

Apache ShardingSphere 高可用功能详解 & 实操演练

赵锦超
2022.08.06

赵锦超

Apache ShardingSphere Committer

SphereEx 研发工程师

- 从事过电商、金融行业，热爱开源
- 目前专注于 Apache ShardingSphere 高可用 & 分布式治理的相关研发工作



目录

01

Apache ShardingSphere 高可用介绍

02

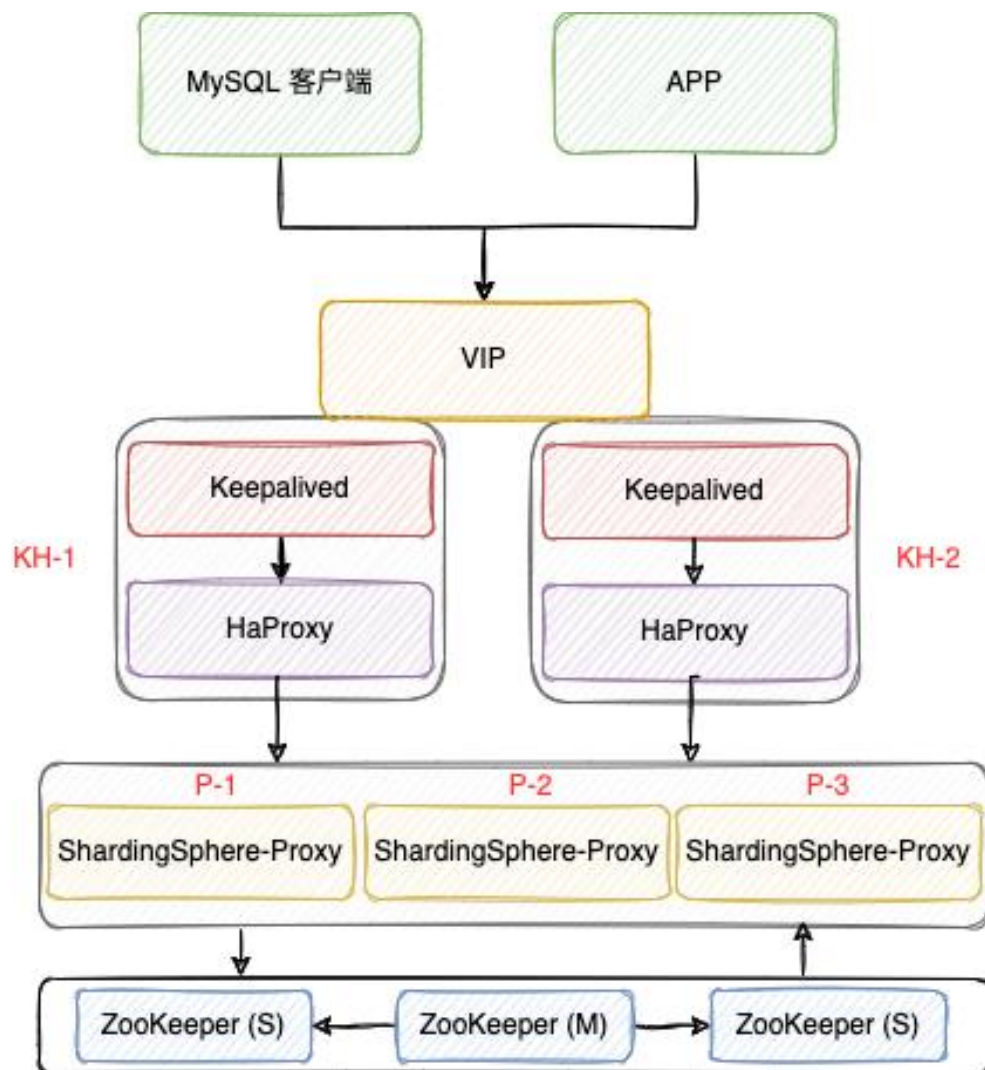
Apache ShardingSphere 高可用源码解析

03

