

计算原理视角中的数据库课程认知分析*

职为梅 宋伟 朱真峰**

郑州大学计算机与人工智能学院, 郑州 450001

摘要 教材往往以知识和技能为主体,其背后的抽象模式或者系统化思考较为欠缺。《伟大的计算原理》一书中提出的六类原理:通信、计算、记忆、协作、评估和设计,正好可以为计算机类专业课程提供一种新的知识提取方式。基于此六大原理,本文对数据库原理与技术课程进行重新审视和系统分析,深入探讨了其中的计算原理表现形式,以及这些原理之间的内在联系;进一步地,教学团队把分析结果在大约 500 名本科生中进行了初步的教学尝试和验证,学生对知识的整体理解和把握能力提升了约 45%。本文在探讨信息管理内涵式发展的同时,有助于教育工作者从一个新的视角对学生进行系统能力培养,这对提升学生的深入思考 and 创新能力具有重要意义。

关键字 计算原理,数据库,内涵式发展,系统能力培养

Cognitive Analyses of Database Course from the Perspective of Computing Principles

Weimei ZHI Wei SONG Zhenfeng ZHU

School of Computer Science and Artificial Intelligence

Zhengzhou University,

Zhengzhou 450001, China;

iezfzhu@zzu.edu.cn

Abstract—Teaching materials often focus primarily on knowledge and skills, with a relative lack of underlying abstract patterns or systematic thinking. The book *Great Principles of Computing* proposes six categories of principles: Communication, Computation, Recollection, Coordination, Evaluation, and Design, these principles offer a novel methodology for knowledge extraction in computer science courses. Building on these six fundamental principles, this paper reexamines and systematically analyzes the course *Database Principles and Technologies*, delving into the manifestations of computational principles within it and the intrinsic connections among these principles. Furthermore, the teaching team conducted a preliminary teaching trial and validation of the analysis results among approximately 500 undergraduate students, resulting in an approximate 45% improvement in students' overall comprehension and mastery of knowledge. While exploring the connotative development, this study can assist educators in cultivating students' systematic capabilities from a novel perspective, and these teaching experiments are of great significance in enhancing students' abilities of deep thinking and innovations.

Keywords—Computational Principles, database systems, connotative development, systematic competency cultivation

1 引言

系统化分析可以从多个角度展开。论文[1]从数据库的定义出发,对数据库的知识点进行了系统化梳理,在打破一般数据库原理课程章的基础上,从节的角度重新组织,并给出了相应的思维导图。论文[2]深入探讨了数据库原理背后的临界知识,即规律、原

理等内容,从深层的思维模式进行了分析,认为数据库中蕴含着分层思维、简约思维、系统思维、底线思维和等价变换等内容,并进一步分析了这些思维模式的普遍存在性。

本文准备从六类计算原理的视角对数据库理论和技术进行重新审视。这些原理之间相辅相成,针对不同的计算机技术,展现出各自的组合与权重^[3]。六大原理简述如下(其可视化如图1所示)^[3]:

