Miniob-001：代码架构及基本说明

**文档修订记录**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **版本** | **修订内容描述** | **确认人/日期** | **查阅人/日期** | **修改人/日期** |
| 1.0 | 初版 |  |  | 张嘉熙 20211017 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

1. 代码架构以及基本说明
2. obclient

客户端代码，包括了控制台函数和字符串处理函数

1. observer

服务器

* 1. event

一些event函数中使用数据结构

* 1. handler

+ handler.h

handle函数中用到的一些数据结构的声明

* 1. net

+ connection\_context.h

+ server\_param.h

+ sever.h

+ server.cpp

服务节点的基本实现，包括启动，关闭，以及初始化节点的功能

* 1. session

基本内容包括了各种会话需要处理的各种信息，例如sql的处理模式

+ session\_stage.h:

1. /\*\*
2. \* seda::stage使用说明：
3. \* 这里利用seda的线程池与调度。stage是一个事件处理的几个阶段。
4. \* 目前包括session,parse,execution和storage
5. \* 每个stage使用handleEvent函数处理任务，并且使用StageEvent::pushCallback注册回调函数。
6. \* 这时当调用StageEvent::done(Immediate)时，就会调用该事件注册的回调函数。
7. \*/
   1. sql

2.5.1. executor //算子执行部分，以select语句为标准，给出了一条语句的基本执行执行流程

2.5.2. optimizer //优化器，目前空白

2.5.3. parser //解析器，目前功能完备

2.5.4. plan\_cache //plan\_cache?目前空白

2.5.5. query\_cache //query\_cache?目前空白

* 1. storage
     1. common //表，库，索引等结构的定义实现
     2. default //disk 中的文件创建、管理
     3. mem //内存部分，需要在此处添加缓存策略
     4. trx //算子、事务部分

二、基本流程