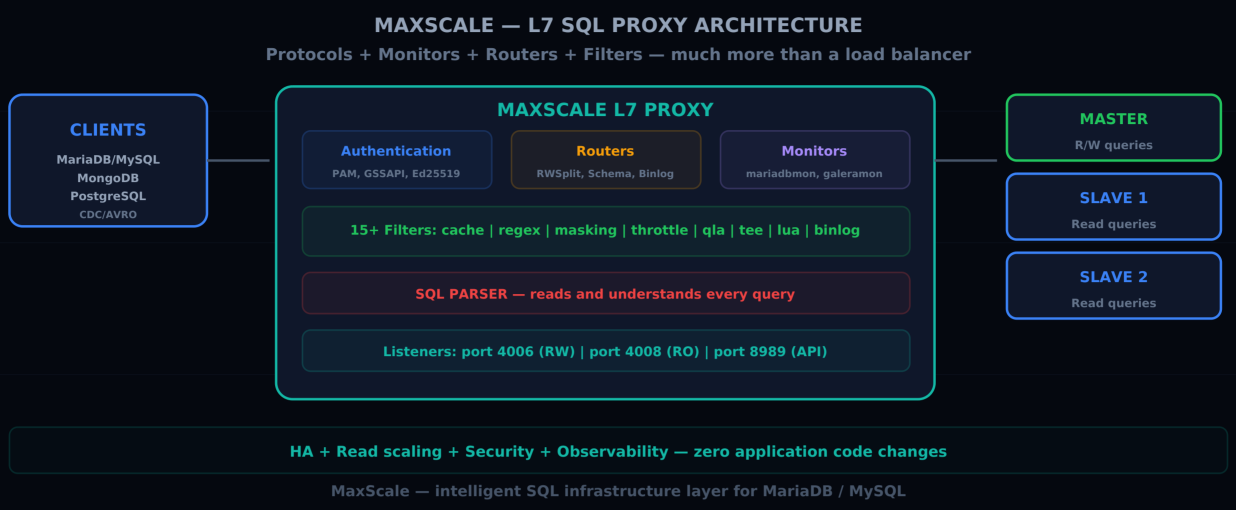
MaxScale：远不止是一个反向 SQL 代理（第一部分）

Sylvain ARBAUDIE · 2025年7月31日

MAXSCALE MARIADB PROXY ARCHITECTURE



不是一个简单的代理

当人们谈到 MaxScale 时，本能反应是将它与 HAProxy 或经典[illegible]进行比较。[illegible]MaxScale 能[illegible]4 层（TCP）代理。它是一个

这个区别类似于一个按地址投递的邮递员（L4）与一个阅读你邮件、按优先级排序、过滤垃圾邮件并只向你展示相关内容的助手（L7）之间的差异。MaxScale 能读懂 SQL。

认证：不仅仅是密码

MaxScale 支持全面的认证机制：

- **MariaDBAuth**：原生 MariaDB / MySQL 认证，带本地凭证缓存
- **PAM**：通过可插拔认证模块（PAM）与企业目录（LDAP、Active Directory）集成
- **GSSAPI/Kerberos**：用于 Windows / Active Directory 环境的 SSO 认证
- **Ed25519**：MariaDB 的公钥认证，比经典 SHA 哈希更安全

认证在监听器级别处理。因此，你可以在不同端口上使用不同的认证方式：PAM 用于内部应用程序，Ed25519 用于管理连接，MariaDBAuth 用于遗留兼容性。

协议：不仅仅是 SQL

MaxScale 是多语言的。它支持多种输入和输出协议：

MariaDBClient / MariaDBBackend

原生 MariaDB / MySQL 协议。这是主要的使用场景：应用程序连接到 MaxScale，就像连接到标准的数据库服务器一样。

CDC / AVRO

变更数据捕获协议以 AVRO 格式流式传输 binlog 变更。这是构建实时数据管道、馈送数据湖或同步外部系统的理想工具。

NoSQL / MongoDB

MaxScale 可以暴露一个兼容 MongoDB 的接口。使用 MongoDB 协议的应用程序可以通过 MaxScale 与 MariaDB 数据库交互。这是一个小众但强大的迁移功能。