通信(Communication):在不同位置之间可靠地传输信息。

计算(Computation):表示可计算性。

* 基金资助: 本文得到郑州大学重点教改项目(编号: 2024ZZUJGXM084): AI 赋能《数据库系统原理》课程的探索与实践资助。

**通讯作者: 朱真峰 iezfzhu@zzu.edu.cn。

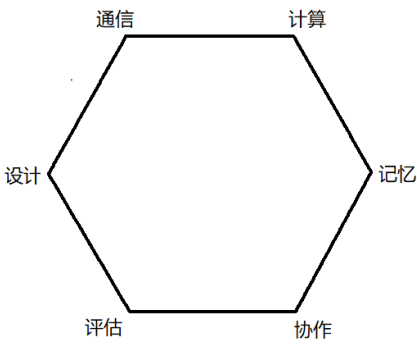


图 1 计算原理中的六大视角

记忆 (Recollection): 信息的表示、存储和检索。

协作 (Coordination): 有效利用多个自主的计算智能体 (agent)。

评估 (Evaluation): 度量系统能否达到预期的计算能力。

设计 (Design): 构建软件系统, 实现可靠性。

2 数据库中六类原理的具体表现

数据库有一个浓缩的定义: 持久存储在计算机中的, 有组织的, 可共享的, 大规模的数据的集合^[4]。这是从数据的角度给数据库进行的定义。数据库也可以从系统的角度进行描述: 数据库系统设计用于管理大量信息。管理方面包含定义信息存储结构和提供信息操作机制; 如果系统出现故障和被非法用户接触, 数据库系统能够确保存储信息的安全; 数据往往由多个用户共享, 系统还需要避免数据操作异常^[1,5]。基于对数据库原理课程的上述理解与描述, 本文进一步从六大计算原理出发, 给出了一个认识数据库的新视角。表 1 进行了简单说明 (同时整合了论文[1, 2]中的两种系统化思维结果), 随后是对各部分的详细解释。

表 1 数据库课程中的六类计算原理表现形式

计算原理 ^[3]	通信	计算	记忆	协作	评估	设计
计算原理在数据库中的具体表现	I/O 通信, 嵌入式 SQL, 网络通信	二分查找, 大数据背景中精度与效率之间的反转	E-R 模型, 持久存储, 索引	事务并发, 封锁协议, 多粒度封锁协议, 分布式系统中数据节点间的协作	I/O 代价, 通信代价, 算法复杂度, 并发吞吐量, 平均响应时间, 资源利用率, 规范化设计理论, 持久存储	概念结构设计, 逻辑结构设计, 物理结构设计, 数据的安全性及完整性, 三级模式及二级映像
数据库中数据的特点 ^[1,4]	大规模, 持久存储	大规模	持久存储, 有组织	共享, 大规模	大规模, 共享, 有组织, 持久存储	大规模, 有组织, 共享
思维模式 ^[2]	系统思维	系统思维, 底线思维	简约思维, 底线思维	系统思维	系统思维, 底线思维, 等价变换	系统思维, 分层思维

2. 1 通信原理

通信原理要求在不同位置之间能够可靠地传输信息^[3]。对于数据库原理与技术, 为了实现持久存储, 我们需要把数据存储于非易失性存储器, 如磁盘等; 而计算数据需要在内存中进行。所以, 这就需要在内存和外存之间反复进行 I/O 操作。为了解决内存和外存访问数据时速度存在数量级差异的问题, 我们需要减少频繁的 I/O 操作: 数据以块的形式进行划分, 并以块为单位, 整体读入 (由外存到内存) 或者读出 (由内存到外存) 数据, 同时也采用了缓冲技术。

对于混合编程 (嵌入式 SQL) 问题。SQL 语言除了可以单独使用外, 也可以嵌入到主语言 (如 C 语言)

进行混合编程。期间, 为了对两种不同的语言进行协调, 这需要借助中介 (主语言变量) 对数据进行必要的处理, 以完成集合式数据和单个数据之间的对接或双向转换。

对于分布式数据库, 完成数据的增、删、改、查等基本操作, 需要在多个数据节点之间进行通信与协作。这些数据节点往往物理地分布于不同的地点, 所以, 此时需要借助通信网络, 完成相应的操作^[6]。

上述操作都具有双向性, 这是通信的一大特点。同时, 也都有一个中间部件, 可以是缓冲区, 主语言变量, 也可以是网络等。

2.2 计算原理

计算即可计算性, 这涉及到能行性和实用性^[3,7]。有时, 理论上可以处理的问题, 实际上未必可行。如, 旅行商问题, 它是一个 NP 类问题, 其时间复杂度达到指数级^[8]。理论上, 只要问题规模有限, 这个复杂度就有上限, 总是可以完成的。但实际处理时, 往往是一个带约束的最优化问题: 人们不可能等到地老天荒! 所以, 至少需要以人的生命为上限, 对其实用性进行度量。为此, 往往需要退而求其次, 计算满意解。

对于数据库而言, 巨大的数据量成为整个应用系统性能的核心所在。此时, 线性存储方式已经不能满足数据量快速增长的现实需求。具有层次结构的存储模式应运而生, 在数据有序的情况下, 这有助于实现快速的二分查找。

