Percona XtraDB Cluster 8.0

Krunal Bauskar
Percona XtraDB Cluster (PXC) Product Lead

PERCONA
LIVE EUROPE
AMSTERDAM

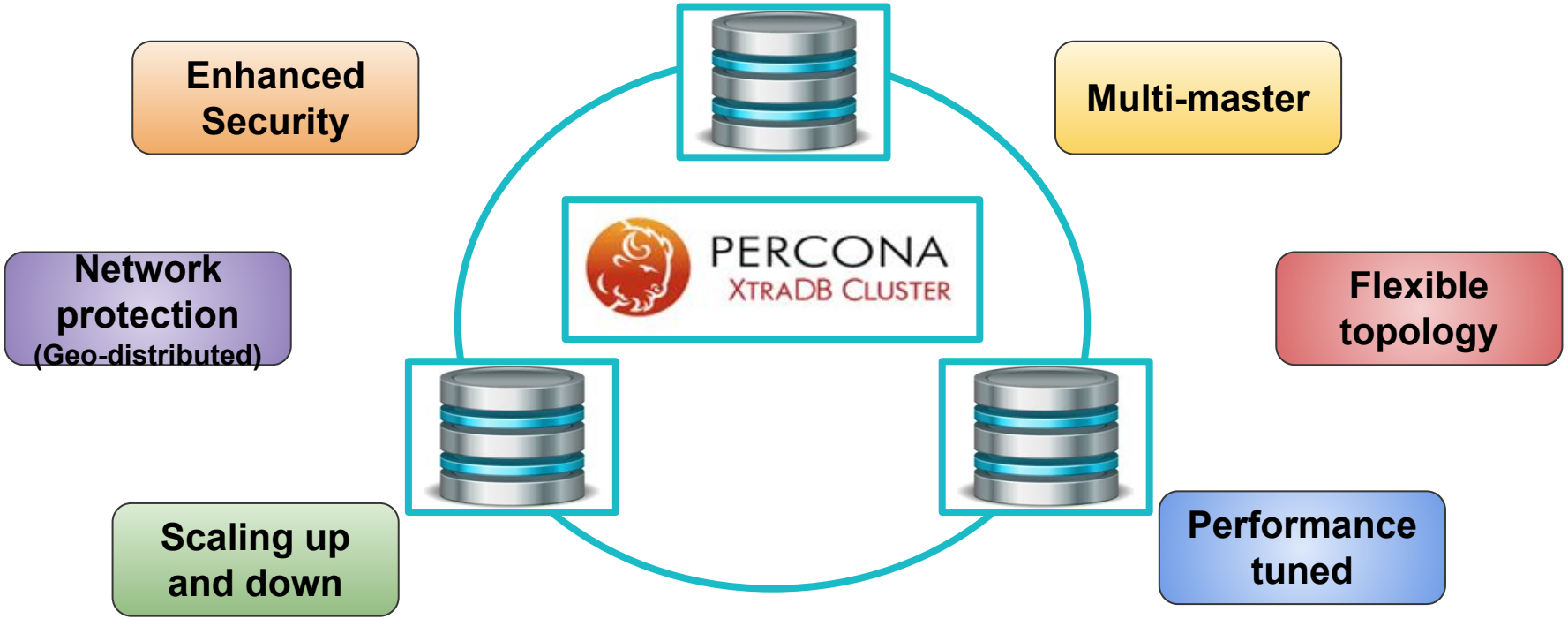
Quick Note About Myself

- Database enthusiast.
- Working with MySQL DB for more than decade.
- Wide interest in data handling and management.
- During my tenure at Yahoo!, Teradata worked on some of the real big-data problems.
- Developed multiple features in InnoDB during 5.7 tenure while working for MySQL/Oracle.

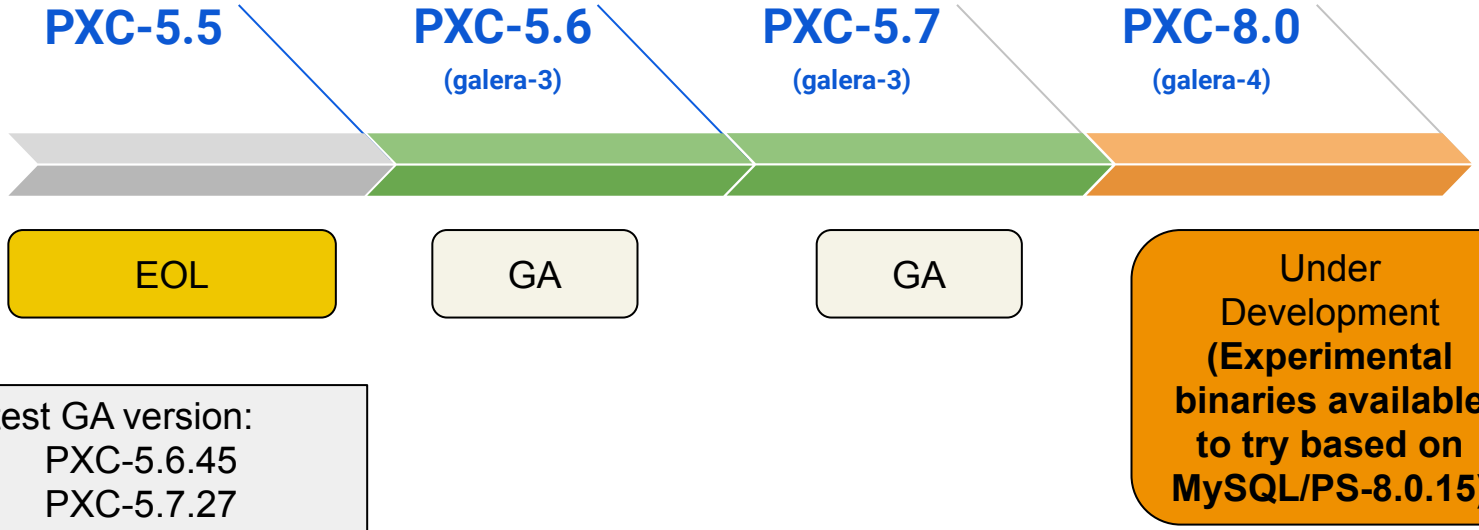
Agenda

- Quick note on “What is Percona XtraDB Cluster (PXC)?”
- Current supported versions of PXC.
- MySQL/PS-8.0 features and how they affect the PXC-8.0.
- What’s new with the PXC-8.0?
- What more to expect with the PXC-8.0?
- Q&A

Percona XtraDB Cluster



Current Supported Versions of PXC



Latest GA version:

- PXC-5.6.45
- PXC-5.7.27

Under-development:

- PXC-8.0.17

MySQL/PS-8.0 Features

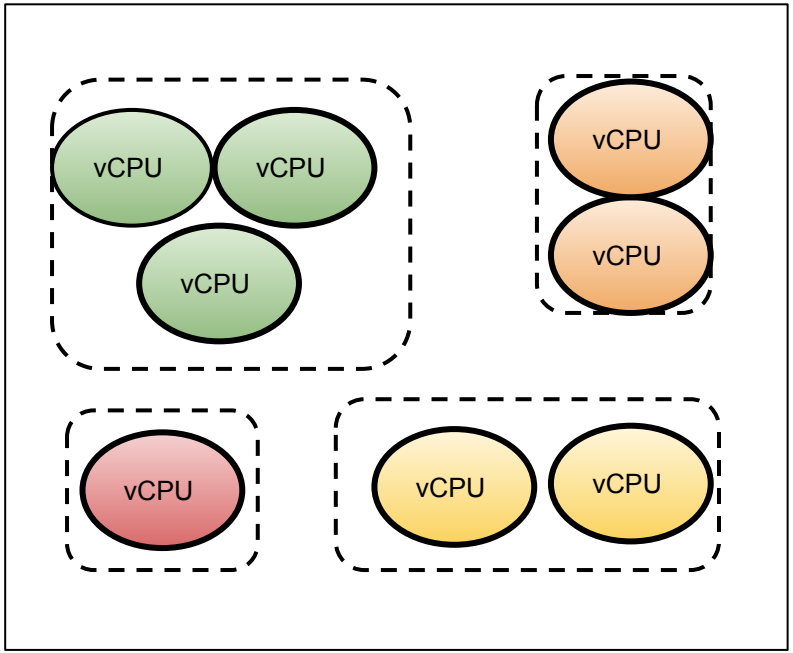
MySQL/PS-8.0 New Features

- Atomic DDL
- Data dictionary
- Introduction of ROLES, RESOURCE GROUP
- CATS scheduling algorithm
- Encryption support
- Cloud compatible features (SET PERSIST)

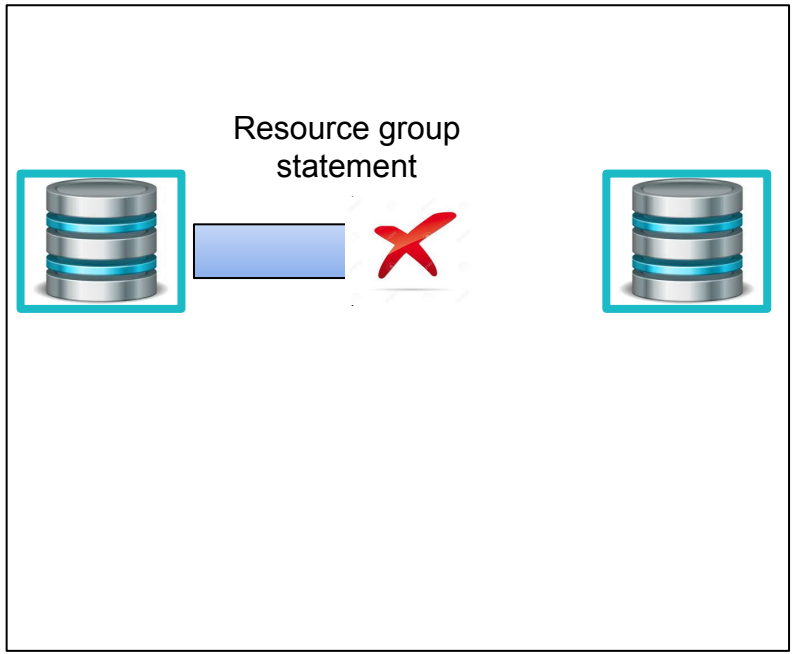
Resource Group and PXC-8.0

Resource Group and PXC-8.0

- Resource Group helps grouping of the available resources like vCPU, etc.
- MySQL threads can be assigned to a resource group for execution.
- Administrator can even associate single statement to a given resource group. (*Say an important high priority query needs to get executed can be assigned to a specialized resource group*).



Resource Group and PXC-8.0

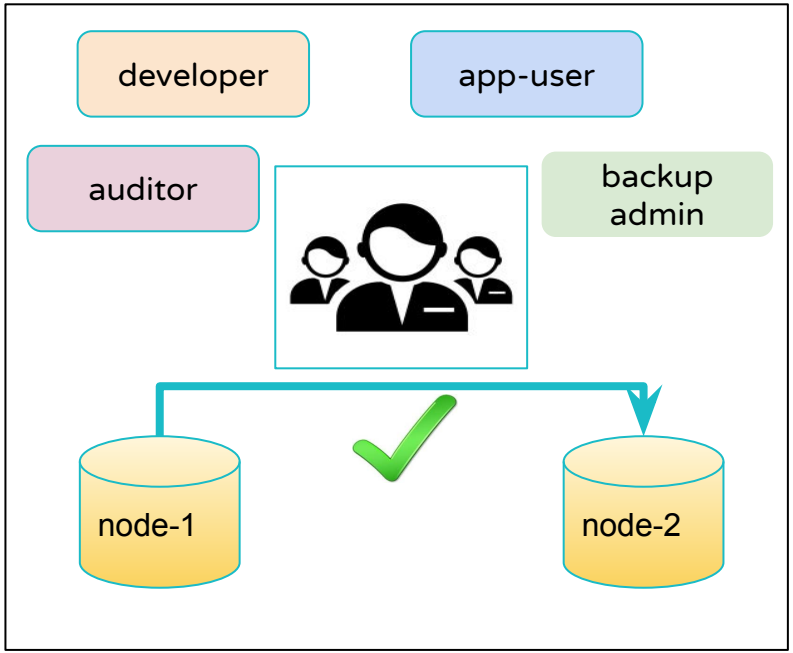


- Given resource group are local to the node, statement to **create/drop/alter resource group are not replicated.**
- Admin can now configure a RESOURCE GROUP for applier threads as actions of applier threads needs to run with HIGH PRIORITY. (Can help reduce Flow-Control).

