

拔尖创新人才培养背景下基于“Docker+OJ”的数据库内核实验改革与探索*

于龙飞 彭朝晖 王运来

山东大学计算机科学与技术学院, 青岛 266000 北京奥星贝斯科技有限公司, 杭州 310013

摘要 围绕在本科生阶段开展“数据库系统”内核实验面临的实验难度过大、学生参与兴趣度不高、考核评价指标不明确等问题,提出了基于“Docker+OJ”的数据库内核实验改革具体实施路径,通过验证分析,本次改革探索对于全面提升学生数据库系统内核实现能力具有非常明显的效果,收到了数据库领域主流企业对于人才培养质量的认可。

关键字 数据库系统, 内核实验, 课程改革

Experimental Reform and Exploration of Database Kernel Based on "Docker+OJ" Under the Background of Top-notch Innovative Talent Training

Yu Longfei Peng Zhaohui Wang Yunlai

School of Computer Science and Technology,
Shandong University
Qingdao 266000, China;
yulongfei@sdu.edu.cn

Oceanbase Technology Co., LTD
Oceanbase
Hangzhou 310013, China
wangyl@oceanbase.com

Abstract—Focusing on the problems of "Database System" kernel experiment in undergraduate stage, such as too difficult experiment, low participation interest of students, unclear assessment and evaluation indicators, this paper proposes a specific implementation path of database kernel experiment reform based on "Docker+OJ". Through verification and analysis, This reform exploration has a very obvious effect on comprehensively improving the kernel realization ability of the student database system, and has received the recognition of the mainstream enterprises in the database field for the quality of talent training.

Keywords—Database System, Kernel Experiment, Course Reform

1 引言

党的二十大报告指出:我们要坚持教育优先发展、科技自立自强、人才引领驱动,加快建设教育强国、科技强国、人才强国,坚持为党育人、为国育才,全面提高人才自主培养质量,着力造就拔尖创新人才^[1]。数据库系统作为信息基础设施的核心部分,对于党政机关、企事业单位、科研机构等至关重要,其核心——数据库管理系统(DBMS)倾注了数据库专家和技术人员数十年研究开发的结果,是最复杂的系统软件之一^[2]。随着国家重点领域卡脖子问题得到社会的广泛

关注,数据库行业对于内核领域人才的需求变得愈加迫切^[3],高校数据库系统实验教学亟需应对新时代变化要求^[4],从“以应用能力培养为主”转向“应用能力、系统研发与分析能力培养并重”转变^[5]。

2 数据库系统内核实验现状剖析

目前,国内拔尖创新人才培养主要以解决国家关键科学和工程领域重大问题为导向^[2],引导学生投入到国家战略急需、产业需求紧迫、关乎国计民生、强调自主可控的核心技术领域中创新实践^[6]。但是多年来,国内高校计算机类本硕毕业生却普遍不具备对数据库系统内核进行调试、修改和优化的能力^[7],究其原因,主要有以下几点:首先,设计开发数据库管理系统的门槛较高,很少有本科生甚至研究生具备完整设计实现数据库管理系统的能^[8];其次,国内高校“数据库系统”系列课程重点在“用”数据库,而不是“造”

***基金资助:** 本文得到教育部首批新文科研究与改革实践项目“基于虚拟现实技术的党史教育教学模式研究与实践”(2021180018),山东大学2023年度教育教学改革研究重点项目“计算机拔尖创新人才培养质量保障机制及自主支撑体系研究与实践”、一般项目“基于国产数据库系统内核实验课程的一流教材建设与改革研究”资助。

数据库^[9],导致在教学内容、实验资源和课程评价等方面存在短板^[10];第三,国内数据库系统市场长期被外国公司垄断,导致国内人才和技术积累不足,相关专业的毕业生就业需求得不到很好的满足^[11];最后,大量开源数据库软件的广泛使用加剧了行业内部竞争,传统商业数据库的盈利模式受到挑战,国内厂商普遍采用“开源内核+商业化外壳”的封装形式^[12],对数据库系统内核研发人才的需求相对有限^[13]。

3 基于“Docker+OJ”的数据库内核实验改革

3.1 改革整体设计

数据库内核实验基于计算机类本科生专业必修课“数据库系统”,开课时间为第三学期,该课程共88个学时,其中实验共16周32个学时,参考的教材为《数据库系统概念(第7版,本科教学版)》(Abraham Silberschatz等著,杨冬青等译,机械工业出版社)。改革前,实验包含2个部分,分别为“SQL实验”和

