

# 关系数据库设计原理与分析

温立辉

(河源职业技术学院 电子与信息工程学院, 广东 河源 517000)

**摘要:** 文章针对数据层存储方案设计问题, 讨论了软件开发生命周期中几大阶段与数据存储设计的关系及每阶段的内容、成果输出, 最后, 举例分析了数据存储方案设计过程中需要遵守的重要准则以及核心要点。

**关键词:** 实体模型; E-R关系; 概念设计; 范式原则

数据库设计涉及信息系统的性能、效率、安全等方面, 好的设计方案能极大地减轻实施阶段的工作量, 减少运维的成本与投入。数据库设计需要从数据模型、负载、数据存储量、并发数等方面着手考虑, 需要遵守相关的原则与规范<sup>[1]</sup>。

## 1 数据模型

数据模型是数据存储方案设计过程中首要考虑的问题, 模型是对客观事物的模拟描述, 模型既是了解事物本质的需要, 也是与不同角色人员间沟通讨论解决方案的重要工具。数据建模作为数据存储设计的第一步, 其输出结果就是得到符合实际需求的数据模型, 再进而得到可实施的执行方案。在理论上来说, 一个完整数据模型包括结构、操作、约束3个方面<sup>[2]</sup>。

(1) 结构: 是指数据存储结构, 如模型中的存储项以及每个项的类型, 以及存储结构所占据的存储空间的大小及存储空间所在位置区域等方面。

(2) 操作: 是指外部对存储结构的读或写方面的操作, 或者增、删、改、查的数据交互过程。

(3) 约束: 是指数据存储结构内部数据项的限制条件, 如数据项各种类型的长度、大小, 不同存储结构间数据项的依赖关系等。

在数据存储领域, 存储结构中存在众多的数据存储模型, 其中相对成熟的数据模型有如下几种<sup>[3]</sup>。

(1) 层次模型: 也叫树状模型, 模型中有一个根节点, 然后在这个节点下生成众多的叶子分支。

(2) 网状模型: 模型中的实体节点纵横交错的发生关联关系, 实体之间没有主次、子父之分, 呈现一个板平的网状。

(3) 关系模型: 以平面表为基础, 表中有行与列, 其中列定义的是存储结构的数据项, 行代表的是存储结构中的数据记录, 是目前关系数据库中的普遍形式。

## 2 数据库设计过程

数据存储方案设计是软件开发生命周期的概要设计阶段必须考虑的问题, 从应用程序的运行数据持久化需求着手到最终数据库环境的建立, 大概包含以下几个方面的过程<sup>[4]</sup>, 每个方面的工作内容及成果输出如图1所示。

(1) 概念定义: 是数据存储方案定义的初始阶段, 在这一过程主要是对客户业务功能数据需求的归纳, 得到业务的

概念模型。概念模型是一个对客观事物抽象表述且与技术无关的模型, 是设计人员与客户交流模型工具。通过概念模型可以清楚地展示数据层的存储功能。

(2) 逻辑定义: 是对概念模型的实现方案的定义, 是数据存储层的一种实现思想, 定义存储模型中的相关内在实现, 对应的成果输出是逻辑模型。

(3) 物理定义: 是根据数据存储方案的逻辑实现思想定义出具体的数据库存储环境, 包括可运行数据库运行脚本、表空间的分配、触发器、存储过程、视图等方面的建立。

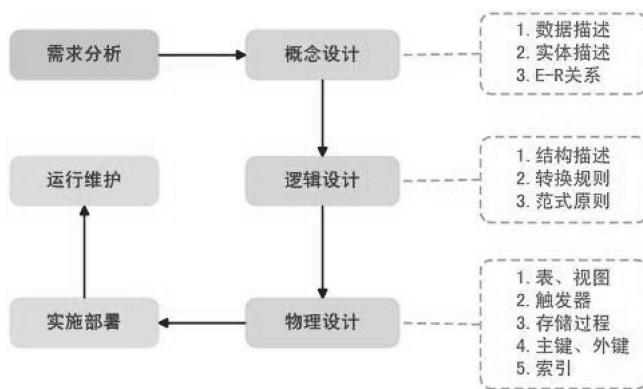


图1 各阶段内容及成果

## 3 范式原则

关系数据库范式原则是进行数据存储设计的一个重要准则, 范式设计原则将直接指向数据库中表的字段定义、字段关系、记录唯一性标识是否合理。当前在关系数据库系统中存在6种范式原则, 其中基础的原则有3种, 分别是: 第一范式(1NF)、第二范式(2NF)、第三范式(3NF)<sup>[5]</sup>, 如图2所示。

(1) 1NF: 要求数据表中字段不可再分, 且字段只能单值存储。这是最基本的范式原则, 所有数据表都必须满足这一条, 否则关系数据库将不支持所设计的数据模型。如果一个“电话”字段可以拆分成“手机”与“座机”两个属性, 或者在同一条记录中“电话”属性字段存储了两个值, 都是不符合该范式的设计要求<sup>[6]</sup>。

