Tungsten Cluster Master Class

Intermediate: Connector (Proxy) Deep Dive

Chris Parker, Customer Success Director, EMEA & APAC

continuent

The MySQL Availability Company

Topics

In this short course, we will

- Explore Tungsten Connector (Proxy)
  - Bridge vs Proxy Mode
  - Types of Routing

Tungsten Clustering

- Core Components
  - Connector
  - Manager
  - Replicator

- Intelligent Proxy
- Routing Modes
  - Bridge Mode (Default)
  - Proxy Mode
- Routing Methods
  - Port Based
  - Host Based
  - SQL Based
  - Direct Read
  - SmartScale
- Cluster-Aware
- Highly-configurable to suit multiple user requirements

https://www.continuent.com/blog/mastering-tungsten-clustering-experience-the-power-of-the-tungsten-connector-intelligent-mysql-proxy/
https://www.continuent.com/blog/how-can-i-tell-which-tungsten-connector-mode-i-am-using-bridge-proxy-direct-or-proxy-smartscale/
https://www.continuent.com/blog/configuring-the-tungsten-connector-for-pci-compliance/

Connector Utilities

- `tpm connector`
  - Establishes a connection via the local Connector.
- If **not** in Bridge mode, tungsten help will list tungsten specific commands that can be executed from the MySQL CLI when connected through connector
- Additional options available through the `connector` command, type `connector help` to see options

```
Shell> connector help
(...)
  graceful-stop    Graceful stop of the application.
  reconfigure      Reconfigure the Connector by re-reading configuration.
  cluster-status   Display current view of the cluster configuration.
  client-list      List current connections and their endpoints
```

QoS

- RW_STRICT
  - This setting indicates that the query is a write and must be directed to a Primary
- RO_RELAXED
  - This setting enables the connector to redirect the query as if it were read-only, and therefore prefer a replica over a primary, and will choose a primary if no replica is available

Load Balancing

- `DefaultLoadBalancer`
  - Always selects the primary data source
- `MostAdvancedSlaveLoadBalancer`
  - Selects the replica that has replicated the most events, by comparing data sources "high water" marks.
- `LowestLatencySlaveLoadBalancer` (Default)
  - Selects the replica data source that has the lowest replication lag, or *appliedLatency*
- `RoundRobinSlaveLoadBalancer`
  - Selects a replica in a round robin manner, by iterating through them using internal index.
- `HighWaterSlaveLoadBalancer`
  - Default, and mandatory, for SmartScale routing.
  - Given a session high water (usually the high water mark of the update event), selects the first replica that has higher or equal high water mark, or the primary if no replica is online or has replicated the given session event.

Bridge Mode

- Bridge mode is the default routing method if no other method is configured during install
- Eliminates the need for the `user.map` file for user authentication
- Acts as a router at the network layer only, connecting sessions directly between the client and the MySQL database
- A simpler method for connectivity, but with limitations:
  - No Automatic Read/Write Splitting
  - No Seamless Reconnects
- Very Fast – No SQL Parsing
- Low Memory Requirements Per Session

Enabling Bridge Mode

- Enabled by default, or controlled by the following parameter:

```
connector-bridge-mode=true|false
```

Proxy Mode

- Simple Proxy
  - Default when Bridge mode = false + No other routing options are configured
- No automatic R/W Splitting
- R/W Splitting can be achieved by
  - Port Based Routing
  - Host Based Routing
  - SQL Based Routing
  - Smartscale
  - Direct Reads
- Seamless reconnects
- Requires handling of the `user.map` file for User Authentication
- **Warning**: Large Packets = Lots of Memory!

Mixed Mode

- Known as “Fall Back Bridge Mode” you can configure the connector to revert to Bridge mode if the Connector is unable to Authenticate using Proxy Mode
- When enabling, specify `RW_STRICT` or `RO_RELAXED` QoS
- If the Connector cannot successfully authenticate a user in the `user.map` file, a connection will be established based on the QoS setting
- Enabled, in Proxy Mode, with the following property:

```
property=fallBackBridgeMode=RW_STRICT|RO_RELAXED
```

Port-Based Routing

Manual routing of reads vs writes based on port number the client connects TO - i.e. all queries to port are sent to a replica

- Specify a read only port in the configuration:
  - e.g. `application-readonly-port=3307`
- Calls to a Connector via `application-port` will always resolve to the primary
- Calls to a Connector via the `application-readonly-port` will select an available replica, or the primary if no replica is available

Bridge Mode	
Proxy Mode	

Enabling Port Based Routing

- Enabled by the use of the following properties:

```
application-readonly-port=<port>
```

- To enable in combination with Proxy Mode, you will also need to disable Bridge mode using the following parameter, and configure the `user.map` file for Authentication:

```
connector-bridge-mode=false
```

Host Based Routing

Manual routing of reads vs writes based on IP address/hostname the client connects TO - i.e. all queries to specific IP are sent to a replica.

- Two different hostnames and associated IP addresses need to be created on each Connector host.
- Clients connecting to one hostname will be routed to the current primary for writing, and connections to the other hostname will be redirected to an available replica using the current load-balancing algorithm.
- Requires edits to the `user.map` file to define QoS per host:

```
@hostoption master.host1 qos=RW_STRICT
@hostoption slave.host1 qos=RO_RELAXED
```

Bridge Mode	
Proxy Mode	

Enabling Host Based Routing

- Configure network to enable multiple hostnames for each host
- Update `user.map` with valid user credentials and routing based on QoS per host.