另外, 数据库中的数据之所以重要, 是因为其中有不可估量的价值。其中信息的快速获得, 催生了数据挖掘。数据库挖掘中存在大量算法, 都期望从大量数据中高效产生有价值的信息。这自然也会用到数据结构课程中的算法时间复杂度, 以此来度量可计算性或者计算效率。值得注意的是, 数据库往往涉及大数据。数据量的增长, 使得数据与处理(或者程序)之间的地位发生反转, 从少量数据时的以程序为中心, 变化成大规模数据中的以数据为中心。衡量算法性能的一对指标, 精度和效率, 也在此时发生变化, 从以精度为主, 转换为以效率为主, 以便能够快速分析出数据中的整体变化趋势, 进行及早决策^[9]。

可计算性还表现在数学文献的理论基础上。关系型数据库中的操作涉及集合论, 关系代数和一阶谓词逻辑, 等等。并在其上构建了一目运算: 选择和投影, 二目运算: 集合并、集合差、集合乘(笛卡尔积、自然连接等)和集合除运算, 等等。这为 SQL 语言的有效执行奠定了理论基础。

总之, 计算原理从问题能够有效解决的角度, 说明处理大规模数据时关注焦点的变化, 等等。

2.3 记忆原理

记忆原理涉及信息的表示、存储和检索^[3]。对于关系型数据库, 最终的关系表, 不管来自 E-R 图, 还是规范化理论的分析结果, 关系表之间存在一定的冗余或者共有属性。在一个特定的数据库应用系统里, 任意两个关系表 X 和 Y (同时表示属性集), 从属性构成角度, 存在两种集合意义上的关系: $X \cap Y \neq \emptyset$ 和 $X \cap Y = \emptyset$ 。前者表示有共有属性, 后者是没有共有属性。同时, 不能存在一张表完全包含另一张表的情况, 即 $X \subseteq Y$ 或者 $X \supseteq Y$ 。但总会存在一条关系链, 把多张关系表通过两两之间有交集(共有属性)的方式, 整体关联在一起。例如, 高校信息管理系统中的学生表 S,

课表 C 和选修关系表 SC, 其中表 S 和表 C, 可以通过表 SC (SC 的外码分别参照表 S 和表 C 的主码) 进行关联, 从而可以检索特定学生(姓名)选修特定课程(课程名)的信息。这种数据的有组织性描绘了数据的内在联系, 并决定了具体的存储内容。

为了实现持久存储, 数据库引入了冗余技术, 即, 把数据库中的数据, 乃至整个系统进行备份, 存储在另外的物理空间中去。一般地, 为了确保安全, 应用系统还会采用具有独立故障模式的存储设备。

另外, 通过 B+树等索引方式, 可以实现对大量数据的快速定位和访问。

总之, 存储原理涉及数据的有效组织与持久存储, 以及索引等相关内容。

2.4 协作原理

协作是指有效利用多个自主的计算智能体^[3]。数据库中的数据是共享的。当多个用户(或者多台机器)同时访问特定数据库中的数据时, 难免会出现资源竞争。例如, 同时在线抢购某件热门商品, 或者消费券。在一定时间内, 需求人员(用户)和商品之间往往存在僧多粥少的情况。到底哪些人员能够获得, 哪些人员暂时不能获取, 这就需要专门的分配机制。首先, 多人同时操作时, 数据库系统引入了事务操作, 用于保证相互之间不会造成干扰(具体涉及事务的 ACID 特性), 使得每个用户或者机器终端有一种独享数据的错觉。其次, 还需要并发控制机制, 使得并发操作可串行化, 为此, 数据库中引入了加锁方法和相应的锁协议。当并发控制涉及的数据块大小不同时, 需要使用多粒度封锁协议, 等等。

对于分布式系统而言, 数据的各种操作, 常常需要联合多个数据节点, 并考虑在局部操作和全局操作之间取得平衡, 这也需要多节点之间的协作。

总之, 协作需要在多个对等的事物之间进行协调。在数据库中, 这些事物可以是事务, 也可以是分布式系统中的数据结点。

2.5 评估方法

评估方法用于度量系统能否达到预期的计算能力^[3]。当不需要访问数据时, 数据库中的数据存储于外部设备中。对数据进行查询、插入、删除和修改时, 则需要把数据调入内存。由于内外存处理数据速度的差异, 从而产生了查询数据时的特殊度量方式: IO 代价。这是数据库版的时间复杂度。查询是对数据进行插入、删除和修改的基础。于是, 有了代数优化和物理优化, 都期望能够更加高效地查询到相应的数据, 或者高效确认相应数据的是否存在。