“应用开发与实现”,每个实验16学时,改革后,提高实验教学整体难度,在不增加实验总学时数的情况下,引入“数据库内核实验”,作为占比较重的实验内容,在此基础上对实验课程进行了如表1设计。

(1) 优化整合

主要是调整“SQL实验”和“应用开发与实现”实验难度和学时安排,为数据库内核实验预留足够学时。

① SQL实验部分,对SQL实验内容进行“合并同类项”,更加强调SQL基础知识的贯通理解,以了解熟悉SQL语句的主要语法为主,降低复杂查询的考察强度,通过8个实验涵盖SQL语法的数据定义与操作、基本和复合查询、统计函数与视图、索引优化与事务等核心知识点,强化学生SQL语法的编写能力。

② 综合课程设计实验部分,由原本的单人独立完成变为为组队完成,要求使用git仓库托管小组的代码,通过小组分工合作,实现一个具有实际作用、前后端分离的数据库管理系统,最后由组长进行项目汇报。

表1 实验课程改革整体设计

原实验课程		对比指标	改革后实验课程		
实验1: SQL实验	实验2: 应用开发与实现	名称	调整后实验1: SQL实验	新增实验2: 数据库内核实验	调整后实验3: 应用开发与实现
2-9周	10-17周	周数	2-6周	7-12周	13-17周
16	16	课时	10	12	10
8	1	实验项目数	8	5	1
50%	50%	分数占比	25%	37.5%	37.5%
否	否	是否分组	否	是	是
-	-	小组人数	-	1-3人	1-3人
平台自动评测	报告展示	评分办法	平台自动评测	平台自动评测	报告展示
答疑	答疑+编程小梯子	教学辅助	答疑	答疑+随堂解析+线下专题突破	答疑+编程小梯子
需要	需要	实验报告	需要	需要	需要
平台自动查重	-	防作弊	平台自动查重	在线实验平台、代码比对、版权	-

(2) 课赛结合

为了激发学生挑战自我的积极性,将数据库系统内核理论教学、实验环节与数据库系统大赛时间同步。

① 授课顺序上,将教材第三部分“数据管理实现技术”中的“存储管理”“索引”“查询处理和查询优化”“事务管理”等章节前置,时间安排上与国内主要的数据库系统大赛初赛阶段时间基本吻合,实验安排和数据库系统大赛赛程相互衔接,学生沿着课堂学习、实验验证与参赛突破三个台阶逐步攀登,实现知识点在小周期内循环迭代转化为实践能力的应用效果。

② 考察与奖励机制上,学生通过组队参加竞赛的成绩可以按比例换算为课程实验的成绩,进入决赛环节的队伍实验部分以满分计算,激励学生勇于突破创

造佳绩,大赛决赛的成绩还可以作为保研的重要依据,增加学生参与热情。

(3) 线上线下混合

通过在线实验平台管理学生的实验项目,结合线下答疑与研讨解决学生实际困难。

① 发挥在线实验平台的管理效能,将各个学生分组的代码仓库进行集中统一管理,对源代码进行聚类性和相似性比对,从源头管控实验代码抄袭的问题^[14];鼓励学生在学习借鉴开源代码的优秀实现方式,并要求学生将最终的代码在Github或Gitee等平台上公开,利用开源社区规范引导学生形成学术诚信和学术自觉,要求学生所有公开代码添加开源许可、版权声明

和.gitignore文件,避免发布敏感信息和出现版权问题。

② 建立线下定期答疑与学生互助帮扶机制,组织实验指导教师和助教每周定期线下答疑,解决学生的个性化问题,同时加强学生党员、骨干示范帮扶作用和 student 间跟踪监控评估与反馈,及时跟进学业基础差、实验进度慢、信心不足的学生,确保学生团队在整体上保质保量完成全部实验任务。

3.2 实验内容设计

数据库系统内核实验相比于经典的 SQL 查询实验和数据库系统应用开发实验有很大不同,一是要求学生能够读懂完整项目的框架结构,能够定位目标代码并理解重要函数的具体功能,二是要求学生利用所学 SQL 语法知识和数据库管理系统的原理,能够找到题目所需问题切入点并能够熟练运用所学的编程技巧完善代码功能,三是要求学生能够搭建项目调试环境并理解 Makefile 文件,具备初步管理中大型项目的能力。基于以上考虑,按照“先观察、后修改、再优化、能探索”的目标层层递进,结合数据库大赛相关项目案例,设计了如下实验内容。