Apache ShardingSphere HA & MySQL MGR 实战演练

Apache ShardingSphere 高可用介绍

ShardingSphere-Proxy 高可用



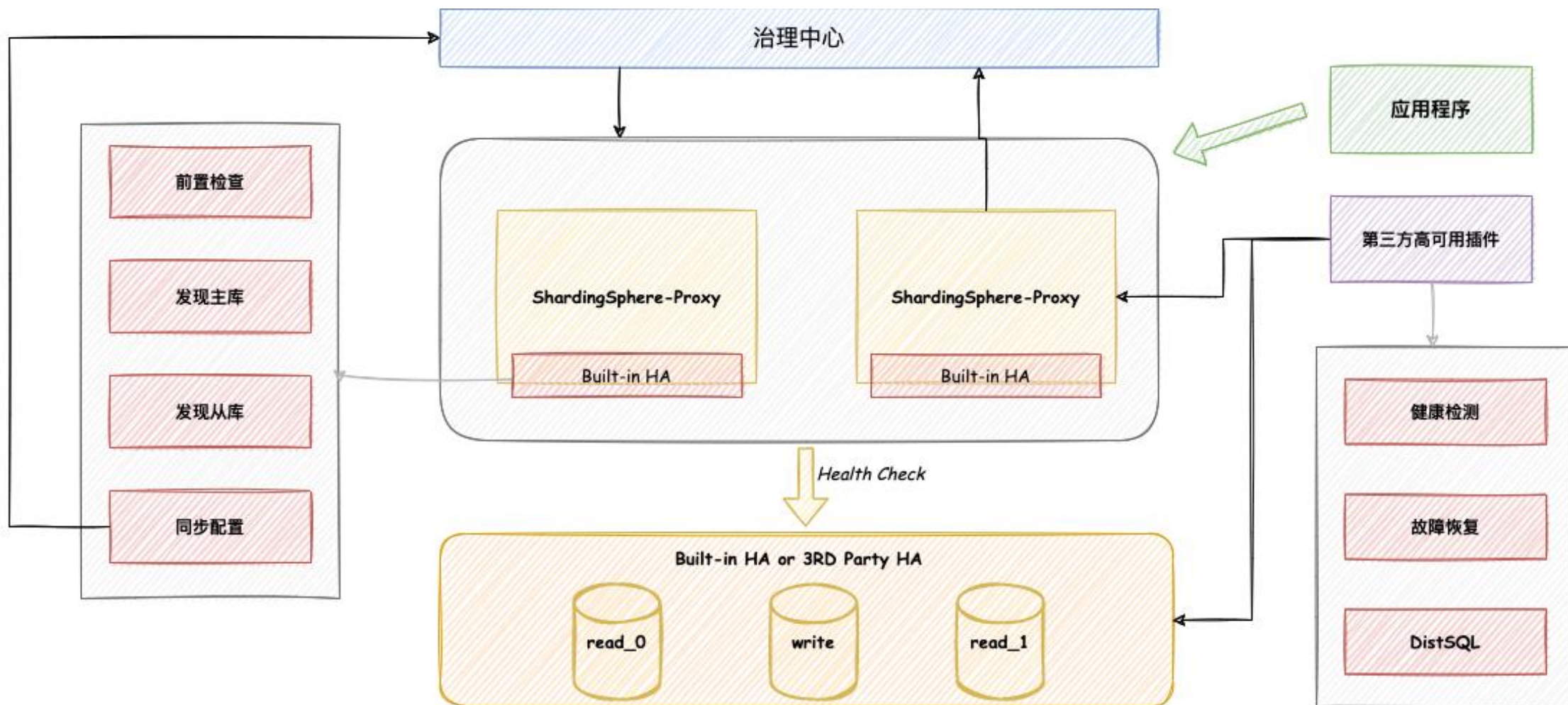
Apache ShardingSphere 高可用介绍

Apache ShardingSphere 本身不提供数据库高可用的能力，它通过第三方提供的高可用方案感知数据库主从关系的切换。Apache ShardingSphere 提供数据库发现的能力，自动感知数据库主从关系，并修正计算节点对数据库的连接。