PostgreSQL（实验性）

输入端的 PostgreSQL 协议支持正在开发中。目标是允许 PostgreSQL 应用程序连接到 MariaDB 后端。

监控器：拓扑智能

监控器是 MaxScale 的眼睛。它们定期查询后端服务器，以自动检测集群的拓扑和健康状态。

mariadbmon

用于 MariaDB / MySQL 复制拓扑的主要监控器。它检测主节点、从节点、中继节点和复制状态。它处理自动故障转移：如果主节点宕机，一个从节点被提升，其他从节点被重新配置为从新主节点复制。

galeramon

Galera 集群的专用监控器。它检测集群状态（Primary、Non-Primary、Disconnected）、节点数量、状态 UUID，并相应地处理路由。

路由器：路由智能

路由器是 MaxScale 的大脑。它们根据查询类型和检测到的拓扑来决定将每个查询发送到哪里。

readwritesplit

最[illegible]器。[illegible]PDATE [illegible]DELETE [illegible]CREATE [illegible]ELECT[illegible]节点。分[illegible]节点。事务完整地发送到主节点。

readconnroute

一个更简单的路由器，在可用服务器之间分配连接（而非查询）。适用于不需要查询级读/写分离的读密集型工作负载。

schemarouter

将查询路由到承载请求[illegible]的[illegible]。在[illegible]库分片 `asia` 在服务器 B 上。

binlogrouter

将 MaxScale 变成复制中继。MaxScale 作为主节点的从节点，而真正的从节点连接到 MaxScale 而非主节点。这减少了主节点的负载并集中了 binlog 分发。

kafkarouter (CDC)

将 binlog 事件发送到 Apache Kafka。每次数据库修改都作为 Kafka 消息发布，允许消费者实时响应。

过滤器：超过 15 个模块

过滤器在客户端和服务器之间拦截并转换 SQL 流。MaxScale 提供了超过 15 个：

- **qlafilter**：完整的查询日志记录（审计）
- **regexfilter**：通过正则表达式重写查询
- **cache**：带自动失效的查询缓存
- **throttlefilter**：按用户限制查询速率
- **masking**：动态掩码敏感数据（邮箱、卡号）
- **topfilter**：收集最慢查询
- **commentfilter**：注入 SQL 注释用于追踪
- **tee**：将流复制到第二个后端（影子测试）
- **namedserverfilter**：基于规则路由到特定服务器
- **hintfilter**：解释 SQL 提示以强制路由
- **luafilter**：执行 Lua 脚本实现自定义逻辑
- **binlogfilter**：按 schema 或表过滤 binlog 事件

部署：Docker 和 Kubernetes 就绪

MaxScale 提供官方 Docker 镜像：

```
docker run -d --name maxscale \
  -v /path/to/maxscale.cnf:/etc/maxscale.cnf \
  -p 4006:4006 -p 8989:8989 \
  mariadb/maxscale:latest
```

对于 Kubernetes，MaxScale 可以根据使用场景部署为 StatefulSet 或 Deployment。REST API 便于与健康检查和就绪探针集成。

业务价值

超越技术功能，MaxScale 提供具体的业务价值：

- 高可用性：秒级自动故障转移，对应用程序透明
- 读扩展性：无需修改应用程序代码即可添加从节点
- 安SQL 过滤、数据掩码、集中认证
- 可观测性：日志记录、指标、REST API 用于监控
- 更简便的迁移：多协议支持用于技术过渡

MaxScale 不是一个简单的反向代理。它是一个智能的基础设施层，位于应用程序和 MariaDB / MySQL 数据库之间，在不修改一行应用程序代码的情况下带来弹性、安全性和灵活性。

本[illegible]。