表 2 实验内容设计表

实验名称		实验学时	答疑学时	实验要求
实验环境准备	实验一: Visual Studio Code + Docker 开发数据库内核	2	2	完成实验任务,在线提测代码,提交实验报告
SQL 基本语法优化与实现	实验二: 元数据校验与删除表实验	2	2	
	实验三: 插入多行数据实验	2	2	
SQL 高级特性优化与实现	实验四: 数据更新实验	3	3	
	实验五: LRU 算法实验	3	3	

(1) 实验环境准备

实验目的:熟悉利用主流工具管理项目代码、搭建调试环境、运行调试样例,观察数据库运行内部机理。

主要内容:

① 以小组为单位,创建远程代码仓库,通过 git 工具拉取开源数据库管理系统源代码,通过 push 命令将本地源代码同步到远程仓库,赋权 OJ 平台测试账号(用于提交测试)和实验课程指导教师监督账号(用于源代码相似性比对)。

② 搭建并配置 Docker 环境,使用 Visual Studio Code 集成开发工具远程到容器环境中,安装编译所需

依赖库和调试所需扩展插件,配置自动化编译和调试脚本,对源代码进行首次编译运行并观察文件和目录变化情况,运行数据库系统服务端,并用客户端发送测试 SQL 命令,通过断点观察数据库运行状态以及函数堆栈、变量、文件等变化情况。

(2) SQL 基本语法优化与实现

实验目的:通过实现基本的 SQL 语法功能,掌握 SQL 语句在数据库中执行的整个流程,能够修改和优化语法功能。

主要内容:

① 实现 DROP TABLE 语法:阅读并参照源码中 CREATE TABLE 的实现过程,编写代码实现 DROP TABLE 的功能。学生需要理解数据库在创建表的过程中都执行了哪些操作,然后通过逆向思维逐个删除对应的表文件,考察学生参考现有代码进行修改编程的能力。

② 实现 UPDATE 语法:阅读并理解 INSERT 和 DELETE 的实现过程,编写代码实现 UPDATE 单行记录的功能,并能够处理 WHERE 条件。学生需要掌握一行记录从表中读取和插入表中新的记录的整个过程,根据不同的技术路线,可以选择一行记录原位置读取并更新的方式,或者先删除原纪录再插入新纪录的方式,考察学生参考现有代码,选择技术路线并实现新功能的能力。

(3) SQL 高级特性优化与实现

实验目的:通过实现数据库系统的高级特性,提升对系统的理解能力,能在优化高级特性的基础上,探索实现高效的调度算法。

主要内容:

① 实现 JOIN 语法:根据课程教材中相关章节的实现原理,实现多表的 INNER JOIN,并且能够处理带有 ON 条件的多表 JOIN 操作,需要理解表与表之间 JOIN 的关系代数原理,并且对多表情况下 JOIN 顺序选择问题进行优化,避免内存爆炸问题,考察学生对于代码运行时间和资源占用的优化能力。

② 实现缓冲区 LRU 算法:根据课程教材中关于存储管理的实现原理,实现内存缓冲区页面调度算法,需要对 SQL 查询历史进行记录,使用高效的查找策略定位最近使用的数据块,并能够使用锁机制处理并发情况下数据换入换出访问冲突等问题,考察学生处理复杂算法问题时的快速学习和探索能力。