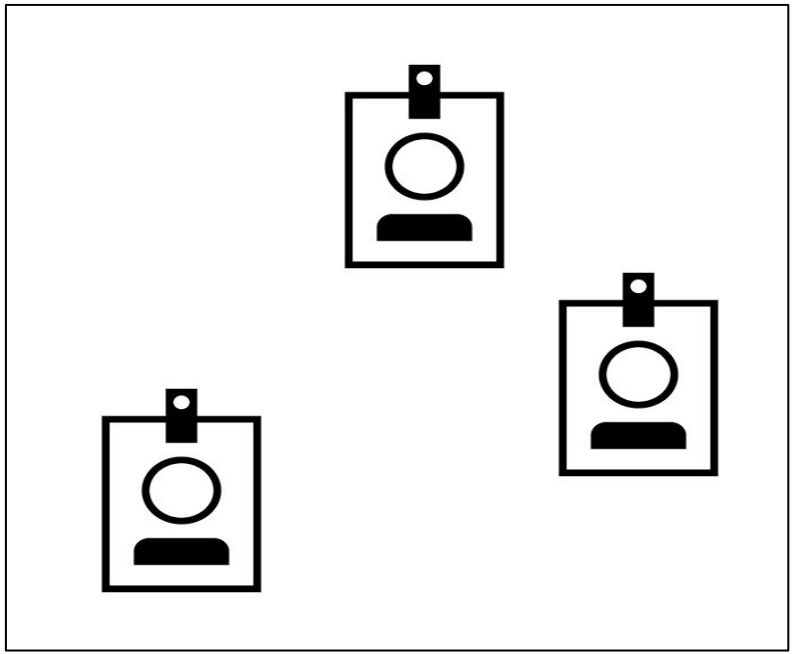
Roles and PXC-8.0

Roles and PXC-8.0

- MySQL-8.0 introduced ROLES to facilitate user and privilege management. (CREATE/DROP/REVOKE/GRANT/)
- **ROLES management statements are replicated across the PXC cluster.**
- MySQL needs ROLES to be activated before use. PXC replicates ROLE ACTIVATION too (**done through `SET DEFAULT ROLE` command**)



Roles and PXC-8.0

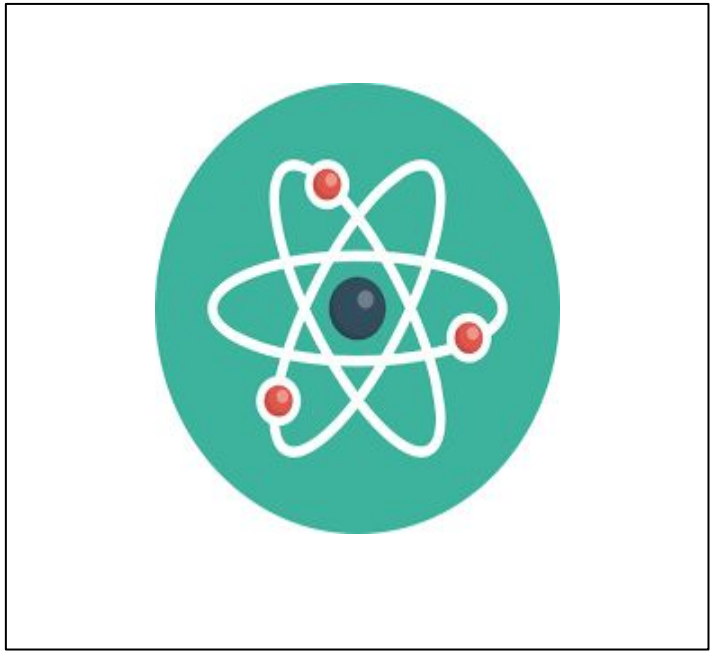


- PXC is already working on pre-defining some of the roles for general action.
- Like we tried to define mysql.pxc.sst.role that can be assigned to backup-user (needed for SST)*.

Atomic DDL and PXC-8.0

Atomic DDL and PXC-8.0

- MySQL-8.0 introduced support for Atomic DDL.
- PXC-5.7 replicates non-atomic statement (like DDL) through TOI/RSU and atomic statement (like DML) through write-set based replication [allowing DML transactions to rollback and killed].
- **PXC-8.0 continues to replicate DDL through TOI/RSU too.**



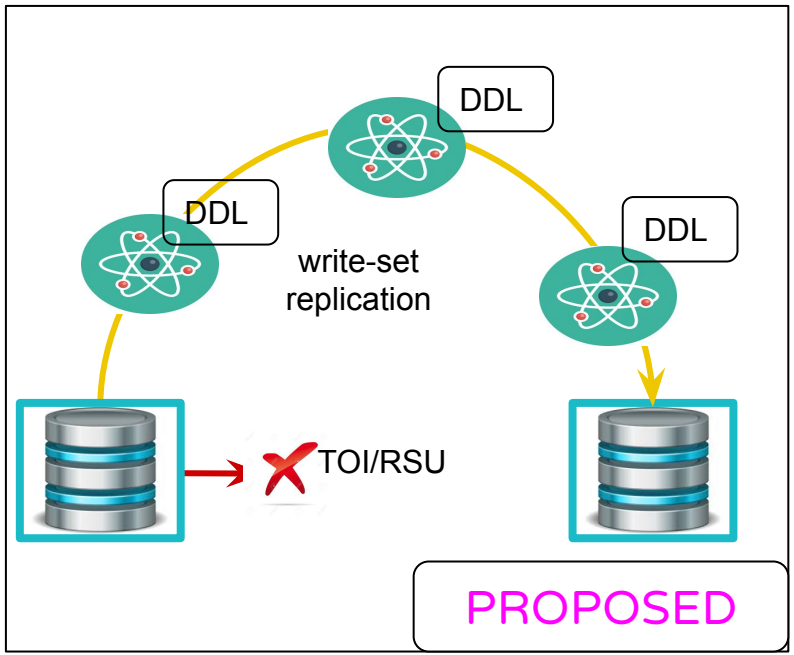
Atomic DDL and PXC-8.0

Important differences to note:

- `table local to node: t1 (local to n1)`
- `table replicated on cluster: t2 (replicated on n1 and n2)`

`drop table t1, t2;`	**on-n1**	**on-n2**
PXC-5.7	`t1, t2 dropped`	`t2 dropped`
PXC-8.0	`t1, t2 dropped`	`no-table-dropped`

Atomic DDL and PXC-8.0



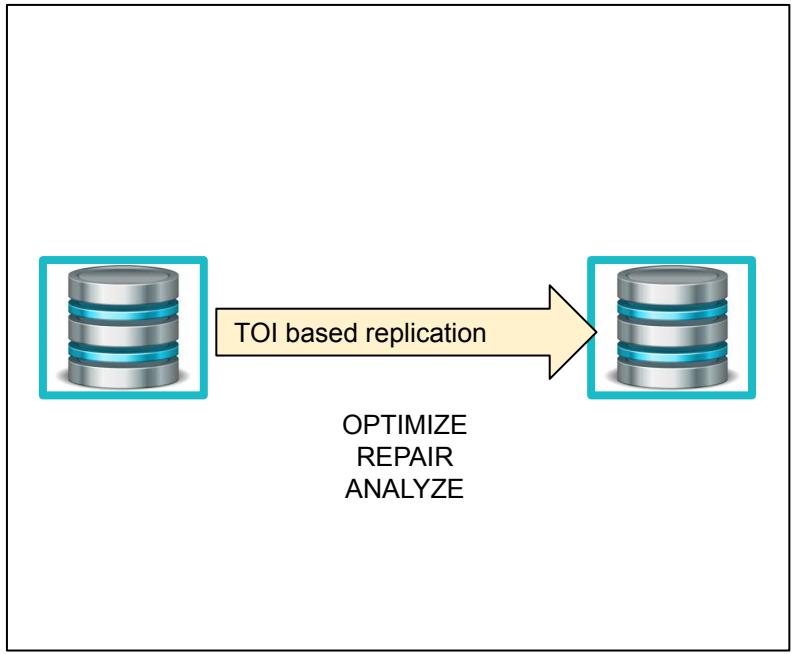
- Eventually, PXC plans to make use of the atomic nature of DDL and execute the DDL through write-set replication too.
- This will also take-care of blocking DDL issue that exists due to TOI based execution.

Non-Atomic DDL and PXC-8.0

Non-Atomic DDL and PXC-8.0

- MySQL-8.0 continues to support some non-atomic DDL operations
  - OPTIMIZE
  - REPAIR
  - ANALYZE

- **PXC continues to support them and there is no change in semantics of these non-atomic DDLs.**



Spatial Reference System (SRS) and PXC-8.0

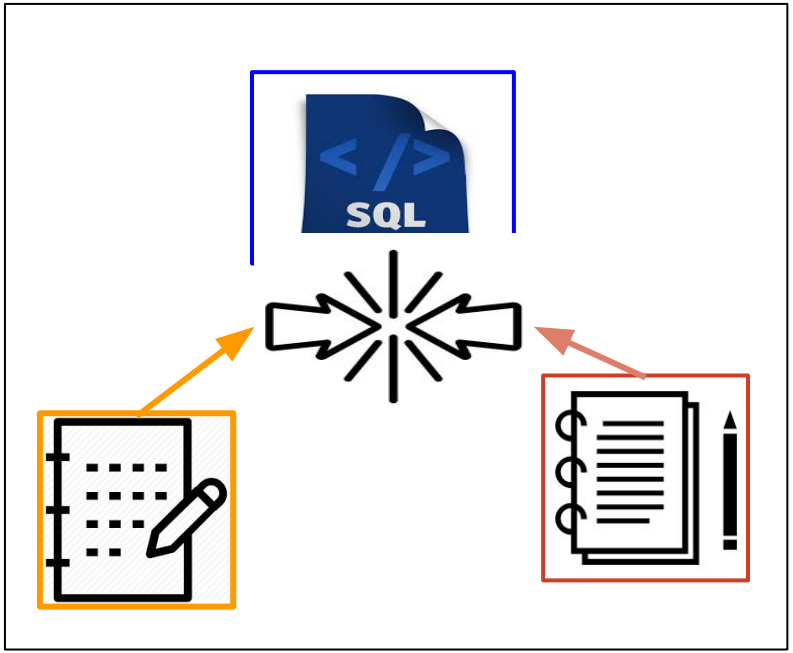
Spatial Reference System (SRS) and PXC-8.0

- MySQL-8.0 introduced SRS support. SRS can be created and dropped using CREATE/DROP command.
- **PXC cluster replicates these SRS commands.**

XID Consistency and PXC-8.0

XID Consistency and PXC-8.0

- Now that DDL are transactional, MySQL assigns XID to each DDL statement.
- MySQL has its own logic to assign XID and PXC (wsrep) has its own logic to assign XID.
- With 2 sub-systems in place it can cause duplicate XID generation that can eventually cause problem with recovery (especially with new improved recovery logic of MySQL-8.0 that checks for unique XID).



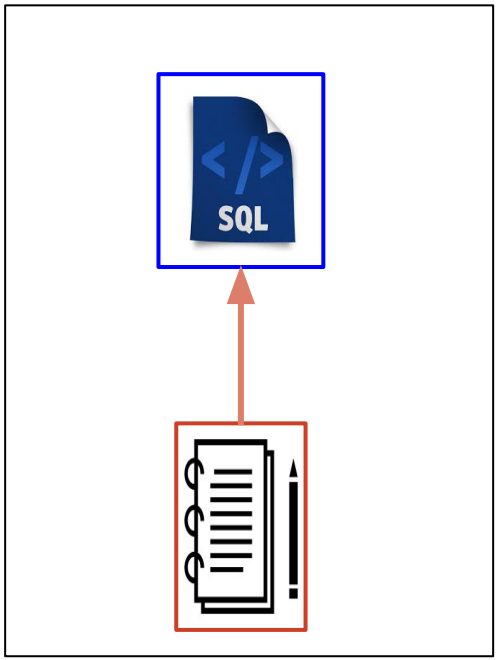
XID Consistency and PXC-8.0

```
|  mysql-bin.000001  |    759  |  Gtid  |       2  |    824  |   SET  @@SESSION.GTID_NEXT=
'9725d6d2-5ff7-11e9-a3b4-9cb6d0ba1a1d:1'    |
....
| mysql-bin.000001 |  980 | Xid  |       2 |        1011 |  COMMIT /* xid=12 */


--------------------------------------------------------------------------------


|  mysql-bin.000001  |  3027  |  Gtid  |      2  |    3092  |   SET  @@SESSION.GTID_NEXT=
'1bc0f40c-8930-ee16-770b-15117ed32fd0:12'  |
....
| mysql-bin.000001 | 3248 | Xid  |       2 |        3279 |  COMMIT /* xid=12 */
```

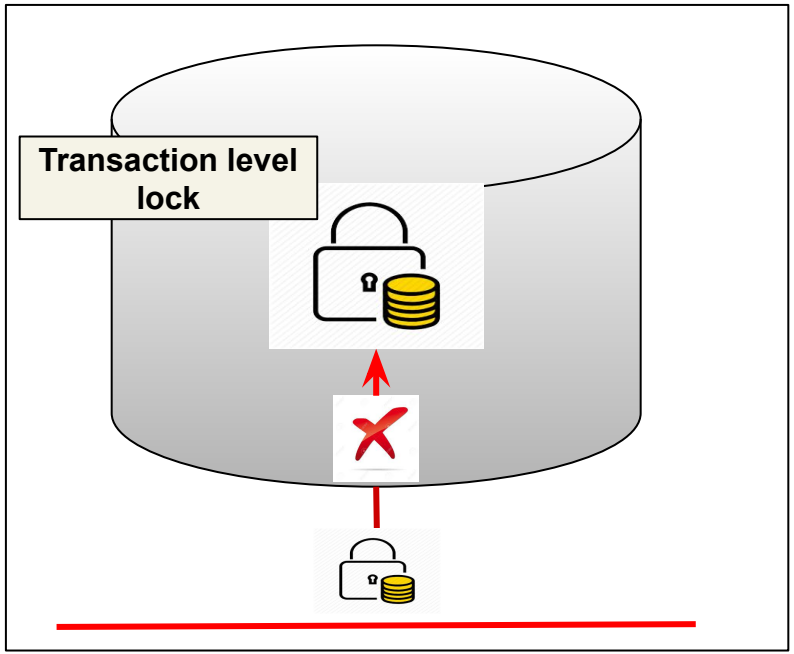
XID Consistency and PXC-8.0



- While running in cluster-mode:
  - Only PXC sub-system will be used for assigning XID.
  - **Statements that are not replicated (ALTER TABLE DISCARD/IMPORT TABLESPACE) are not logged to binlog.**
- **If session turns off cluster-mode (wsrep_on=off) then it will automatically set sql_log_bin=off too.**
- This ensures consistent XID generation from PXC sub-system avoiding XID inconsistency.

