数据库设计与性能优化

傅仁毅　著

华中科技大学出版社
中国·武汉

内 容 提 要

本书介绍了数据库基本知识、数据库设计的规范化问题、用 PowerDesigner 设计数据库的过程及数据库性能的基本原则、步骤和常规方法，还详细介绍了 SQL Server 2005 及 Oracle 10G 的优化过程。

本书是作者多年从事数据库系统工作，特别是在数据库性能优化方面的经验之作。本书既介绍了一般性能优化经验，又介绍了最常用的两个数据库系统性能优化的过程；既有实用的数据库技术及数据库设计工具，又有作者在多个大型 ERP 系统的性能优化过程中的经验。读者可直接把相关问题的解决方案应用到设计与性能优化过程中，在实际数据库设计与性能优化过程中得到帮助。

图书在版编目(CIP)数据

数据库设计与性能优化/傅仁毅　著.—武汉：华中科技大学出版社，2010.9
ISBN 978-7-5609-6545-1

Ⅰ.J…　Ⅱ.①桂…　②李…　Ⅲ.JAVA 语言-程序设计　Ⅳ.TP312

中国版本图书馆 CIP 数据核字(2010)第 172751 号

数据库设计与性能优化　　傅仁毅　著

策划编辑：谢燕群
责任编辑：朱建丽
封面设计：潘　群
责任校对：李　琴
责任监印：熊庆玉
出版发行：华中科技大学出版社(中国·武汉)
　　武昌喻家山　　邮编：430074　　电话：(027)87557437
录　　排：华中科技大学惠友文印中心
印　　刷：武汉市新华印刷有限责任公司
开　　本：710mm×1000mm　1/16
印　　张：25
字　　数：515 千字
版　　次：2011 年 5 月第 1 版第 1 次印刷
定　　价：39.80 元



本书若有印装质量问题，请向出版社营销中心调换
全国免费服务热线：400-6679-118　竭诚为您服务
版权所有　侵权必究

前　言

目前，计算机系统的运行速度越来越快，内存和硬盘等存储设备的容量越来越大，为大型应用系统提供了硬件基础。数据库技术是计算机领域发展最快、应用最广、影响最大的几项技术之一，在生活和工作中起着越来越重要的作用。

优秀的数据库设计是高效的应用系统的根本，而在多年的数据积累后，如何保证应用系统的良好性能又是至关重要的。在数据库设计及数据库性能优化方面，多数书籍介绍的技术陈旧，内容脱离实际、理论偏多，对读者的实际工作帮助不大。本书讲授了实用的数据库技术及数据库设计工具，并介绍了作者在多个大型ERP系统的性能优化过程中的经验。其中，既有一般性能优化经验，又有最常用的两个数据库系统性能优化的过程。读者可直接把书中相关问题的解决方案应用到设计与性能优化过程中。

全书共分9章，第1章介绍数据库基本概念、数据库管理系统、数据模型及数据库设计；第2章介绍关系数据库，包括关系数据模型、关键字、关系运算、结构化查询语言、数据定义、查询、数据更新、视图、数据安全性；第3章介绍数据库规范化，即如何设计一个好的数据库；第4章~第6章分别介绍使用PowerDesigner进行概念数据模型、逻辑数据模型及物理数据模型的设计，并最终生成到具体的数据库系统中；第7章介绍数据库优化概述及常规方法；第8章介绍SQL Server 2005的优化，包括其优化设计、分割与并行、索引与SQL代码的优化、执行计划的重复利用等；第9章介绍Oracle 10G的优化，包括环境优化、初始参数优化、代码优化及资源竞争优化等。

希望本书能为读者在实际数据库设计与性能优化过程中提供参考与帮助。

作　者

2010年12月

目　录

第 1 章　数据库设计的相关概念……(1)
1.1　什么是数据库……(1)
1.2　数据库管理系统……(2)
1.3　数据模型……(6)
1.4　数据库设计……(6)
第 2 章　关系数据库……(10)
2.1　关系数据模型……(10)
2.2　关键字……(10)
2.3　关系运算……(11)
2.4　结构化查询语言……(12)
2.5　数据定义……(12)
2.6　查询……(15)
2.7　数据更新……(20)
2.8　视图……(22)
2.9　数据安全性……(23)
第 3 章　数据库规范化……(25)
3.1　规范化的提出……(25)
3.2　规范化的含义……(27)
3.3　第一范式……(28)
3.4　第二范式……(29)
3.5　第三范式……(30)
3.6　BCNF……(31)
3.7　第四范式……(31)
3.8　第五范式……(33)
3.9　规范化步骤……(33)
第 4 章　概念数据模型设计……(35)
4.1　实体关系模型的基本要素……(35)
4.2　新建概念数据模型……(36)
4.3　创建新实体……(38)
4.4　添加实体属性……(39)

