

MariaDB Galera Cluster

VS

Percona Xtradb Cluster



PERCONA
XtraDB Cluster

ÁLVARO CAMARGO NÚÑEZ

ADMINISTRACIÓN DE SISTEMAS INFORMÁTICOS EN RED

ÍNDICE

1. INTRODUCCIÓN.....	3
2. ¿QUÉ ES MARIADB GALERA CLÚSTER? CARACTERÍSTICAS Y BENEFICIO.....	4
3. ESCENARIO MARIADB GALERA CLÚSTER.....	6
4. INSTALACIÓN DE MARIADB GALERA CLÚSTER.....	8
5. CONFIGURACIÓN DE MARIADB GALERA CLÚSTER.....	10
6. CREACIÓN DEL CLÚSTER.....	14
7. PRUEBAS DE REPLICACIÓN EN MARIADB GALERA CLÚSTER.....	16
8. ¿QUÉ ES PERCONA XTRADB CLÚSTER? CARACTERÍSTICAS Y BENEFICIOS.....	27
9. ESCENERIO PERCONA XTRADB CLÚSTER.....	30
10. INSTALACIÓN DE PERCONA XTRADB CLÚSTER.....	31
11. CONFIGURACIÓN DE PERCONA XTRADB CLUSTER.....	34
12. CREACIÓN DEL CLÚSTER.....	36
13. PRUEBAS DE REPLICACIÓN EN PERCONA XTRADB CLÚSTER.....	38
14. MARIADB GALERA CLÚSTER VS PERCONA XTRADB-CLÚSTER.....	55

1. INTRODUCCIÓN

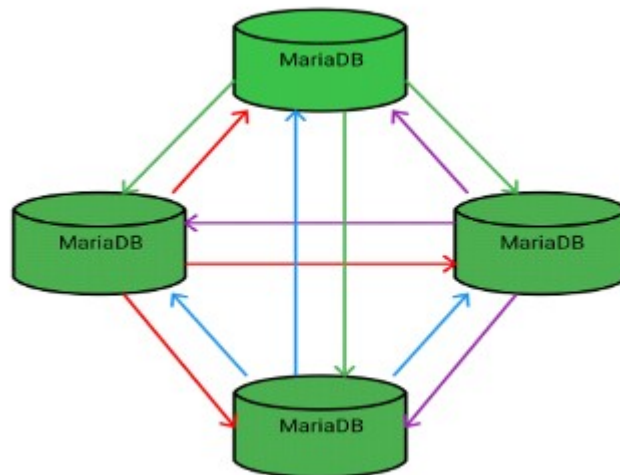
En este proyecto realizaremos 2 clústeres síncrono multimaestro para MariaDB. Un clúster será referente a MariaDB Galera Clúster y el otro a Percona XtraDB Clúster .

MariaDB Galera Cluster es un clúster síncrono multimaestro para MariaDB. Está disponible solo en Linux y solo es compatible con los motores de almacenamiento XtraDB / InnoDB .

MariaDB es un sistema gestor de bases de datos (**SGBD**), es decir, un conjunto de programas que permiten modificar, almacenar, y extraer información de una base de datos. Disponiendo de otro tipo de funcionalidades como la administración de usuarios, y recuperación de la información si el sistema se corrompe, entre otras.

MariaDB surge a raíz de la compra, de la compañía desarrolladora de otro (SGBD) llamado MySQL, por la empresa Sun Microsystems. El desarrollador original, decide tomar el código fuente original de MySQL y genera un derivado con mejoras y cambios a los que llama MariaDB. Permitiendo así la existencia de una versión de este producto con licencia GPL (General Public License).

2. ¿QUÉ ES MARIADB GALERA CLÚSTER?. CARACTERÍSTICAS Y BENEFICIOS.



MariaDB Galera Clúster es un clúster síncrono multimaestro para MariaDB. Está disponible solo en Linux y solo es compatible con los motores de almacenamiento XtraDB / InnoDB . Es un software avanzado de agrupación de bases de datos que no proporciona pérdida de datos, gran tiempo de actividad y escalabilidad para el crecimiento.

* Características.

- Replicación síncrona
- Topología multimaestro activa-activa
- Leer y escribir en cualquier nodo del clúster
- Control automático de pertenencia, los nodos fallidos caen del clúster
- Unión automática de nodos
- Replicación paralela verdadera, a nivel de fila
- Conexiones directas con el cliente, aspecto y sensación nativos de MariaDB

* Beneficios

Las características anteriores producen varios beneficios para una solución de agrupación de DBMS, las cuales son:

- No hay retraso de esclavos
- No hay transacciones perdidas
- Escalabilidad tanto de lectura como de escritura
- Latencias de cliente más pequeñas

3. ESCENARIO GALERA CLÚSTER.

El escenario que vamos a montar para mostrar el uso de Galera Clúster estará compuesto por 4 máquinas virtuales haciendo uno de Vagrant. Cada máquina tendrá una IP privada la cual será 192.168.3.X.

Servername	IP adress
Galera	192.168.3.17 0
nodo1	192.168.3.17 1
nodo2	192.168.3.17 2
Nodo3	192.168.3.17 3

VagrantFile:

```
Vagrant.configure("2") do |config|
```

```
config.vm.define :galera do |galera|
```

```
galera.vm.box = "debian/buster64"
```

```
galera.vm.hostname = "galera"
```

```
galera.vm.network :public_network, :bridge=>"eth0"
```

```
galera.vm.network :private_network, ip: "192.168.3.170"
```

```
end
```

```
config.vm.define :nodo1 do |nodo1|
```

```
nodo1.vm.box = "debian/buster64"
```

```
nodo1.vm.hostname = "nodo1"
```

```
nodo1.vm.network :public_network, :bridge=>"eth0"
```

```
nodo1.vm.network :private_network, ip: "192.168.3.171"
```

```
end
```

```
config.vm.define :nodo2 do |nodo2|
  nodo2.vm.box = "debian/buster64"
  nodo2.vm.hostname = "nodo2"
  nodo2.vm.network :public_network, :bridge=>"eth0"
  nodo2.vm.network :private_network, ip: "192.168.3.172"

  end

config.vm.define :nodo3 do |nodo3|
  nodo3.vm.box = "debian/buster64"
  nodo3.vm.hostname = "nodo3"
  nodo3.vm.network :public_network, :bridge=>"eth0"
  nodo3.vm.network :private_network, ip: "192.168.3.173"

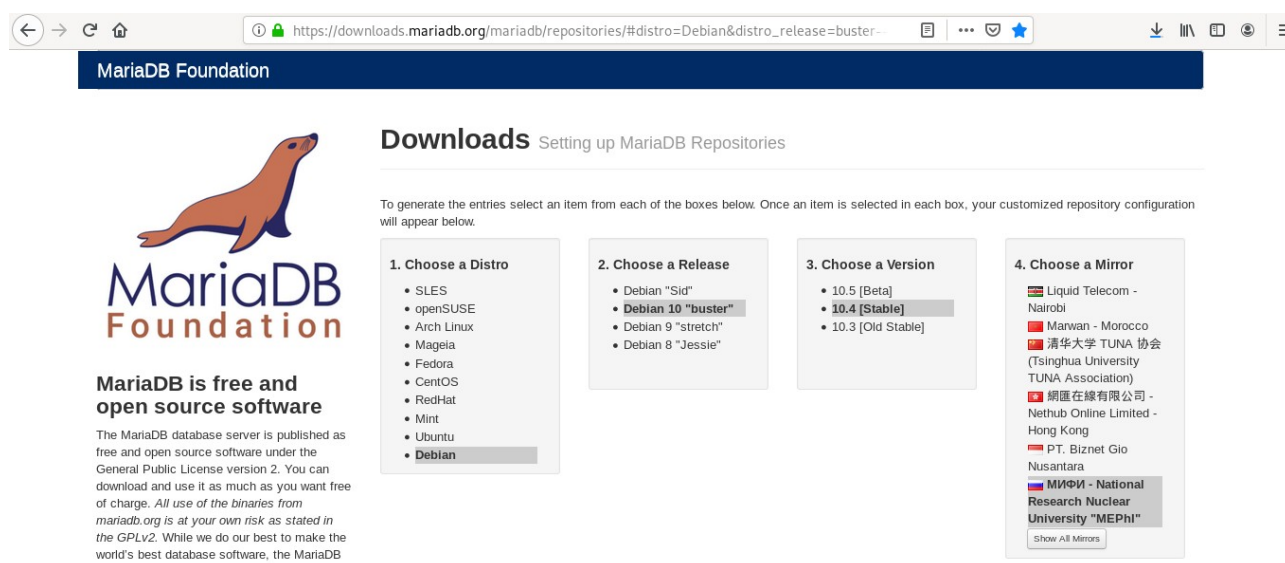
  end

config.vm.box = "debian/buster64"
end
```

4. INSTALACIÓN DE MARIADB GALERA CLUSTER

A continuación mostraremos el repositorio oficial de MariaDB e instalamos el software requerido. Los siguientes comandos que mostraremos debe ejecutarse además en los 3 nodos.