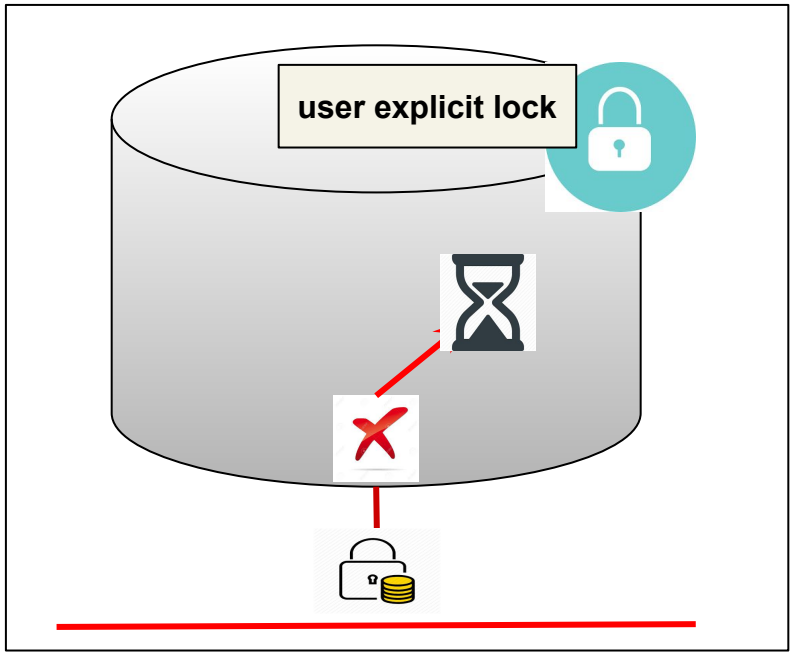
DD-tables Explicit Locks and PXC-8.0

DD-tables Explicit Locks and PXC-8.0



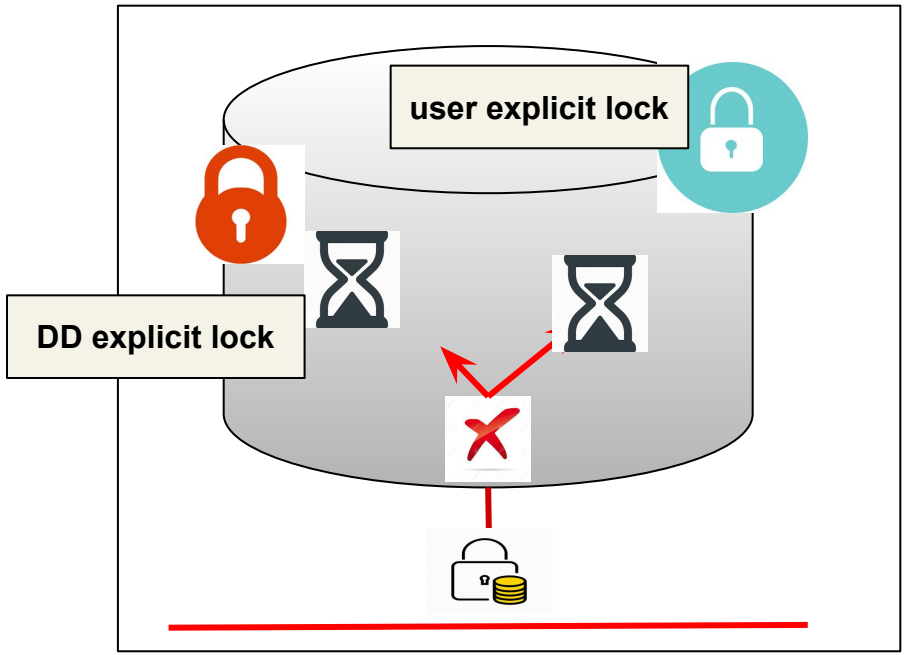
- In PXC replicated transaction takes priority. If replicated transaction traces a local transaction with conflicting lock(s) then such local transaction is force aborted and rolled back. This is applicable to transaction or statement level locks only.

DD-tables Explicit Locks and PXC-8.0



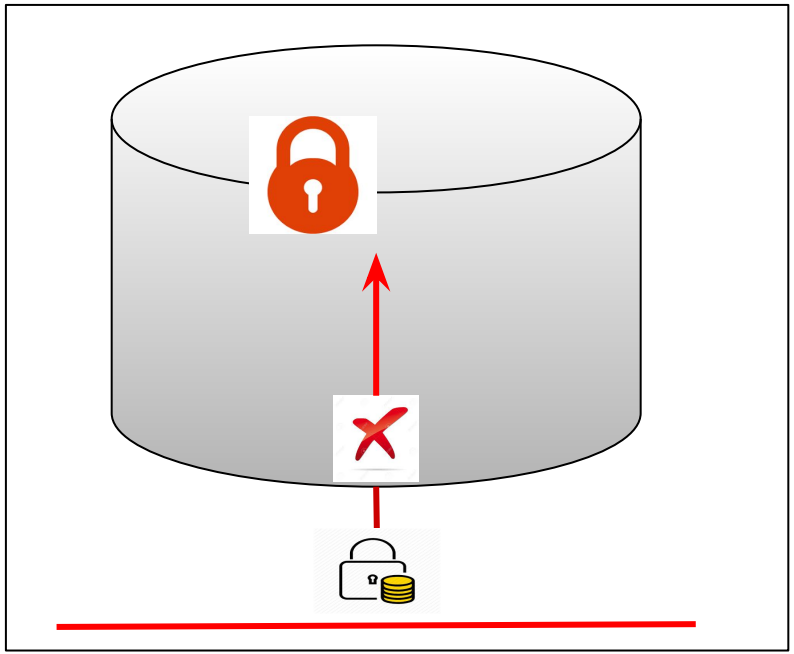
- If there are user established explicit locks (taken through commands like LOCK TABLE) then replicated transaction waits for user to explicitly clear these locks.
- PXC doesn’t recommend user to use EXPLICIT locks as it limits ability to force abort local transaction there-by delaying application of a replicated transaction.

DD-tables Explicit Locks and PXC-8.0



- With introduction of DD-tables, MySQL started using explicit locks for DD-tables updates. These explicit locks are not user driven and meant only for DD tables updates.
- If a local session is holding explicit lock then such local sessions are **non-preemptable** causing applier to wait thereby delaying application of replicated transaction. This can eventually stall complete cluster.

DD-tables explicit locks and PXC-8.0



- PXC now classifies explicit locks as
  - **Preemptable (DD-explicit)**
  - **Non-preemptable (user-explicit)**

Internal DD-tables explicit locks as preemptable thereby allowing background applier thread to kill the said transaction despite it holding so called EXPLICIT locks.

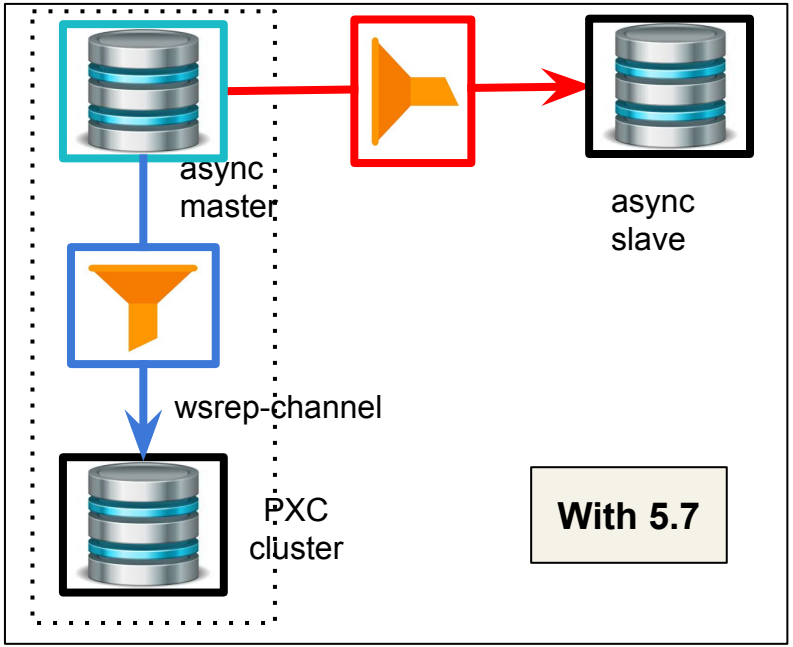
- No change in user held explicit lock. Applier waits for user to explicitly release locks.

Replication Filters and PXC-8.0

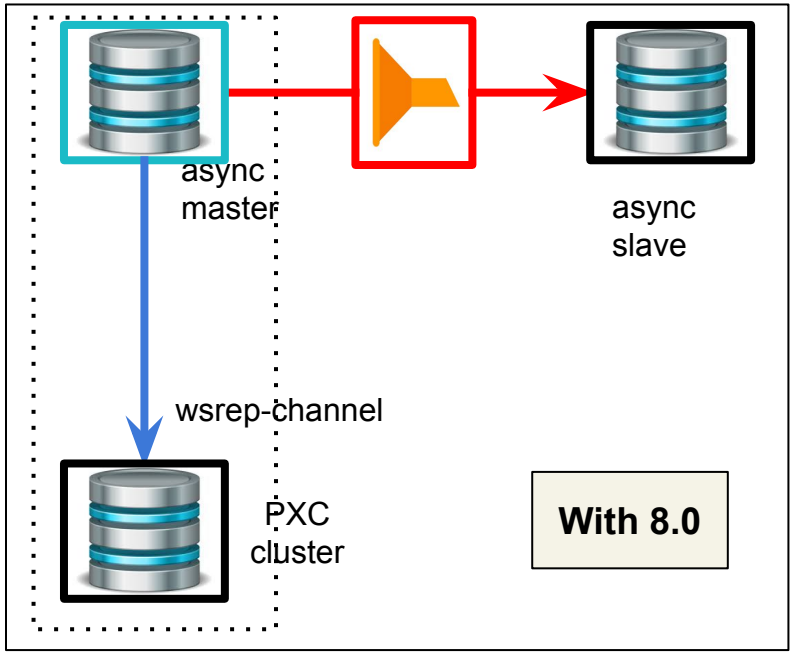
Replication Filters and PXC-8.0

- MySQL-5.7 has a concept of global replication filter.
- MySQL-8.0, introduces channel based filter.
- A link between 2 server is deemed as channel and now with 8.0 user can define separate filter for each link.
  - master (m1) -> slave (s1) : f1
  - master (m2) -> slave (s1) : f2

- PXC links are also deemed as channel and PXC till date inherited global replication filter.