目前支持的高可用方案：

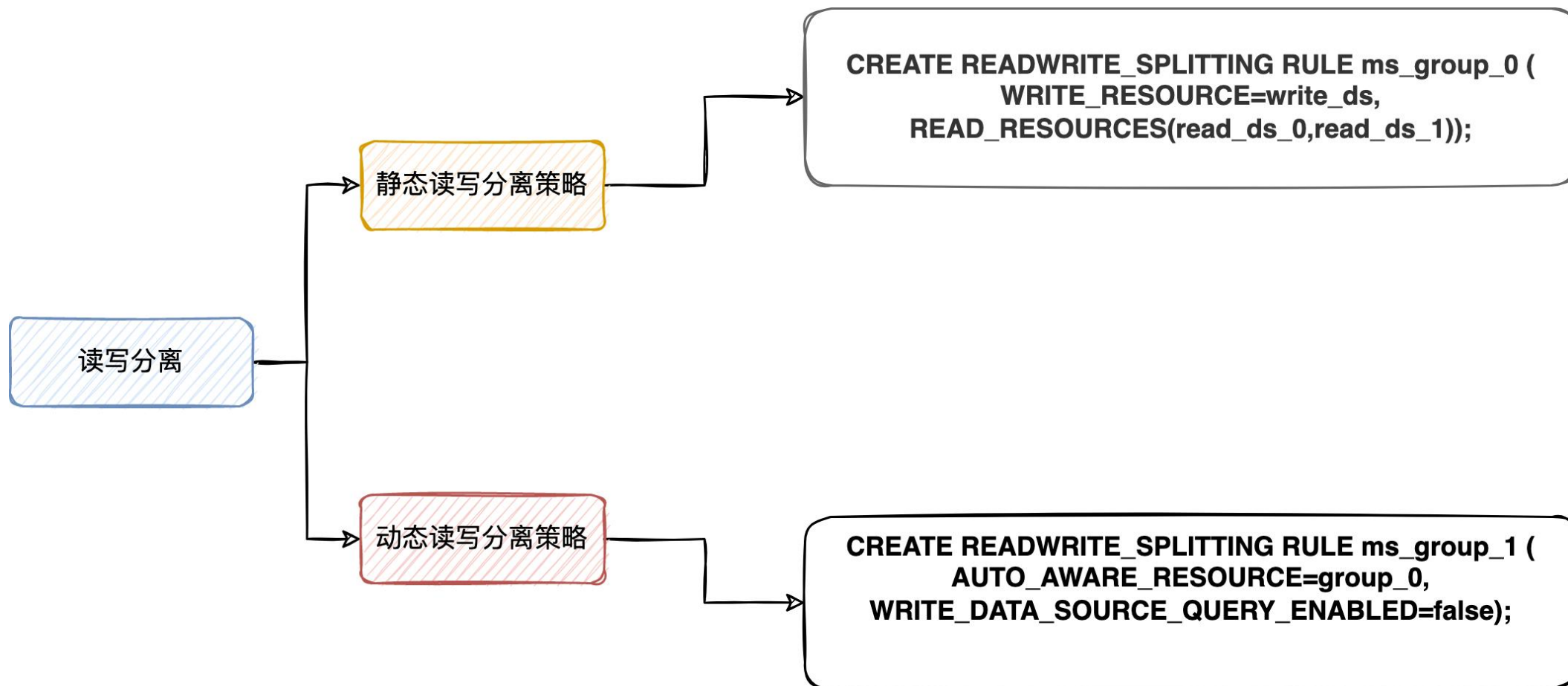
- MySQL MGR 单主模式
- MySQL 主从复制模式
- openGauss 主从复制模式。