Para averiguar el repositorio debe ser configurado, puede consultarse el sitio <https://downloads.mariadb.org/mariadb/repositories/>



- Instalamos los siguientes paquetes.

```
vagrant@galera:~$ sudo apt-get install software-properties-common dirmngr
```

```
vagrant@galera:~$ sudo apt-key adv --fetch-keys 'https://mariadb.org/mariadb_release_signing_key.asc'
```

```
Executing: /tmp/apt-key-gpghome.E6EdrSbQ1c/gpg.1.sh --fetch-keys https://mariadb.org/mariadb_release_signing_key.asc
```



```
gpg:                requesting                key                from
'https://mariadb.org/mariadb_release_signing_key.asc'
```

```
gpg: key F1656F24C74CD1D8: public key "MariaDB Signing Key <signing-
key@mariadb.org>" imported
```

```
gpg: Total number processed: 1
```

```
gpg:                imported: 1
```

```
vagrant@galera:~$
```

- Añadimos el siguiente repositorio.

```
vagrant@galera:~$ sudo add-apt-repository 'deb [arch=amd64]
http://mirror.mephi.ru/mariadb/repo/10.4/debian buster main'
```

Una vez que se importa la clave, se agrega el repositorio y actualizamos, puede instalar MariaDB con:

```
vagrant@galera:~$ sudo apt update
```

```
vagrant@galera:~$ sudo apt-get install mariadb-server
```

5. CONFIGURACIÓN DE MARIADB GALERA CLÚSTER

En todos los nodos, se edita o se crea el archivo
/etc/mysql/mariadb.conf.d/galera.cnf

```
sudo nano /etc/mysql/mariadb.conf.d/galera.cnf
```

Se agrega el siguiente contenido

En la máquina galera:

```
vagrant@galera:~$ cat /etc/mysql/mariadb.conf.d/galera.cnf
```

```
[mysqld]
```

```
binlog_format=ROW
```

```
default-storage-engine=innodb
```

```
innodb_autoinc_lock_mode=2
```

```
bind-address=0.0.0.0
```

```
# Galera Provider Configuration
```

```
wsrep_on=ON
```

```
wsrep_provider=/usr/lib/galera/libgalera_smm.so
```

```
# Galera Cluster Configuration
```

```
wsrep_cluster_name="db_cluster_camargo"
```

```
wsrep_cluster_address="gcomm://  
192.168.3.171,192.168.3.172,192.168.3.173"
```

```
# Galera Synchronization Configuration
```

```
wsrep_sst_method=rsync
```

```
# Can use mariabackup: apt install mariadb-backup
```

```
# https://mariadb.com/kb/en/library/mariadb-backup-overview/#using-  
mariadb-backup-for-galera-ssts
```

```
#wsrep_sst_method=mariabackup
```

```
#wsrep_sst_auth=USER:PASSWORD
```

```
# Galera Node Configuration

wsrep_node_address="192.168.3.170"

wsrep_node_name="galera"

vagrant@galera:~$

En nodo1:

vagrant@nodo1:~$ cat /etc/mysql/mariadb.conf.d/galera.cnf

[mysqld]

binlog_format=ROW

default-storage-engine=innodb

innodb_autoinc_lock_mode=2

bind-address=0.0.0.0

# Galera Provider Configuration

wsrep_on=ON

wsrep_provider=/usr/lib/galera/libgalera_smm.so

# Galera Cluster Configuration

wsrep_cluster_name="db_cluster_camargo"

wsrep_cluster_address="gcomm://
192.168.3.170,192.168.3.172,192.168.3.173"

# Galera Synchronization Configuration

wsrep_sst_method=rsync

# Can use mariabackup: apt install mariadb-backup

# https://mariadb.com/kb/en/library/mariadb-backup-overview/#using-
mariadb-backup-for-galera-ssts

#wsrep_sst_method=mariabackup

#wsrep_sst_auth=USER:PASSWORD
```

```
# Galera Node Configuration

wsrep_node_address="192.168.3.171"

wsrep_node_name="nodo1"

vagrant@nodo1:~$
```

En nodo2:

```
vagrant@nodo2:~$ cat /etc/mysql/mariadb.conf.d/galera.cnf

[mysqld]

binlog_format=ROW

default-storage-engine=innodb

innodb_autoinc_lock_mode=2

bind-address=0.0.0.0

# Galera Provider Configuration

wsrep_on=ON

wsrep_provider=/usr/lib/galera/libgalera_smm.so

# Galera Cluster Configuration

wsrep_cluster_name="db_cluster_camargo"

wsrep_cluster_address="gcomm://
192.168.3.170,192.168.3.171,192.168.3.173"

# Galera Synchronization Configuration

wsrep_sst_method=rsync

# Can use mariabackup: apt install mariadb-backup
```

```
# https://mariadb.com/kb/en/library/mariadb-backup-overview/#using-
mariadb-backup-for-galera-ssts

#wsrep_sst_method=mariabackup

#wsrep_sst_auth=USER:PASSWORD

# Galera Node Configuration

wsrep_node_address="192.168.3.172"

wsrep_node_name="nodo2"

vagrant@nodo2:~$

En nodo3:

vagrant@nodo3:~$ cat /etc/mysql/mariadb.conf.d/galera.cnf

[mysqld]

binlog_format=ROW

default-storage-engine=innodb

innodb_autoinc_lock_mode=2

bind-address=0.0.0.0

# Galera Provider Configuration

wsrep_on=ON

wsrep_provider=/usr/lib/galera/libgalera_smm.so

# Galera Cluster Configuration

wsrep_cluster_name="db_cluster_camargo"

wsrep_cluster_address="gcomm://
192.168.3.170,192.168.3.171,192.168.3.172"

# Galera Synchronization Configuration

wsrep_sst_method=rsync

# Can use mariabackup: apt install mariadb-backup
```

```
# https://mariadb.com/kb/en/library/mariadb-backup-overview/#using-
mariadb-backup-for-galera-ssts

#wsrep_sst_method=mariabackup

#wsrep_sst_auth=USER:PASSWORD

# Galera Node Configuration

wsrep_node_address="192.168.3.173"

wsrep_node_name="nodo3"

vagrant@nodo3:~$
```

Tenemos que tener en cuenta que `wsrep_node_address` y `wsrep_node_name` deben coincidir con la dirección IP y el nombre del nodo en donde se está creando el archivo.

6. CREACIÓN DEL CLÚSTER

Opcionalmente, para asegurar la instalación de MariaDB y borrar las bases de datos de pruebas se ejecuta el siguiente comando solamente en uno de los nodos en nuestro caso lo realizaremos en la máquina Galera.

```
sudo mysql_secure_installation
```

Se detiene el servicio de MariaDB en todos los nodos ejecutando el siguiente comando en cada uno de ellos

```
sudo systemctl stop mysql
```

Se verifica que el servicio esté detenido

```
vagrant@galera:~$ sudo systemctl is-active mysql
```

```
inactive
```

```
vagrant@galera:~$
```

Ahora, a continuación se crea el nuevo cluster de Galera ejecutando el siguiente comando solo en la máquina Galera.

```
vagrant@galera:~$ sudo galera_new_cluster
```

Se verifica que el clúster haya inicializado. El valor será de 1 ya que el servicio de Mariadb en las máquinas están inactivo.

```
vagrant@galera:~$ mysql -u root -p -e "SHOW STATUS LIKE 'wsrep_cluster_size'"
```

```
Enter password:
```

```
+-----+-----+
| Variable_name | Value |
+-----+-----+
| wsrep_cluster_size | 1 |
+-----+-----+
```

```
vagrant@galera:~$
```

A continuación se inicia el servicio de MariaDB normalmente en los otros nodos .

```
sudo systemctl start mysql
```

Y a comprobamos que los nodos hayan sido agregados al clúster

```
vagrant@galera:~$ mysql -u root -p -e "SHOW STATUS LIKE 'wsrep_cluster_size'"
```

```
Enter password:
```

```
+-----+-----+
| Variable_name | Value |
+-----+-----+
```

```
| wsrep_cluster_size | 4 |
```

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

```
vagrant@galera:~$
```

Debemos de recordar que la replicación de Galera solo funciona sobre bases de datos en InnoDB. En las base de datos de system como 'mysql' (MyISAM) no son replicadas, excepto en la replicación inicial.

7. PRUEBAS DE REPLICACIÓN EN MARIADB GALERA CLUSTER

Se crea la base de datos por ejemplo auditorio en la máquina Galera y crearemos una serie de tablas e introducimos datos a dichas tablas.

```
root@galera:/home/vagrant# mysql -u root -p
```

```
Enter password:
```

```
Welcome to the MariaDB monitor. Commands end with ; or \g.
```

```
Your MariaDB connection id is 42
```

```
Server version: 10.4.12-MariaDB-1:10.4.12+maria~buster-log mariadb.org  
binary distribution
```

```
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.
```

```
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```



```
MariaDB [(none)]> create database auditorio;
```

```
Query OK, 1 row affected (0.013 sec)
```

```
MariaDB [(none)]> use auditorio;
```

```
Database changed
```

```
MariaDB [auditorio]> create table compositores
```

```
-> (
```

```
-> ID varchar(5),
```

```
-> Nombre varchar(50),
```

```
-> FechaNacimiento date,
```

```
-> Nacionalidad varchar(30),
```

```
-> URLBiografia varchar(100),
```

```
-> constraint fk_compositores primary key(ID),
```

```
-> constraint nacimiento check (year(fechanacimiento) between '1290' AND  
'1920' ),
```

```
-> constraint url check(URLBiografia REGEXP 'www\.[a-z]*\.(com|es|org|  
net)$');
```

```
Query OK, 0 rows affected, 1 warning (0.204 sec)
```

```
MariaDB [auditorio]>
```

```
MariaDB [auditorio]> insert into compositores values ('01','LUDWING VAN  
BEETHOWEN',STR_TO_DATE('1770-12-16','%Y-%m-  
%d'),'alemana','www.ludwingvanbeethoven.org');
```

```
Query OK, 1 row affected (0.014 sec)
```

```
MariaDB [auditorio]> insert into compositores values ('02','WOLFGANG  
AMADEUS MOZART',STR_TO_DATE('1756-01-27','%Y-%m-  
%d'),'austriaca','www.mozart.com');
```

Query OK, 1 row affected (0.027 sec)

```
MariaDB [auditorio]> insert into compositores values ('03','ANTONIO  
VIVALDI',STR_TO_DATE('1678-03-04','%Y-%m-  
%d'),'italiana','www.antoniovivaldi.net');
```

Query OK, 1 row affected (0.008 sec)

