MONITORING CLUSTER STATUS

From the database client, you can check the status of write-set replication throughout the cluster using standard queries. Status variables that relate to write-set replication have the prefix wsrep_, meaning that you can display them all using the following query:

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

+------------------------+-------+
| Variable_name          | Value |
+------------------------+-------+
| wsrep_protocol_version | 5     |
| wsrep_last_committed   | 202   |
| ...                    | ...   |
| wsrep_thread_count     | 2     |
+------------------------+-------+
```

Note: See Also: In addition to checking status variables through the database client, you can also monitor for changes in cluster membership and node status through wsrep_notify_cmd.sh. For more information on its use, see Notification Command.

CHECKING CLUSTER INTEGRITY

The cluster has integrity when all nodes in it receive and replicate write-sets from all other nodes. The cluster begins to lose integrity when this breaks down, such as when the cluster goes down, becomes partitioned, or experiences a split-brain situation.

You can check cluster integrity using the following status variables:

- wsrep_cluster_state_uuid shows the cluster state UUID, which you can use to determine whether the node is part of the cluster.

  ```
  SHOW GLOBAL STATUS LIKE 'wsrep_cluster_state_uuid';

  +--------------------------+--------------------------------------+
  | Variable_name            | Value                                |
  +--------------------------+--------------------------------------+
  | wsrep_cluster_state_uuid | d6a51a3a-b378-11e4-924b-23b6ec126a13 |
  +--------------------------+--------------------------------------+
  ```

  Each node in the cluster should provide the same value. When a node carries a different value, this indicates that it is no longer connected to rest of the cluster. Once the node reestablishes connectivity, it realigns itself with the other nodes.

- wsrep_cluster_conf_id shows the total number of cluster changes that have happened, which you can use to determine whether or not the node is a part of the Primary Component.

  ```
  SHOW GLOBAL STATUS LIKE 'wsrep_cluster_conf_id';

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

```
+-----------------------+-------+
| wsrep_cluster_conf_id | 32    |
+-----------------------+-------+
```

Each node in the cluster should provide the same value. When a node carries a different, this indicates that the cluster is partitioned. Once the node reestablish network connectivity, the value aligns itself with the others.

- wsrep_cluster_size shows the number of nodes in the cluster, which you can use to determine if any are missing.

```
SHOW GLOBAL STATUS LIKE 'wsrep_cluster_size';

+--------------------+-------+
| Variable_name      | Value |
+--------------------+-------+
| wsrep_cluster_size | 15    |
+--------------------+-------+
```

You can run this check on any node. When the check returns a value lower than the number of nodes in your cluster, it means that some nodes have lost network connectivity or they have failed.

- wsrep_cluster_status shows the primary status of the cluster component that the node is in, which you can use in determining whether your cluster is experiencing a partition.

```
SHOW GLOBAL STATUS LIKE 'wsrep_cluster_status';

+----------------------+---------+
| Variable_name        | Value   |
+----------------------+---------+
| wsrep_cluster_status | Primary |
+----------------------+---------+
```

The node should only return a value of `Primary`. Any other value indicates that the node is part of a nonoperational component. This occurs in cases of multiple membership changes that result in a loss of quorum or in cases of split-brain situations.

> **Note: See Also**: In the event that you check all nodes in your cluster and find none that return a value of `Primary`, see Resetting the Quorum.

When these status variables check out and return the desired results on each node, the cluster is up and has integrity. What this means is that replication is able to occur normally on every node. The next step then is checking node status to ensure that they are all in working order and able to receive write-sets.

CHECKING THE NODE STATUS

In addition to checking cluster integrity, you can also monitor the status of individual nodes. This shows whether nodes receive and process updates from the cluster write-sets and can indicate problems that may prevent replication.

- wsrep_ready shows whether the node can accept queries.

  ```
  SHOW GLOBAL STATUS LIKE 'wsrep_ready';

  +---------------+-------+
  | Variable_name | Value |
  +---------------+-------+
  | wsrep_ready   | ON    |
  +---------------+-------+
  ```

  When the node returns a value of `ON` it can accept write-sets from the cluster. When it returns the value `OFF`, almost all queries fail with the error:

  ```
  ERROR 1047 (08501) Unknown Command
  ```

- wsrep_connected shows whether the node has network connectivity with any other nodes.

  ```
  SHOW GLOBAL STATUS LIKE 'wsrep_connected';

  +-----------------+-------+
  | Variable_name   | Value |
  +-----------------+-------+
  | wsrep_connected | ON    |
  +-----------------+-------+
  ```

  When the value is `ON`, the node has a network connection to one or more other nodes forming a cluster component. When the value is `OFF`, the node does not have a connection to any cluster components.