4.5 确定实体标识符 ..... (41)
4.6 新建关系 ..... (42)
4.7 新建业务规则 ..... (46)
4.8 新建数据项 ..... (49)
4.9 新建域 ..... (51)
4.10 模型检查 ..... (53)
4.11 概念数据模型到物理数据模型的转换 ..... (54)
第 5 章 逻辑数据模型设计 ..... (60)
5.1 新建逻辑数据模型 ..... (60)
5.2 新建实体 ..... (61)
5.3 新建标识符 ..... (64)
5.4 新建关系 ..... (65)
5.5 新建业务规则 ..... (67)
5.6 新建域 ..... (69)
5.7 模型检查 ..... (71)
5.8 逻辑数据模型到物理数据模型的转换 ..... (71)
第 6 章 物理数据模型设计 ..... (75)
6.1 新建物理数据模型 ..... (75)
6.2 新建表 ..... (76)
6.3 新建外键/引用 ..... (80)
6.4 新建域 ..... (83)
6.5 新建业务规则 ..... (87)
6.6 新建索引 ..... (90)
6.7 新建视图 ..... (92)
6.8 新建存储过程 ..... (95)
6.9 新建触发器 ..... (97)
6.10 创建数据库脚本 ..... (99)
6.11 不同数据库脚本的生成 ..... (108)
6.12 生成数据库测试数据 ..... (111)
第 7 章 数据库优化概述及常规方法 ..... (117)
7.1 引言 ..... (117)
7.2 数据库性能的基本要素 ..... (117)
7.3 数据库优化的基本原则 ..... (118)
7.4 数据库优化的基本步骤 ..... (119)
7.5 数据库优化的建议 ..... (120)

7.6 锁与并发控制的优化考虑 ……(121)
7.7 日志的优化考虑 ……(123)
7.8 操作系统的优化考虑 ……(123)
7.9 硬件优化的考虑 ……(124)
第 8 章 SQL Server 2005 的优化 ……(125)
8.1 SQL Server 2005 系统架构 ……(125)
8.2 SQL Server 2005 的数据库设计 ……(126)
8.3 SQL Server 2005 数据表分割与并行运行 ……(128)
8.4 tempdb 系统数据库的优化考虑 ……(130)
8.5 索引的优化考虑 ……(131)
8.6 SQL 代码的优化 ……(153)
8.7 执行计划的重复利用 ……(166)
8.8 事务与锁的优化考虑 ……(172)
第 9 章 Oracle 10G 的优化 ……(178)
9.1 Oracle 系统架构 ……(178)
9.2 Oracle 环境优化 ……(181)
9.3 Oracle 初始参数优化 ……(183)
9.4 SQL 代码优化 ……(189)
9.5 资源竞争优化 ……(207)
参考文献 ……(218)

第 1 章　数据库设计的相关概念

1.1　什么是数据库

简单地说，数据库（Database）是按照数据结构来组织、存储和管理数据的“仓库”。J. Martin 对数据库进行了比较完整的定义：数据库是存储在一起的相关数据的集合，这些数据是结构化的，已去掉有害的或不必要的冗余数据，并为多种应用服务的数据；数据的存储独立于使用它的程序；对数据库插入新数据，修改和检索原有数据均能按一种公用的和可控制的方式进行。当某个系统中存在结构上完全分开的若干数据库时，则该系统包含一个“数据库集合”。

从上述定义可以看出，数据库是存储在一起的相关数据的集合。在这个数据集合中，已去掉有害的或不必要的冗余数据，这种结构化的集合即为数据库本身，也是数据库系统的核心和管理对象。

图 1-1 中的数据集包含冗余数据，在订单数据表中，包括每一个客户的客户地址，这个地址为冗余数据。实际上，由订单数据表中的客户 ID 字段即可知道客户地址。应尽量消除此类冗余数据，冗余数据不但浪费存储空间，需要更多的数据录入及维护的人力成本，还会浪费计算机的计算资源；更为严重的是，冗余数据会导致数据的不一致。如图 1-1 所示的订单数据表中，如果某客户地址改变了，就得保证此表中涉及的该客户的每条记录中的客户地址字段都要做相应的修改，否则，有可能出现同一客户具有不同地址的情况，这种情况一般是不允许的。

客户信息表

客户 ID	客户名	客户地址
123	王华	学院路 100 号
456	李国	科技路 20 号
789	张兴	振兴路 30 号

订单数据表

订单号	产品号	客户 ID	客户地址
1001	112	123	学院路 100 号
1003	102	456	科技路 20 号
1005	131	123	学院路 100 号

