

## 数据库调优课程研究报告之索引研究

|  |  |
| --- | --- |
| 院 系： | 软件学院 |
| 组 长 学 号： | 19212010054 |
| 组 长： | 吴俊 |
| 小 组 成 员： | 吴俊 |
| 完 成 日 期： | 2019年12 月 9 日 |

## 目录

[数据库调优课程研究报告之索引研究 1](#_Toc17284)

[目录 1](#_Toc20287)

[图目录 2](#_Toc15004)

[表目录 3](#_Toc7245)

[一、 系统概述 4](#_Toc23951)

[二、 系统平台 5](#_Toc12727)

[三、 系统数据库设计 6](#_Toc19622)

[四、 系统索引算法 7](#_Toc22551)

[4.1 线性哈希(Linear Hash) 7](#_Toc31162)

[4.2 扩展哈希(Extensible Hash) 7](#_Toc32483)

[4.3 B+树索引(B+ Tree Index) 7](#_Toc20322)

[五、 实验设置 8](#_Toc9060)

[六、 实验过程 9](#_Toc26252)

[七、 结论 10](#_Toc28791)

## 图目录

[图 三-1 总体E-R图 6](#_Toc13082)

## 表目录

**错误！未找到图形项目表。**

# 系统概述

本文设计实现了一个比较不同索引算法性能特点的系统。该系统是由Java开发的Spring Boot Web应用程序，其中线性哈希算法(Linear Hash)，扩展哈希算法(Extensible Hash)以及B+树索引算法(B+Tree Index)均由Java实现，系统实现了每种索引算法的增加、删除和查询数据的方法；系统模拟生成的数据均以设计的表名作为文件名，以csv格式保存在文件中，每个文件代表一张数据库表，文件中的每一行代表一条数据记录，每条数据记录所在的行代表其在磁盘上的位置，以这种方式模拟一个数据库；系统以图表的形式可视化实验过程中不同索引算法在不同实验条件下的性能，不仅可以看出不同索引算法随实验条件改变而变化的整体趋势，而且可以对比索引算法之间的性能差异。

本文设计了三种不同的实验方案，分别是：探索索引种类与数据量的关系，探索索引种类与索引字段类型的关系以及探索不同索引种类的查询范围。在探索索引种类与数据量的关系实验中，系统保持索引字段不变，观察随着插入或删除的数据量的变化，不同索引算法插入，查询和删除数据的用时变化；在探索索引种类与索引字段类型的关系实验中，系统保持每次插入的数据量不变，观察不同索引算法在索引字段变化时的表现；在探索不同索引种类的查询范围实验中，系统保持索引字段和数据量不变，改变查询要求，观察不同索引算法能否完成任务。

# 系统平台

系统开发语言：Java

系统开发工具：Jetbrain IntelliJ Idea

系统开发框架：Spring Boot，Echarts

系统开发环境：

Windows 10 专业版 Intel(R) Core(TM)i5-8250U CPU @1.60GHz 1.80GHz

16.0GB内存，64位操作系统

系统运行环境：

Windows 10 专业版 Intel(R) Core(TM)i5-8250U CPU @1.60GHz 1.80GHz

16.0GB内存，64位操作系统

# 系统数据库设计

本系统设计的关系数据库应用场景是大学图书馆的图书管理系统。系统共设计了五张数据表，分别是图书信息表(Book)，学生信息表(Student)，教师信息表(Teacher)，图书借阅表(BookBorrow)以及图书归还表(BookReturn)。为了方便比较不同索引算法的性能，系统并未将这些表设计到数据库中，而是保存为多个csv文件，一张表对应一个文件名与表名相同的csv文件，数据保存在文件中，文件中的一行代表一条数据记录，数据记录所在的行代表其磁盘地址，以这种方式模拟一个图书管理系统的数据库。

