FromDual Annual Company Meeting

Athens, 2013

Galera Cluster for MySQL

http://www.fromdual.com

About FromDual GmbH (LLC)

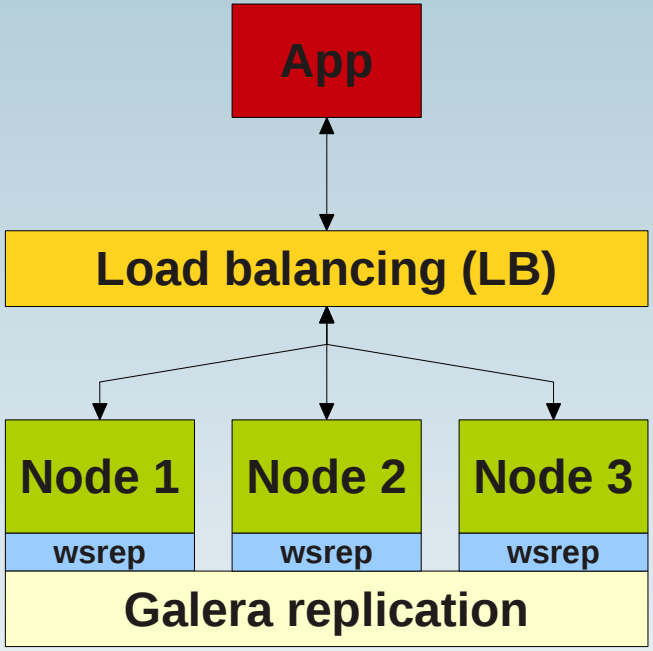
www.fromdual.com

- FromDual provides neutral and independent:
  - Consulting for MySQL
  - Support for MySQL and Galera Cluster
  - Remote-DBA Services for MySQL
  - MySQL Training
- Oracle Silver Partner (OPN)
- Member of SOUG, DOAG, /ch/open

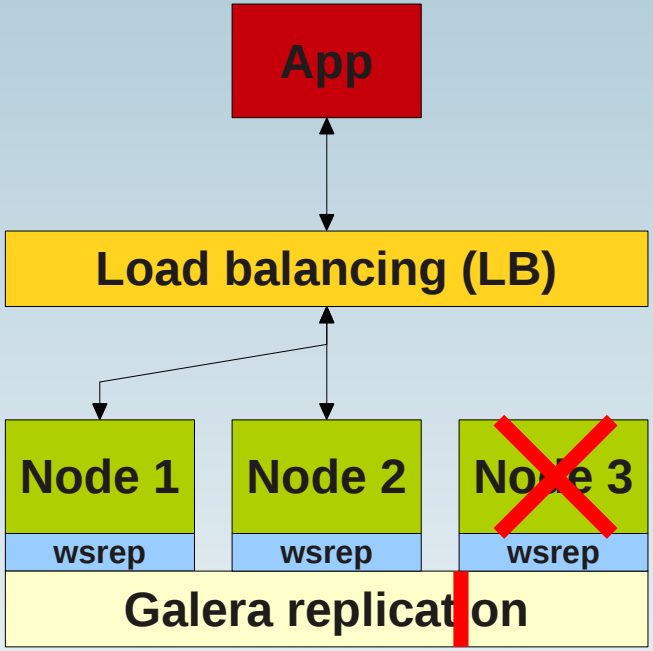
www.fromdual.com

Our customer

Galera Cluster



Galera Cluster



Advantages / Disadvantages

- + Synchronous replication
  → No lost transaction
- + Based on InnoDB SE
- + Active-active real multi-master topology
  → Read and write to any cluster node is possible!
- + Automatic membership control
- + True parallel replication, on row level
  → No slave lag
- + Read scalability (Read Scale-Out!) and write improvements (+ SSD)
- + Rolling Restart (Upgrade of Hardware, O/S, DB release, etc.)
- - No original MySQL binaries → Codership MySQL binaries
- - Be aware of Hot Spots on rows: Higher probability of deadlocks
- - Initial full sync (SST) blocks for reading and writing → 3 nodes

Galera Cluster Set-up

Split-Brain (sb)

It indicates data inconsistencies originating from the maintenance of two separate data sets with overlap in scope, because a failure condition based on servers not communicating and synchronizing their data to each other.

