

# 1 Sharding-JDBC 概览

## 1.1 认识ShardingSphere

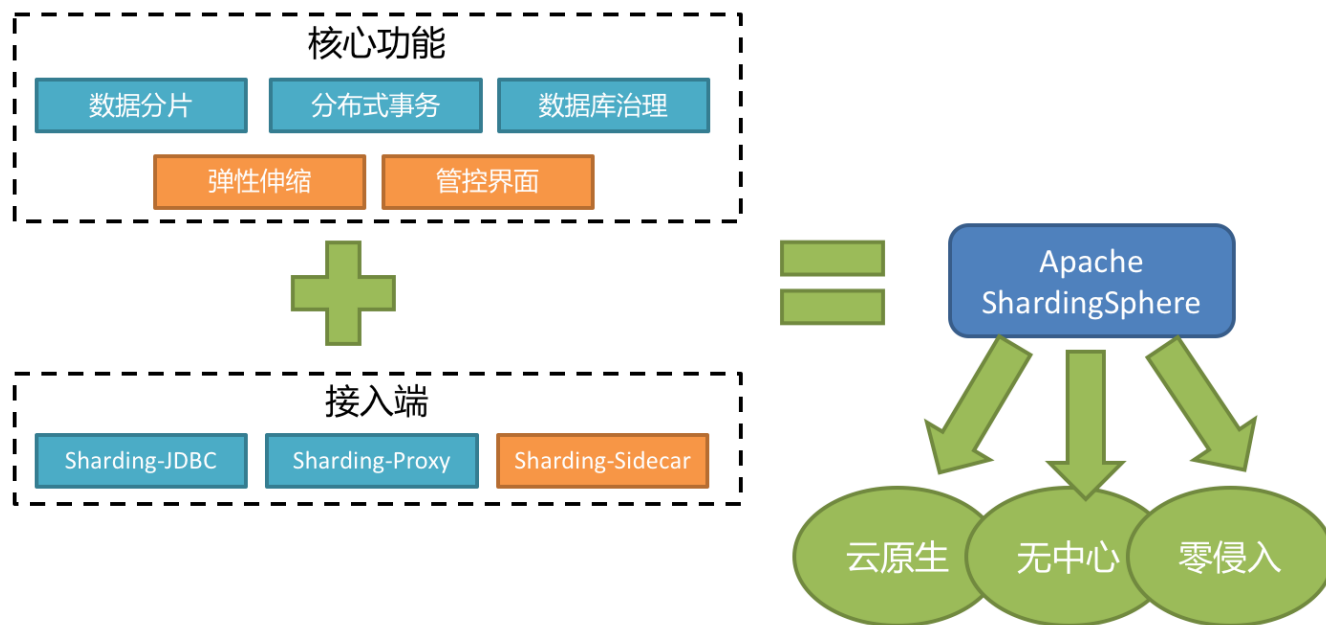


ShardingSphere是一套开源的分布式数据库中间件解决方案组成的生态圈，它由Sharding-JDBC、Sharding-Proxy和Sharding-Sidecar（计划中）这3款相互独立的产品组成。他们均提供标准化的数据分片、分布式事务和数据库治理功能，可适用于如Java同构、异构语言、容器、云原生等各种多样化的应用场景。

当前版本：3.0

官网地址：[https://shardingsphere.apache.org/index\\_zh.html](https://shardingsphere.apache.org/index_zh.html)