3.3 基于 Docker 的实验环境设计

为了让零基础学生也能够逐步掌握数据库内核开发的必备技能,能够创建和管理实验环境也是非常重要,将实验环境的配置本身作为第一项实验内容,使

用带有图形界面的主流开发工具,通过命令行或配置文件的形式,构建统一的调试环境,可以让学生很直观的理解开发过程的全貌。

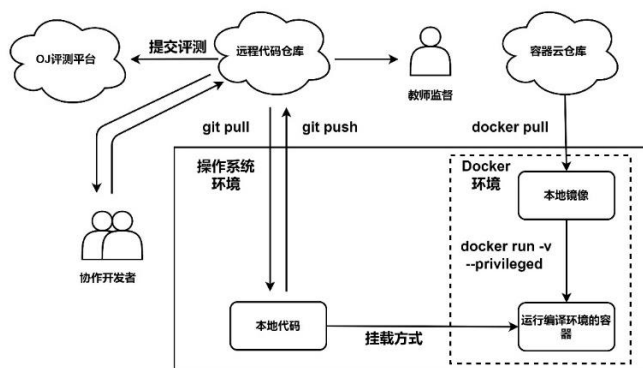


图 1 基于 Docker 的实验环境示意图

(1) 设计原则

① 内核编译需要独立的环境,学会对环境进行配置可以在后期的调试中,尽可能减少环境带来的干扰,且让学生学会构造和运行维护这样的环境本身也是计算机系统能力的一部分。

② 选择使用 Docker 这种跨平台的轻量级虚拟化技术,能够兼顾学生不同类型、不同配置的实验开发设备,提供稳定和统一的实验环境基准。

③ 容器化技术作为目前较为主流的云计算技术,与数据库系统未来的技术演进息息相关,便于学生提前了解当今技术趋势。

(2) 配置方法

① 从官方地址下载并安装 Docker 客户端。

② 将本地代码挂载到运行的容器中,使用 `docker run -v` 命令将本地代码目录映射到容器目录指定路径,可以确保容器出现异常时,代码不会受到影响,使用 `--privileged` 参数授权对挂载代码进行调试。

③ 在 Visual Studio Code 集成开发工具中安装 Docker 扩展插件,通过 `attach` 方式进入容器环境中,安装 `libevent`、`jsoncpp`、`googletest` 等依赖库,使用编译命令在容器中对源代码进行编译,编译通过后运行数据库服务端。

④ 从容器中另外开启一个命令行终端,使用客户端命令登录服务端,输入 SQL 语句测试数据库运行情况。

⑤ 使用容器进行实验过程中,如果启动容器时忘记增加一些参数,但是环境已经配置好不想从头重复操作,可以使用“`docker commit -t 容器名:版本号`”

的方式提交更改,版本号可以自行命名,启动新容器时指定基于容器的某个版本号即可。

3.4 基于 OJ 的考核评测设计

(1) 考核方式

每个小组的代码仓库添加了实验课程指导教师监督账号作为观察者,可以获取源代码并进行代码级的查重比对,发现学术不端问题一经确认,实验部分以 0 分记录。同时为了更加客观综合评价实验成绩,对于考核方式进行了如下设计,实验部分满分为 100 分:

① 未参加数据库大赛的小组,需按要求完成全部实验内容,并且结合代码实现思路,从整体上进行技术分享报告,实验成绩计算方法为:总分 = 5 个实验总分 \times 50% + 技术分享报告得分 \times 50%。

② 参加数据库大赛的小组,在完成与实验内容同等类型、数量的赛题基础上,未进入全国 100 强的小组,根据初赛阶段最终成绩计算,总分 = 5 个与实验同等类型和难度赛题得分 \times 50% + 其他赛题得分换算的比例分 \times 50% + 技术分享报告额外奖励分 \times 20%。

③ 参加数据库大赛的小组,进入初赛阶段全国 100 强的队伍,数据库内核实验部分记为满分。

④ 进入决赛阶段的队伍(全国 50 强),本课程全部实验记为满分。

(2) 评测方式

评测使用标准的单元测试工具,可以在本地运行单元测试代码自行进行评测,也可以通过 OJ 进行在线评测。

评测流程为:编译源码、获取测试用例文件、启动数据库 server、执行测试用例、对比执行结果与预先设置的结果文件、输出结果。

评测之前,首先将本地已经完成实验要求的代码通过 `git push` 命令推送到远程仓库,将仓库地址、`commit id` 和仓库分支名称复制到 OJ 平台提测弹窗内,即可自动发起在线评测请求,稍等片刻 OJ 平台将返回测试结果。在线评测时,由于每个源码仓库已经添加了 OJ 平台测试账号,OJ 平台将自动拉取仓库中的代码,并启动后台评测服务,代码运行的结果可以直观的展示给学生。

4 改革成效

在本科生阶段推进基于“Docker+OJ”的数据库内核实验,通过“优化整合、课赛结合、线上线下混合”的改革思路探索出一系列方式简洁、难度递进、激励先进、兼顾整体的改革举措,从不同方面体现了良好的实验效果。