```
MariaDB [auditorio]> insert into compositores values ('04','FRÉDERIC  
CHOPIN',STR_TO_DATE('1810-03-01','%Y-%m-  
%d'),'polaca','www.fredericchopin.com');
```

Query OK, 1 row affected (0.009 sec)

```
MariaDB [auditorio]> insert into compositores values ('05','MANUEL DE  
FALLA',STR_TO_DATE('1876-11-23','%Y-%m-  
%d'),'espanola','www.manueldefalla.com');
```

Query OK, 1 row affected (0.010 sec)

```
MariaDB [auditorio]> insert into compositores values ('06','FRANZ  
SCHUBERT',STR_TO_DATE('1797-01-31','%Y-%m-  
%d'),'austriaca','www.franzschubert.com');
```

Query OK, 1 row affected (0.010 sec)

```
MariaDB [auditorio]> insert into compositores values ('07','LOUISS  
SPOHR',STR_TO_DATE('1784-04-05','%Y-%m-  
%d'),'alemana','www.louisspohr.com');
```

Query OK, 1 row affected (0.010 sec)

```
MariaDB [auditorio]> insert into compositores values ('08','ROBERT SCHUMANN',STR_TO_DATE('1810-06-08','%Y-%m-%d'),'alemana','www.robertschumann.es');
```

Query OK, 1 row affected (0.014 sec)

```
root@galera:/home/vagrant# mysql -u root -p
```

Enter password:

Welcome to the MariaDB monitor. Commands end with ; or \g.

Your MariaDB connection id is 43

Server version: 10.4.12-MariaDB-1:10.4.12+maria~buster-log mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

```
MariaDB [(none)]> use auditorio;
```

Reading table information for completion of table and column names

You can turn off this feature to get a quicker startup with -A

Database changed

```
MariaDB [auditorio]> select * from compositores;
```

```
+----+-----+-----+-----+-----+
| ID | Nombre                | FechaNacimiento | Nacionalidad | URLBiografia |
+----+-----+-----+-----+-----+
```

01	LUDWING VAN BEETHOWEN	1770-12-16	alemana	
www.ludwingvanbeethoven.org				
02	WOLFGANG AMADEUS MOZART	1756-01-27	austriaca	
www.mozart.com				
03	ANTONIO VIVALDI	1678-03-04	italiana	
www.antoniovivaldi.net				
04	FRÉDÉRIC CHOPIN	1810-03-01	polaca	
www.fredericchopin.com				
05	MANUEL DE FALLA	1876-11-23	espanola	
www.manueldefalla.com				
06	FRANZ SCHUBERT	1797-01-31	austriaca	
www.franzschubert.com				
07	LOUISS SPOHR	1784-04-05	alemana	
www.louisspohr.com				
08	ROBERT SCHUMANN	1810-06-08	alemana	
www.robertschumann.es				

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

8 rows in set (0.001 sec)

MariaDB [auditorio]>

Se observa que en el nodo1 se ha replicado la base de datos, la tabla y los datos.

```
root@nodo1:/home/vagrant# mysql -u root -p
```

Enter password:

Welcome to the MariaDB monitor. Commands end with ; or \g.

Your MariaDB connection id is 42

Server version: 10.4.12-MariaDB-1:10.4.12+maria~buster-log mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> use auditorio;

Reading table information for completion of table and column names

You can turn off this feature to get a quicker startup with -A

Database changed

MariaDB [auditorio]> select * from compositores;

```
+----+-----+-----+-----+-----+
| ID | Nombre                | FechaNacimiento | Nacionalidad | URLBiografia
|
+----+-----+-----+-----+-----+
| 01 | LUDWING VAN BEETHOWEN | 1770-12-16      | alemana      |
www.ludwingvanbeethoven.org |
| 02 | WOLFGANG AMADEUS MOZART | 1756-01-27      | austriaca    |
www.mozart.com           |
| 03 | ANTONIO VIVALDI        | 1678-03-04      | italiana      |
www.antoniovivaldi.net  |
| 04 | FRÉDÉRIC CHOPIN        | 1810-03-01      | polaca       |
www.fredericchopin.com  |
| 05 | MANUEL DE FALLA        | 1876-11-23      | espanola     |
www.manueldefalla.com   |
| 06 | FRANZ SCHUBERT         | 1797-01-31      | austriaca    |
www.franzschubert.com   |
```

```
| 07 | LOUISS SPOHR          | 1784-04-05      | alemana      |
www.louisspohr.com      |
```

```
| 08 | ROBERT SCHUMANN        | 1810-06-08      | alemana      |
www.robertschumann.es  |
```

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

8 rows in set (0.003 sec)

MariaDB [auditorio]>

Además insertaremos nuevos datos desde nodo1.

```
root@nodo1:/home/vagrant# mysql -u root -p
```

Enter password:

Welcome to the MariaDB monitor. Commands end with ; or \g.

Your MariaDB connection id is 42

Server version: 10.4.13-MariaDB-1:10.4.13+maria~buster-log mariadb.org
binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

```
MariaDB [(none)]> use auditorio;
```

Reading table information for completion of table and column names

You can turn off this feature to get a quicker startup with -A

Database changed

```
MariaDB [auditorio]> create table interpretes
```

```
-> (
```

```
-> id_interprete varchar(10),  
-> nombre_interprete varchar(50),  
-> constraint pk_interpretes primary key (id_interprete),  
-> constraint contar check(length(id_interprete)>='5'));
```

Query OK, 0 rows affected, 1 warning (0.058 sec)

Comprobamos desde nodo2 la replicación de la nueva tabla e insertaremos datos a dicha tabla.

```
root@nodo2:/home/vagrant# mysql -u root -p
```

Enter password:

Welcome to the MariaDB monitor. Commands end with ; or \g.

Your MariaDB connection id is 41

Server version: 10.4.13-MariaDB-1:10.4.13+maria~buster-log mariadb.org
binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

```
MariaDB [(none)]> use auditorio;
```

Reading table information for completion of table and column names

You can turn off this feature to get a quicker startup with -A

Database changed

```
MariaDB [auditorio]> insert into interpretes values ('123456','MAURICE  
ANDRE');
```

Query OK, 1 row affected (0.031 sec)

```
MariaDB [auditorio]> insert into interpretes values ('1er456','CLAUDIO ARRAU');
```

```
Query OK, 1 row affected (0.016 sec)
```

```
MariaDB [auditorio]> insert into interpretes values ('654tgb','DANIEL BARENBOIM');
```

```
Query OK, 1 row affected (0.017 sec)
```

```
MariaDB [auditorio]> insert into interpretes values ('543frtyh','YURI BASHMET');
```

```
Query OK, 1 row affected (0.015 sec)
```

```
MariaDB [auditorio]> insert into interpretes values ('12654f','HEINZ HOLLIGER');
```

```
Query OK, 1 row affected (0.030 sec)
```

```
MariaDB [auditorio]> insert into interpretes values ('qazwsxedc','VLADIMIR HOROWITZ');
```

```
Query OK, 1 row affected (0.016 sec)
```

```
MariaDB [auditorio]> insert into interpretes values ('1mkonji','YEHUDI MENUHIN');
```

```
Query OK, 1 row affected (0.025 sec)
```

A continuación añadiremos tablas de la base de datos desde el nodo3 y comprobaremos desde la máquina principal galera si se a realizado la replicación.


```
root@nodo3:/home/vagrant# mysql -u root -p
```

```
Enter password:
```

```
Welcome to the MariaDB monitor.  Commands end with ; or \g.
```

```
Your MariaDB connection id is 17
```

```
Server version: 10.4.13-MariaDB-1:10.4.13+maria~buster-log mariadb.org  
binary distribution
```

```
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.
```

```
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

```
MariaDB [(none)]> use auditorio;
```

```
Reading table information for completion of table and column names
```

```
You can turn off this feature to get a quicker startup with -A
```

```
Database changed
```

```
MariaDB [auditorio]> create table solistas
```

```
-> (
```

```
-> IDinterprete_solista varchar(10),
```

```
-> nombre_artistico varchar(50),
```

```
-> instrumento varchar(100),
```

```
-> constraint pk_solistas primary key (IDinterprete_solista),
```

```
    -> constraint solistas_interpretes foreign key (IDinterprete_solista)  
references interpretes(id_interprete));
```

```
Query OK, 0 rows affected, 1 warning (0.059 sec)
```

```
MariaDB [auditorio]>
```

```
root@galera:/home/vagrant# mysql -u root -p
```

```
Enter password:
```

```
Welcome to the MariaDB monitor.  Commands end with ; or \g.
```

```
Your MariaDB connection id is 44
```

```
Server version: 10.4.13-MariaDB-1:10.4.13+maria~buster-log mariadb.org  
binary distribution
```

```
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.
```

```
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

```
MariaDB [(none)]> use auditorio;
```

```
Reading table information for completion of table and column names
```

```
You can turn off this feature to get a quicker startup with -A
```

```
Database changed
```

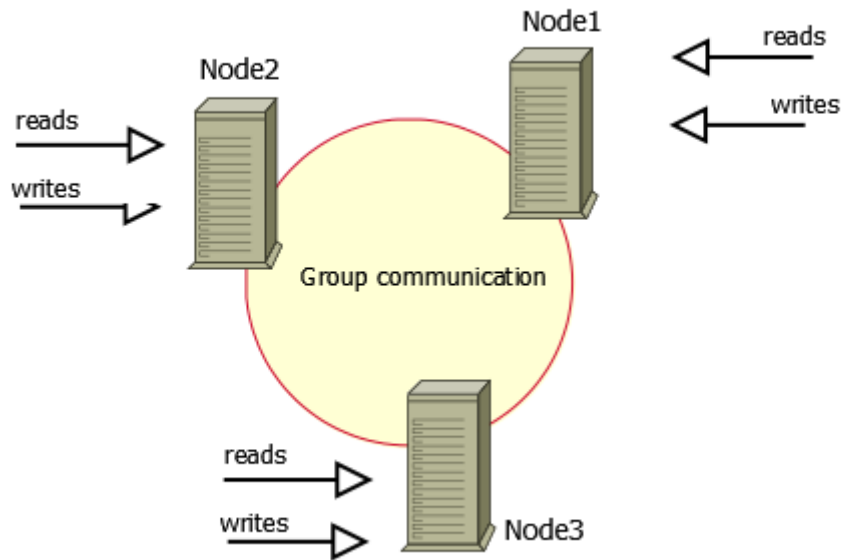
```
MariaDB [auditorio]> describe solistas;
```

