MaxScale

The Pluggable Router

Massimiliano Pinto
Vilho Raatikka

© SkySQL Ab 2014
MaxScale Objectives

- Highly scalable
- Transparent to the application
- Highly available
- Extendible
- Flexible
- Modules specify what MaxScale is: router / firewall / data extract tool / …
- Open source, GPL2 software since 8th of Jan. 2014
How? In a nutshell

- Client authenticates once – with MaxScale
- Allows multiple protocols for clients and servers
- Alter, block or log client requests and results
- Configuration and state aware monitors
- Content-aware or unaware routers
- Core of MaxScale is multi-threaded, event-driven network IO processor
Typical Scenarios

- MySQL Replication with
  - Read connection load balancing
  - Read/Write statement splitting

- Galera Cluster
  - Connection load balancing and conflict avoidance
MySQL Replication with R/W split

For applications that have been designed to work with a single server and require read scalability.

MaxScale monitors the state of each node and selects only available nodes.

MaxScale creates 2 connections, one for R/W on the Master node and one R/O load balanced on the Slave nodes.

Each application uses only 1 connection.
Galera Cluster using Connection Load Balancing

For applications that can use the “all-master” capabilities of Galera

MaxScale monitors the state of each Galera Cluster node and selects only synced nodes

Each application uses only 1 connection

MaxScale load balances the client connections and whenever possible writes to one node avoiding conflicts

© SkySQL Ab 2014
Inside MaxScale
MySQL Local Authentication

Users are loaded at start
"read user info"

Backends

connect/auth

MaxScale Core
Authentication
Protocol
Router
Filter & logging
Monitor
Inside MaxScale
MySQL Local Authentication

Users are loaded at start

“read user info”

Backends
Inside MaxScale
Basic Read Load Balancing

```
select * from T1
```

select a

select a candidate server

© SkySQL Ab 2014
Inside MaxScale
Read/Write Splitting

“call my_proc(1)”

Call parser and examine parse tree
Extending with filters

- May be connected to form arbitrary chains
- Allows inspection, modification and rejection of requests and results
  - blacklist
  - data mining / log
Extending with filters
Extending with filters

Protocol → Filter → Filter → Router → Protocol

Routing Hint

© SkySQL Ab 2014
Extending with filters

```
| Protocol | Filter | Filter | Router | Protocol |
```

```
| INSERT INTO T1 (....) |
```

© SkySQL Ab 2014
Extending with filters

![Diagram showing the extension of filters in a network with protocol, filter, router, and monitor components.]

- Protocol
- Filter
- Filter
- Router
- Protocol

Log Entry

INSERT INTO T1 (...)

© SkySQL Ab 2014
Extending with filters

Protocol → Filter → Filter → Router → Protocol

INSERT INTO T1 (...)

Log Entry

Router → Protocol

© SkySQL Ab 2014
Extending with filters

Protocol → Filter → Filter → Router → Protocol

INSERT INTO T1 (...)

Routing Hint

Log Entry
Extending with filters

Protocol → Filter → Filter → Router → Protocol

INSERT INTO T1 (....)

Log Entry

INSERT INTO DWH.T1 (....)

Routing Hint
Extending with filters

Protocol → Filter → Filter → Router → Protocol

Log Entry

Router → Protocol

INSERT INTO T1 (...)

Routing Hint

INSERT INTO DWH.T1 (...)

© SkySQL Ab 2014
Extending with filters

Protocol → Filter → Filter → Router → Protocol

Log Entry

INSERT INTO T1 (...)

INSERT INTO DWH.T1 (...)

© SkySQL Ab 2014
Get involved

- Check on GitHub: https://github.com/skysql/MaxScale
- MariaDB Source: https://downloads.mariadb.org/
- Google groups: https://groups.google.com/forum/#!forum/maxscale
- Binary Tarball: http://downloads.skysql.com/files/SkySQL/MaxScale
- SkySQL website: http://www.skysql.com

Thank you!