(2) 2NF: 要求每个数据实体中必须有唯一标识, 如果唯一标识为联合标识, 则其他普通属性不能存在对联合标识的部分依赖。本范式解决的是数据表中的唯一标识问题,

及非主属性对码的部分依赖。如一个表中有：“开户人”“开户银行”“开户人联系电话”“开户行地址”“帐户余额”等字段，其中“开户人”“开户银行”为联合标识（联合主键），因普通属性“开户人联系电话”却依赖“开户人”属性，“开户行地址”属性依赖“开户行”属性，因而存在普通属性对联合标识的部分依赖，不符合该要求。

(3) 3NF: 要求数据实体的普通属性间不能存在传递依赖，本范式所要解决的是普通属性的互相依赖问题。如一个表中有如下属性：“职工号”“职工姓名”“所属部门”“部门办公电话”“部门经理”因为普通属性“部门办公电话”“部门经理”“所属部门”三者间存在依赖关系，“所属部门”属性可直接决定“部门办公电话”与“部门经理”，因而数据表该不符合该要求<sup>[7]</sup>。

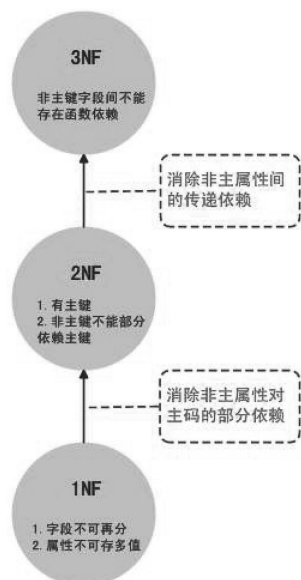


图2 范式设计原则

#### 4 性能与理论范式平衡点

数据库领域有两种岗位，一种是数据库管理员岗位

(Database Administrator, DBA)，另一种是数据库编程开发人员岗位(Database Programming, DBP)。作为数据库管理员，希望所有的存储方案能尽可能是范式原则理想状态下最佳实现，以便以日后数据维护方便、清晰；而作为一名数据库开发人员，在为应用系统设计存储方案时（数据库设计方案），总是希望尽可能地提高存储方案性能与效率，但在提高性能与效率的同时，总是不可避免地要以违反理论范式为代价。就如编程界在提高应用程序实时并发数量级与维护数据准确性一样，程序实时并发数量级提高了，数据的准确度就得不到保障，若要提升数据的精准度，则不得不降低应用系统的实时并发数，究竟应用程序实时并发数量级与数据精准度，谁轻谁重要，在业界没有一个统一的标准，但有一个重要原则就是根据实际需求来取舍两者的权重。

同样的道理在进行数据数据库存储方案设计时，并不能一概而论，是提升存储方案的性能效率重要，还是保证理论范式原则完美、完整性重要；一般来说，在进行表实体设计时，为了提升性能，避免频繁的表联结操作，一般会考虑在主要业务实体中适当冗余其他次要业务的关键信息，以降低表联结操作对系统性能所造成的额外开销。这虽然在一定程度上破坏了实体的范式原则，却是允许和可接受的，特别是在大数据业务实体中，进行响应优化时特别重要；当然为了保证业务实体的科学合、理性，也不能完全抛开范式原理，一味追求性能，从而造成实体过分冗余而导致存储方案变得非常糟糕而难以维护。

#### 5 结语

数据存储方案是信息系统开发必不可少的环节，方案优势将会成为系统开发周期中的一个风险点，合理的存储方案将极大地降低开发过程中的风险，反之则会极大地提升系统开发风险。数据存储设计过程中既要考虑客户的需求，同时也要考虑相关的理论原则，只有两者相结合，才能提供合理、合格的存储方案。

#### [参考文献]

- [1]姚万鹏.软件开发中数据库设计理论的实践[J].电子技术与软件工程, 2016(18): 201.
- [2]黄辉.浅谈数据库设计与安全性策略[J].西部皮革, 2016(18): 69.
- [3]王庆利, 王璐.计算机软件数据库设计的原则及应用探究[J].数字技术与应用, 2016(5): 187.
- [4]温立辉.数据库高级应用技术[M].北京: 北京理工大学出版社, 2016.
- [5]王延武.浅谈数据库设计方法[J].山东工业技术, 2016(18): 139.
- [6]刘霜.数据库设计对SQL Server数据库性能优化分析[J].中国新通信, 2016(15): 3-4.
- [7]姚万鹏.软件开发中数据库设计理论的实践[J].电子技术与软件工程, 2016(18): 201.

## Principles and analysis of relational database design

Wen Lihui

(Electronics and Information Engineering Department of Heyuan Polytechnic, Heyuan 517000, China)

**Abstract:** Aiming at the design problem of a relational database table entity, this paper discusses the relationship between several major stages of software development life cycle and data storage design and the output of each stage of content. Finally, an example is given to analyze the important rules to be observed in the design of data storage scheme and the core points.

**Key words:** entity model; E-R relation; conceptual design; paradigm principle