```
+-----+-----+-----+-----+-----+-----+  
| Field          | Type          | Null | Key | Default | Extra |  
+-----+-----+-----+-----+-----+-----+  
| IDinterprete_solista | varchar(10) | NO   | PRI | NULL    |      |  
| nombre_artistico   | varchar(50) | YES  |     | NULL    |      |  
| instrumento        | varchar(100)| YES  |     | NULL    |      |  
+-----+-----+-----+-----+-----+-----+
```

```
3 rows in set (0.002 sec)
```

```
MariaDB [auditorio]>
```

8. ¿QUÉ ES PERCONA XTRADB CLUSTER?. CARACTERÍSTICAS Y BENEFICIOS.



Un clúster consta de nodos, donde cada nodo contiene el mismo conjunto de datos sincronizados entre nodos. La configuración recomendada es tener al menos 3 nodos, pero también puede tener 2 nodos. Cada nodo es una instancia regular del servidor MySQL (por ejemplo, el servidor Percona). Puede convertir una instancia de MySQL Server existente en un nodo y ejecutar el clúster utilizando este nodo como base. También puede separar cualquier nodo del clúster y usarlo como una instancia regular de MySQL Server.

*Beneficios

- Cuando ejecuta una consulta, se ejecuta localmente en el nodo. Todos los datos están disponibles localmente, sin necesidad de acceso remoto.
- Sin gestión central. Puede perder cualquier nodo en cualquier momento y el clúster continuará funcionando sin pérdida de datos.
- Buena solución para escalar una carga de trabajo de lectura. Puede poner consultas de lectura en cualquiera de los nodos.

Inconvenientes

- Sobrecarga de aprovisionamiento nuevo nodo. Cuando agrega un nuevo nodo, tiene que copiar el conjunto de datos completo de uno de los nodos existentes. Si es de 100 GB, copia 100 GB.
- Esto no se puede usar como una solución efectiva de escalado de escritura.
- Puede haber algunas mejoras en el rendimiento de escritura cuando ejecuta el tráfico de escritura en 2 nodos frente a todo el tráfico en 1 nodo, pero no puede esperar mucho. Todas las escrituras todavía tienen que ir en todos los nodos.
- La replicación solo funciona con el motor de almacenamiento InnoDB. Cualquier escritura en tablas de otros tipos no se replica.
- Tiene varios duplicados de datos: para 3 nodos tiene 3 duplicados.

Componentes

Percona XtraDB Cluster se basa en el servidor Percona que se ejecuta con el motor de almacenamiento XtraDB. Utiliza la biblioteca Galera, que es una implementación de la API de replicación de conjunto de escritura (wsrep) desarrollada por Codership Oy. El método de transferencia de datos predeterminado y recomendado es a través de Percona XtraBackup.

Características del clúster XtraDB de Percona

Replicación sincrónica	Los datos se escriben en todos los nodos simultáneamente, o no se escriben en caso de falla incluso en un solo nodo
Replicación multimaestro	Cualquier nodo puede desencadenar una actualización de datos.
Replicación paralela verdadera	Varios subprocesos en el esclavo que realiza la replicación en el nivel de fila
Aprovisionamiento automático de nodos	Simplemente agrega un nodo y se sincroniza automáticamente.
Consistencia de los datos	No más nodos no sincronizados.
Modo estricto PXC	Evita el uso de funciones experimentales y no compatibles
Script de configuración para ProxySQL	Percona XtraDB Cluster incluye la herramienta proxysql-admin que configura automáticamente los nodos de Percona XtraDB Clúster usando ProxySQL.
Configuración automática de encriptación SSL	Percona XtraDB Cluster incluye la variable pxc-encrypt-cluster-traffic que permite la configuración automática del cifrado SSL
Rendimiento optimizado	El rendimiento del Percona XtraDB Clúster está optimizado para escalar con una creciente carga de trabajo de producción

9. ESCENARIO PERCONA XTRADB CLUSTER.

Tendremos 3 servidores . A continuación mostraremos el nombre del servidor y su correspondiente IP.

Servername	IP adress
pxc1	192.168.4 .170
pxc2	192.168.4 .171
pxc3	192.168.4 .172

VagrantFile:

```
Vagrant.configure("2") do |config|
  config.vm.define :pxc do |pxc|
    pxc.vm.box = "debian/buster64"
    pxc.vm.hostname = "pxc"
    pxc.vm.network :public_network, :bridge=>"eth0"
    pxc.vm.network :private_network, ip: "192.168.5.170"

    end

  config.vm.define :pxc1 do |pxc1|
    pxc1.vm.box = "debian/buster64"
    pxc1.vm.hostname = "pxc1"
    pxc1.vm.network :public_network, :bridge=>"eth0"
    pxc1.vm.network :private_network, ip: "192.168.5.171"

    end

  config.vm.define :pxc2 do |pxc2|
    pxc2.vm.box = "debian/buster64"
    pxc2.vm.hostname = "pxc2"
    pxc2.vm.network :public_network, :bridge=>"eth0"
    pxc2.vm.network :private_network, ip: "192.168.5.172"

    end

  config.vm.box = "debian/buster64"
end
```

Actualizamos los servidores.

```
sudo apt-get update
```

```
sudo apt-get upgrade
```

10. INSTALACIÓN DE PERCONA XTRADB CLUSTER

— PREREQUISITOS

Si no estamos ejecutando un servidor DNS, debemos de configurar su archivo / etc / hosts en cada nodo.

```
GNU nano 2.7.4  
Modified
```

File: /etc/hosts

```
192.168.5.170 pxc  
192.168.5.171 pxc2  
192.168.5.172 pxc3
```

Debemos de asegurarnos de tener los siguientes puertos abiertos. Percona XtraDB Clúster los requiere para la comunicación.

3306

4444

4567

4568

Si anteriormente tenemos instalado MySQL en el servidor, puede haber un perfil de AppArmor que evitará que los nodos de Percona XtraDB Cluster se comuniquen entre sí. La mejor solución es eliminar el paquete apparmor por completo:

```
$ sudo apt-get remove apparmor
```

Si necesitamos tener AppArmor habilitado debido a políticas de seguridad o por otras razones, es posible deshabilitar o ampliar el perfil MySQL.

- Instalamos Percona_XtraDB-Cluster en pxc,pxc2 y pxc3

- Primeramente instalamos los siguiente paquete para descargar contenidos de percona.

```
sudo apt-get install -y wget gnupg2 lsb-release
```

```
sudo apt-get -y install wget
```

Obtenemos el paquete del repositorio en todos los servidores:

```
wget https://repo.percona.com/apt/percona-release_latest.generic_all.deb
```

Instalamos el paquete de repositorio descargado con dpkg:

```
sudo dpkg -i percona-release_latest.generic_all.deb
```

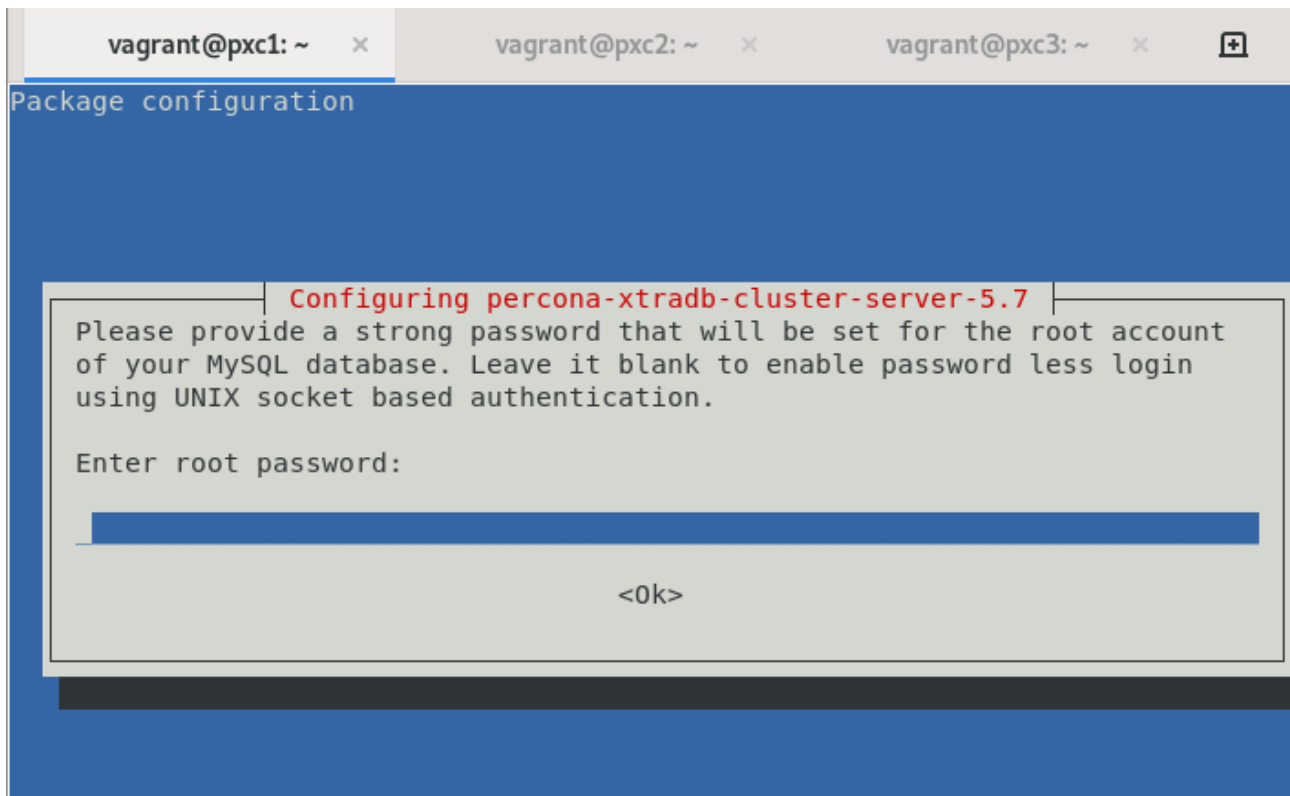
Actualizamos.