Replication Filters and PXC-8.0

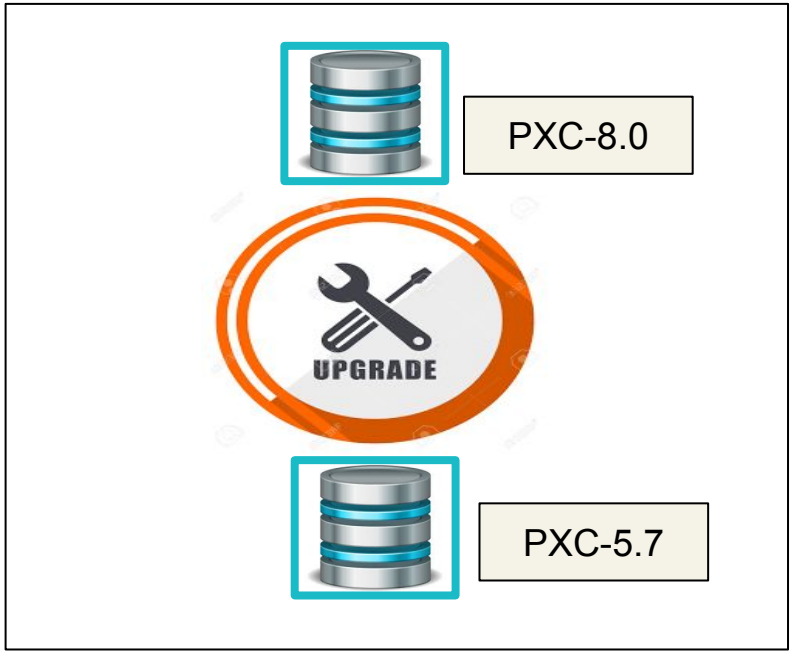


- Applying filter for PXC channel actually doesn't make much sense since it is multi-master solution and each node needs to see all the objects and changes.
- Starting PXC-8.0, wsrep channel is created but no more configurable. User can't set replication filter or use normal channel commands against this filter.

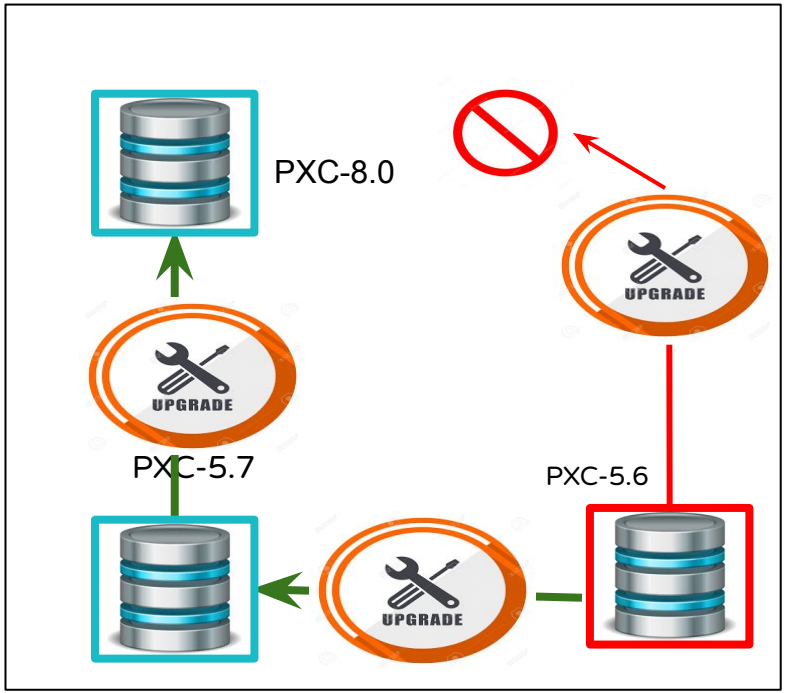
Upgrade Dependency with PXC-8.0

Upgrade Dependency with PXC-8.0

- Given REDO and DD changes, MySQL-8.0 data-directory is not compatible with MySQL-5.7.
- MySQL recommends running upgrade for even minor releases.
- PXC can support upgrade from **5.7-DONOR to 8.0-JOINER** but not vice-versa. SST script has been updated to enforce the needed checks.



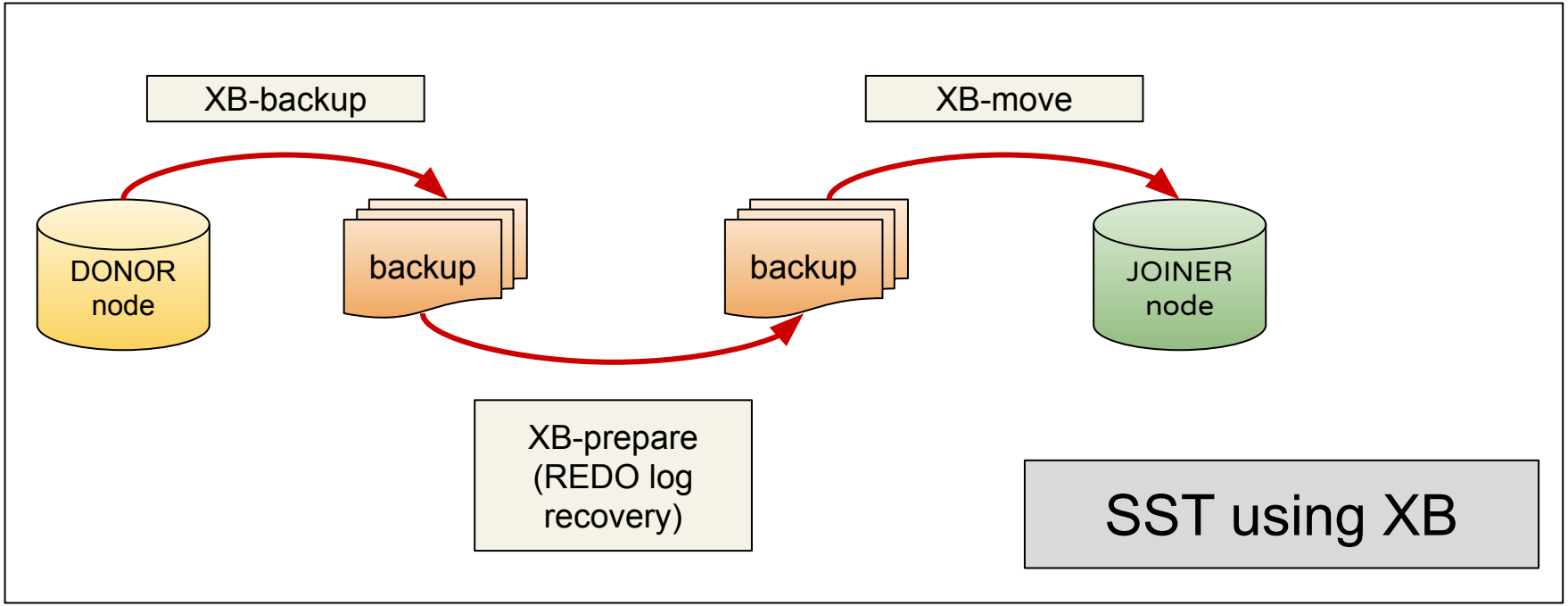
Upgrade Dependency with PXC-8.0



- PXC will not support upgrade from **5.6 to 8.0 (5.6-DONOR to 8.0-JOINER)**. User should upgrade to 5.7 and then continue to upgrade further.
- Percona recommends upgrading to PXC-5.7.25+ before upgrading to 8.0 to ensure proper enforcement of upgrade checks.

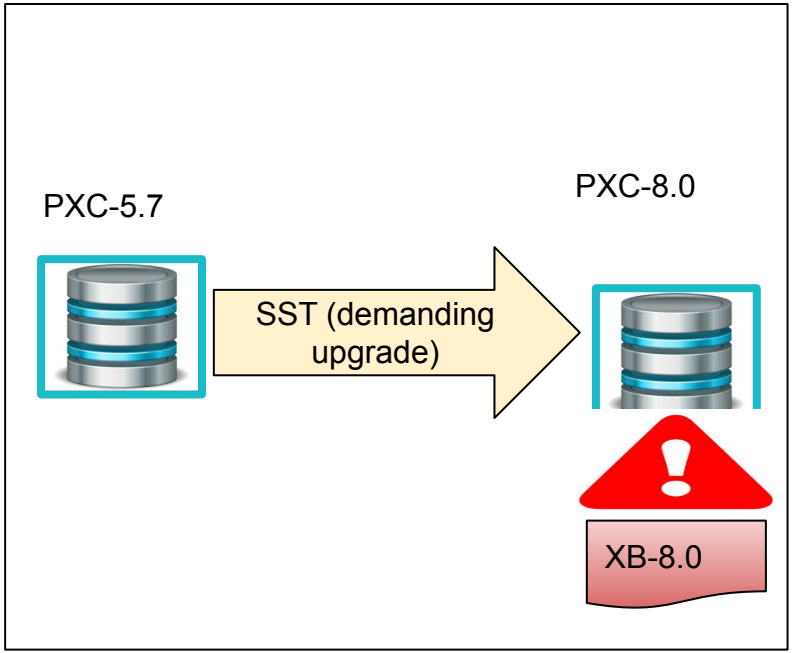
Upgrade Post SST with PXC-8.0

Upgrade Post SST and PXC-8.0

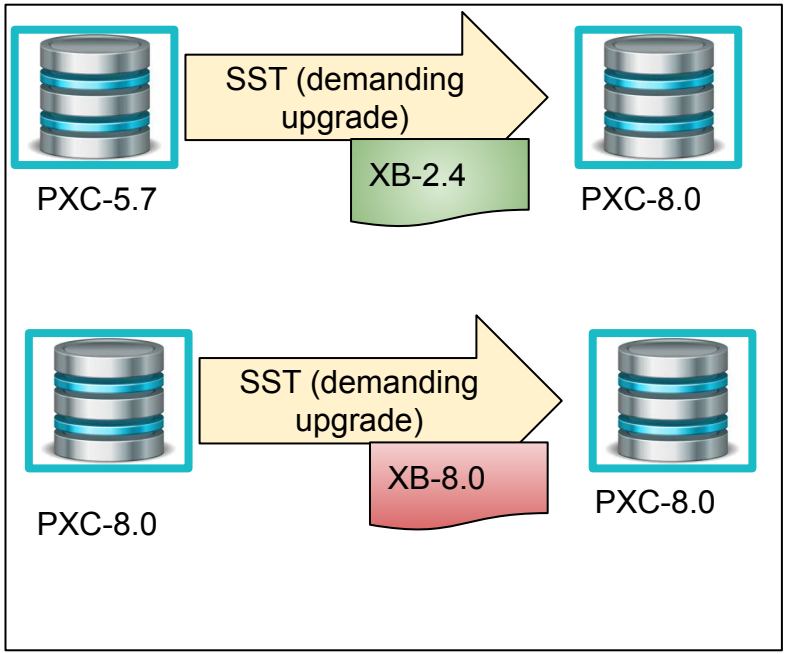


Upgrade Post SST and PXC-8.0

- MySQL-8.0 has changed REDO log format.
- This means copying over live 5.7-data-directory and trying to recover it with XB-8.0 binaries (that are MySQL-8.0 compatible) will not work.
- This requirement demands:
  - **If DONOR is < 8.0 then use XB-2.4 to prepare.**
  - **If DONOR is > 8.0 then use XB-8.0 to prepare.**



Upgrade Post SST and PXC-8.0



- Node can't have 2 packages of XB installed on the same node.
- Also, in the past we have seen user facing restriction in using different XB binaries due to its dependency on PXC.
- Both of these pain-point are now addressed with PXC-8.0.

- **PXC-8.0 package will ship its own version of XB binaries to allow auto-node provisioning from 5.7-DONOR to 8.0-JOINER.**

Upgrade Post SST and PXC-8.0

- **PXC-8.0 will only support XB based SST.**

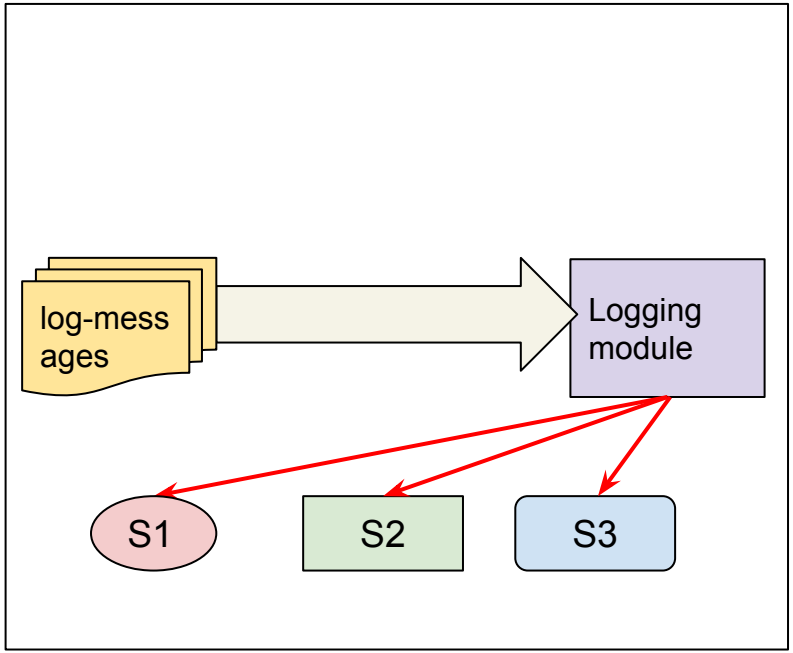
- mysqldump is deprecated in 5.7 and has been removed from PXC-8.0.

