.tribes.group.interceptors.MessageDispatc
h15Interceptor"/>
          </Channel>

          <Valve
className="org.apache.catalina.ha.tcp.ReplicationValve"
                 filter=""/>
          <Valve
className="org.apache.catalina.ha.session.JvmRouteBinderValve"/>

          <Deployer
className="org.apache.catalina.ha.deploy.FarmWarDeployer"
                    tempDir="/tmp/war-temp/"
                    deployDir="/tmp/war-deploy/"
                    watchDir="/tmp/war-listen/"
                    watchEnabled="false"/>

          <ClusterListener
className="org.apache.catalina.ha.session.ClusterSessionListener"/>
        </Cluster>
```

# 4. Appendices

## 4.1 References

MySQL Manual (version 5.6) *18.1.2.1 Setting the Replication Master Configuration* Retrieved 05 15, 2016, from https://dev.mysql.com/doc/refman/5.6/en/replication-howto-masterbaseconfig.html

MySQL Manual (version 5.6) *17.1.1.2 Setting the Replication Slave Configuration* Retrieved 05 15, 2016, from https://dev.mysql.com/doc/refman/5.6/en/replication-howto-slavebaseconfig.html

MySQL Manual (version 5.6) *17.1.1.3 Creating a User for Replication* Retrieved 05 15, 2016, from https://dev.mysql.com/doc/refman/5.6/en/replication-howto-repuser.html

MySQL Manual (version 5.6) *17.1.1.4 Obtaining the Replication Master Binary Log Coordinates* Retrieved 05 15, 2016, from https://dev.mysql.com/doc/refman/5.6/en/replication-howto-masterstatus.html

MySQL Manual (version 5.6) *17.1.1.5 Creating a Data Snapshot Using mysqldump* Retrieved 05 15, 2016, from https://dev.mysql.com/doc/refman/5.6/en/replication-howto-mysqldump.html

MySQL Manual (version 5.6) *17.1.1.10 Setting the Master Configuration on the Slave* Retrieved 05 15, 2016, from https://dev.mysql.com/doc/refman/5.6/en/replication-howto-slaveinit.html

MySQL Manual (version 5.6) *18.1.2.5 Setting Up Replication Slaves* Retrieved 05 16, 2016, from http://dev.mysql.com/doc/refman/5.7/en/replication-setup-slaves.html

Apache *Apache HTTP Server Version 2.4* Retrieved 05 20, 2016, from https://httpd.apache.org/docs/2.4/platform/windows.html Aoache *workers.properties configuration* Retrieved 05 21,2016, from http://tomcat.apache.org/connectors-doc/reference/workers.html