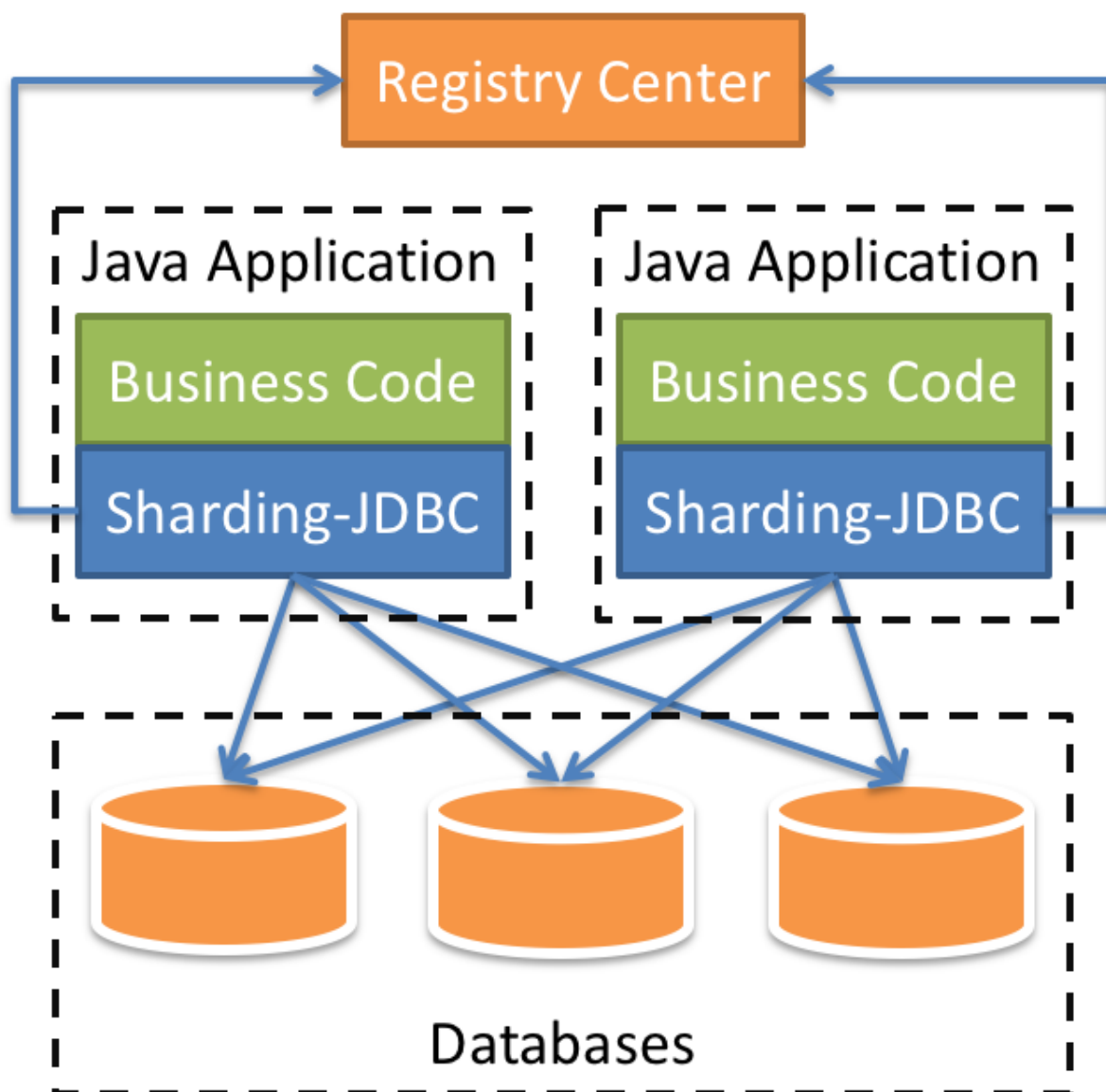
ShardingSphere构成：



## 1.2 认识Sharding-JDBC

定位为轻量级Java框架，在Java的JDBC层提供的额外服务。它使用客户端直连数据库，以jar包形式提供服务，无需额外部署和依赖，可理解为增强版的JDBC驱动，完全兼容JDBC和各种ORM框架。

- 适用于任何基于Java的ORM框架，如：JPA, Hibernate, Mybatis, Spring JDBC Template或直接使用JDBC。
- 基于任何第三方的数据库连接池，如：DBCP, C3P0, BoneCP, Druid, HikariCP等。
- 支持任意实现JDBC规范的数据库。目前支持MySQL，Oracle，SQLServer和PostgreSQL。