- rsync is not supported due to
  - REDO log format change.
  - Updated Galera-4 (G-4) protocol.

Error Logging with PXC-8.0

Error Logging with PXC-8.0

- MySQL-8.0 improved the error/info/warning logging framework.
- User can route all the messages through logging component(s) configurable through log_error_services.
- This features delays logging of the message during server start till the logging module is loaded. Messages are cached and once the the logging module is loaded messages are routed accordingly.



Error Logging with PXC-8.0

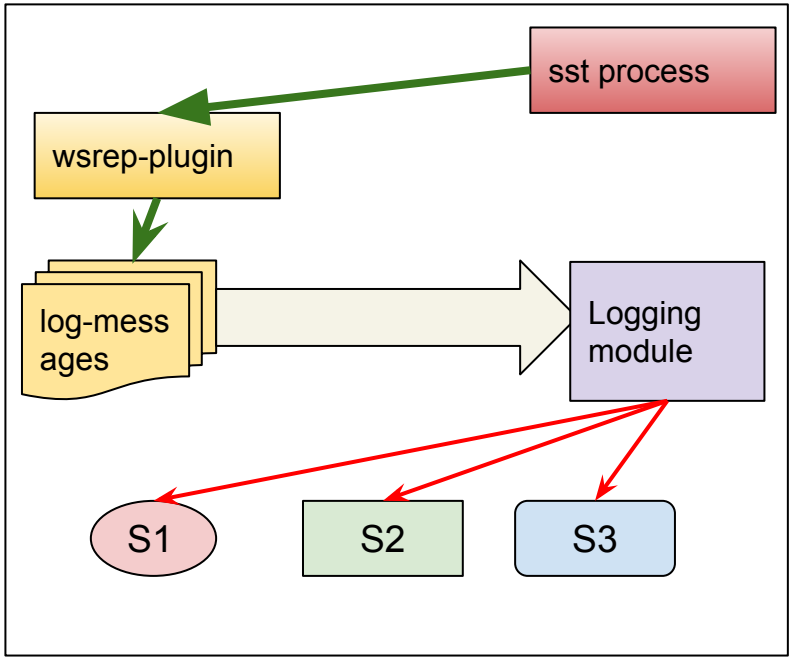
- MySQL-8.0 also emits the module name representing where the message is originating from

2019-05-14T07:22:50.264963Z 3 [Note] [MY-011089] [Server] Data dictionary restarting version '80014'.

2019-05-14T07:22:50.791202Z 3 [Note] [MY-012357] [InnoDB] Reading DD tablespace files

2019-05-14T07:22:50.794591Z 3 [Note] [MY-012356] [InnoDB] Validated 6/6 tablespaces

Error Logging with PXC-8.0



- PXC-8.0 inherits this new logging framework and PXC messages are now logged with module name too.

  `PXC defines 3 modules:`

  - `WSREP`
  - `GALERA`
  - `WSREP-SST`

- Also, worth noting that SST originating messages from an external process (wsrep-xtrabackup-v2) are now redirected to mysqld process (wsrep-plugin) for consistent logging and processing.

Error Logging with PXC-8.0

2019-05-14T09:07:36.728088Z 0 [Note] [MY-000000] [WSREP-SST] (debug) Cleaning up temporary directories

2019-05-14T09:07:36.736437Z 0 [Note] [MY-000000] [Galera] 0.0 (n1): State transfer to 1.0 (n2) complete.

2019-05-14T09:07:36.736459Z 0 [Note] [MY-000000] [Galera] Shifting DONOR/DESYNCED -> JOINED (TO: 0)

2019-05-14T09:07:36.736758Z 0 [Note] [MY-000000] [Galera] Member 0.0 (n1) synced with group.

2019-05-14T09:07:36.736771Z 0 [Note] [MY-000000] [Galera] Shifting JOINED -> SYNCED (TO: 0)

2019-05-14T09:07:36.736859Z 2 [Note] [MY-000000] [WSREP] Synchronized with group, ready for connections

2019-05-14T09:07:36.736887Z 2 [Note] [MY-000000] [WSREP] Setting wsrep_ready to true

2019-05-14T09:07:36.736913Z 2 [Note] [MY-000000] [WSREP] wsrep_notify_cmd is not defined, skipping notification.

2019-05-14T09:07:48.011987Z 0 [Note] [MY-000000] [Galera] 1.0 (n2): State transfer from 0.0 (n1) complete.

2019-05-14T09:07:48.012500Z 0 [Note] [MY-000000] [Galera] Member 1.0 (n2) synced with group.

Error Logging with PXC-8.0

- As mentioned above, if there is no error during the startup, MySQL will delay logging/processing of the messages till log module is loaded.

- In PXC, log module is loaded post SST and so if there is no error message during SST then appearance of first log message may take time.

In the case below, after starting node-2, messages started appearing after 32 seconds (timestamp and order are maintained)

```
2019-05-15T07:59:36.250167Z 0 [Warning] [MY-010139] [Server] Changed
limits: max_open_files: 1024 (requested 8161)

....

2019-05-15T08:00:08.957156Z 0 [Note] [MY-011245] [Server] Plugin mysqlx
reported: 'Scheduler 'network', create threads'
```

CATS and PXC-8.0

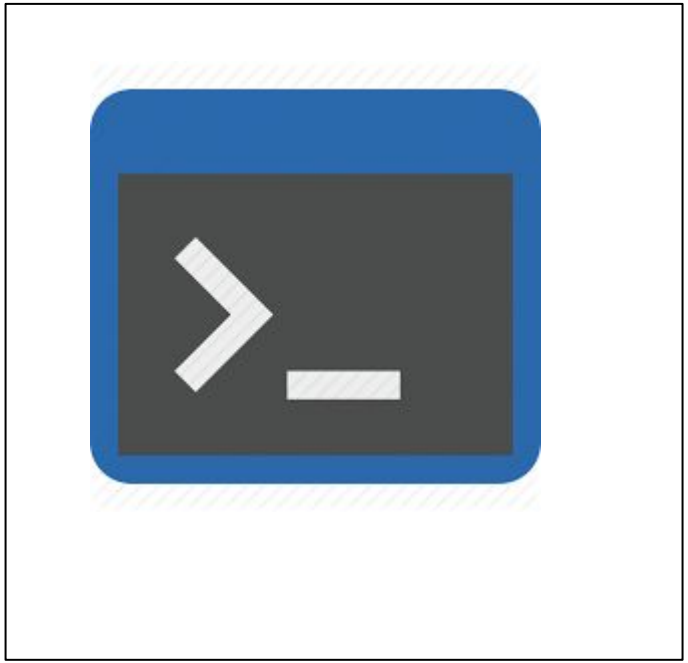
CATS and PXC-8.0

- MySQL-8.0 introduced Contention Aware Transaction Scheduling algorithm. A transaction that can unblock the majority of the waiting/blocked transactions, is first to get hold of the lock.
- PXC has a concept of applier that needs to take priority, irrespective of the number of waiting locks/transactions.
- PXC-8.0 doesn't yet support CATS instead continues to use FIFO scheduling.

XPlugin with PXC-8.0

XPlugin and PXC-8.0

- In MySQL-8.0, x-plugin is enabled by default (to be used with new mysql shell).
- X-Plugin loading sequence (during boot) fire queries against server.
- PXC can’t start accepting queries till PXC is in SYNCED mode.
- This means with PXC, X-plugin is loaded once node enter synced state.
- If SST or IST is involved, sync may take longer and so would start of mysqlx plugin.



Deprecated and Removed with PXC-8.0

Deprecated and Removed with PXC-8.0

- **mysqldump**: deprecated in PXC-5.7
  - Logical backup. No takers
- **rsync**: limited support
  - REDO log format change, G-4 protocol
- **wsrep_force_binlog_format**: deprecated in PXC-5.7
  - PXC-8.0 operates only with binlog_format=ROW.
- **wsrep_convert_lock_to_trx**: deprecated in PXC-5.7.
- **wsrep_preordered**: deprecated in PXC-5.7 *(With performance fix this is no more needed)*.

Deprecated and Removed with PXC-8.0

- **innodb_disallow_writes:** deprecated in PXC-5.7
  - Deprecated in favor of innodb_read_only.
  - Was used by rsync. rsync is no more supported.
- **session level binlog_format=STATEMENT**: deprecated in PXC-5.7.
- **wsrep_drupal_282555_workaround**: deprecated in PXC-5.7.
  - Auto-increment bug that caused duplicate value generation is now fixed.

Create Table As Select and PXC-8.0

Create Table As Select and PXC-8.0

- Known to introduce GTID inconsistency with MySQL-8.0 so blocked with gtid_mode=ON.
- PXC doesn't recommend use of CTAS (already blocked with pxc_strict_mode=ENFORCING).
- CTAS further leads to XID inconsistency with MySQL-8.0. Reported here (bug#93948).
- **PXC-8.0 will now replicate CTAS through TOI (Total Order Isolation) just like other DDLs.**

Create Table As Select and PXC-8.0

- This has the following repercussions:
  - `CREATE TABLE dest SELECT * from src;`

wsrep-replicate-myisam=off	n1-node	n2-node
PXC-5.7	**dest table created and loaded**	**dest table created but not loaded**
PXC-8.0	**dest table created and loaded**	**dest table created and loaded**

source table local	n1-node	n2-node
PXC-5.7	**dest table created and loaded**	**dest table created and loaded**
PXC-8.0	**dest table created and loaded**	**dest table not-created**

SET PERSIST (mysqld-auto.cnf) and PXC-8.0

SET PERSIST (mysqld-auto.cnf) and PXC-8.0

- MySQL-8.0 allows the user to persist the changed configuration across restart. To facilitate this, it creates a file named mysqld-auto.cnf.
- This file is not copied over as part of SST from DONOR to JOINER.

What’s New with PXC-8.0?

What's New with PXC-8.0?

- Internal SST user (No more wsrep_sst_auth).
- Auto-upgrade.
- Avoid setting up automatic slave if DONOR is slave.
- Additional stats through SHOW STATUS.

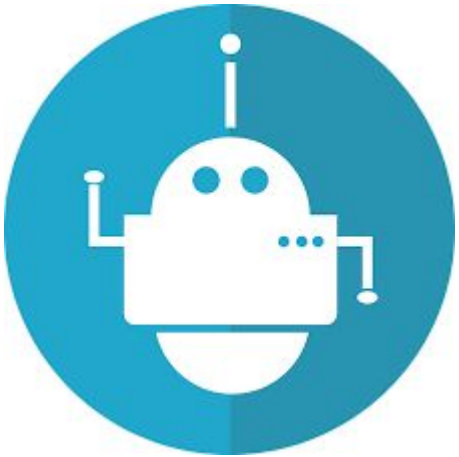
Support for Galera-4 (G-4)

Internal SST User

Internal SST User

- For doing SST, PXC demands presence of the user on DONOR node that can then be used to take backup from DONOR.
- This creates 3 problems for the administrator:
  - **Administrator needs to create the said user and grant it correct privileges.**
  - **User is retained even post SST (as next SST may need it).**
  - **Credentials for the said user are specified in human readable format in configuration file. Raising Security Alarm.**

Internal SST User



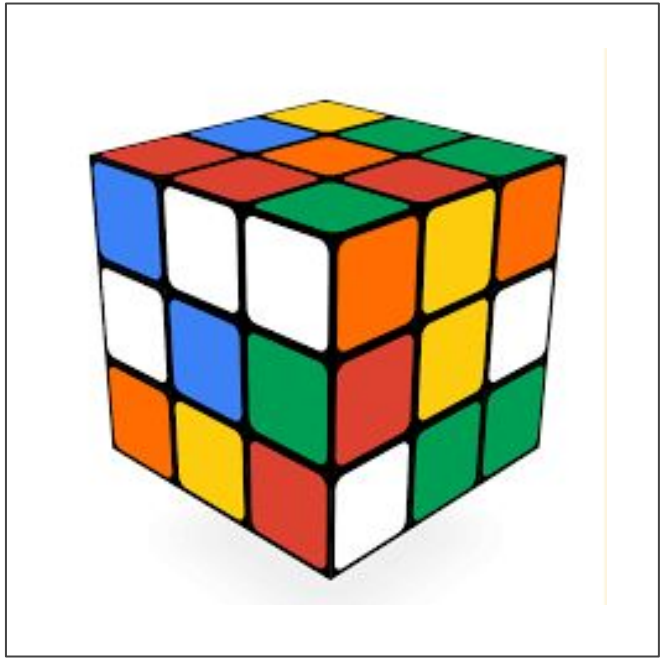
- PXC-8.0 got rid of this. ~~wsrep_sst_auth~~ is now removed from PXC-8.0.
- PXC-8.0 introduces concept of internal-sst-user (mysql.pxc.sst.user).
- On DONOR:
  - This user is created on donor when sst action starts and is removed on completion of sst action.
  - User is assigned a random on the fly generated password.
  - Needed privileges are worked out internally and assigned to user to take backup.