如下为该数据库的E-R图：

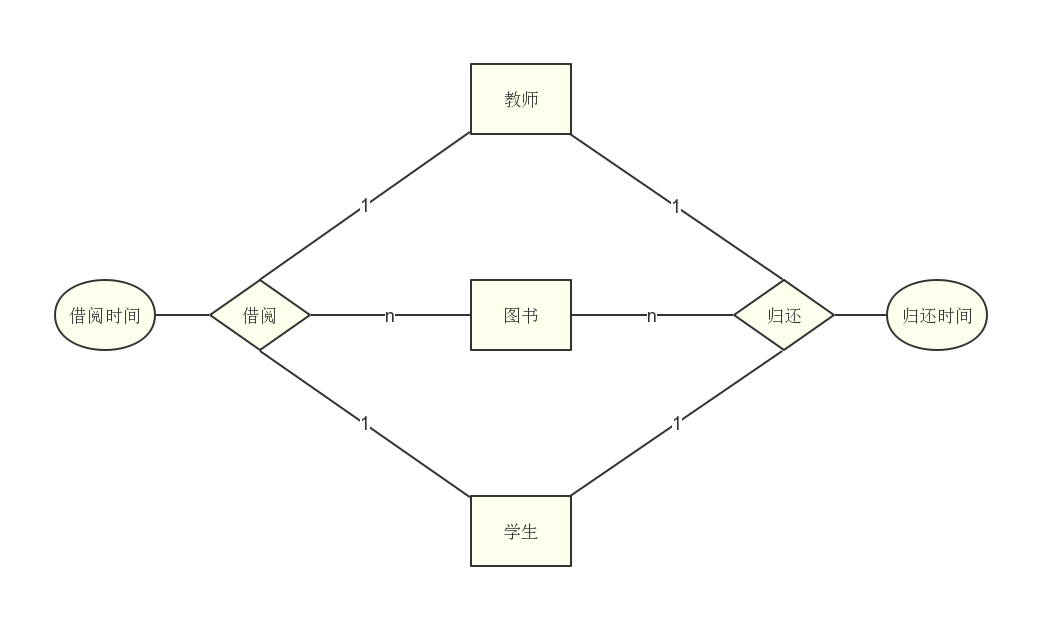


图 三-1总体E-R图

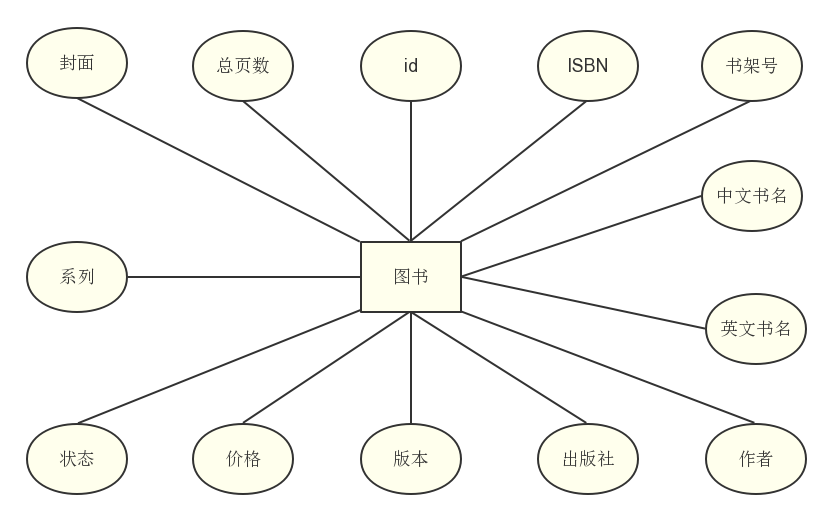


图 三-2图书E-R图

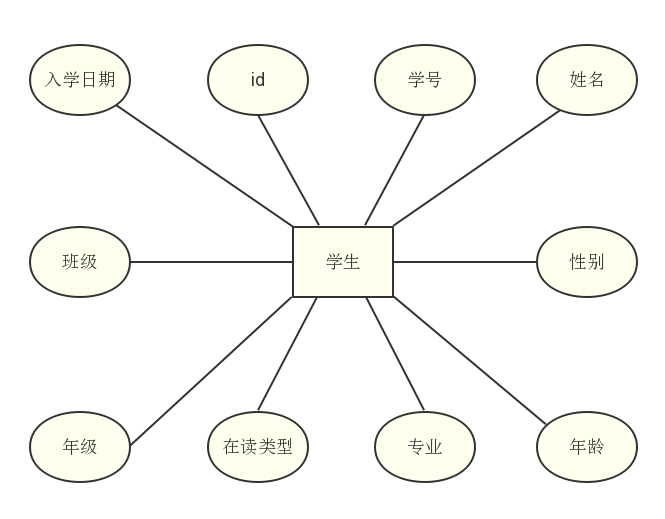


图 三-3学生E-R图

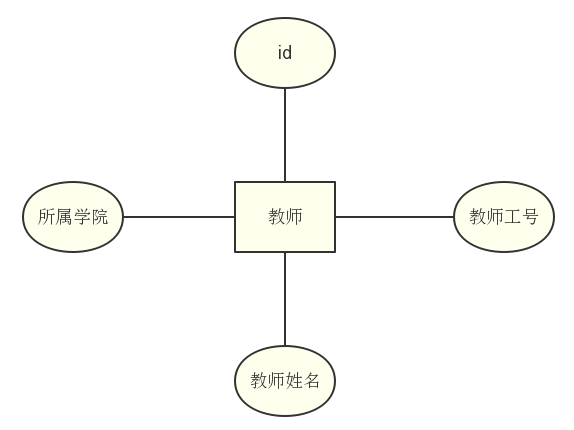


图 三-4教师E-R图

数据表设计如下：

表 三-1图书信息表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 约束 | 描述 |
| id | int | primary key | 主键，记录唯一标识 |
| book\_isbn | varchar(20) | not null | 图书ISBN编号 |
| book\_bookcase\_no | varchar(20) | not null | 图书书架号 |
| book\_name\_cn | varchar(30) | not null | 中文图书名 |
| book\_name\_en | varchar(30) | not null | 英文图书名 |
| book\_author | varchar(20) | not null | 图书作者 |
| book\_press | varchar(20) | not null | 出版社 |
| book\_version | int | not null | 图书版本 |
| book\_price | float | not null | 图书单价 |