Apache ShardingSphere 高可用介绍



Apache ShardingSphere 高可用介绍

ShardingSphere 读写分离





目录

01

Apache ShardingSphere 高可用介绍

02

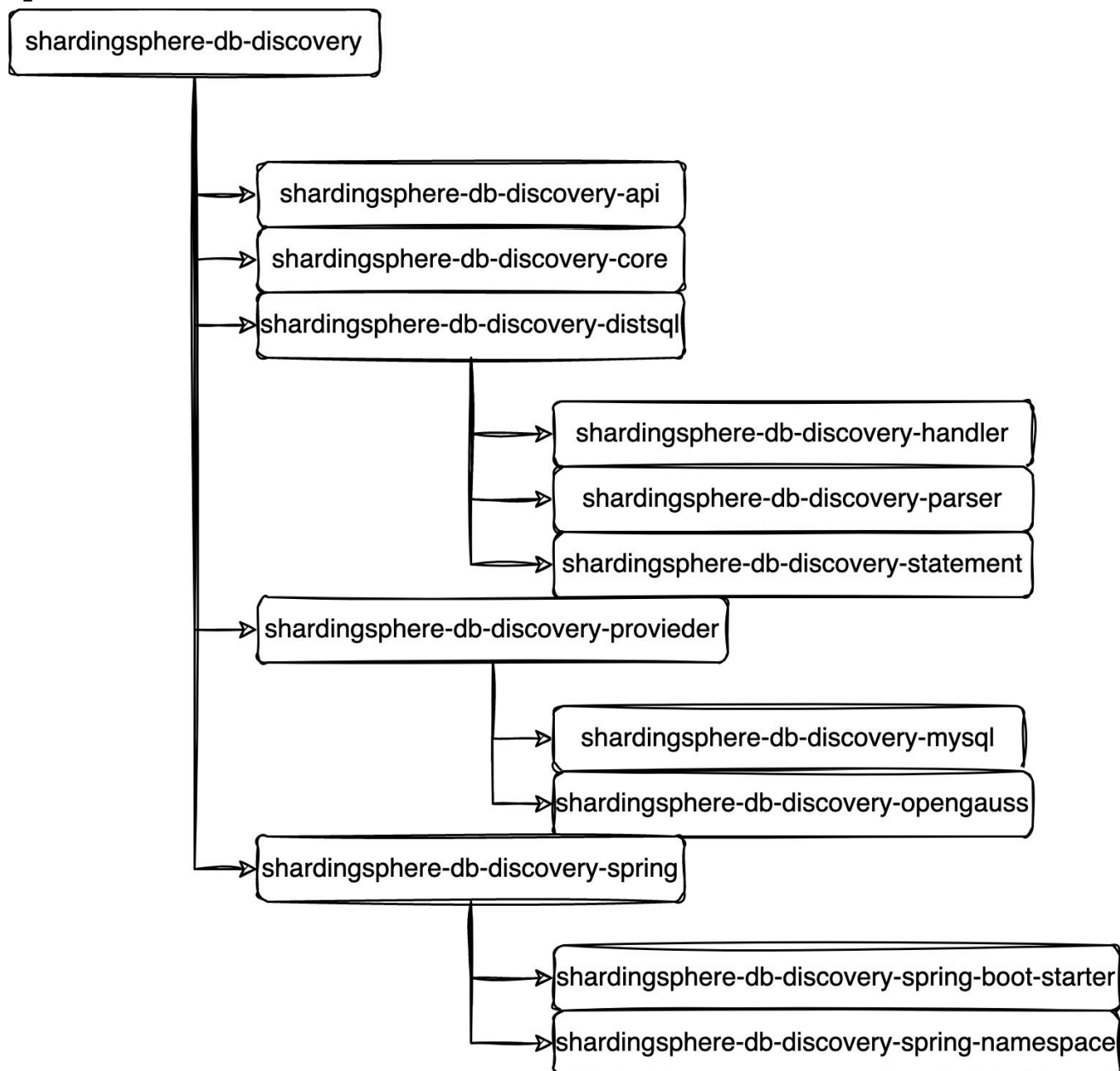
Apache ShardingSphere 高可用源码解析

03

Apache ShardingSphere HA & MySQL MGR 实战演练

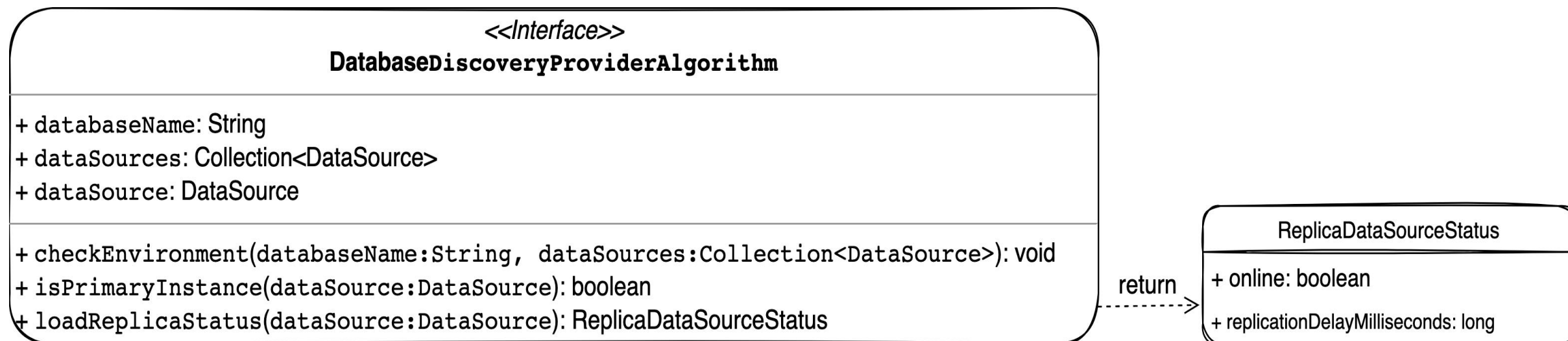
Apache ShardingSphere 高可用源码解析

数据库发现模块结构



Apache ShardingSphere 高可用源码解析

DatabaseDiscoveryProviderAlgorithm SPI 接口核心方法



Apache ShardingSphere 高可用源码解析