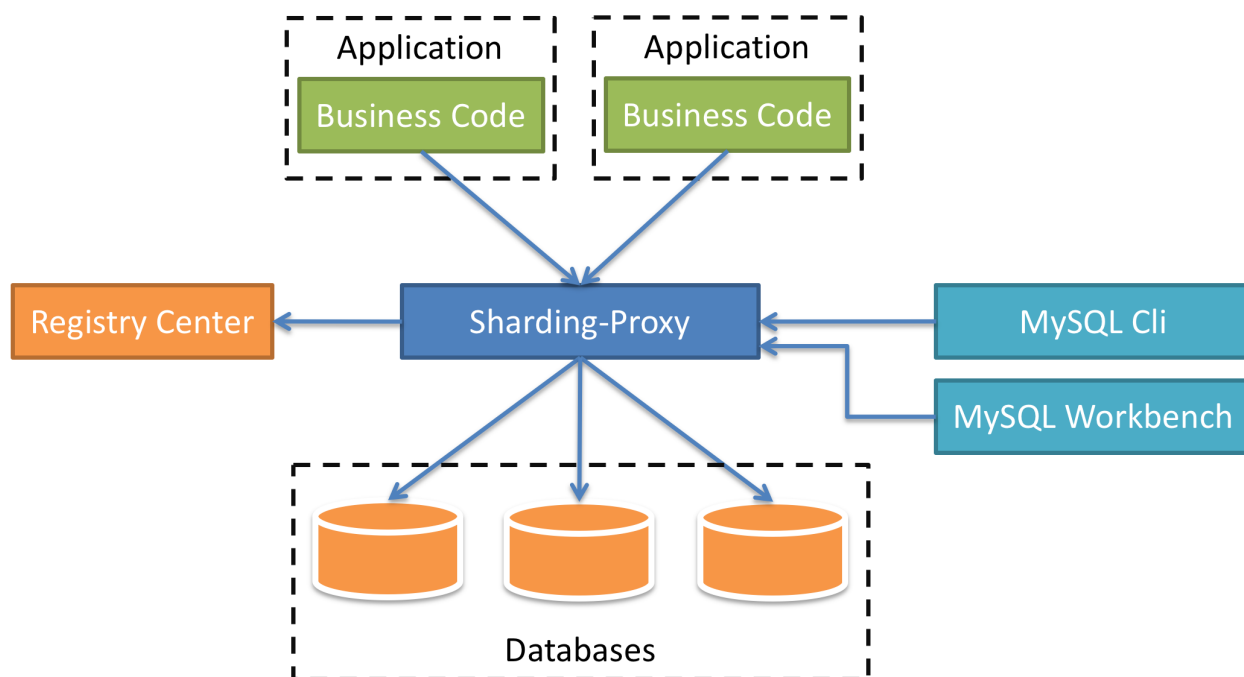


### 1.3 认识Sharding-Proxy

定位为透明化的数据库代理端，提供封装了数据库二进制协议的服务端版本，用于完成对异构语言的支持。目前先提供MySQL版本，它可以使用任何兼容MySQL协议的访问客户端(如：MySQL Command Client, MySQL Workbench等)操作数据，对DBA更加友好。