图 1-1　包含冗余数据的数据表

有时，由于应用系统建设的时间不同，订单系统与财务系统可能分别是建立在不同的计算机系统中的不同数据库，如果两个系统完全独立，则会出现冗余数据，需要更多的人力成本及措施来处理它们之间数据的不一致性，对两个系统进行整合并最终共享同一数据库，是较好解决这一问题的办法。

随着时间的推移，业务条件与业务需求会有所变化，此时，数据库的管理者希望数据库能够做出相应变化，并且，应用程序不必因为数据库的变化而修改，即应用程序与数据库之间不存在相互依赖的关系，这就是数据的独立性。数据的独立性有两个层面的含义：物理独立性与逻辑独立性。物理独立性是指数据库的物理设备更换、物理位置的改变等物理结构的变化，不影响数据库的逻辑结构，不影响应用程序，不会导致应用程序的修改。逻辑独立性是指数据定义的修改、数据间联系的变更等数据库逻辑结构的改变，不会导致应用程序的修改。数据的独立性依赖于数据库管理系统来实现，目的是使应用程序尽可能不受数据的影响。

总之，在数据库的设计伊始，就应努力消除冗余数据，保证数据的一致性与独立性，尽可能整合数据以达到共享的目的。

1.2 数据库管理系统

数据库管理系统(Database Management System，DBMS)是一种操纵和管理数据库的软件系统，用于建立、使用和维护数据库。DBMS 负责解释并完成用户或程序对数据库提出的各种请求，负责对数据库的维护并保证数据的完整性及一致性。

(1) 数据定义语言(Data Definition Language，DDL)：DBMS 提供相应 DDL 来定义数据库结构，刻画数据库框架，并保存在数据字典（Data Dictionary）中，数据字典是一类特殊的数据库，称为元数据（Metadata），实际上是定义数据的数据。结构化查询语言(the Structured Query Language，SQL)是大部分 DBMS 中最常用的一种 DDL。

(2) 数据操作语言（Data Manipulate Language）：DBMS 提供数据操作语言，供用户实现对数据库的基本操作，如检索、插入、修改和删除。

(3) 数据库运行管理：DBMS 为保障数据的安全性、完整性和并发控制等特性，对数据库运行进行有效的控制和管理，确保数据正确有效。如在多用户环境下的并发控制、安全性检查和存取限制控制、完整性检查和执行、运行日志的管理、事务的管理和自动恢复等，这些功能保证了数据库系统的正常运行。

(4) 数据组织与存储：DBMS 分类组织、存储和管理各种数据，包括数据字典、用户数据、存取路径等，确定文件结构和存取方式在物理上组织这些数据，实现数据之间的联系，以提高存储空间利用率及存取效率。

(5) 数据库维护：数据库初始数据的装入，数据库的转存、恢复、重组织，系统性能监视、分析。

(6) 数据库传输：DBMS 与操作系统、网络中其他软件系统及数据库之间的协调，提供处理数据的传输，实现用户程序与 DBMS 之间的通信。

目前常用的数据库管理系统有以下几种。

1. Oracle

Oracle 是一个功能强大、面向对象的关系型数据库系统，支持 Internet 应用。在管理信息系统、企业数据处理、因特网及电子商务等领域有着广泛的应用。由于 Oracle 在数据安全性与数据完整性控制方面的优越性，以及跨操作系统、跨硬件平台的数据互操作能力，使得许多用户将 Oracle 作为其应用系统的数据处理系统。Oracle 不仅具有完整的数据管理功能，还提供了一套界面友好、功能齐全的数据库开发工具。Oracle 使用 PL/SQL 语言执行各种操作，支持面向对象的功能，具有开放性、可移植性、可伸缩性等功能。目前，最新版 Oracle 11G 的自动 SQL 优化功能可以让 Oracle 自动将 3 倍于原有性能的 profile 应用到 SQL 语句上。此外，Oracle 11G 的自动内存优化功能，使得所有内存可以通过仅设定一个参数来实现全表自动优化。只要告知有多少内存可用，Oracle 就可以自动指定多少内存分配给 PGA、多少内存分配给 SGA 和多少内存分配给操作系统进程。

2. SQL Server

SQL Server 是一个开放式的关系型数据库管理系统，具有友好的交互界面，使用 T-SQL 语言完成数据操作。SQL Server 2008 具有可靠性、可伸缩性、可用性、可管理性等特点，为用户提供完整的数据库解决方案；可以对整个数据库、数据文件和日志文件进行加密，而不需要改动应用程序；可以审查数据操作，审查不只包括对数据修改的所有信息，还包括关于什么时候对数据进行读取的信息，从而提高了可靠性和安全性。