| book\_series | varchar(30) |  | 图书所属系列 |
| book\_cover | varchar(30) |  | 图书封面 |
| book\_total\_page | int | not null | 图书总页数 |
| book\_status | int | not null | 图书状态，0表示未借出，1表示已借出 |

表 三-2 学生信息表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 约束 | 描述 |
| id | int | Primary key | 主键，记录唯一标识 |
|  |  |  |  |
|  |  |  |  |

# 系统索引算法

## 4.1 线性哈希(Linear Hash)

## 4.2 扩展哈希(Extensible Hash)

## 4.3 B+树索引(B+ Tree Index)

# 实验设置

在探索索引种类与数据量的关系实验中，系统保持索引字段不变，分别一次性插入一千，一万，十万，一百万和一千万条记录，然后记录不同索引算法所消耗的时间，并反馈到前端显示。每次数据插入是分开的，在每次插入数据后，还可以查询指定的记录，并统计不同索引算法查询所消耗的时间，并反馈到前端显示。在插入一千万条记录后，系统分别一次性删除一，十，一百，一千和一万条数据记录，统计不同索引算法消耗的时间并反馈到前端显示。

在探索索引种类与索引字段类型的关系实验中，系统从某张表中选取两个不同类型的字段分别建立索引，不同类型的字段是指值稠密型字段和值稀疏型字段，值稠密型字段是指所有数据中该字段的值出现的种类很少，一般是枚举类型，值稀疏型字段与之相反。然后系统每次插入一千条数据，

# 实验过程

# 结论