- 向应用程序完全透明，可直接当做MySQL使用。

- 适用于任何兼容MySQL协议的客户端。



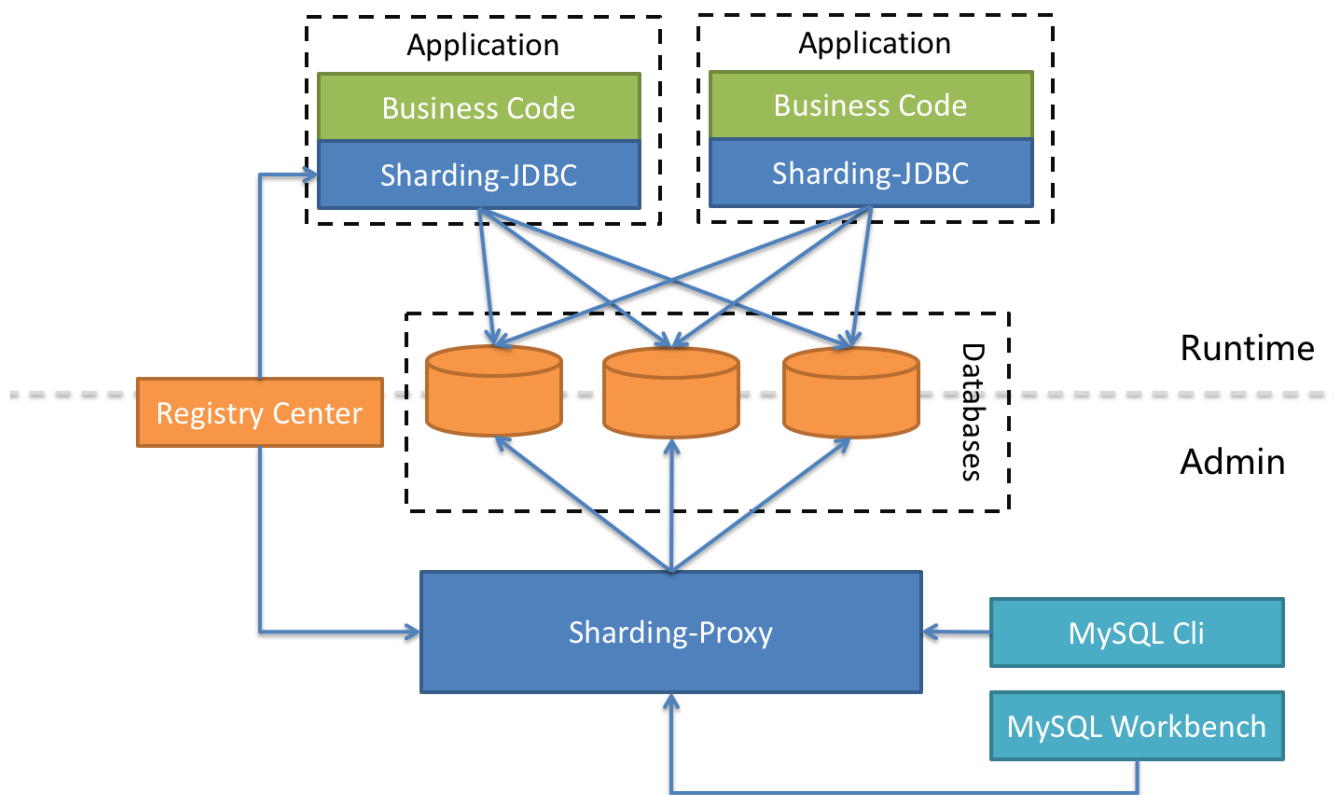
## 1.4 三个组件对比认识

	<i>Sharding-JDBC</i>	<i>Sharding-Proxy</i>	<i>Sharding-Sidecar</i>
数据库	任意	MySQL	MySQL
连接消耗数	高	低	高
异构语言	仅Java	任意	任意
性能	损耗低	损耗略高	损耗低
无中心化	是	否	是
静态入口	无	有	无

## 1.5 混合架构

Sharding-JDBC采用无中心化架构，适用于Java开发的高性能的轻量级OLTP应用；Sharding-Proxy提供静态入口以及异构语言的支持，适用于OLAP应用以及对分片数据库进行管理和运维的场景。

ShardingSphere是多接入端共同组成的生态圈。通过混合使用Sharding-JDBC和Sharding-Proxy，并采用同一注册中心统一配置分片策略，能够灵活的搭建适用于各种场景的应用系统，架构师可以更加自由的调整适合于当前业务的最佳系统架构。