- To enable in combination with Proxy Mode, you will also need to
  - Update `user.map` with valid user credentials and routing based on QoS per host
  - Disable Bridge mode using the following parameter

    ```
    connector-bridge-mode=false
    ```

SQL Based Routing

Manual routing of reads vs writes based on inline SQL comments - routing to primary or replica is controlled by the comment contents.

- With SQL-based routing, the redirection of queries and operations through the Connector is controlled by hints about the desired QoS which are provided in the comments of individual statements.
- Unless otherwise specified, statements will go to the current primary to be executed, or whatever replica is selected by the read-write splitting configuration, if enabled.
- To specify that a statement can be executed on the replica, place a comment before the statement:

```
/* TUNGSTEN USE qos=RO_RELAXED */ SELECT * FROM TABLENAME
```

- This style of comment indicates to the connector that the **specific query** that follows should go to a replica. If unavailable, the query may still be executed on the primary.

```
-- TUNGSTEN USE qos=RO_RELAXED
```

- This style of comment indicates to the connector that **all queries that follow** should go to a replica. If a replica is unavailable, any query may still be executed on the primary.

Bridge Mode	
Proxy Mode	

SQL Based Routing

- If you force the Connector to send traffic to a replica using `qos=RO_RELAXED`, then any write operations that follows will also go to the replica until you tell the Connector to go back to the primary by indicating `qos=RW_STRICT`.
- The application is fully responsible for where the traffic is routed to.
- If care is not taken, the application could send writes to a replica this way which is unacceptable from a clustering perspective.
  - If a non-authoritative user is used, the write will fail as the replica will be in read-only mode. However, if the user is a power user, MySQL will allow the write regardless of read-only setting!!
- **All writes must go to the primary or they will be lost to a non-authoritative node, and may corrupt the data badly.**

Enabling SQL Based Routing

- Enabled by use of the following properties

```
connector-bridge-mode=false
property=selective.rwsplitting=true
```

- Update `user.map` with valid user credentials.

Direct Read

Automatic R/W splitting routing of reads vs writes on a per-user basis.

- “Blind” Read-Write splitting
- SELECT… & SHOW… will route to any replica
- INSERT, UPDATE and SELECT FOR UPDATE… will always route to a primary
- Autocommit Only
  - SET {SESSION|GLOBAL} autocommit=1;
- Routing is enabled on a per-user basis, controlled by `user.map` entries
- **RISKS**
  - Potentially reads outdated data
  - SELECT function_that_writes(); would go to a replica and could cause data differences

Bridge Mode	
Proxy Mode	

Enabling Direct Read

- Enabled by the use of the following properties:

```
connector-bridge-mode=false
```

- Update `user.map` with valid user credentials and add an `@direct <user>` directive for each user that you wish to route via the direct method.

Smartscale

Automatic R/W splitting with intelligent session awareness of stale replica data.

- Automatically selects Primary or Replica based on the type of operation
  - Writes ALWAYS go to primary.
  - Reads routed to first replica that has received the last write, within the current session
  - WARNING: **ALL** transactions go to a primary, so any tool that wraps all queries in a transaction will not benefit from Smartscale, i.e. Hibernate
  - WARNING: **ALL** ephemeral objects will only exist on the primary
- Session ID can be set via different option-specific keyword strings:
  - DATABASE
  - USER
  - CONNECTION
  - PROVIDED_IN_DB_NAME (Schema = “mydb?sessionid=1234”)
  - The above are not placeholders. Use the exact string as shown.

Bridge Mode	
Proxy Mode	

Smartscale Session ID

- **DATABASE** - Applications will see write operations made to the same database as it is connected to. Reads from other databases might be outdated depending on the replica latency.
- **USER** - All connections that use the same user will read data consistent with the writes made by the current user. Other users data might be outdated.
- **CONNECTION** - Only writes made by the current connection are guaranteed to be read consistently. Writes from other connections might be outdated.
- **PROVIDED_IN_DB_NAME** - Allows you to specify a variable sessionid in the database connection string . An application, typically PHP, can pass its own session id to make smart scale even more efficient.

Enabling Smartscale

- Enabled by use of the following properties:

```
connector-bridge-mode=false
connector-smartscale=true
connector-smartscale-sessionid=DATABASE|USER|CONNECTION|PROVIDED_IN_DBNAME
```

- Update `user.map` with valid user credentials.
- All users that are required to connect through SmartScale need an additional MySQL GRANT as follows:

```
GRANT REPLICATION CLIENT ON *.* TO <user>@<host>;
```

Connector Summary

- Many, many more properties are available that can be tuned
- Highly configurable and powerful
- Select your choice of routing carefully and test!
  - Some methods will require application changes
  - Some methods may make incorrect assumptions if you misconfigure the routing
  - Check on the use of `BEGIN..END` transactions to determine how the Connector would handle
- Extensive online documentation
  - https://docs.continuent.com/tungsten-clustering-6.1/connector.html

Summary

What we have learnt today

- Discussed Tungsten Connector
  - Reviewed different modes (Bridge vs Proxy)
  - Explored routing methods

Next Steps

In the next session we will

- Explore the use of SSL within a cluster
  - How to configure it
  - Considerations
  - Step by Step implementation

THANK YOU FOR LISTENING

continuent.com

Chris Parker, Customer Success Director, EMEA & APAC