- **Optimistic approach:**
  - **Simply let the partitioned nodes work as usual**
  - **Provides a greater level of availability**
  - **At the cost of sacrificing correctness**
  - **Automatic or manual clean-up might be required**

    → MySQL Master/Master Replication
- **Pessimistic approach:**
  - **Sacrifice availability in exchange for consistency.**
  - **Once a network partitioning has been detected, access to the sub-partitions is limited in order to guarantee consistency.**
  - **Quorum-consensus approach. Allows sub-partition with a majority to remain available**
  - **The remaining sub-partitions should fall down to an auto-fencing mode.**

    → Galera Cluster, active/passive failover Cluster, MySQL NDB Cluster

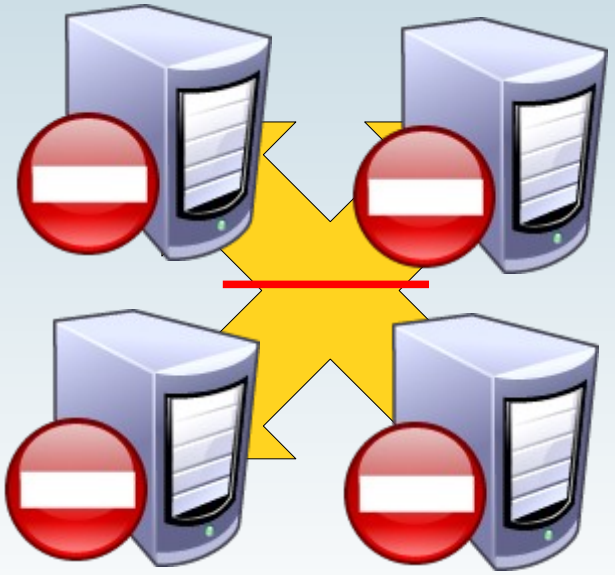
Quorum

- Quorum comes from politics:

  “Minimum number of members necessary to conduct the business of that group”

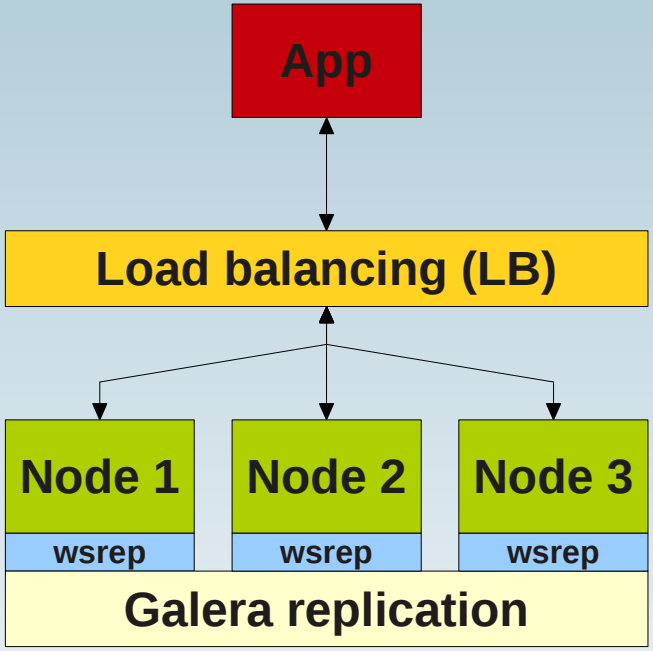
- In short for Cluster:
  - MORE than half of the nodes must be available
  - Otherwise they will shut-down themselves!
- Otherwise: Split-Brain!!! (which is bad)
- 2 Nodes connected in series have higher probability for failure than just one node!
- Quorum: FLOOR(n/2+1)
- Nodes gracefully leaving the Cluster do not count for the quorum!