JOINER doesn't need said user for completion of SST.

Auto-Upgrade

Auto-Upgrade

- MySQL recommends running upgrade even for minor version upgrade.
- With PXC DONOR and JOINER setup this could be challenging.

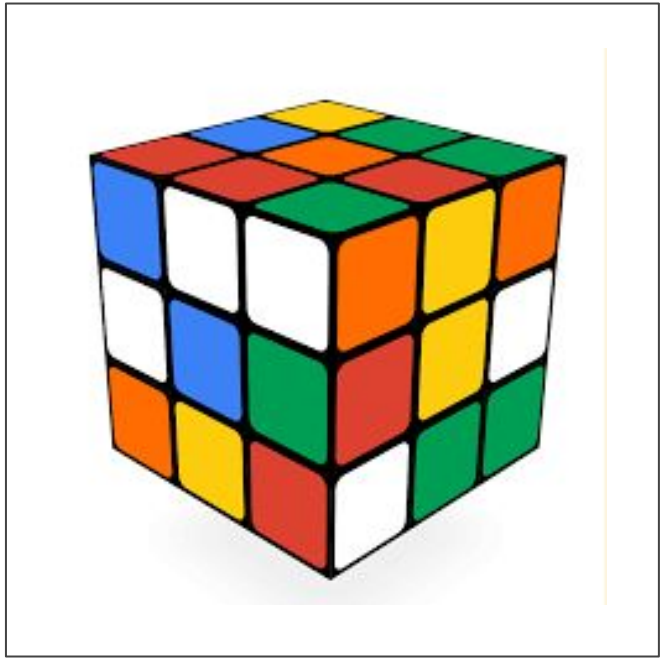


Auto-Upgrade

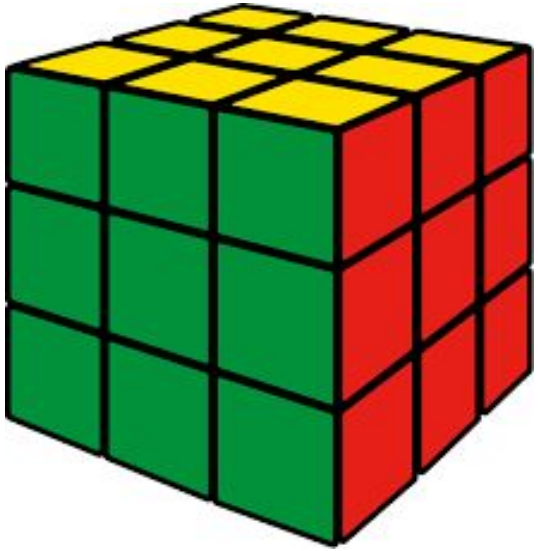
- DONOR (5.7) -> JOINER (8.0)
- DONOR donates 5.7 data-directory that is then prepared on JOINER side using XB-2.4 (*now PXC-8.0 ships binaries for XB-2.4*).
- Said prepared data-directory is then copied over at needed location (by XB move stage).

Data-directory is still 5.7 compatible and 8.0 binaries can't work with it. There is need for upgrade.

[MySQL-8.0.16 now provides inherent upgrade option]



Auto-Upgrade



- PXC introduces **auto-upgrade**.
- Simply boot the JOINER. JOINER and DONOR will handshake to see if the versions are compatible ensuring DONOR version <= JOINER version.
- On successful SST, JOINER will automatically restart the server for mysql-upgrade (or inherent upgrade with mysql-8.0.16+) and upgraded data-directory is then handed over to main flow for further processing.

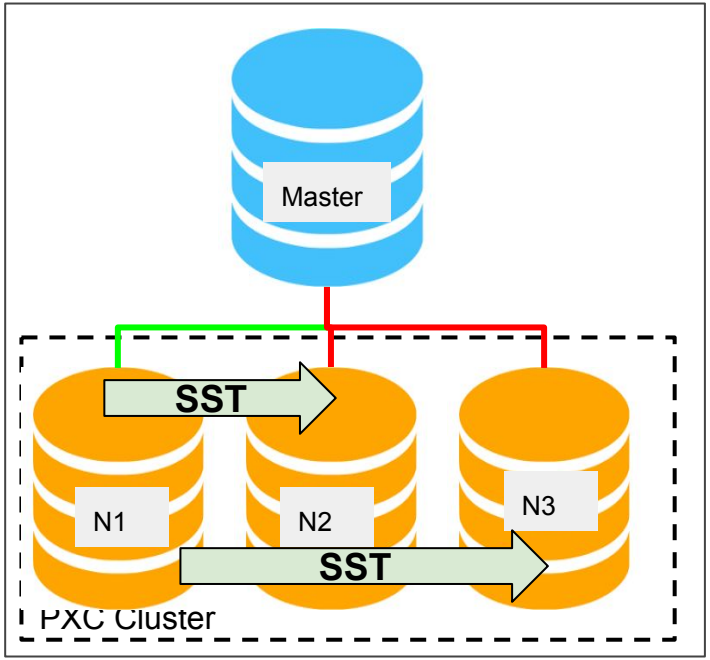
Takes care of minor version upgrade too.

Helpful with Cloud operations.

Accidentally Configuring Multiple Slaves Post SST

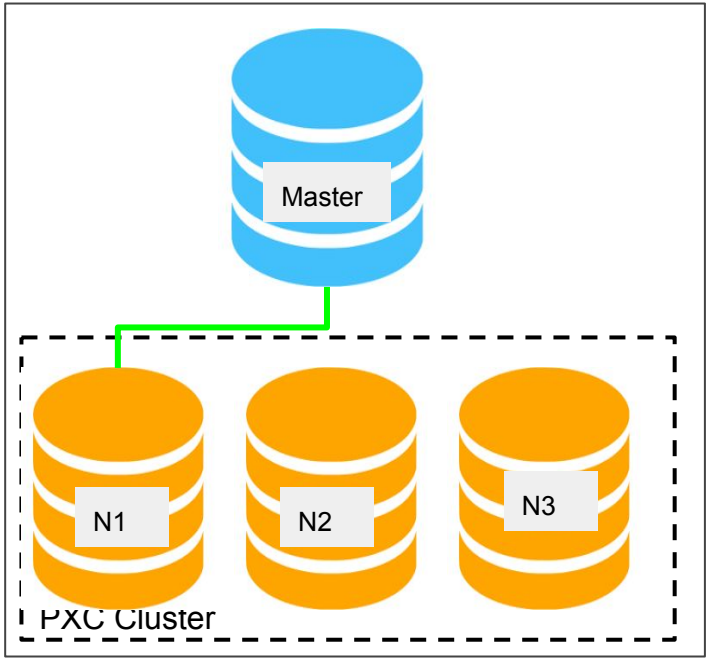
Accidentally Configuring Multiple Slave Post SST

- One other problem that is often reported post SST is automatic spawn-up of multiple slaves when user has configured only one of the PXC node as slave.
- If DONOR node is acting as active async slave then post SST the slave_master_info is copied over there-by creating an unwanted additional async slave.
- No more unconfigured slaves with PXC-8.0.



Accidentally Configuring Multiple Slave Post SST

- PXC-8.0 as part of post-processing executes RESET SLAVE ALL to clear the slave information that is copied over as part of SST.
- This all happens behind the scenes as part of SST process. So SST process is not merely copying over of the data-directory.
- PXC-8.0 is equipped with **SMART SST**.



Additional Stats Through SHOW STATUS

Additional Stats Through SHOW STATUS

- PXC has the concept of monitor. Monitor is configured with conditions that decide which transactions are allowed to pass through it and execute in critical section.
- For example:
  - **Apply Monitor** allows parallel application of the transaction.
  - **Commit Monitor** allows only one transaction (ordered by sequence number) to proceed with action.
  - **Local Monitor** is meant to order local action including sync-wait/pause.

- These monitors play an important role in understanding which transactions/threads are operating in which part of critical section.

Additional Stats Through SHOW STATUS

- Stats for this monitor is now exposed through SHOW STATUS

```
| wsrep_monitor_status (L/A/C) | [ (17, 10), (3, 3), (3, 3) ] |
....
| wsrep_monitor_status (L/A/C) | [ (21, 10), (3, 3), (3, 3) ] |
```

- L/A/C represent Local/Apply/Commit and pair represent (last_entered_, last_left_)
- Say for L = 17, 10 that suggest last_left_= 10 and currently local monitor is being held by write-set with seqno=11 and last_entered_=17 (that is waiting) to enter critical section.

Galera-4

Galera-4

- Streaming replication. Now user can replicate large transactions.
- System tables to monitor state of cluster.
- Improved wsrep-debug to control logging granularities.
- Inbuilt predefined synchronization functions based on GTID.
- Ability to suppress/ignore some of the DML/DDL application errors.