checkEnvironment(final String databaseName, final Collection<DataSource> dataSources) 方法

组复制插件

```
SELECT PLUGIN_STATUS FROM information_schema.PLUGINS WHERE PLUGIN_NAME='group_replication'
```

单主模式

```
SELECT VARIABLE_VALUE FROM performance_schema.global_variables WHERE VARIABLE_NAME='group_replication_single_primary_mode'
```

组名一致性

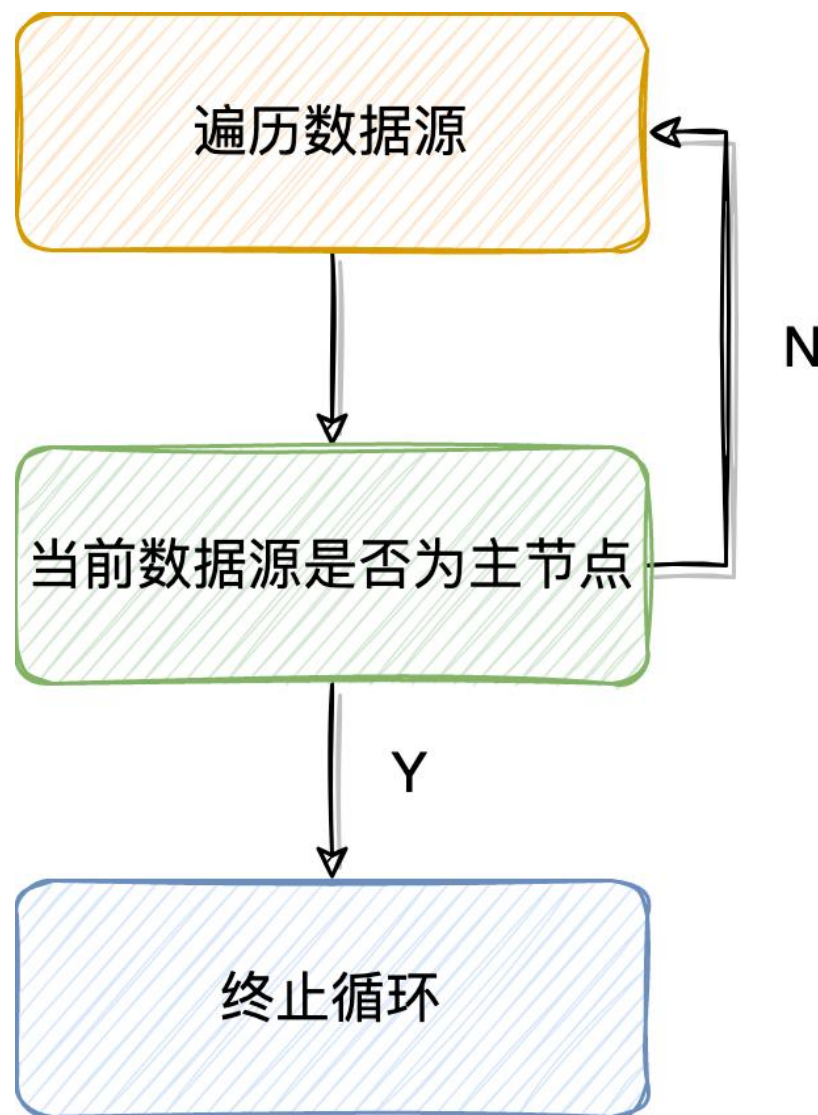
```
SELECT VARIABLE_VALUE FROM performance_schema.global_variables WHERE VARIABLE_NAME='group_replication_group_name'
```

