

Database System Project

Jiarong Wei

魏嘉荣

目录

- [Project 要求](#)
- [评分规范](#)
- [Q & A](#)
- [附录1：设计指导](#)

目标

- 设计并实现一个单用户 SQL 数据库：MiniSQL
 - 允许用户通过字符界面输入SQL语句实现
 - 表的建立/删除
 - 索引的建立/删除
 - 表记录的插入/删除/查找
- 通过对 MiniSQL 的设计与实现，提高学生的系统编程能力，加深对数据库管理系统事先技术的理解。

需求

- 表

- 一个表最多可以定义32个属性
- 各属性可以指定是否为 `unique`
- 支持单属性的主键定义

- 索引

- 对于表的主属性自动建立 `B+` 树索引
- 对于声明为 `unique` 的属性可以通过 `SQL` 语句由用户指定建立/删除 `B+` 树索引
- 所有 `B+` 树索引都是单属性单值

需求

- 数据类型
 - integer
 - char(n)
 - $1 \leq n \leq 255$
 - float
- 数据操作
 - 可以通过指定用 **and** 连接的多个条件进行查询
 - 支持等值查询和区间查询
 - 支持每次一条记录的插入操作
 - 支持每次一条或多条记录的删除操作

需求

- 支持标准的SQL语句
 - 创建表
 - 删除表
 - 创建索引
 - 删除索引
 - 选择
 - 插入
 - 删除
 - 退出系统
 - 执行 SQL 脚本
- 每一条 SQL 语句以分号结尾
- 一条 SQL 语句可以一行或多行
- 关键字均为小写

评分规范

- Project 得分由三个部分组成
 - 报告 50%
 - 总体设计报告（中期） 20%
 - 详细设计报告 30%
 - 验收 50%

报告

- 提交截止时间

- 总体设计报告（中期） 2017.6.6
- 详细设计报告 2017.6

- 提交方式

- 总体设计报告以小组提交，详细设计报告以个人提交
- 总体设计报告只需要组长提交一份即可
- 报告提交至 **FTP** 上的对应目录下

报告 - 总体设计报告

- 系统概述
 - 目的
 - 背景
 - 功能
 - 环境
 -
- 测试方案及样例
- 系统设计
 - 各个模块
- 组员信息以及分工

报告 - 详细设计报告

- 针对各个模块
 - 设计原理
 - 设计思路
 - 结构图
 - 接口描述
 -

验收

- 流程
 - 演示
 - 功能测试：基本的功能测试
 - 压力测试：针对大数据量
 - 设计陈述：小组成员陈述各自的设计
 - 随机提问
- 时间
 - 2017.6

