

**数据库课程设计报告**

**——RDBMS设计报告**

**学 院：** 计算机与信息技术学院

**专 业：**  计算机科学与技术

**小组成员：** 李若森 13281132

宋世平 13281037

**指导教师：** 徐薇

**北京交通大学**

**2015年 12月 30日**

目录

[一、需求分析 4](#_Toc439242922)

[(一) 系统设计需求 4](#_Toc439242923)

[(二) 功能及指令设计 4](#_Toc439242924)

[二、算法模型与功能设计 4](#_Toc439242925)

[(一) 存储结构设计 4](#_Toc439242926)

[(二) 语义树生成处理 5](#_Toc439242927)

[(三) 表义树转化计算 5](#_Toc439242928)

[(四) 各功能拆分实现 5](#_Toc439242929)

[三、函数关系设计 6](#_Toc439242930)

[(一) 函数层级 6](#_Toc439242931)

[(二) 函数调用关系 6](#_Toc439242932)

[四、功能展示 6](#_Toc439242933)

[附录： 10](#_Toc439242934)

# 一、 需求分析

## 系统设计需求

1. 设计完成数据库管理系统。
2. 采用文件形式进行数据存储。
3. 系统采用交互式命令解释型。
4. 系统完成对命令的解释、执行，并返回完成结果。
5. 提供可执行功能及对应指令格式。

## 功能及指令设计

1. 指令格式采用SQL语句标准格式
2. 支持的功能及指令格式(仅单表、不可递归调用指令、不支持NULL值):

|  |
| --- |
| 1. CREATE TABLE |
| **e.g. CREATE TABLE example (Col1 INT, Col2 DOUBLE, Col3 STRING);** |
| 1. DROP TABLE |
| **e.g. DROP TABLE example;** |
| 1. SELECT |
| **e.g. SELECT \* FROM example;** |
| 1. INSERT INTO |
| **e.g. INSERT INTO examples VALUES( 10, 3.14, 'hello' );** |
| 1. UPDATE |
| **e.g. UPDATE example SET Col = 'col2' WHERE Col = 'col1';** |
| 1. DELETE FROM |
| **e.g. DELETE FROM example WHERE Col = 'col';** |
| 1. HELP |
| 1. QUIT |

# 二、 算法模型与功能设计

## 存储结构设计

结构体内部采用vector<vector<string>>存储库中数据，对每列数据采用标记区分其数据类型(int(0)，double(1)，string(2))，对行列分别序列化分解，方便随机访问。

## 语义树生成处理

通过LEX提取SQL语句中的关键字，并将结果传递给YACC。然后通过YACC进行句法分析，依据LEX给出的关键字信息分拆语句为一个个最小终端标记，方便后续处理。在此过程后，经处理的语句最终会生成一个由孩子兄弟表示法表示的一颗多叉树，每个节点都有不同的所对应的标记表示该节点的类型以及存储的值。

具体生成过程详见附件1（语义树生成）。

## 表义树转化计算

1. 表义树转化：

在前一步生成语义树的基础上，提取关键结点（如"+", "=", "AND"等运算符，1, 1.5, ‘string’等确值及变量(列)名）将WHERE条件部分转化为方便表达式计算的表义树。

1. 表义树计算：

后序遍历表义树可得WHERE部分的后缀表达式(逆波兰表达式)，其中按照参数及返回值不同分为三级二元运算:

1. 布尔值参与计算，返回布尔值("AND", "OR", "XOR")
2. 数值参与计算，返回布尔值("=", "<>", ">", "<", ">=", "<=")
3. 数值参与计算，返回数值("+", "-", "\*", "/")

表义树计算过程严格按照优先级由高到低计算，WHRER部分条件表达式最终必然返回布尔值。

将行中数据带入表义树计算，依据返回的布尔值真假对符合条件的行进行记录并最终返回。

## 各功能拆分实现

1. 建表/增行

建表过程先实现顺序增列，后多次执行增行操作。

增行过程利用Map实现由列名到列序列号的转化，随机访问在新行中补充数据。

1. 查询

逐行遍历，套入WHERE条件表达式中，表义树计算筛选获得符合条件行。

语义树分解获得被查询列，与对应行共同构成待查询表，其中若行为表达式形式，则将对应行列数据带入，并返回表达式值作为列值构表。

1. 更新数据

在查询的基础上利用Map将列名转化为列序号，并进行更新。

1. 删除数据

利用WHERE条件表达式查询出行，将符合条件的行直接删除。

# 三、 函数关系设计

## 函数层级

1. 功能接口

根据待实现功能(如CREATE TABLE等)进行调用的功能模块主接口。

1. 语义树生成

将输入指令按照既定规则转化为树形结构进行存储。

1. 语义树分析

将生成语义树按照需求划分、转化为多组链表、树状结构，并加以简化和处理分析。

1. 原子化功能实现(原子表级操作)

各功能主模块中需要实现的功能分解为直接对表/元素进行操作的过程。

## 函数调用关系

详见附件2（函数调用关系图）。

# 四、 功能展示

|  |
| --- |
| CREATE TABLE: |
|  |
| DROP TABLE: |
|  |

|  |
| --- |
| SELECT: |
|  |
| INSERT INTO: |
|  |

|  |
| --- |
| UPDATE: |
|  |
| DELETE FROM: |
|  |

|  |
| --- |
| HELP: |
|  |

# 附录：

## 附件列表：

1. YACC逻辑图.zip
2. 函数调用关系图.zip
3. RDBMS.zip (源代码打包)