```
sudo apt-get update
```

A continuación instamos Percona-Xtradb-Clúster.

```
sudo apt-get -y install percona-xtradb-cluster-57
```

En la instalación nos pedirá que asignemos una contraseña para el usuario root en mysql.



Percona XtraDB Cluster se distribuye con varias UDF (funciones definidas por el usuario) útiles de Percona Toolkit.

Ejecutamos los siguientes comandos para crear estas funciones:

```
mysql -e "CREATE FUNCTION fnv1a_64 RETURNS INTEGER SONAME 'libfnv1a_udf.so'"
```

```
mysql -e "CREATE FUNCTION fnv_64 RETURNS INTEGER SONAME 'libfnv_udf.so'"
```

```
mysql -e "CREATE FUNCTION murmur_hash RETURNS INTEGER SONAME 'libmurmur_udf.so'"
```

Detenemos mysqld en todos los servidores.

```
sudo systemctl stop mysql
```

```
sudo systemctl status mysql
```

● mysql.service - LSB: Start and stop the mysql (Percona XtraDB Cluster) daemon

Loaded: loaded (/etc/init.d/mysql; generated; vendor preset: enabled)

Active: inactive (dead) since Tue 2020-04-21 14:53:59 GMT; 9s ago

Docs: man:systemd-sysv-generator(8)

Process: 1492 ExecStop=/etc/init.d/mysql stop (code=exited, status=0/SUCCESS)

Process: 415 ExecStart=/etc/init.d/mysql start (code=exited, status=0/SUCCESS)

11. CONFIGURACIÓN DE PERCONA_XTRADB-CLÚSTER

Configuramos los servidores. Comenzamos en el primer nodo pxc, modificando el siguiente archivo /etc/mysql/my.cnf. El estado inicial del archivo seria asi;

```
vagrant@pxc:~$ sudo cat /etc/mysql/my.cnf
```

```
#  
# The Percona XtraDB Cluster 5.7 configuration file.  
#  
# * IMPORTANT: Additional settings that can override those from this file!  
# The files must end with '.cnf', otherwise they'll be ignored.  
# Please make any edits and changes to the appropriate sectional files  
# included below.  
#
```

```
!includedir /etc/mysql/conf.d/
```

```
!includedir /etc/mysql/percona-xtradb-cluster.conf.d/
```

A continuación mostraremos la modificación de dicho archivo de configuración. Al final del archivo agregamos las siguientes líneas.

PXC

```
[mysqld]
```

```
server_id=1
```

```
datadir=/var/lib/mysql
```

```
user=mysql
```

```
# Path to Galera library
```

```
wsrep_provider=/usr/lib/libgalera_smm.so
```

```
# La URL de conexión del clúster contiene las direcciones IP del nodo pxc, el  
nodo pxc1 y el nodo pxc2
```

```
wsrep_cluster_address=gcomm://192.168.4.170,192.168.4.171,192.168.4.172
```

```
# Para que Galera funcione correctamente, el formato binlog debe ser ROW  
binlog_format=ROW
```

```
# MyISAM storage engine has only experimental support
default_storage_engine=InnoDB
# Este modo de bloqueo de autoincremento InnoDB es un requisito para
Galera
innodb_autoinc_lock_mode=2
# Dirección de nodo y nombre
wsrep_node_address=192.168.4.170
wsrep_node_name=pxc
# SST method
wsrep_sst_method=xtrabackup-v2
# Nombre de cluster
wsrep_cluster_name=ppnet_cluster
# Autenticación para el método SST
wsrep_sst_auth="sstuser:password"
root@pxc1:/home/vagrant#
```

PXC1

```
[mysqld]
server_id=2
datadir=/var/lib/mysql
user=mysql
# Path to Galera library
wsrep_provider=/usr/lib/libgalera_smm.so
# La URL de conexión del clúster contiene las direcciones IP del nodo pxc, el
nodo pxc1 y el nodo pxc2
wsrep_cluster_address=gcomm://192.168.4.170,192.168.4.171,192.168.4.172
# Para que Galera funcione correctamente, el formato binlog debe ser ROW
binlog_format=ROW
# MyISAM storage engine has only experimental support
default_storage_engine=InnoDB
# Este modo de bloqueo de autoincremento InnoDB es un requisito para
Galera
innodb_autoinc_lock_mode=2
# Dirección de nodo y nombre
wsrep_node_address=192.168.4.171
wsrep_node_name=pxc1
# SST method
wsrep_sst_method=xtrabackup-v2
# Nombre de cluster
wsrep_cluster_name=ppnet_cluster
# Autenticación para el método SST
wsrep_sst_auth="sstuser:password"

root@pxc2:/home/vagrant#
```

PXC2

```
[mysqld]
server_id=3
datadir=/var/lib/mysql
```

```

user=mysql
# Path to Galera library
wsrep_provider=/usr/lib/libgalera_smm.so
# La URL de conexión del clúster contiene las direcciones IP del nodo pxc, el
nodo pxc1 y el nodo pxc2
wsrep_cluster_address=gcomm://192.168.4.170,192.168.4.171,192.168.4.172
# Para que Galera funcione correctamente, el formato binlog debe ser ROW
binlog_format=ROW
# MyISAM storage engine has only experimental support
default_storage_engine=InnoDB
# Este modo de bloqueo de autoincremento InnoDB es un requisito para
Galera
innodb_autoinc_lock_mode=2
# Dirección de nodo y nombre
wsrep_node_address=192.168.4.172
wsrep_node_name=pxc2
# SST method
wsrep_sst_method=xtrabackup-v2
# Nombre de cluster
wsrep_cluster_name=ppnet_cluster
# Autenticación para el método SST
wsrep_sst_auth="sstuser:password"

```

```
root@pxc3:/home/vagrant#
```

12. CREACIÓN DEL CLÚSTER

Iniciamos el primer nodo PXC

```

root@pxc:/home/vagrant# /etc/init.d/mysql bootstrap-pxc
[ ok ] Bootstrapping Percona XtraDB Cluster database server: mysqld already
running.

```

Iniciamos sesión en mysql.

```
root@pxc1:/home/vagrant# mysql -u root -p
```

Comprobamos el número de máquinas del clúster.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

```
mysql> show status like "wsrep_cluster_size";
+-----+-----+
```

```
| Variable_name | Value |
+-----+-----+
| wsrep_cluster_size | 1 |
+-----+-----+
1 row in set (0.01 sec)
```

Antes de comenzar el segundo PXC2 y el tercer nodo PXC3, necesitamos crear el usuario sstuser que configuramos en el archivo my.cnf con su contraseña correspondiente en todos los nodos.

```
CREATE USER 'sstuser'@'localhost' IDENTIFIED BY 'password';
```

```
GRANT PROCESS, RELOAD, LOCK TABLES, REPLICATION CLIENT ON *.*
TO 'sstuser'@'localhost';
```

```
FLUSH PRIVILEGES;
```

Iniciamos mysql en el segundo PXC2 y tercer nodo PXC3. En cada uno de ellas realizamos los pasos anteriores los cuales realizamos en PXC1.

Iniciamos servicios de mysql en PXC1 y PXC2.

```
vagrant@pxc1:~$ sudo systemctl start mysql
```

```
vagrant@pxc2:~$ sudo systemctl start mysql
```

Si no obtenemos errores, eso significa que hemos agregado el nodo al clúster. Para comprobarlo ejecutamos el siguiente comando y deberíamos ver que de una máquina en el clúster debería de aparecernos 3.

```
vagrant@pxc:~$ sudo mysql -p
```

Enter password:

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 14

Server version: 5.7.29-32-57-log Percona XtraDB Cluster (GPL), Release rel32, Revision 1b99a32, WSREP version 31.43, wsrep_31.43

Copyright (c) 2009-2020 Percona LLC and/or its affiliates

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

```
mysql> show status like "wsrep_cluster_size";
```

```
+-----+-----+
| Variable_name | Value |
+-----+-----+
| wsrep_cluster_size | 3 |
+-----+-----+
```

```
1 row in set (0.00 sec)
```

```
mysql>
```

13. PRUEBAS DE REPLICACIÓN EN PERCONA XTRADB CLUSTER.

Comprobamos que funciona correctamente nuestro clúster.

- Creamos una base de datos en pxc.