  Note: The reason for a loss of connectivity can also relate to misconfiguration. For instance, if the node uses invalid values for the wsrep_cluster_address or wsrep_cluster_name parameters.

  Check the error log for proper diagnostics.

- wsrep_local_state_comment shows the node state in a human readable format.

  ```
  SHOW GLOBAL STATUS LIKE 'wsrep_local_state_comment';

  +---------------------------+--------+
  | Variable_name             | Value  |
  +---------------------------+--------+
  | wsrep_local_state_comment | Joined |
  +---------------------------+--------+
  ```

When the node is part of the Primary Component, the typical return values are `Joining`, `Waiting on SST`, `Joined`, `Synced` or `Donor`. In the event that the node is part of a nonoperational component, the return value is `Initialized`.

> **Note:** If the node returns any value other than the one listed here, the state comment is momentary and transient. Check the status variable again for an update.

In the event that each status variable returns the desired values, the node is in working order. This means that it is receiving write-sets from the cluster and replicating them to tables in the local database.

CHECKING THE REPLICATION HEALTH

Monitoring cluster integrity and node status can show you issues that may prevent or otherwise block replication. These status variables will help in identifying performance issues and identifying problem areas so that you can get the most from your cluster.

> **Note:** Unlike other the status variables, these are differential and reset on every `FLUSH STATUS` command.

Galera Cluster triggers a feedback mechanism called Flow Control to manage the replication process. When the local received queue of write-sets exceeds a certain threshold, the node engages Flow Control to pause replication while it catches up.

You can monitor the local received queue and Flow Control using the following status variables:

- wsrep_local_recv_queue_avg shows the average size of the local received queue since the last status query.

  ```
  SHOW STATUS LIKE 'wsrep_local_recv_queue_avg';

  +--------------------------+----------+
  | Variable_name            | Value    |
  +--------------------------+----------+
  | wsrep_local_recv_que_avg | 3.348452 |
  +--------------------------+----------+
  ```

  When the node returns a value higher than `0.0` it means that the node cannot apply write-sets as fast as it receives them, which can lead to replication throttling.

  > **Note:** In addition to this status variable, you can also use wsrep_local_recv_queue_max and wsrep_local_recv_queue_min to see the maximum and minimum sizes the node recorded for the local received queue.

- wsrep_flow_control_paused shows the fraction of the time, since `FLUSH STATUS` was last called, that the node paused due to Flow Control.

  ```
  SHOW STATUS LIKE 'wsrep_flow_control_paused';

  +---------------------------+----------+
  | Variable_name             | Value    |
  +---------------------------+----------+
  | wsrep_flow_control_paused | 0.184353 |
  +---------------------------+----------+
  ```

  When the node returns a value of `0.0`, it indicates that the node did not pause due to Flow Control during this period. When the node returns a value of `1.0`, it indicates that the node spent the entire period paused. If the time between `FLUSH STATUS` and `SHOW STATUS` was one minute and the node returned `0.25`, it indicates that the node was paused for a total 15 seconds over that time period.

  Ideally, the return value should stay as close to `0.0` as possible, since this means the node is not falling behind the cluster. In the event that you find that the node is pausing frequently, you can adjust the wsrep_slave_threads parameter or you can exclude the node from the cluster.

- wsrep_cert_deps_distance shows the average distance between the lowest and highest sequence number, or seqno, values that the node can possibly apply in parallel.

  ```
  SHOW STATUS LIKE 'wsrep_cert_deps_distance';

  +--------------------------+---------+
  | Variable_name            | Value   |
  +--------------------------+---------+
  | wsrep_cert_deps_distance | 23.8889 |
  +--------------------------+---------+
  ```

  This represents the node's potential degree for parallelization. In other words, the optimal value you can use with the wsrep_slave_threads parameter, given that there is no reason to assign more slave threads than transactions you can apply in parallel.

DETECTING SLOW NETWORK ISSUES

While checking the status of Flow Control and the received queue can tell you how the database server copes with incoming write-sets, you can check the send queue to monitor for outgoing connectivity issues.

Note: Unlike other the status variables, these are differential and reset on every `FLUSH STATUS` command.

wsrep_local_send_queue_avg show an average for the send queue length since the last `FLUSH STATUS` query.

```
SHOW STATUS LIKE 'wsrep_local_send_queue_avg';

+----------------------------+----------+
| Variable_name              | Value    |
+----------------------------+----------+
| wsrep_local_send_queue_avg | 0.145000 |
+----------------------------+----------+
```

Values much greater than `0.0` indicate replication throttling or network throughput issues, such as a bottleneck on the network link. The problem can occur at any layer from the physical components of your server to the configuration of the operating system.

Note: In addition to this status variable, you can also use wsrep_local_send_queue_max and wsrep_local_send_queue_min to see the maximum and minimum sizes the node recorded for the local send queue.