Galera-4 is coming with PXC-8.0

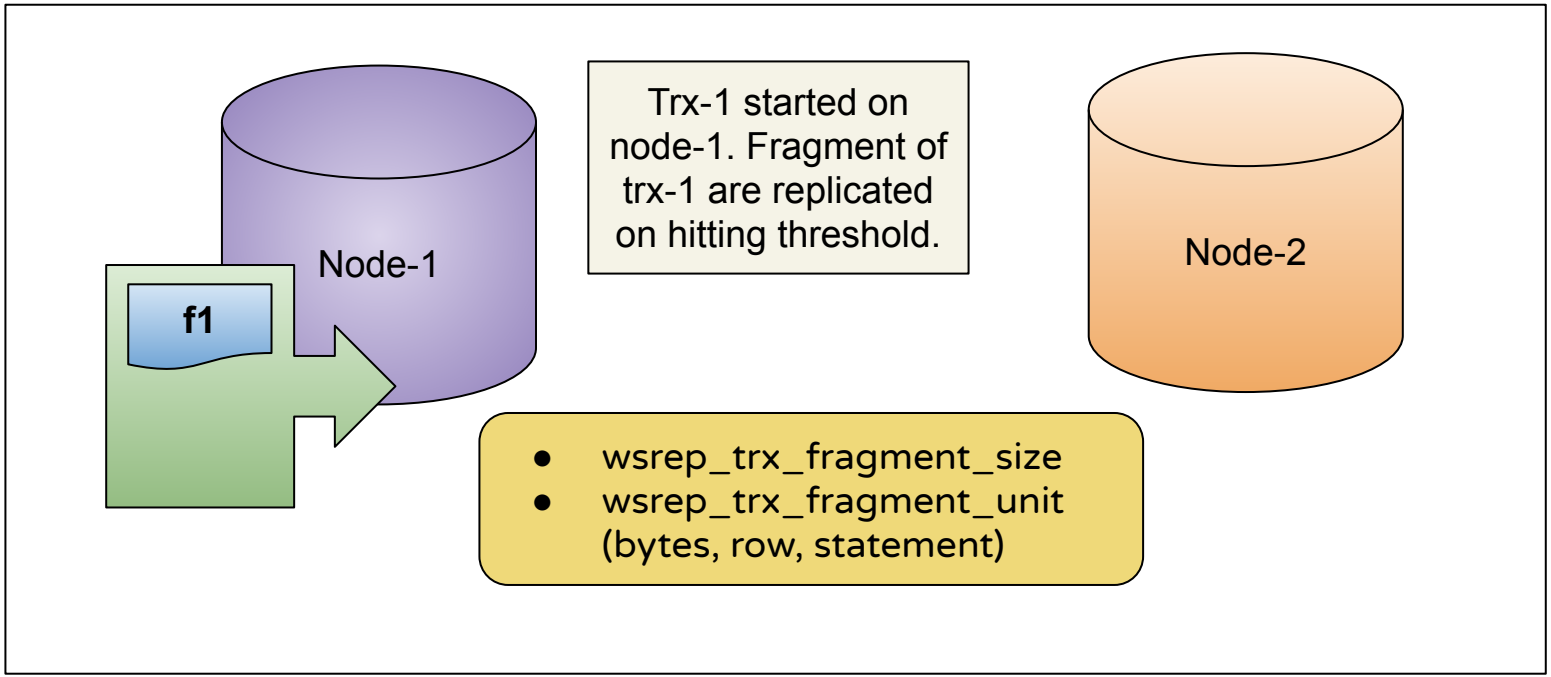
Galera-4 - Streaming Replication

Galera-4 - Streaming Replication

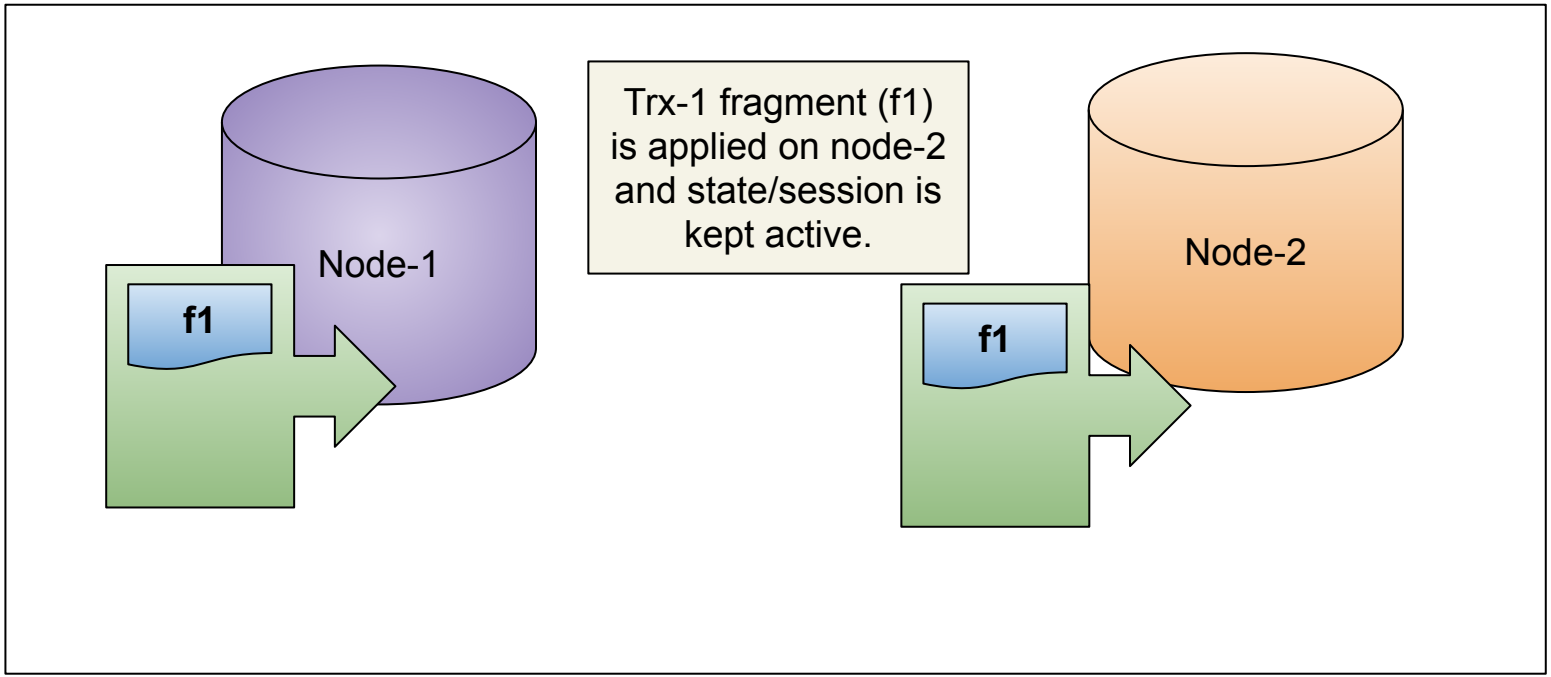
- Galera-3 added cap on how big a single transaction to replicate is. Defined through wsrep_max_ws_rows/wsrep_max_ws_size
- This was too restrictive with production workload and user had to handle this in application.
- Technologically it was limiting due to the time it takes to replicate (network bandwidth hogging) and apply (execution time) larger transaction.

Galera-4 introduces Streaming Replication that replicates transaction in small chunk