```
vagrant@pxc:~$ sudo su
```

```
root@pxc:/home/vagrant# mysql -u root -p
```

```
Enter password:
```

```
Welcome to the MySQL monitor.  Commands end with ; or \g.
```

```
Your MySQL connection id is 17
```

```
Server version: 5.7.29-32-57-log Percona XtraDB Cluster (GPL), Release rel32,  
Revision 1b99a32, WSREP version 31.43, wsrep_31.43
```

```
Copyright (c) 2009-2020 Percona LLC and/or its affiliates
```

```
Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.
```

```
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.
```

```
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

```
mysql> create database auditorios;
```

```
Query OK, 1 row affected (0.01 sec)
```

```
Creamos tablas e insertamos datos.
```

```
mysql> create table compositores
```

```
-> (
```

```
-> ID varchar(5),
```

```
-> Nombre varchar(50),
```

-> FechaNacimiento date,
-> Nacionalidad varchar(30),
-> URLBiografia varchar(100),
-> constraint fk_compositores primary key(ID),
-> constraint nacimiento check (year(fechanacimiento) between '1290' AND '1920'),
-> constraint url check(URLBiografia REGEXP 'www\.[a-z]*\.(com|es|org|net)\$');

Query OK, 0 rows affected (0.06 sec)

```
mysql> insert into compositores values ('01','LUDWING VAN BEETHOWEN',STR_TO_DATE('1770-12-16','%Y-%m-%d'),'alemana','www.ludwingvanbeethoven.org');
```

Query OK, 1 row affected (0.04 sec)

```
mysql> insert into compositores values ('02','WOLFGANG AMADEUS MOZART',STR_TO_DATE('1756-01-27','%Y-%m-%d'),'austriaca','www.mozart.com');
```

Query OK, 1 row affected (0.02 sec)

```
mysql> insert into compositores values ('03','ANTONIO VIVALDI',STR_TO_DATE('1678-03-04','%Y-%m-%d'),'italiana','www.antoniovivaldi.net');
```

Query OK, 1 row affected (0.01 sec)

```
mysql> insert into compositores values ('04','FRÉDÉRIC CHOPIN',STR_TO_DATE('1810-03-01','%Y-%m-%d'),'polaca','www.frederic Chopin.com');
```

Query OK, 1 row affected (0.01 sec)


```
mysql> insert into compositores values ('05','MANUEL DE FALLA',STR_TO_DATE('1876-11-23','%Y-%m-%d'),'espanola','www.manueldefalla.com');
```

Query OK, 1 row affected (0.01 sec)

```
mysql> insert into compositores values ('06','FRANZ SCHUBERT',STR_TO_DATE('1797-01-31','%Y-%m-%d'),'austriaca','www.franzschubert.com');
```

Query OK, 1 row affected (0.01 sec)

```
mysql> insert into compositores values ('07','LOUISS SPOHR',STR_TO_DATE('1784-04-05','%Y-%m-%d'),'alemana','www.louisspohr.com');
```

Query OK, 1 row affected (0.02 sec)

```
mysql> insert into compositores values ('08','ROBERT SCHUMANN',STR_TO_DATE('1810-06-08','%Y-%m-%d'),'alemana','www.robertschumann.es');
```

Query OK, 1 row affected (0.00 sec)

```
mysql>
```

```
mysql>
```

- Comprobamos que se ha realizado la replicación de la tabla y datos insertados anteriormente desde pxc desde pxc1.

```
vagrant@pxc1:~$ sudo su
```

```
root@pxc1:/home/vagrant# mysql -u root -p
```

```
Enter password:
```

```
Welcome to the MySQL monitor. Commands end with ; or \g.
```

Your MySQL connection id is 12

Server version: 5.7.29-32-57-log Percona XtraDB Cluster (GPL), Release rel32, Revision 1b99a32, WSREP version 31.43, wsrep_31.43

Copyright (c) 2009-2020 Percona LLC and/or its affiliates

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

```
mysql> show databases
```

```
-> ;
```

```
+-----+
| Database      |
+-----+
| information_schema |
| auditorios     |
| mysql          |
| performance_schema |
| sys           |
+-----+
```

```
5 rows in set (0.02 sec)
```

```
mysql>
```

```
root@pxc1:/home/vagrant# mysql -u root -p
```

```
Enter password:
```

```
ERROR 1045 (28000): Access denied for user 'root'@'localhost' (using password: YES)
```

```
root@pxc1:/home/vagrant# mysql -u root -p
```

```
Enter password:
```

```
Welcome to the MySQL monitor. Commands end with ; or \g.
```

```
Your MySQL connection id is 14
```

```
Server version: 5.7.29-32-57-log Percona XtraDB Cluster (GPL), Release rel32, Revision 1b99a32, WSREP version 31.43, wsrep_31.43
```

```
Copyright (c) 2009-2020 Percona LLC and/or its affiliates
```

```
Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.
```

```
Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.
```

```
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

```
mysql> use auditorios;
```

```
Reading table information for completion of table and column names
```

```
You can turn off this feature to get a quicker startup with -A
```

```
Database changed
```

```
mysql> select * from compositores;
```

```

+----+-----+-----+-----+-----+
| ID | Nombre                | FechaNacimiento | Nacionalidad | URLBiografia
|
+----+-----+-----+-----+-----+
| 01 | LUDWING VAN BEETHOWEN | 1770-12-16      | alemana      |
www.ludwingvanbeethoven.org |
| 02 | WOLFGANG AMADEUS MOZART | 1756-01-27      | austriaca    |
www.mozart.com          |
| 03 | ANTONIO VIVALDI        | 1678-03-04      | italiana     |
www.antoniovivaldi.net |
| 04 | FRÉDÉRIC CHOPIN        | 1810-03-01      | polaca       |
www.fredericchopin.com  |
| 05 | MANUEL DE FALLA        | 1876-11-23      | espanola     |
www.manueldefalla.com   |
| 06 | FRANZ SCHUBERT         | 1797-01-31      | austriaca    |
www.franzschubert.com   |
| 07 | LOUISS SPOHR           | 1784-04-05      | alemana      |
www.louisspohr.com      |
| 08 | ROBERT SCHUMANN        | 1810-06-08      | alemana      |
www.robertschumann.es   |

```

```

+----+-----+-----+-----+-----+
8 rows in set (0.00 sec)

```

mysql>

- Insertamos otra tabla a la base de datos creada en pxc desde pxc1.

mysql> create table interpretes

-> (

-> id_interprete varchar(10),

```
-> nombre_interprete varchar(50),  
-> constraint pk_interpretes primary key (id_interprete),  
-> constraint contar check(length(id_interprete)>='5'));
```

Query OK, 0 rows affected (0.05 sec)

mysql>

- Comprobamos desde pxc2 insertando datos a la tabla creada anteriormente en pxc1.

```
root@pxc2:/home/vagrant# mysql -u root -p
```

Enter password:

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 12

Server version: 5.7.29-32-57-log Percona XtraDB Cluster (GPL), Release rel32, Revision 1b99a32, WSREP version 31.43, wsrep_31.43

Copyright (c) 2009-2020 Percona LLC and/or its affiliates

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

```
mysql> use auditorios;
```

Reading table information for completion of table and column names

You can turn off this feature to get a quicker startup with -A

Database changed

```
mysql> insert into interpretes values ('123456','MAURICE ANDRE');
```

Query OK, 1 row affected (0.03 sec)

```
mysql> insert into interpretes values ('1er456','CLAUDIO ARRAU');
```

Query OK, 1 row affected (0.01 sec)

```
mysql> insert into interpretes values ('654tgb','DANIEL BARENBOIM');
```

Query OK, 1 row affected (0.02 sec)

```
mysql> insert into interpretes values ('543frtyh','YURI BASHMET');
```

Query OK, 1 row affected (0.01 sec)

- Insertamos nuevas tablas y datos desde pxc2 y mostramos desde pxc.

```
root@pxc2:/home/vagrant# mysql -u root -p
```

Enter password:

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 12

Server version: 5.7.29-32-57-log Percona XtraDB Cluster (GPL), Release rel32,
Revision 1b99a32, WSREP version 31.43, wsrep_31.43

Copyright (c) 2009-2020 Percona LLC and/or its affiliates

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

```
mysql> use auditorios;
```

Reading table information for completion of table and column names

You can turn off this feature to get a quicker startup with -A

```
mysql> create table solistas
```

```
-> (
```

```
-> IDinterprete_solista varchar(10),
```

```
-> nombre_artistico varchar(50),
```

```
-> instrumento varchar(100),
```

```
-> constraint pk_solistas primary key (IDinterprete_solista),
```

```
    -> constraint solistas_interpretes foreign key (IDinterprete_solista)
references interpretes(id_interprete));
```

Query OK, 0 rows affected (0.05 sec)

```
mysql>
```

```
mysql> insert into solistas values ('124223','RUBEN FERNANDEZ','flauta');
```

Query OK, 1 row affected (0.01 sec)

```
mysql> insert into solistas values ('123456','MAURICE ANDRE','trompeta');
```

```
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into solistas values ('1er456','CLAUDIO ARRAU','piano');
```

```
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into solistas values ('654tgb','BARENBOIM','piano');
```

```
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into solistas values ('543frtyh','YURI BASHMET','violin');
```

```
Query OK, 1 row affected (0.01 sec)
```

```
mysql>
```

```
root@pxc:/home/vagrant# mysql -u root -p
```

```
Enter password:
```

```
Welcome to the MySQL monitor.  Commands end with ; or \g.
```

```
Your MySQL connection id is 18
```

```
Server version: 5.7.29-32-57-log Percona XtraDB Cluster (GPL), Release rel32,  
Revision 1b99a32, WSREP version 31.43, wsrep_31.43
```

```
Copyright (c) 2009-2020 Percona LLC and/or its affiliates
```

```
Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.
```

```
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective
```


owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

```
mysql> use auditorios;
```

Reading table information for completion of table and column names

You can turn off this feature to get a quicker startup with -A

Database changed

```
mysql> select * from solistas;
```

```
+-----+-----+-----+
| IDinterprete_solista | nombre_artistico | instrumento |
+-----+-----+-----+
| 123456          | MAURICE ANDRE   | trompeta   |
| 124223          | RUBEN FERNANDEZ | flauta     |
| 1er456          | CLAUDIO ARRAU   | piano      |
| 543frtyh        | YURI BASHMET    | violin     |
| 654tgb          | BARENBOIM       | piano      |
+-----+-----+-----+
```

5 rows in set (0.00 sec)

```
mysql>
```

- Añadimos HAproxy para comprobar la integración de balanceo de carga.
- Antes debemos de crear nuestra máquina que tendrá instalado el HAproxy, añadimos al vagrantfile:

```
config.vm.define :balanceador do |balanceador|  
  
  balanceador.vm.box = "debian/buster64"  
  
  balanceador.vm.hostname = "balanceador"  
  
  balanceador.vm.network :public_network, :bridge=>"eth0"  
  
  balanceador.vm.network :private_network, ip: "192.168.5.173"  
  
end
```

La configuración del clúster ahora está completa, pero debemos preparar el clúster para que funcione con haproxy.

Dado que MYSQL se verifica a través de httpchk, hay una utilidad que viene con Percona XtraDB Cluster llamada "clustercheck" que evita el haproxy.

Configuramos clustercheck

- Crear el usuario de clustercheck

```
GRANT PROCESS ON *.* TO 'clustercheckuser'@'localhost' IDENTIFIED BY  
'clustercheckpassword!';
```

- Instalamos xinetd

```
sudo apt-get install xinetd
```

- Reiniciamos servicios

```
sudo systemctl restart xinetd
```

- En nuestra nueva máquina que hará de balanceador de carga tendremos que instalar HAproxy.