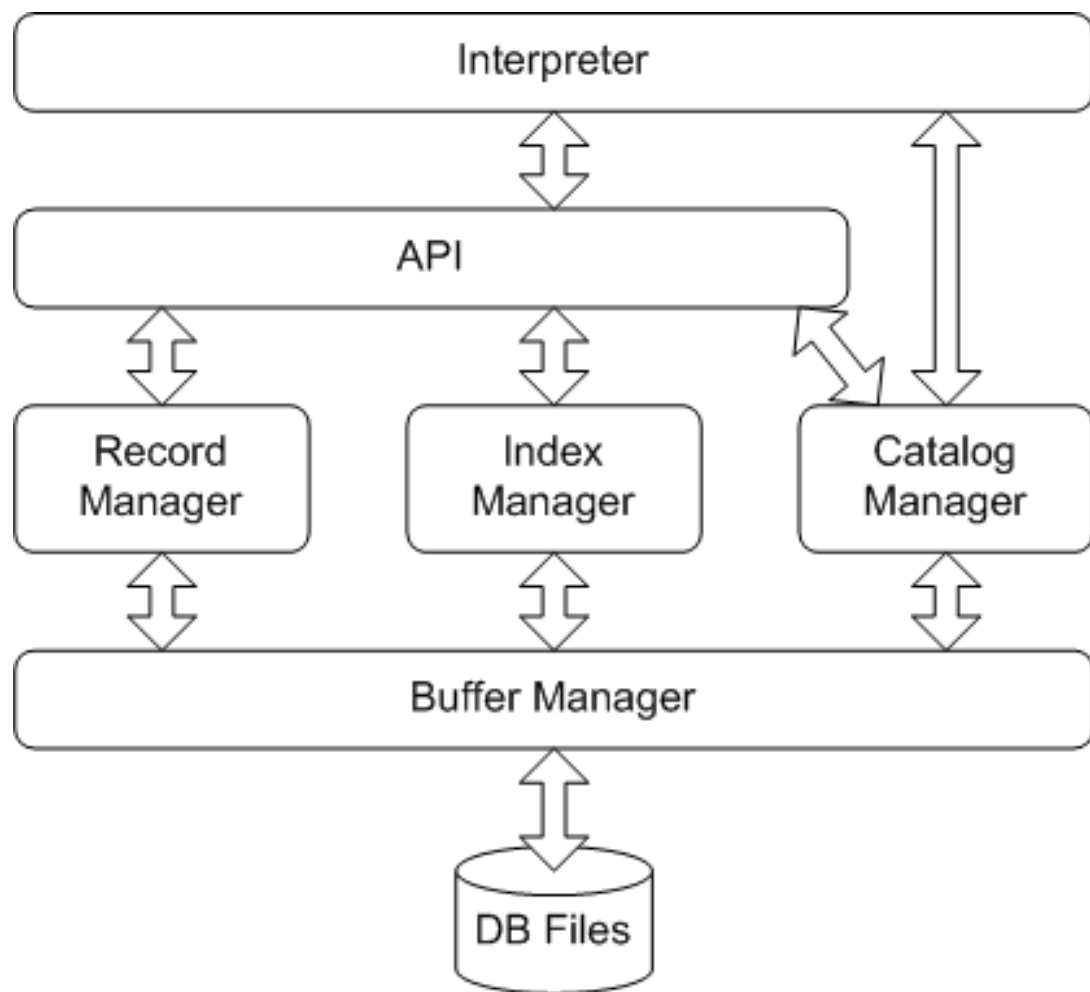
Q & A

- Q: “好复杂呀~可以抱大腿吗？” “好简单呀~可以单挑吗？”
- A: Project 分组完成，每组人数 1-4 人。组长将小组成员信息在 2017.5.16 前告诉我 (然后就可以结对编程了呢)
- Q: “用什么语言好呢？” “PHP是世界上最好的语言~”
- A: 使用 C / C++ / Java / C# 完成

设计指导

仅供参考！！

系统架构



Interpreter

- 解析 SQL 语句 判断语句是否合法
判断表是否存在/语法
- 调用具体 API 执行 SQL 语句
- 返回/输出结果

API

- 提供执行 SQL 语句的接口
- 对接 Interpreter 和其他模块

Catalog Manger

- 数据库的元数据信息
 - 表的定义信息（表名、列数、主键、索引等）
 - 属性的定义信息（数据类型、是否唯一等）
 - 索引的定义信息（所属表、索引建立在哪个属性上等）
 - 提供访问及操作上述信息的接口

Record Manager

- 管理记录表中数据的数据文件
 - 数据文件的创建
 - 数据文件的删除
 - 记录的插入
 - 记录的删除
 - 记录的查询
 - 提供相应接口

Buffer Manager

- 负责缓冲区的管理
 - 数据块的写入、写出
 - 实现缓冲区的替换算法（e.g. LRU）
 - 记录块状态（是否被修改过）
 - 锁定缓冲区某块，不允许替换出去
- 缓冲区与文件系统交互的单位是块
 - 一般可定为 4KB 或 8KB

Index Manager

- 负责 B+ 树索引的实现
 - B+ 树索引的创建
 - B+ 树索引的删除
 - 等值查询
 - 插入键值
 - 删除键值
 - 提供相应的接口

DB Files

- Catalog 文件
- 数据文件
- 索引文件