Making Back-Ups with Galera Cluster
Codership Training
Introduction

Making Back-Ups with Galera Cluster
Introductions

Codership Oy

Creators & Developers of Galera Cluster

Employees in Multiple Countries

Galera Cluster

Released Initially in May 2007

Over 1.5 Million Downloads

Russell Dyer, Presenter

KB Editor, Documentation, Instructor (MySQL, MariaDB)

Writer (O'Reilly Books)
Tutorial Outline

Back-Up Basics
Using Standard Replication
Using Galera Arbitrator

Restoring Nodes
Back-Up Plan
Back-Up Basics

Making Back-Ups with Galera Cluster
Backup Principles

**POLICIES**

Make Thorough Back-Ups
- Synchronize & Copy Binary Logs
- Copy Configuration Files

Make Daily & Continuous Backups

Store Backups in Multiple Locations

**PRAXIS**

Back-Up Methods
- Physical Back-Ups
- Logical Back-Ups

Verify Back-Ups

Practice Recovering
Physical Back-Ups

**PRO POINTS**
- More Intuitive & Simple
- Faster than Other Methods

**CONTRA POINTS**
- Usually Have to Stop `mysqld`
- Won’t Detect Corrupted Files
- Not Useful for Migrations – Same Storage Engine

# Logical Backups

<table>
<thead>
<tr>
<th><strong>PRO POINTS</strong></th>
<th><strong>CONTRA POINTS</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Produces Text Files with SQL Statements</td>
<td>Slower &amp; Requires Table Locks</td>
</tr>
<tr>
<td>Full or Specific Back-Ups</td>
<td>Uses a Local Drive, not across Network</td>
</tr>
<tr>
<td>Can Be Used for Migration</td>
<td></td>
</tr>
<tr>
<td>Independent of Storage Engines</td>
<td></td>
</tr>
</tbody>
</table>

Simple Galera Node Back-Up Procedure

Stop MySQL Daemon
Run Back-Up Utility on Down Node
  mysql\ldump
  rsync
Start MySQL Daemon

Logical Back-Up

```
mysqldump -p -u admin_backup / --flush-logs / --all-databases / > /backups/backup-20191015.sql
```

Physical Back-Up

```
cd /backups/temp/
rsync -a /var/lib/mysql/ .
tar -czf .. /backup-20191015.tgz *
```

Documentation on `mysqldump`: https://mariadb.com/kb/en/mariadb/mysqldump/
Simple Galera Node Back-Up Demonstration
Using Standard Replication

Making Back-Ups with Galera Cluster
Using a Galera Slave

Galera Nodes can be Master to Slave

Slave may be Used for Back-Ups

Extra Requirements for Galera Master & Slave

   - Enable Binary Logs
   - Extra Parameter Settings

Galera Master Configuration

Set `server-id` and `wsrep_gtid_domain_id` & — Same Value on All Nodes

Set `gtid_domain_id` to Unique Values — Different Value than `wsrep_gtid_domain_id`

Enable `wsrep_gtid_mode` and `log-slave-updates` — All Nodes

Enable Binary Log on All Nodes

---

Excerpt from Database Configuration File

```
[mysqld]
...
server-id = 01
gtid_domain_id = 1
wsrep_gtid_domain_id = 0
wsrep_gtid_mode = ON
log_slave_updates = ON
log-bin = /var/lib/mysql/master-bin
log-bin-index = /var/lib/mysql/master-bin.index
```

Galera Options: https://galeracluster.com/library/documentation/mysql-wsrep-options.html
Galera Slave Configuration

Set server-id to Unique Value

Add read-only ON to Prevent Writes

Restart `mysqld`

Excerpt from Database Configuration File

```sql
[mysqld]
...
server-id = 02
gtid_domain_id = 99
log-bin = /var/log/mysql/slave-bin
log-bin-index = /var/log/mysql/slave-bin.index
binlog_format = MIXED
relay-log-index = /var/lib/mysql/slave-relay-bin.index
relay-log = /var/lib/mysql/slave-relay-bin
read-only = 1
innodb-read-only = 1
```
Galera Slave Preparation

Load Data from Master

Execute **CHANGE MASTER** Statement

Execute **START SLAVE** on Slave

### Logical Back-Up

```
mysql -p -u admin_backup / --flush-logs --all-databases / > /backups/backup-20191015.sql
```

Executed from Command-Line

```
mysql -p -u root < backup-20191015.sql
```

Executed from Command-Line

```
CHANGE MASTER TO
MASTER_HOST='172.31.31.202',
MASTER_PORT=3306,
MASTER_USER='replicator',
MASTER_PASSWORD='rover123';
```