```
sudo apt-get -y install haproxy
```

Abrimos el archivo de configuración `/etc/haproxy/haproxy.cfg` como vemos a continuación.

```
global
```

```
    log /dev/log      local0
```

```
    log /dev/log      local1 notice
```

```
    chroot /var/lib/haproxy
```

```
    stats socket /run/haproxy/admin.sock mode 660 level admin expose-fd
listeners
```

```
    stats timeout 30s
```

```
    user haproxy
```

```
    group haproxy
```

```
    daemon
```

```
    # Default SSL material locations
```

```
    ca-base /etc/ssl/certs
```

```
    crt-base /etc/ssl/private
```

```
    # Default ciphers to use on SSL-enabled listening sockets.
```

```
    # For more information, see ciphers(1SSL). This list is from:
```

```
    # https://hynek.me/articles/hardening-your-web-servers-ssl-ciphers/
```

```
    # An alternative list with additional directives can be obtained from
```

```
    # https://mozilla.github.io/server-side-tls/ssl-config-generator/?
server=haproxy
```

```
ssl-default-bind-ciphers
ECDH+AESGCM:DH+AESGCM:ECDH+AES256:DH+AES256:ECDH+AES128
:DH+AES:RSA+AESGCM:RSA+AES:!aNULL:!MD5:!DSS
```

```
ssl-default-bind-options no-sslv3
```

```
defaults
```

```
log global
```

```
mode http
```

```
option httplog
```

```
option dontlognull
```

```
timeout connect 5000
```

```
timeout client 50000
```

```
timeout server 50000
```

```
errorfile 400 /etc/haproxy/errors/400.http
```

```
errorfile 403 /etc/haproxy/errors/403.http
```

```
errorfile 408 /etc/haproxy/errors/408.http
```

```
errorfile 500 /etc/haproxy/errors/500.http
```

```
errorfile 502 /etc/haproxy/errors/502.http
```

```
errorfile 503 /etc/haproxy/errors/503.http
```

```
errorfile 504 /etc/haproxy/errors/504.http
```

- Añadimos la siguiente configuración.