组数据库 URL

```
SELECT MEMBER_HOST, MEMBER_PORT, MEMBER_STATE FROM performance_schema.replication_group_members
```

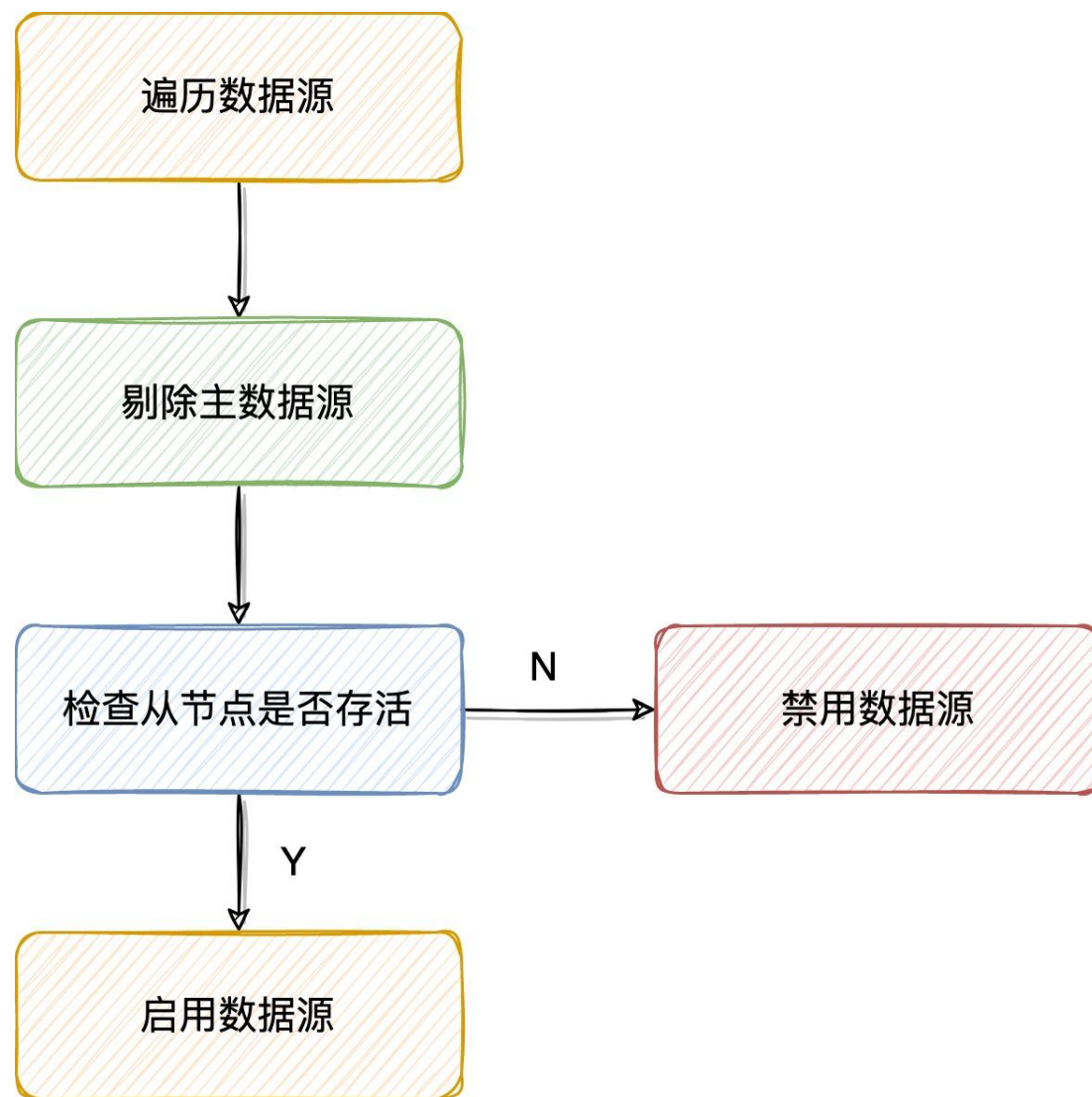
Apache ShardingSphere 高可用源码解析

isPrimaryInstance(final DataSource dataSource) 方法



Apache ShardingSphere 高可用源码解析

ReplicaDataSourceStatus loadReplicaStatus(final DataSource replicaDataSource) 方法





目录

01

Apache ShardingSphere 高可用介绍

02

Apache ShardingSphere 高可用源码解析

03

Apache ShardingSphere HA & MySQL MGR 实战演练

Apache ShardingSphere HA & MySQL MGR 实战演练

准备工作：

- ShardingSphere-Proxy 5.1.2 or master branch & cluster mode
- MySQL MGR 集群 8.x
- ZooKeeper 3.7.0

Apache ShardingSphere HA & MySQL MGR 实战演练

```
mysql> use db_ha;
Database changed
mysql> ADD RESOURCE ds_0 (
  ->   HOST=10.9.122.82,
  ->   PORT=33306,
  ->   DB=ha_test,
  ->   USER=root,
  ->   PASSWORD=sphereEx@2021
  -> );
Query OK, 0 rows affected (0.56 sec)

mysql>
mysql> ADD RESOURCE ds_1 (
  ->   HOST=10.9.122.82,
  ->   PORT=33307,
  ->   DB=ha_test,
  ->   USER=root,
  ->   PASSWORD=sphereEx@2021
  -> );
Query OK, 0 rows affected (0.15 sec)

mysql>
mysql> ADD RESOURCE ds_2 (
  ->   HOST=10.9.122.82,
  ->   PORT=33308,
  ->   DB=ha_test,
  ->   USER=root,
  ->   PASSWORD=sphereEx@2021
  -> );
Query OK, 0 rows affected (0.08 sec)

mysql> CREATE DB_DISCOVERY RULE replica_ds (
  ->   RESOURCES(ds_0, ds_1, ds_2),
