

一、概述

1.1、目的

1.2、背景

1.3、参考资料

1.4、术语

二、整体设计

2.1、

三、功能描述

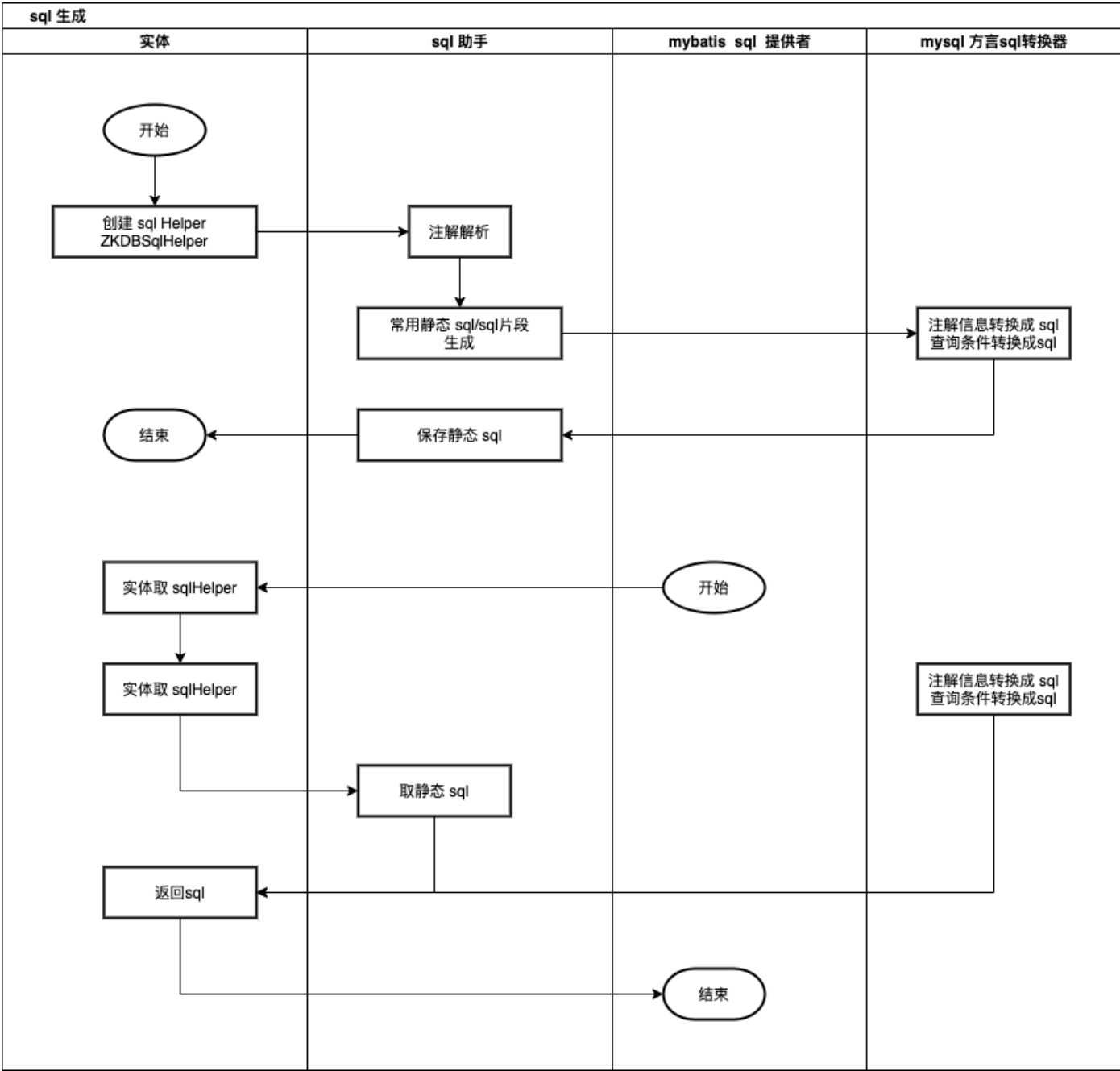
3.1 通用功能包

- 排序枚举类型
- 排序类
- json Java 数据类型及 ARJson 类型处理器

3.2 sql 生成

注解、注解提供器、查询条件、实体基础类、zk sql 提供者、sql 转换器、myBatis sql 提供者

- 整体设计思路如下图：



3.2.1 注解

- 类注解 @ZKTable
含：表名、表别名、排序字段数组
- 字段注解 @ZKColumn
- 字段修改注解 @ZKUpdate
字段注解的成员变量
- 字段查询注解 @ZKQuery
字段注解的成员变量

3.2.2 注解提供器

3.2.3 查询条件

- 查询逻辑操作符
com.zk.db.commons.ZKDBOptLogic
枚举类型，定义 and 和 or 两种查询逻辑；
- 查询比较操作符
com.zk.db.commons.ZKDBOptComparison
定义常见的查询类型：=、!=、>、>=、<、<=、in not、in like、not like、is null、is not null
- 查询条件接口
com.zk.db.commons.ZKDBQuery
- 查询字段条件
com.zk.db.commons.ZKDBQueryCol
com.zk.db.commons.ZKDBQueryColSql
- 查询 where 条件
com.zk.db.commons.ZKDBQueryWhere