```
#First part
```

```
listen stats
```

```
#Listen on port 9000
```

```
bind :9000
```

#This is the virtual URL to access the stats page

stats uri /haproxy_stats

stats enable

#login: username and password

stats auth camargo2:camargo2

#This allows you to take down and bring up back end servers

stats admin if TRUE

#Second part

frontend pxc-front

bind *:3307

mode tcp

default_backend pxc-back

backend pxc-back

mode tcp

balance roundrobin

option httpchk

server pxc 192.168.5.170:3306 check port 9200 inter 400 rise 3 fall 3

server pxc1 192.168.5.171:3306 check port 9200 inter 400 rise 3 fall 3

server pxc2 192.168.5.172:3306 check port 9200 inter 400 rise 3 fall 3

#Third part

frontend pxc-onenode-front

bind *:3306

mode tcp

default_backend pxc-onenode-back

backend pxc-onenode-back

mode tcp

balance roundrobin

option httpchk

server pxc 192.168.5.170:3306 check port 9200 inter 400 rise 3 fall 3

server pxc1 192.168.5.171:3306 check port 9200 inter 400 rise 3 fall 3 backup

server pxc2 192.168.5.172:3306 check port 9200 inter 400 rise 3 fall 3 backup

- Guardamos y reiniciamos servicios.(Tenemos que tener en cuenta abrir os puertos oportunos.

sudo systemctl restart haproxy

- Abrimos el navegador e introducimos lo siguiente:

192.168.5.173:9000/haproxy_stats

HAProxy version 1.8.19-1+deb10u2, released 2020/04/01
Statistics Report for pid 14781

> General process information

pid = 14781 (process #1, nbproc = 1, nthread = 1)
uptime = 0d 0h00m09s
system limits: memmax = unlimited; ulimit-n = 4039
maxsock = 4039; maxconn = 2000; maxpipes = 0
current conns = 1; current pipes = 0/0; conn rate = 1/sec
Running tasks: 1/14; idle = 99 %

active UP
active UP, going down
active DOWN, going up
active or backup DOWN
active or backup DOWN for maintenance (MAINT)
active or backup SOFT STOPPED for maintenance

backup UP
backup UP, going down
backup DOWN, going up
not checked

Display option:
Scope:
• Hide 'DOWN' servers
• Refresh now
• CSV export

External resources:
• Primary site
• Updates (v1.8)
• Online manual

Note: "NOLE"/"DRAIN" = UP with load-balancing disabled.

Queue		Session rate			Sessions				Bytes		Denied		Errors			Warnings		Server												
Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle	
Frontend	0	0	1	1	-	1	1	2 000	2			395	262	0	0	0					OPEN									
Backend	0	0	0	0		0	0	200	0	0	0s	395	262	0	0	0					9s UP		0	0	0	0	0	0		

pxc-front

Queue	Session rate			Sessions				Bytes		Denied		Errors			Warnings		Server													
Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle	
Frontend	0	0	0	0	-	0	0	2 000	0			0	0	0	0	0					OPEN									

pxc-back

Queue	Session rate			Sessions				Bytes		Denied		Errors			Warnings		Server												
Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle
pxc	0	0	0	0	-	0	0	0	0	0	?	0	0	0	0	0	0	0	0	0	9s UP	L7OK/200 in 38ms	1	Y	-	0	0	0s	-
pxc1	0	0	0	0	-	0	0	0	0	0	?	0	0	0	0	0	0	0	0	0	9s UP	L7OK/200 in 36ms	1	Y	-	0	0	0s	-
pxc2	0	0	0	0	-	0	0	0	0	0	?	0	0	0	0	0	0	0	0	0	9s UP	L7OK/200 in 36ms	1	Y	-	0	0	0s	-
Backend	0	0	0	0		0	0	200	0	0	?	0	0	0	0	0	0	0	0	0	9s UP		3	3	0	0	0	0s	

Choose the action to perform on the checked servers: Apply

pxc-onenode-front

Queue	Session rate			Sessions				Bytes		Denied		Errors			Warnings		Server												
Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle

192.168.5.173:9000/haproxy_stats

stats		Queue			Session rate			Sessions				Bytes		Denied		Errors			Warnings		Server										
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle	
Frontend				1	1	-	1	1	2 000	2			395	262	0	0	0					OPEN									
Backend	0	0		0	0		0	0	200	0	0	0s	395	262	0	0	0	0	0	0	0	9s UP		0	0	0					

pxc-front		Queue			Session rate			Sessions				Bytes		Denied		Errors			Warnings		Server										
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle	
Frontend				0	0	-	0	0	2 000	0			0	0	0	0	0	0				OPEN									

pxc-back		Queue			Session rate			Sessions				Bytes		Denied		Errors			Warnings		Server									
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle
<input type="checkbox"/>				0	0	-	0	0	200	0	0	?	0	0	0	0	0	0	0	0	0	9s UP	L7OK/200 in 38ms	1	Y	-	0	0	0s	-
<input type="checkbox"/>	pxc1	0	0	-	0	0	0	0	200	0	0	?	0	0	0	0	0	0	0	0	0	9s UP	L7OK/200 in 36ms	1	Y	-	0	0	0s	-
<input type="checkbox"/>	pxc2	0	0	-	0	0	0	0	200	0	0	?	0	0	0	0	0	0	0	0	0	9s UP	L7OK/200 in 36ms	1	Y	-	0	0	0s	-
Backend	0	0		0	0		0	0	200	0	0	?	0	0	0	0	0	0	0	0	0	9s UP		3	3	0			0s	

Choose the action to perform on the checked servers :

pxc-onenode-front		Queue			Session rate			Sessions				Bytes		Denied		Errors			Warnings		Server										
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle	
Frontend				0	0	-	0	0	2 000	0			0	0	0	0	0	0				OPEN									

pxc-onenode-back		Queue			Session rate			Sessions				Bytes		Denied		Errors			Warnings		Server									
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle
<input type="checkbox"/>	pxc	0	0	-	0	0	0	0	200	0	0	?	0	0	0	0	0	0	0	0	0	9s UP	L7OK/200 in 38ms	1	Y	-	0	0	0s	-
<input type="checkbox"/>	pxc1	0	0	-	0	0	0	0	200	0	0	?	0	0	0	0	0	0	0	0	0	9s UP	L7OK/200 in 46ms	1	-	Y	0	0	0s	-
<input type="checkbox"/>	pxc2	0	0	-	0	0	0	0	200	0	0	?	0	0	0	0	0	0	0	0	0	9s UP	L7OK/200 in 60ms	1	-	Y	0	0	0s	-
Backend	0	0		0	0		0	0	200	0	0	?	0	0	0	0	0	0	0	0	0	9s UP		1	1	2			0s	

Choose the action to perform on the checked servers :

14. MariaDB Galera Cluster VS Percona XtraDB Cluster

MARIADB GALERA CLUSTER

- Desarrollado por Codership.
- Incluido en MariaDB.

Galera Cluster es un clúster de base de datos multimaestro síncrono, basado en la replicación síncrona y Mysql/ InnoDB. Cuando el clúster galera está en uso, puede dirigir las lecturas y escrituras a cualquier nodo y puede perder cualquier nodo individual sin interrupción en las operaciones y sin la necesidad de manejar procedimientos complejos de recuperación por error.

- Replicación síncrona, la aplicación es asíncrona.

Características MariaDB Galera Cluster

- Totalmente integrado en el servidor MariaDB (todo en una descarga)-
- Solo un envío con Galera 4.
- Puede hacer actualizaciones continuas de Galera 3 a Galera 4.
- Sugerencias de nuevas características de Galera 4, existen:
- * Replicación de transmisión (Soporte para grandes transacciones).
- * Apoyo de compromiso grupal.

* Uso de bloqueos de respaldo.

- Todo en el servidor Mariadb está disponible para usar en el clúster Galera.

PERCONA XTRADB CLUSTER

- Galera parchado desarrollado por Percona.

- Disponible desde 2012.

- Características adicionales.

* Soporte extendido FPS.

* Percona XtraBackup integrado

* ProxySql integrado como equilibrador de carga.

* Actúa en “modo estricto” por ejemplos, no permitir MySAM, tablas sin claves primarias, registro de formato binlog ROW en un archivo (no tablas).

* Configuración automática de cifrado SSL.

* Corrección de errores.

* Mejoras innodb(a través de XtraDB) en torno a la búsqueda de texto completo, contadores de fragmentación de página.

* Integración con la supervisión y gestión de Percona.

* Cifrado a través de Keying_vault o plugins de codificación.

* Mejoras de rendimiento.

SIMILITUDES ENTRE AMBOS

- Mysql/MariaDB

- Método de replicacion

- Todos los nodos tienen los mismos datos.

- Ambos requiere InnoDB/XtraDB como motor de almacenamiento.

- Topología multimaestro activa-activa.

- * Escribir en múltiples nodos.
- * No es necesaria una conmutación por error compleja / externa.

- Membresía de nodo: unirse / salir automáticamente.
- Ejecutar escrituras en orden total global.
- Consistencia de los datos.
- Prevención cerebral dividida en quórum.

CASOS DE USO SIMILAR

- Entornos con estrictos requisitos de durabilidad.
- Escribir en múltiples nodos simultáneamente manteniendo los datos consistentes.
- Reducir el tiempo de conmutación por error.

LIMITACIONES SIMILARES

- Transacciones grandes y de larga duración.
 - * Mayor probabilidad de fallas.
 - * Replicación no eficiente de transiciones grandes.
- Zona activa de carga de trabajo (actualización del mismo conjunto de filas en paralelo).
- A menudo escribir en un nodo es la mejor opción.

BINLOG & GCACHE

GALERA/PXC

- Utiliza el evento de fila binlog.
- No requiere registro binario.
- Escribe el evento en Gcache (tamaño configurable).

NODE PROVISIONING / APROVISIONAMIENTO DE NODOS

GALERA/PXC

- Tiene transferencia de instantáneas de estado (SST).
- Percona XtraBackup (Recomendado).
- Rsync.

- MysqlDump.
- Tranferencia de estado incremental (IST) usando Gcache.

GTID VS SEQNO

GALERA/PXC

- Tiene un Seqno que es un número incremental.

MANEJO DE PARTICIONES

GALERA/PXC

- Un nodo particionado rechazará lecturas/escrituras (configurable).

CONTROL DE FLUJO

Evitar que un nodo más lento se quede demasiado atrás.

GALERA/PXC

- Bloquear todas las escrituras en el clúster cuando un nodo alcanza un límite.
- Se envía un mensaje de control de flujo.
- Valores predeterminados bajo; Galera:16, PXC:100.
- Comunicar a otros que dejen de escribir.

WAN SUPPORT

Los sistemas basados en galera tienen características :

- Quorum ponderado.
- Ajustes de comunicación de red ajustables.
- Reducir el tráfico de red con segmentos.

SOPORTE DEL SISTEMA OPERATIVO

*GALERA

FreeBSD & Linux

*PERCONA XTRADB CLUSTER

LINUX

HERRAMIENTAS QUE SE INTEGRAN CON GALERA CLUSTER PARA MYSQL

- Mysql
- MongoDB
- PostreSql
- MariaDB
- Oracle

CON PXC

- Debian
- Centos
- Red Hat Enterprise Linux
- Goland

ALTERNATIVAS A GALERA Y PXC

- Cassandra
- Apache Aurora
- CockroachDB

HEMA CHANGES -DDL (cambios de esquema/ DDL)

* GALERA CLUSTER / PXC

- Aislamiento total de la orden.
- Todas las escrituras serán bloqueadas durante ello.
- Las escrituras en otros nodos serán terminadas.
- Soluciones alternativas:

- * pt_online schema_change
- * wsrep -osu_method =RSU

- Más trabajo operacional.
- No para todas las DDL.

DIFERENCIAS PERCONA XTRADB CLUSTER vs. GALERA CLUSTER

Galera Cluster: Un cluster multimaestro basado en replicación sincrónica para Mysql. Es una solución fácil de usar y de alta disponibilidad, que proporciona un alto tiempo de actividad del sistema, sin pérdida de datos y escalabilidad para el crecimiento futuro.

Se puede mantener funcionando 24/7.

Servidor Percona es un reemplazo directo de código abierto para Mysql. Es un reemplazo gratuito, totalmente compatible, mejorado para Mysql que proporciona un rendimiento, escalabilidad e instrumentación superiores. Sus algoritmos de autoajuste y el soporte para hardware es extremadamente alto rendimiento ofrecen un excelente rendimiento y confiabilidad.

PXC tiene características adicionales:

- Soporte extendido de PFS.
- Cambios SST/Xtrabackup
- Corrección de errores
- Modo estricto PXC
- Integración ProxySQL
- Mejoras de Rendimiento.

En PXC evitar características experimentales / no compatibles.

- Solo permite operaciones innodb.
- Evite cambiar el formato binlog !=ROW.
- Requiere clave primaria en tablas.
- Deshabilitar funciones no compatibles.

- GET_LOCK, LOCK TABLES, CTAS
- FLUSH TABLES <tables> WITH READ LOCK (Tablas con bloqueo de lectura).
- tx_isolation=SERIALIZABLE

PXC INCLUYE PROXYSQL COMO BALANCEADOR DE CARGA

- Herramienta de configuración ProxySql-admin.
- Programación de proxySql.
 - * Controles de salud.
 - * Reconfigura nodos.
- Modo de mantenimiento PXC.
 - * Decir al equilibrador de carga que reequilibre la carga.

TENEMOS EN CUENTA UNA SERIES DE LIMITACIONES

GALERA/PXC

No funciona como se esperaba:

- InnoDB/XtraDB solamente.
- tx_isolation=SERIALIZABLE
- Get lock
- Tablas de bloqueo
- Select para actualizar.
- Cuidado con ALTER TABLE ... import/export.
- Tamaño máximo de transacción limitado.

PROBLEMAS CONOCIDOS

GALERA/PXC

* Se bloquea debido al procesamiento de subprocesos de fondo.

- mysql-wsrep#306: procedimiento almacenado abortado
- mysql-wsrep#305: programador de evento
- mysql-wsrep#304: Funciones de alcance local como CURRENT_USER()

* Varios problemas con DDL

mysql-wsrep#301: ejecutar SHOW CREATE TABLE en varios nodos con DDL puede provocar un bloqueo

mysql-wsrep#275: Abortar las hojas de trx que se pueden abrir en la memoria caché puede provocar un bloqueo

* DDL concurrentes utilizando wsrep_OSU_method = RSUcrash / problemas de inconsistencia

- mysql-wsrep # 283 y mysql-wsrep # 282

*Problemas de apagado:

- mysql-wsrep # 303: la limpieza durante el apagado no borra los bloqueos EXPLICIT MDL (FTWRL)

- mysql-wsrep # 273: No se limpia al cerrar el mensaje si iniciamos el servidor convariable

- mysql-wsrep # 279 desconocido: Intentando acceder a los cables de binloghandler obsoletos estrellarse35 / 60

Galera y Percona XtraDB Cluster utilizan su propio conjunto de pruebas MTR para probar la recuperación, el cambio de estado miembro, el manejo de consultas, la concurrencia, el estrés, etc.

pquery

Para probar Percona XtraDB Cluster y Galera, mejoramos nuestro marco de clúster pquery existente. Este marco iniciará un clúster de 3 nodos y ejecutará pquery contra estos nodos del clúster.

CARACTERÍSTICAS PQUERY

- Más de 20 Coredumps por hora
- Creación de casos de prueba completamente automática
- Nuceo C ++
- 120 segundos por tiempo de ejecución de prueba
- Miles de líneas SQL ejecutadas por prueba
- Compatible con problemas esporádicos
- Automatización de alta gama
- Reducción ultra rápida de casos de prueba
- Framework completo

RESUMEN

	GALERA	PXC
Aprovisionamiento automático del nodo	✓	✓
Integración de balanceador de carga		✓
Hacer cumplir las mejoras prácticas		✓
Manejo de particiones	✓	✓
Tecnología Madura	✓	✓
Multimaestro	✓	✓
OS support	✓	✓
Funcionamiento y Rendimiento		✓
WAN Support	✓	✓