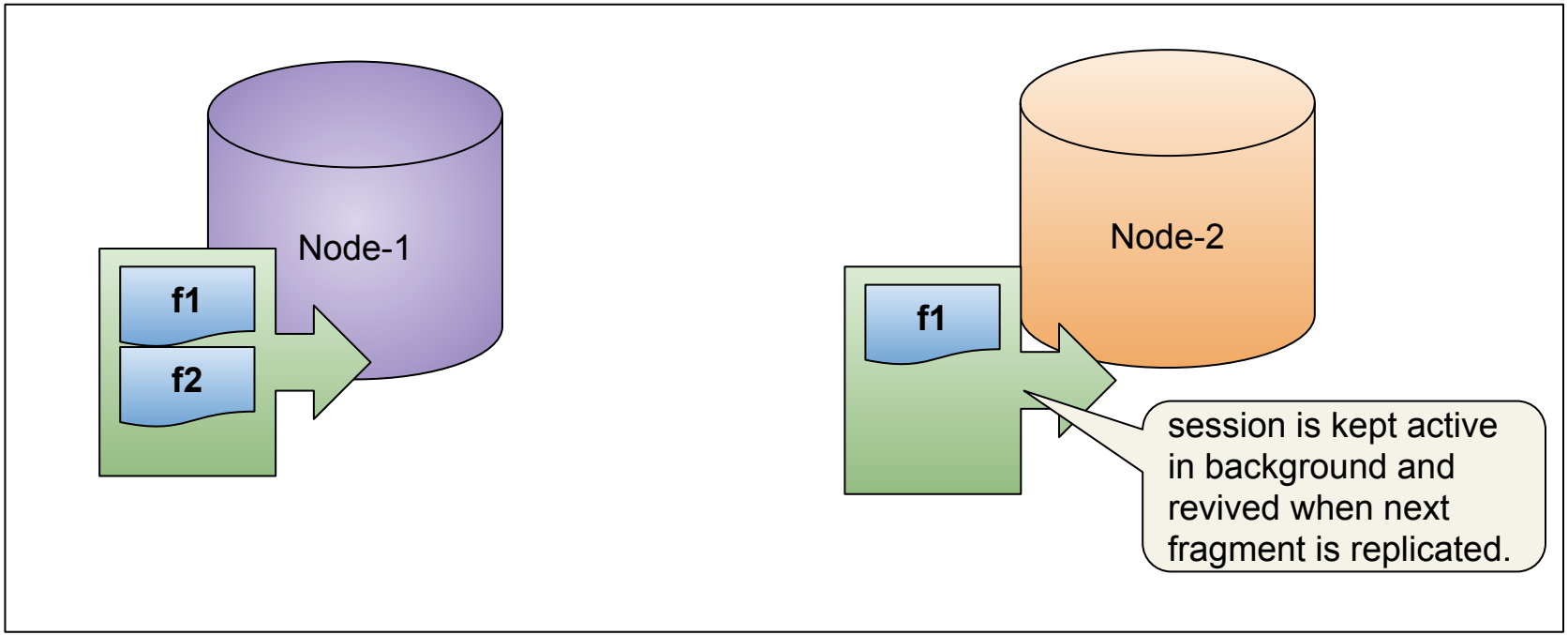
Galera-4 - Streaming Replication



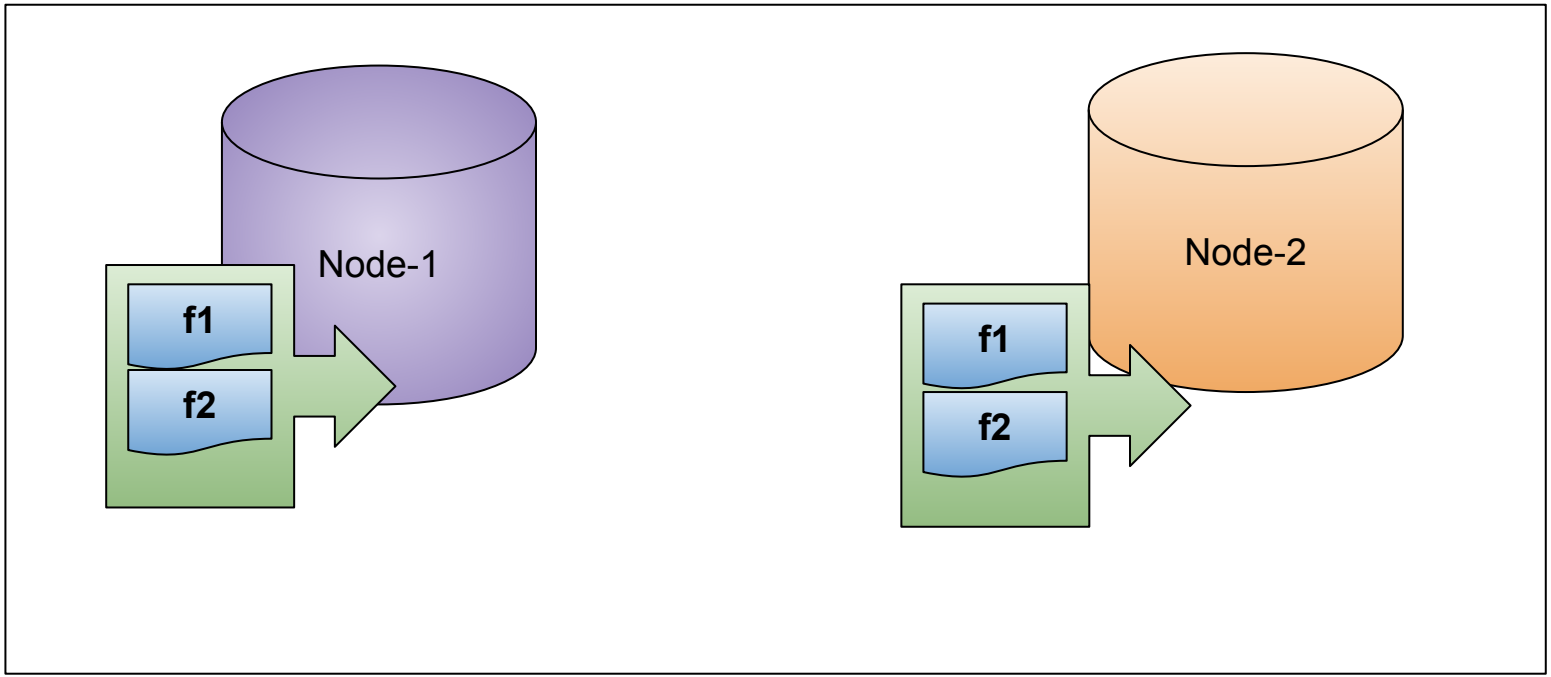
Galera-4 - Streaming Replication



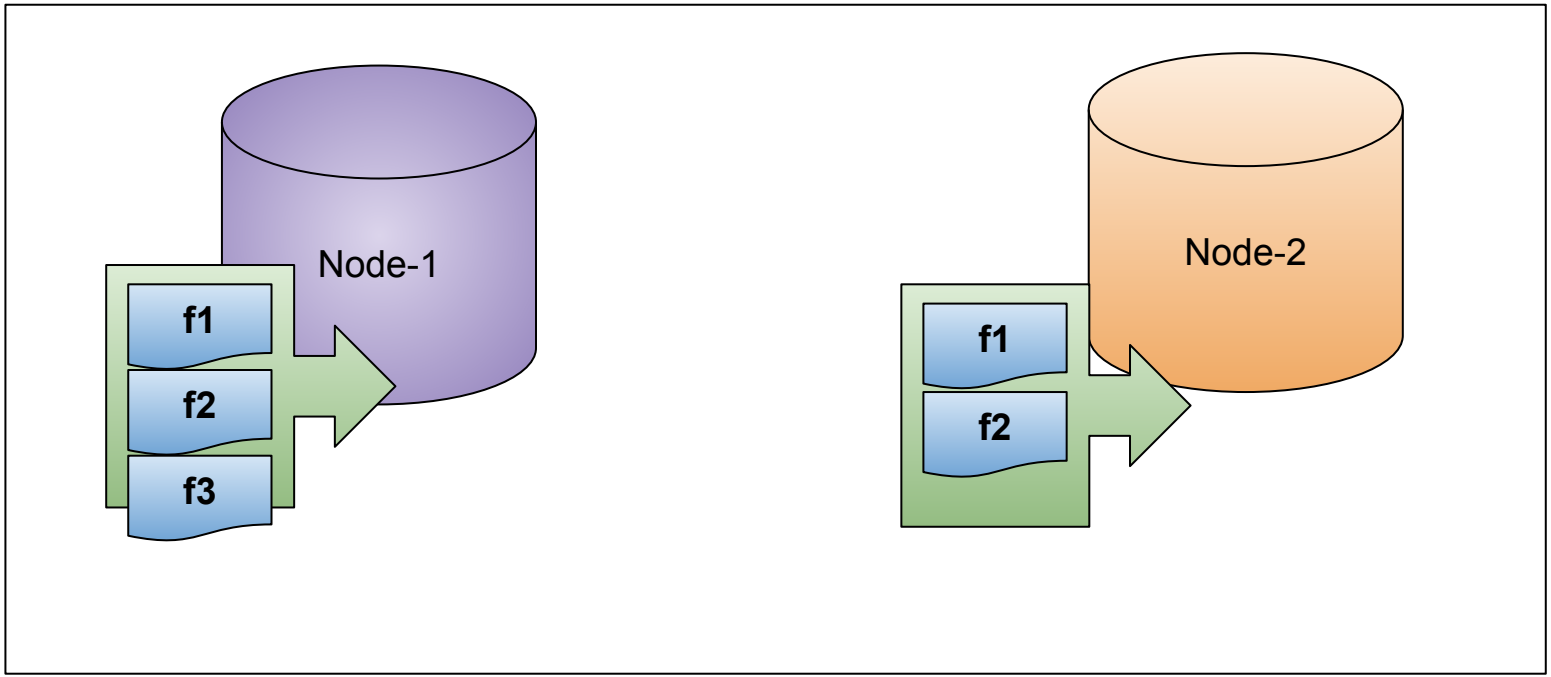
Galera-4 - Streaming Replication



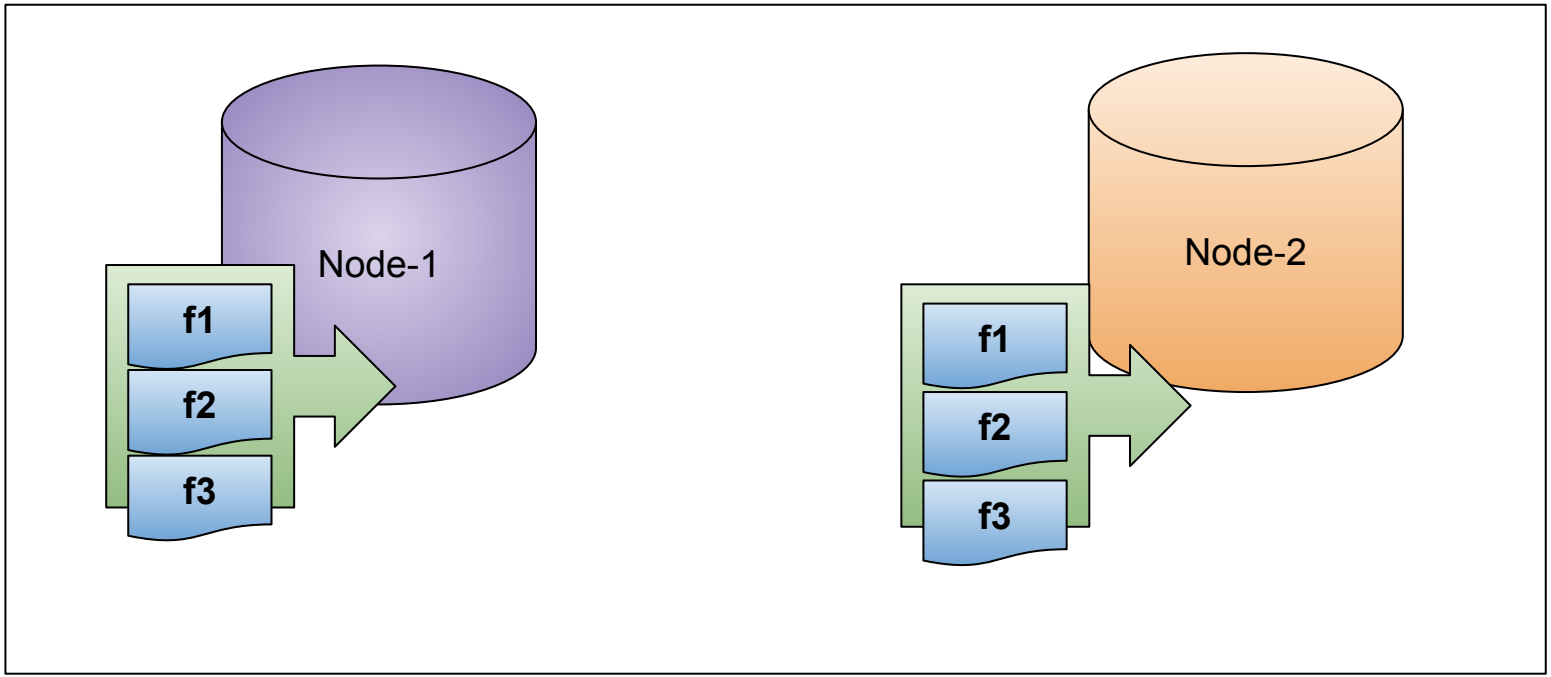
Galera-4 - Streaming Replication



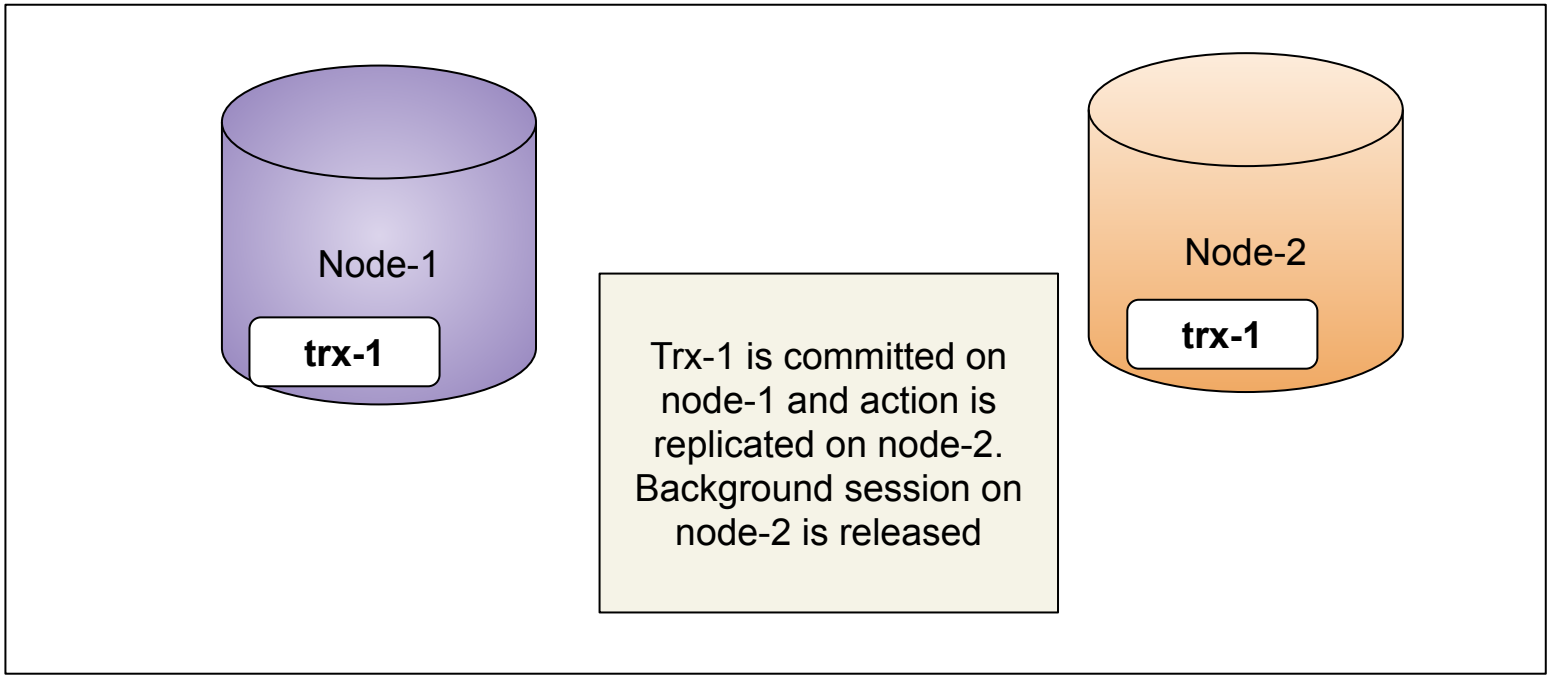
Galera-4 - Streaming Replication



Galera-4 - Streaming Replication



Galera-4 - Streaming Replication



Galera-4 - System Tables

Galera-4 - System Tables

- Galera-4 also introduces 3 system tables that capture cluster state

```
| wsrep_cluster                | (cluster information)
| wsrep_cluster_members        | (cluster node(s) information)
| wsrep_streaming_log          | (fragment cached for SR)
```

Galera-4 - Improved Logging Infrastructure

Galera-4 - Improved Logging Infrastructure

- `wsrep_debug` is multi-valued variable with PXC-8.0
  - 0: NONE
  - 1: SERVER
  - 2: TRANSACTION
  - 3: STREAMING
  - 4: CLIENT

- Significant improvement in logging information. User can control granularities based on wsrep_debug values.

Galera-4 - Synchronization Function

Galera-4 - Synchronization Function

- Inbuilt predefined synchronization functions based on GTID
  - `wsrep_last_written_gtid()`
    - Indicates last written gtid (GTID:seqno) by given session.
    - Now user can tell which action was last executed by this session.
  - `wsrep_last_seen_gtid()`
    - Indicate last committed gtid.
  - `wsrep_sync_wait_upto('gtid', ['timeout'])`
    - Wait for cluster node to reach certain state (or timeout) before proceeding with local action.

Galera-4 - Ignore Application Error

Galera-4 - Ignore Application Error

- **`wsrep_ignore_apply_errors`**: helps ignore some common errors on replication (similar to skip-slave-error)
    - 0: No error skipped. 5.7 compatible behavior
    - 1: Ignore some of the DDL errors (drop-database, drop-table, drop-index, alter-table)
    - 2: Skip DML error (Only delete errors are ignored)
    - 4: Ignore all the DDL errors.

This functionality is still under development and semantics are likely to change.

More to Come….

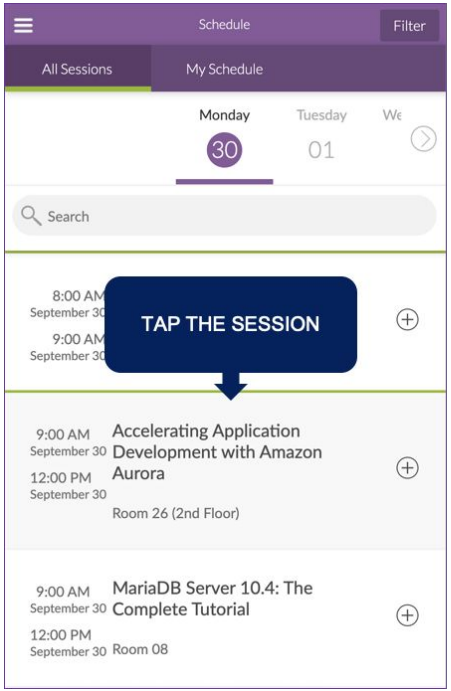
More to Come ….

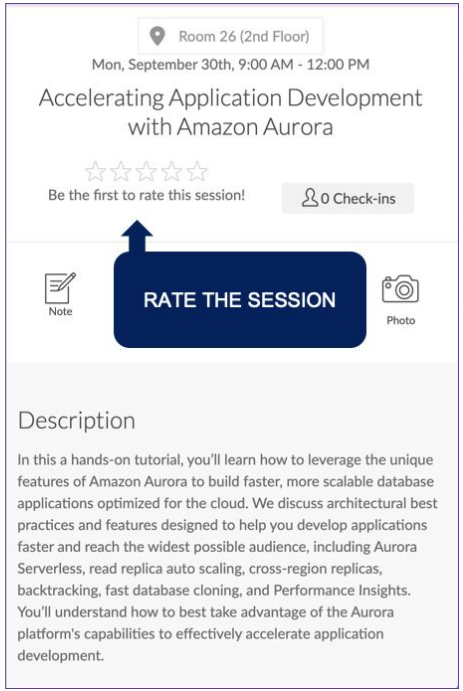
More features are being worked out.

- Improved resilient cluster.
- Cloud friendly cluster.
- Inheriting complete encryption support form MySQL/PS upstream.
- Improved packaging.
- …..

Thanks for Listening. Any Questions?

Rate My Session





We’re Hiring!

Percona’s open source database experts are true superheroes, improving database performance for customers across the globe.

Our staff live in nearly 30 different countries around the world, and most work remotely from home.

Discover what it means to have a Percona career with the smartest people in the database performance industries, solving the most challenging problems our customers come across.