## 1.6 功能列表

### 数据分片

- 分库 & 分表
- 读写分离
- 分布式主键

### 分布式事务(Doing)

- XA强一致事务
- 柔性事务

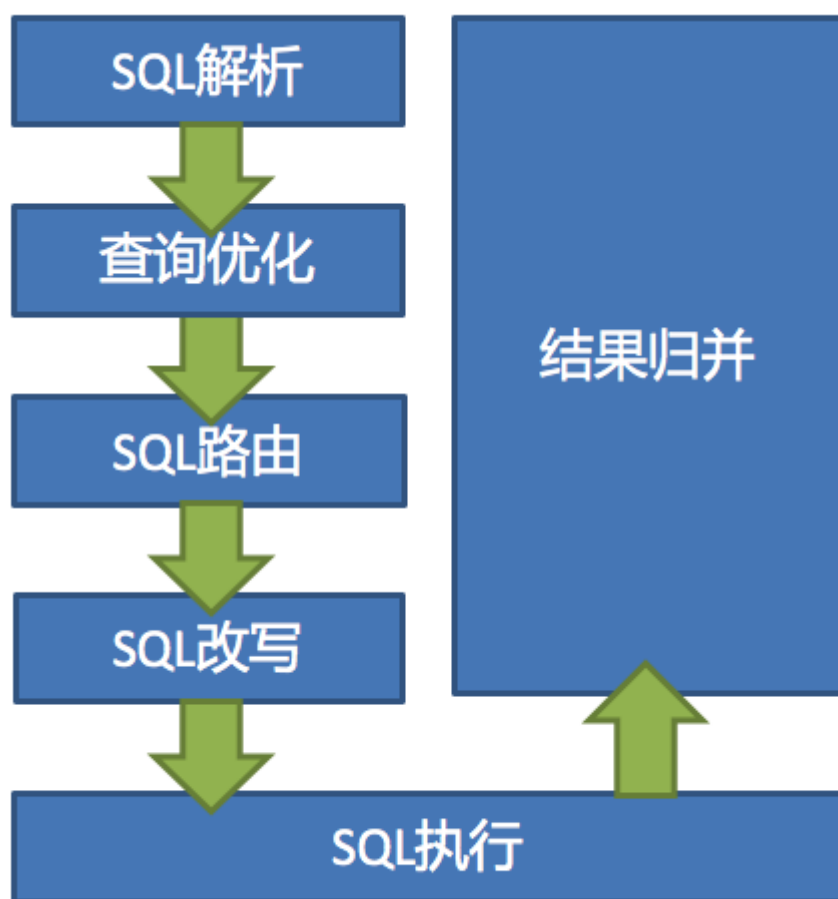
### 数据库治理

- 配置动态化
- 熔断 & 禁用

- 调用链路追踪
- 弹性伸缩 (Planning)

## 1.7 数据分片工作原理

ShardingSphere的3个产品的数据分片主要流程是完全一致的。核心由 SQL解析 => 执行器优化 => SQL路由 => SQL改写 => SQL执行 => 结果归并 的流程组成。



### SQL解析

分为词法解析和语法解析。先通过词法解析器将SQL拆分为一个个不可再分的单词。再使用语法解析器对SQL进行理解，并最终提炼出解析上下文。解析上下文包括表、选择项、排序项、分组项、聚合函数、分页信息、查询条件以及可能需要修改的占位符的标记。

### 执行器优化

合并和优化分片条件，如OR等。

## SQL路由

根据解析上下文匹配用户配置的分片策略，并生成路由路径。目前支持分片路由和广播路由。

## SQL改写

将SQL改写为在真实数据库中可以正确执行的语句。SQL改写分为正确性改写和优化改写。

## SQL执行

通过多线程执行器异步执行。

## 结果归并

将多个执行结果集归并以便于通过统一的JDBC接口输出。结果归并包括流式归并、内存归并和使用装饰者模式的追加归并这几种方式。

## 1.8 规划线路图