对于分布式系统而言, I/O 代价只是其一, 我们还需要关注通信代价, 这也是制约分布式系统整体性能的一个关键因素。

另外, 在关系型数据中, 最终数据表的形成, 有两种方式, 一是通过对 E-R 图进行等价转换, 二是通过规范化理论进行模式分解。对于实际的数据库, 涉及到的属性往往不会少于 10 个。此时, 先分析得到 E-R 图, 再进行转换, 最后通过关系模式的规范化设计理论对当前的模式进行理论分析, 能够产生更有保障的一系列关系表。这里涉及到关系数据库模式的理论评估问题。良好的模式需要尽可能在表之间的减少冗余, 只保留必要的属性。规范化理论能够提供有力的模式合理性评估方法, 可以用于检验或者优化现有模式。

对于并发控制环节, 多个事务并发操作, 只要控制得当, 不但不会造成数据操作上的混乱, 还能够提高并发吞吐量和资源的利用率, 并能减少事务的平均响应时间。

另外, 数据库中的数据必须能够持久存储。基于冗余技术的持久存储, 能够保证数据极难被完全破坏, 这也涉及对数据的安全性评估。这里存在来自概率论的理论支撑: 多个备份数据间以独立的故障模式进行存储, 且不同备份之间互不影响时, 数据被彻底破坏的概率是多份数据同时遭受破坏的各自概率的乘积; 随着备份数量的增加, 该概率值会趋向于 0^[2]。

总之, 上述这些指标, 都可用于评估数据库特定方面的系统性能。

2.6 设计模式

设计则是通过构建有针对性的软件系统, 实现可靠性^[3]。数据库的价值在于设计开发针对特定应用的数据库管理系统。在开发此类系统时, 一般要经历可行性分析、需求分析、概念结构设计、逻辑结构设计、物理结构设计和系统实现及维护等几个步骤。这里包含三个设计阶段。在需求分析的基础上, 概念结构设计产生全局 E-R 图。进一步地, 逻辑结构设计按照一定原则将概念模式转换为某个具体 DBMS 支持的逻辑结构, 如关系数据库中的关系表。之后, 开发人员还需要设计数据库的内模式, 涉及文件结构、各种存取路径、存储空间分配、记录的存储格式等, 为逻辑数据结构选取合适的物理结构。

数据库应用系统的操作, 还涉及数据库的完整性和安全性相关内容。完整性包含实体完整性、参照完整性和用户完整性等。安全性包含用户标识与鉴别、存取控制、视图机制、审计和数据加密等内容。通过对数据添加访问时约束条件, 以及对数据库接触人员的授权控制, 能够设计出安全可靠的应用系统。

如果需要调整数据库中的某个模式时, 数据库可以通过三级模式及二级映像, 减少待调整模式对其他模式的干扰, 从而提高数据的逻辑独立性和物理独立性。

总之, 良好的设计能够带来系统的可靠性。在数据库中, 设计表现在应用系统的开发过程, 安全性与完整性等相关内容。

3 计算原理之间的内在关系

数据库中展现的这六类计算原理: 通信、计算、记忆、协作、评估和设计, 两两之间都存在一些关联。分析如下:

通信与计算: 通信涉及不同语言间的混合编程。关系型数据库中的 SQL 语言能够完成数据的常见操作, 同时也能够与嵌入的主语言进行通信, 完成一些更加细致的数据处理。

通信与记忆: 数据库中的数据, 存储在外存中, 需要时再调入内存。为了便于在内外存之间完成数据交换, 往往以数据块, 或者页的形式进行模块化处理。这些模块在外存中的存储与查找, 形成了 I/O 通信的基础。

通信与协作: 通信往往发生在不同的有自主能力的智能体之间。要想完成稍稍复杂的操作, 协作必不可少。

通信与评估: 分布式系统需要通过控制节点 (Control Node) 在多个数据节点之间进行数据处理^[6]。性能的好坏, 可以用通信代价进行评估。另外, I/O 操作中数据块的多少, 用于衡量查询算法的性能。