Quorum



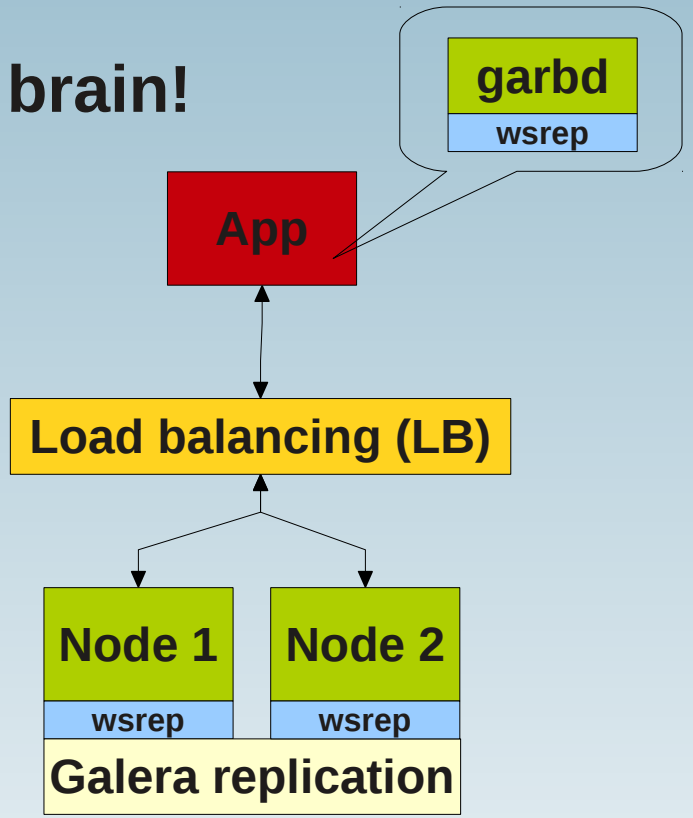
3 node Cluster

- **Standard (recommended) set-up:**



2 + 1 node Cluster

- 2 nodes is bad → split brain!
- Minimalistic set-up:
  2 + 1
- Problem: SST

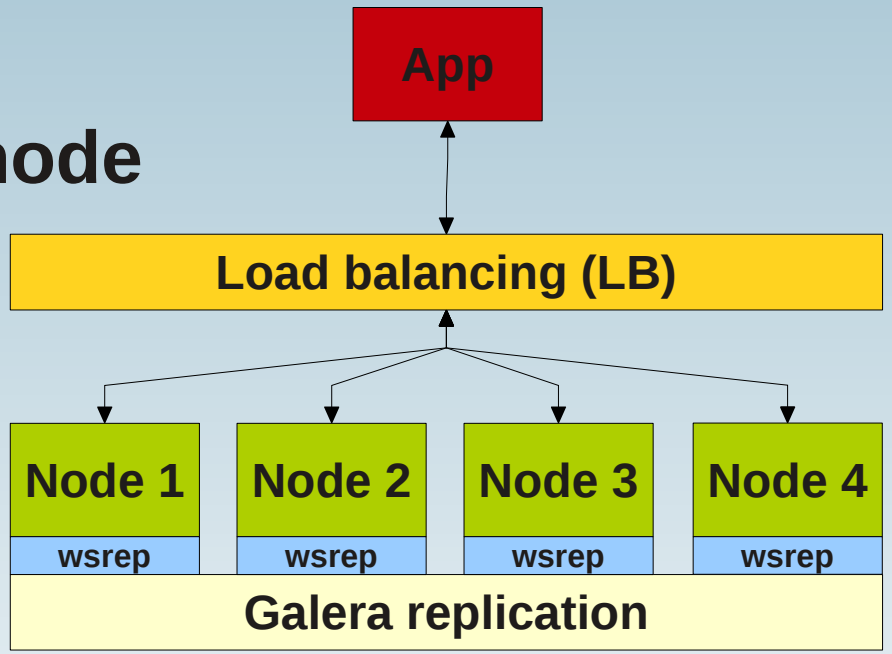


- "Our M/S-Replication has only 2 nodes as well!" or
- "I do not want to spend too much in Hardware!"

  → 2 + 1 = 2 Galera Nodes + 1 Galera Arbitrator

4 and more node Cluster

- Good for (read) Scale-out
  - Backup-node
  - Dedicated SST-Donor node
  - reporting Node, etc.
- Good to have an odd number of nodes!
  - If not → weighted Quorum?
  - Even number: Split Brain!
- Biggest Cluster, just for fun: 17 nodes!