Executed from *mysql* Client


Monitoring Replication

Regularly Check Status on Master

Includes Binary Log File Name & Position

Check Often Status of Replication on Slave

Demonstration of Backing-Up a Galera Slave
Using Galera Arbitrator

Making Back-Ups with Galera Cluster
Deciding Vote among Even Number of Nodes
Avoids Split-Brain
Requests Consistent Application State Snapshot
Used for Making Back-Ups

Galera Arbitrator: https://galeracluster.com/library/documentation/arbitrator.html
Back-Ups with Galera Arbitrator

Arbitrator Receives Back-Up Request
  Manual or Automated (e.g., cron)
Node is Chosen for SST – Donor
  Desynchronized from Cluster
Back-Up Script is Run
Donor Node is Resynchronized

Galera Cluster

Node 1

Arbitrator

Node 2

Node 3

Desynchronized Temporarily

Back-Up
Configure Galera Arbitrator

Configuration File for Arbitrator

Name of Cluster
IP Addresses of Nodes — Ports Optional
Local IP Addresses (i.e., 0.0.0.0) & Port
Back-Up Node (i.e., Donor)
Naming of Back-Up Script
Path & Name of Log File

Execute `garbd` with `--cfg`

```
group='galera-training'
address="gcomm://172.31.30.39:4567,
          172.31.18.53:4567,
          172.31.26.106:4567"

options="gmcast.listen_addr=tcp://0.0.0.0:4444"
donor="galera-3"

sst='backup_mysqldump'

log='/var/log/garbd.log'
```

Contents of `/etc/garbd.cnf` File

```
garbd --cfg /etc/garbd.cnf
```

Executed from the Command-Line
Demonstration of Back-Ups with Galera Arbitrator
Restoring Nodes

Making Back-Ups with Galera Cluster
Galera Node Failure Scenarios

One Node Crashed in Cluster
   Start Fresh without Restoring Back-Up
All Nodes Down with Good Data
   Restart Most Up-to-Date Node First
Data on All Nodes Lost
   Restore Data from Back-Up

Node Failure and Recovery: https://galeracluster.com/library/documentation/recovery.html
Recovering a Single Node

Uninstall MySQL or MariaDB
Delete Database Directory
Install MySQL or MariaDB and Secure
Edit Configuration File
Start Database

Executed from Command-Line

```
systemctl stop mysqld
yum remove mysql mysql-server
rm -rf /var/lib/mysql
yum install mysql mysql-server
systemctl start mysqld
mysql_secure_installation
vi /etc/my.cnf
systemctl start mysqld
```
Restarting a Cluster

Determine Most Up-to-Date Node

View Each Node’s grastate.dat File

Ensure UUID Values the Same for All Nodes

Find Node with Highest Sequence Number

Restart Most Up-to-Date Node First

Use `mysqld_bootstrap` on MySQL

Use `galera_new_cluster` on MariaDB

Start Other Nodes

Use `systemctl`
Restoring All Nodes

Install Software on All Nodes — Without Data

- Secure Database (e.g., root password)
- Edit Configuration File

Restore Data on One or All Nodes

- On One Node is Simpler
- All Nodes is Potentially Faster

Start Nodes

- Start & Check Seed Node
- Start & Check Other Nodes
Demonstration of Recovering a Galera Node & Cluster
Back-Up & Restoration Plan

Making Back-Ups with Galera Cluster
Take Inventory

Assemble Information on Nodes

- List of Key Software and Versions
- Keep Printed Copies of Configuration Files

Assess & Assign Staff

- DBAs with Needed Skills
- Who Does Back-Ups — At Least Two
- Most Skilled for Restoring Nodes & Cluster
Develop a Back-Up Plan

Make a Back-Up Schedule
- Which Days & Times
- Which Nodes Used
- Where are Copies Kept Off-Site

Regularly Look for Trouble
- Review Error Logs for Warnings & Error Messages
- Reads Messages from Codership for Security Vulnerabilities
- Keep Database and Galera Cluster Software Up-to-Date
Verification & Restoration

Write a Verification Schedule

- Check File Sizes & Contents – Different DBA
- Ensure Configuration Files are Copied

Create a Restoration Plan

- Practice Restoring Nodes – All DBAs
- Use Test Servers to Assemble New Cluster
Conclusion

Making Back-Ups with Galera Cluster
Additional Resources

Codership Library (galeracluster.com/library)

  Documentation (/library/documentation)
  Knowledge Base (/library/kb)
  FAQ (/library/faq)
  Training (/library/training)
    Videos (/library/training/videos)
    Tutorials (/library/training/tutorials)