3.2.3.1 myBatis 查询条件实现类

- com.zk.db.mybatis.commons.ZKDBQueryScript
字段列的查询条件，包含一个 ZKDBQuery 对象；会添加 mybatis 脚本包裹；如 <if> 的脚本逻辑判断

3.2.4 基础类

3.2.4.1 实体父类组成

com.zk.db.entity.ZKDBBaseEntity

- 包含一个 sql 助手；ZKDBSqlHelper 以接代实体的 sql；可以取查询相关的映射内容，如查询结果映射等
- sql 片段: 逻辑删除 sql 片段
- 取查询条件组对象 ZKDBQueryWhere 对象

3.2.4.2 dao 父接口

com.zk.db.mybatis.dao.ZKDBBaseDao

3.2.5 zk sqlHelper 助手

- 包含一个类注解映射信息对象 和 sql 转换对象；

3.2.5.1 取表信息

- 取表名

- 取表别名
- sql 片段: 取默认排序 sql 片段

3.2.5.2 保存的静态 sql

sqlHelper 助手初始化时：设置类注解映射信息对象；并将静态 sql 生成保存

- 插入 sql
- 修改 sql
- 根据主键查询 sql
- 逻辑删除 sql
- 物理删除 sql
- sql 片段: 查询字段映射；带表别名
- sql 片段: 取默认排序sql片段；带表别名
- sql 片段: 主键查询条件；带表别名
- sql 片段: 查询条件；带表别名
- sql 片段: 列表查询，只是不含排序部分；带表别名

3.2.5.3 动态生成 sql

通过 sql 转换器和实体类，实时转化

- sql 片段: 取查询结果映射；自定义别名
- sql 片段: 取查询条件；自定义别名
- sql 片段: 取主键查询条件；自定义别名
- sql 片段: 生成排序 sql；可自定义别名，也可指定是否需要默认排序片段
重置了三个方法，详见方法主解

3.2.5.4 其他sql生成过程数据对象

- 取查询条件组对象

3.2.6 sql 转换器

接口：com.zk.db.commons.ZKSqlConvert

3.2.6.1 mysql 下 myBatis 实现：

com.zk.db.mybatis.commons.ZKDBMybatisConvert

主要实现 根据注解 解析查询的条件，ZKDBQueryWhere 对象；

com.zk.db.mybatis.mysql.ZKMySQLSqlConvert

这里会实现 插入、修改、查询、等 sql 转换；是转换为具体 sql 的实现类；

3.2.7 myBatis sql 提供者

myBatis dao 层的注解 sql 提供者使用，定义相关 sql 的方法，通过转换、组合、取类的静态 sql 提供需要的完成 sql

3.3 动态数据源

3.3.1 spring

- 自定义数据源：com.zk.db.dynamic.spring.dataSource.ZKDynamicDataSource 数据源中可以定义多个数据源；如 读、写数据源；
- 自定义事务管理：com.zk.db.dynamic.spring.transaction.ZKDynamicTransactionManager 重写 org.springframework.jdbc.datasource.DataSourceTransactionManager 类 doGegin 方法，根据事务的只读或可写，设置当前事务需要使用的数据源；

3.3.2 myBatis

- 动态数据源插件：com.zk.db.dynamic.mybatis.ZKDynamicDataSourcePlugin
通过 sql 关键字判断是否需要写数据库，来设置当前使用的数据源是读数据源还是写数据源。

3.4 myBatis

3.4.1 主要配置

com.zk.db.mybatis.session.ZKDBConfiguration

com.zk.db.mybatis.spring.ZKDBSqlSessionFactoryBean

3.4.2 插件

- like 参数替换【不建议使用，建议在sql 定义时，添加这个处理；】
com.zk.db.mybatis.plugins.ZKSqlLikeParameter
- 分页
com.zk.db.mybatis.plugins.ZKSqlPage
- sql 运行效率
com.zk.db.mybatis.plugins.ZKSqlRunCost
- 动态数据源插件
com.zk.db.dynamic.mybatis.ZKDynamicDataSourcePlugin
- 自定义 mybatis 查询时内容参数
com.zk.db.mybatis.plugins.ZKCustomParams
插件会给继承自 ZKDBBaseDao 的接口的方法，添加自定义参数 _zkSql 参数，含四个属性，表名、表别名、查询结果映射、查询条件；注意需要结合 主要配置类实现；

3.4.3 dao 层扫描注解

com.zk.db.annotation.ZKMyBatisDao

spring 根据这个注解来加载 dao 层 bean;

3.4.4 类型处理器

- com.zk.db.mybatis.typeHandler.ZKJsonArrayTypeHandler
- com.zk.db.mybatis.typeHandler.ZKJsonTypeHandler