  ->   TYPE(NAME='MySQL.MGR',PROPERTIES('group-name'='558edd3c-02ec-11ea-9bb3-080027e39bd2')),
  ->   HEARTBEAT(PROPERTIES('keep-alive-cron'='0/5 * * * * ?'))
  -> );
Query OK, 0 rows affected (0.39 sec)

mysql> CREATE READWRITE_SPLITTING RULE readwrite_ds (
  ->   AUTO_AWARE_RESOURCE=replica_ds
  -> );
Query OK, 0 rows affected (0.19 sec)
```

Apache ShardingSphere HA & MySQL MGR 实战演练

```
mysql> SHOW DB_DISCOVERY RULES;
```

group_name	data_source_names	primary_data_source_name	discovery_type	discovery_heartbeat
replica_ds	ds_0,ds_1,ds_2	ds_0	{name=replica_ds_MySQL.MGR, type=MySQL.MGR, props={group-name=558edd3c-02ec-11ea-9bb3-080027e39bd2}}	{name=replica_ds_heartbeat, props={keep-alive-cron=0/5 * * * * ?}}

1 row in set (0.00 sec)

```
mysql> SHOW READWRITE_SPLITTING RULES;
```

name	auto_aware_data_source_name	write_data_source_query_enabled	write_data_source_name	read_data_source_names	load_balancer_type	load_balancer_props
readwrite_ds	replica_ds	NULL	ds_0	ds_1,ds_2		

1 row in set (0.01 sec)

```
mysql> SHOW READWRITE_SPLITTING READ RESOURCES;
```

resource	status	delay_time(ms)
ds_1	enabled	0
ds_2	enabled	0

2 rows in set (0.01 sec)

欢迎关注我们!



技术干货



加入交流群

SphereEx 官网: <https://sphere-ex.com>

Apache ShardingSphere Website: <https://shardingsphere.apache.org>

Apache ShardingSphere GitHub: <https://github.com/apache/shardingsphere>

Apache ShardingSphere Slack Channel: <https://apacheshardingsphere.slack.com>

谢谢观看