MySQL Configuration

- **my.cnf**

```
default_storage_engine          = InnoDB
binlog_format                   = row

innodb_autoinc_lock_mode        = 2   # performace?
innodb_locks_unsafe_for_binlog = 1   # how locking is done?!?

innodb_flush_log_at_trx_commit = 0   # performance only!

query_cache_size                = 0
query_cache_type                = 0   # Mutex! Consistency!
```

Galera Configuration

- my.cnf (conf.d/galera.cnf, conf.d/wsrep.cnf)

```
# wsrep_provider                  = none
wsrep_provider                  = …/lib/plugin/libgalera_smm.so

# wsrep_cluster_address           = "gcomm://"
wsrep_cluster_address           = "gcomm://node2,node3"

wsrep_cluster_name              = 'Galera Cluster'
wsrep_node_name                 = 'Node A'

wsrep_sst_method                = mysqldump
wsrep_sst_auth                  = sst:secret
```

Different Operation Scenarios

- Node preparation
  - Create SST user
- Initial Cluster (re-)start
  - Start very 1st node
- Node (re-)start
  - requires SST or IST
- Rolling restart
  - e.g. for upgrades

Node preparation

- Create SST user:
- Start node with Galera disabled:

```
wsrep_provider = none
```

- Create a user for Snapshot State Transfer (SST = initial fully sync)
  - Default user for SST is root! :-(
  - We recommend to use your own user:

```
GRANT ALL PRIVILEGES ON *.* TO 'sst'@'%' IDENTIFIED BY 'secret';
GRANT ALL PRIVILEGES ON *.* TO 'sst'@'localhost' IDENTIFIED BY 'secret';
```

- Stop node again and set:

```
wsrep_provider = …/lib/plugin/libgalera_smm.so
```

Initial Cluster (re-)start

- This procedure is used when the 1st node of a Galera Cluster is started
  - Choose the node with the most accurate (or most recent) data!
- Start 1st node with:

```
wsrep_cluster_address = "gcomm://"
```

or

```
mysqld_safe --wsrep-cluster-address="gcomm://"
```

  - → this tells the node to be the first one!

Node (re-)start

- Start 2nd and 3rd node as follows:

```
wsrep_cluster_address = "gcomm://<ip_first_node,<ip_third_node>"
```

  and

```
wsrep_cluster_address = "gcomm://<ip_first_node,<ip_second_node>"
```

- If it is the very first time:

  → Nodes do a full sync = Snapshot State Transfer (SST) with the 1st node

- If it is NOT the very first time:

  → Node do an incremental sync = Incremental State Transfer (IST) or a SST with the 1st node

- Then at last: Restart 1st node (now he behaves like 2nd and 3rd) with:

```
wsrep_cluster_address = "gcomm://<ip_third_node,<ip_second_node>"
```

- Avoid to start 2 nodes in parallel!

Rolling Restart

- Scenario:
  - Hardware-, O/S-, DB- and Galera-Upgrade
  - MySQL configuration change
  - During full operation!!! (99.999% HA, 5x9 HA)

- → Rolling Restart
  - Start one node after the other in a cycle (Node Restart)
  - New features or settings are used after Rolling Restart is completed

Checking Galera Cluster

- 2 Sources of Information:

- **GLOBAL STATUS:**

  ```
  SHOW GLOBAL STATUS LIKE 'wsrep_%';
  ```

- MySQL Error Log:

  ```
  tail -f error.log
  ```

  - Some information are written to the “other” Error Log. Also look there!