通信与设计: 例如, 安全性设计中, 为了鉴别出合法用户, 常用的方式是用户标识与鉴别, 即用户名加密码, 这种信息须在用户和系统之间进行安全传输。

计算与记忆: 数据的有效存储, 是对数据进行计算分析的基础。层次存储更有助于快速查找数据库中的数据。

计算与协作: 并发控制中, 多个事务之间需要通过封锁协议等完成协作。分布式系统中, 数据的处理往往需要在多个节点之间进行协调处理。

计算与评估: 计算的性能, 可以通过对应的指标进行评估。例如, 基于 I/O 的时间复杂度, 单位时间内的吞吐量, 等等。

计算与设计: 良好的设计, 有助于获得高效的计算性能。例如, 在关系型数据库中, 概念设计产生的 E-R 图和相应的逻辑设计阶段得到的关系表, 能够完

成对复杂数据的有效建模，从而为高效的数据操作奠定基础。

记忆与协作：信息以分布式进行存储时，对数据的处理需要在多个数据节点进行协作完成。对于持久存储问题，如果当前的数据库（系统）被破坏，可以启用备份数据库（系统）。这也是一种交接班式的协作模式，从而保证系统的稳定运行。

记忆与评估：数据的存储模式决定了查询数据时的效率。同时，如 2.5 节分析所示，基于冗余的持久存储技术，能够保证有效降低数据被彻底破坏的可能性。另外，关系模式的规范化设计理论可以对 E-R 模型产生的模式进行评估和优化。

记忆与设计：持久存储通过冗余技术，达到数据难以被毁坏的效果。这本身也是一种有效的安全设计方式。

协作与评估：多个事务之间协作，需要采用封锁协议。同时，协作的好坏，可以通过平均响应时间，单位时间内的吞吐量等进行评估。

协作与设计：与记忆与协作之间的关系类似，当数据以分布式的设计方式进行数据存储时，数据的处理需要在多个数据节点进行协作完成。另外，实现持久存储的冗余设计，也催生了当前的数据库（系统）和备份数据库（系统）之间的交接班式的协作模式。

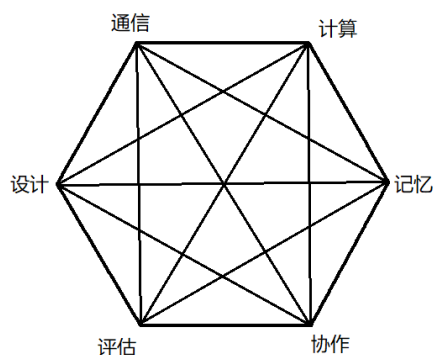


图 2 数据库中的六类计算原理及其内在关系

评估与设计：在物理结构设计中，数据在外存中的存储方式，直接决定了进行数据读取时的效率。对于数据库应用系统中的大量数据，顺序存储会带来低效的数据读取方式。而 B 树，B+树，hash 方法等等，访问数据时会更加高效和便捷。另外，能够保证持久存储的冗余技术，可以归类为设计。这些都是评估关注的问题。

通过上述分析，我们发现：通信、计算、记忆、协作、评估和设计这六类计算原理，在数据库中都有

所体现；同时，两两之间都会直接关联，从而形成六边形的全联通图案，如图 2 所示。

4 测评设计及效果分析

数据库教学过程通过融入第 3 节的内容，能够提升学生对数据库更加深入和系统的认知水平。本节从测评设计和效果方面进一步探讨相关内容。

4.1 测评设计

从第 3 节的分析可知，六类计算原理在数据库中得到充分体现，也在一定程度展现了章节之间的内在联系。在课程平时测试和最终的考试环节，数据库课程组专门设计了一些跨章节的试题，以考查学生对数据库的掌握水平。以下从六个角度分别给出了若干值得深入思考的问题。

通信方面：数据通信的双方具有什么特点？数据库中的通信体现在什么方面？

计算方面：数据库往往面临大规模的数据，如何提升数据的查询效率？有何代价？对于大规模复杂问题和简单的数据问题，从计算的角度分析，在处理效果上有何不同？

记忆方面：数据库中的数据为什么要借助外存进行存储？如何防止小概率事件的发生？这体现了处理问题时的什么思维模式？

协作方面：数据库中的数据往往是一种共享资源。共享时如果控制不当，会出现什么问题？如何解决？这对日常工作有何启示？

评估方面：数据库中的计算复杂度用什么度量？与数据结构课程中的度量方式有何不同？并发操作有什么优势？如何衡量 E-R 模型的优劣？

设计方面：数据库应用系统的开发，需要经历几个阶段，试说明各阶段的工作。数据库面临复杂工程问题，试说明数据库如何拆解和处理复杂问题。

4.2 效果分析

20 世纪 60 年代，针对学习问题，美国心理学家布鲁姆提出了一种多层次认知理论，由浅入深依次是：记忆、理解、应用、分析、评价和创造^[10]。4.1 节中的问题在一定程度上都需要在深入理解数据库理论课程的基础上才能作答。这正好能够考查学生在记忆和理解基础上的更高的学习能力。为了测试效果，针对 4.1 节中的问题，课程组对 2023 级约 500 名学生（五个教学班级，每班约 100 名）进行了测评。测评方式如下：把这 500 名学生分成两组，对其中两个班级（对照组）的学生直接进行测试，另外三个班级（提升组）在引

导和解读后进行测试。测试的平均准确度（得分）如下（每个方面满分都为 10 分）：

表 2 测评效果数据对比

组别	通信	计算	记忆	协作	评估	设计
对照组	5.2	6.5	6.0	5.3	6.3	5.1
提升组	7.3	9.2	9.1	7.5	8.2	8.2

表 2 给出了两组平均的测评数据，从中可以看出明确引入六类计算原理后的学习提升效果。对于每一个计算原理，提升组的平均得分都显著高于对照组得分，分别提升 40.4%，43.1%，51.7%，41.5%，30.2%和 60.8%，总体上平均提升 44.6%。

总之，通过对六种计算原理的引入和分析，能够加深学生对数据库的理解和运用能力，这种深层的分析有助于学生理解数据库的底层逻辑，从而提升创新水平。当然，这种测评的科学性也值得进一步跟踪、探讨和研究。

5 结束语

本文从六类计算原理的内在视角重新梳理了数据库原理与技术课程中的重要内容，给出了数据库课程的一种新的系统化认知方式，并从外在表现方面探讨了不同视角之间的联系。六类计算原理对教学的提升效果也得到了初步验证。本文内容有助于教育工作者培养和提高学生对数据库应用系统的分析、设计和开发能力。计算原理视角下对数据库的认知分析和实践，有待进一步完善。

参考文献

[1] 朱真峰, 田侦. 基于思维导图的数据库原理课程总体认知分析[J]. 计算机教育, 2020, 1(4):92-97

[2] 职为梅, 朱真峰. 数据库原理中蕴含的临界知识及其应用[J]. 计算机技术与教育学报, 2024, 12(2):10-16

[3] 彼得 J. 丹宁, 雷格 H. 马特尔 著, 罗英伟, 高良才, 张伟, 熊瑞勤 译. 伟大的计算机原理[M]. 北京: 机械工业出版社, 2018

[4] 范明, 叶阳东, 邱保志, 职为梅等. 数据库原理教程[M]. 北京: 科学出版社, 2018

[5] Abraham Silberschatz, Henry Korth, Sudarshan. Database System Concepts[M]. New York: McGraw Hill, 2009

[6] 李海翔. 分布式数据库原理、架构与实践[M]. 北京: 机械工业出版社, 2021

[7] 朱真峰. 迭代算法的能行性及实用性[J]. 郑州师范教育, 2015, 4(2): 23-25

[8] Bock S, Bomsdorf S, Boysen N, et al. A Survey on the Traveling Salesman Problem and its Variants in a Warehousing Context[J]. European Journal of Operational Research, 2024, 322(1):1-14

[9] 维克托·迈尔-舍恩伯格, 肯尼思·库克耶 著, 周涛等 译. 大数据时代 生活、工作与思维的大变革[M]. 杭州: 浙江人民出版社, 2013

[10] Benjamin S Bloom, Engelhart Max D, et al. Taxonomy of Educational Objectives: The Classification of Educational Goals, Handbook I: Cognitive domain[M]. New York: David McKay Company, 1956.