Sources


```
120131 07:37:17 mysqld_safe Starting mysqld daemon
...
120131  7:37:18 [Note] WSREP: wsrep_load(): loading provider library
                       'libgalera_smm.so'
120131  7:37:18 [Note] WSREP: Start replication
...
120131  7:37:18 [Note] WSREP: Shifting CLOSED -> OPEN (TO: 0)
120131  7:37:18 [Note] .../mysql/bin/mysqld: ready for connections.
...
120131  7:37:23 [Note] WSREP: Quorum results:
        conf_id    = 2,
        members    = 3/3 (joined/total)
```

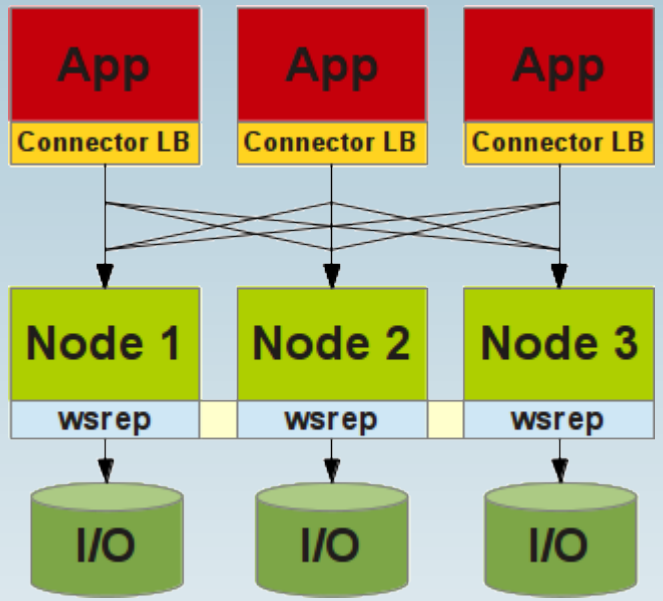
```
SHOW GLOBAL STATUS LIKE 'wsrep%';
+----------------------------+--------------------------------------+
| Variable_name              | Value                                |
+----------------------------+--------------------------------------+
| wsrep_local_state_comment  | Synced (6)                           |
| wsrep_cluster_size         | 3                                    |
| wsrep_cluster_status       | Primary                              |
| wsrep_connected            | ON                                   |
| wsrep_ready                | ON                                   |
+----------------------------+--------------------------------------+
```

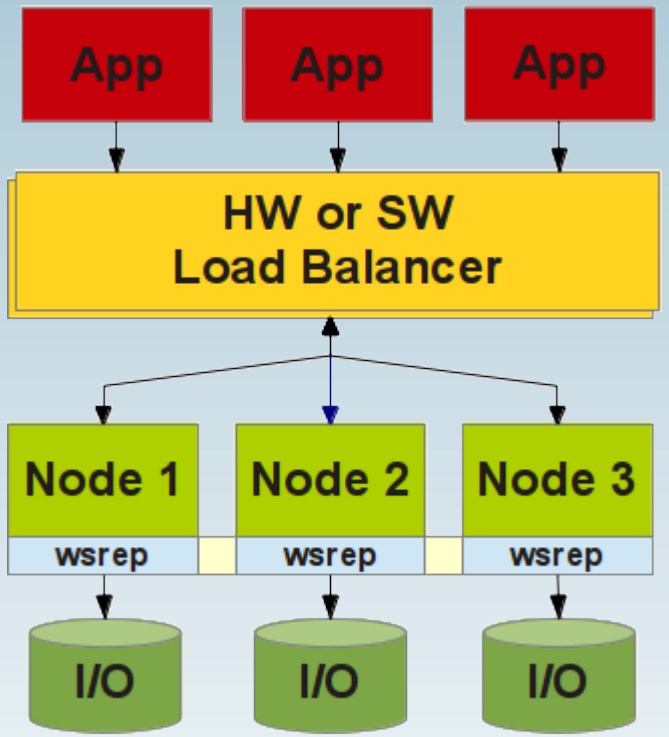
Load Balancing

Load Balancing

- In your Application (on your own)
- Connectors
  - Connector/J
  - PHP: MySQLnd replication and load balancing plug-in
  - `libglb`
- SW Load Balancer
  - GLB, Pen, LVS/IPVS/Ldirector, Ultra Monkey, HAProxy, MySQL Proxy, SQL Relay
- HW Load Balancer

Location of Load Balancing





Q & A

Questions ?

Discussion?

We have time for some face-to-face talks...

- **FromDual provides neutral and independent:**
  - **Consulting**
  - **Remote-DBA**
  - **Support for MySQL, Galera, Percona Server and MariaDB**
  - **Training**

www.fromdual.com