3. DB2

DB2 数据库系统采用多进程、多线程体系结构，可以运行于多种操作系统之上。DB2 提供了高层次的数据利用性、完整性、安全性、可恢复性，以及从小规模到大规模应用程序的执行能力，具有与平台无关的基本功能和 SQL 命令。DB2 还以拥有一个非常完备的查询优化器而著称，其外部连接改善了查询性能，并支持多任务并行查询。DB2 具有很好的网络支持能力，每个子系统可以连接十几万个分布式用户，可同时激活上千个活动线程，对大型分布式应用系统尤为适用。DB2 支持面向对象的编程，支持复杂的数据结构，如无结构文本对象，可以对无结构文本对象进行布尔匹配、最接近匹配和任意匹配等搜索。此外，DB2 支持多媒体应用程序，支持 BLOB 对象，允许在数据库中存取二进制大对象和文本大对象。DB2 9 作为最新版产品，率先实现了可扩展标记语言（XML）和关系数据的无缝交互，而无需考虑数据的格式、平台或位置。

4. MySQL

MySQL 是一个小型关系型数据库管理系统，体积小、速度快，总体拥有成本低，尤其是其开放源码的特点，使得许多中小型网站为了降低网站总体拥有成本而选择 MySQL 作为网站数据库。MySQL 使用 C 和 C++编写，并使用了多种编译器进行测试，保证了源代码的可移植性。此外，MySQL 为多种编程语言提供了 API，支持多线程，充分利用 CPU 资源，支持多用户；优化的 SQL 查询算法，有效地提高查询速度；MySQL 既能够作为一个单独的应用程序应用在客户端服务器网络环境中，也能够作为一个库嵌入到其他的软件中。MySQL 5.5 在 4 核以上的服务器中的性能、复制及表分区功能方面都有很大提升。

5. Sybase

Sybase 数据库系统在企业解决方案、Internet 应用、商务智能及移动与嵌入计算领域为客户提供了先进的技术，支持 Java、XML 及分布事务处理，支持服务器间的失败转移，通过组件集成服务，集成企业内分布异构的数据源。Sybase 数据库系统不用操作系统来管理进程，把与数据库的连接作为自己的一部分来管理。此外，Sybase 数据库的引擎还代替操作系统来管理一部分硬件资源，如端口、内存、硬盘等，绕过了操作系统这一环节，提高了性能。Sybase ASE 15 在可靠性、总体拥有成本和性能方面得到了显著提高，为在任务关键性环境中实现战略敏捷性及持续创新奠定了基础。同时，Sybase 还引入了网格和群集、服务导向型架构及实时消息等新技术。

6. Access

作为 Microsoft Office 组件之一的 Access，是在 Windows 环境下流行的桌面型数据库管理系统。使用 Access 时，无需编写任何代码，只需通过直观的可视化操作就可以完成大部分数据管理任务。Access 数据库包括组成数据库的基本要素，如存储信息的表、显示人机交互界面的窗体、有效检索数据的查询、信息输出载体的报表、提高应用效率的宏、功能强大的模块工具等要素。Access 不仅可以通过 ODBC 与其他数据库相连，实现数据交换和共享，还可以与 Word、Excel 等办公软件进行数据交换和共享，并且通过对象链接与嵌入技术在数据库中链接和嵌入声音、图像等多媒体数据。Access 2010 可以生成 Web 数据库并将它们发布到 SharePoint 网站。此外，用户界面也发生了变化；利用增强的安全功能及与 Windows SharePoint Services 的高度集成，可以进行更有效地管理，并让信息跟踪应用程序比以往更加安全。

选择数据库管理系统时应考虑以下几个方面。

(1) 了解数据库管理系统中的管理语句是否符合国际标准，符合国际标准则便于系统的维护、开发、移植；有没有面向用户的易用的开发工具；所支持的数据库容量是否满足未来几年业务发展的需要。

(2) 了解有无辅助设计工具，辅助设计工具可以帮助开发者根据软件工程方法提供各开发阶段的维护、编码环境，便于复杂软件的开发、维护；有无第四代语言的开

发平台，第四代语言具有非过程语言的设计方法，用户不需编写复杂的过程性代码，易学、易懂、易维护；有无面向对象的设计平台，面向对象的设计思想十分接近人类的逻辑思维方式，便于开发和维护；对多媒体数据类型是否支持，多媒体数据需求是今后发展的趋势，支持多媒体数据类型的数据库管理系统必将减少应用程序的开发和维护工作。

(3) 了解数据库管理系统的性能情况，包括性能评估（响应时间、数据单位时间吞吐量）、性能监控（内外存使用情况、系统输入/输出速率、SQL 语句的执行、数据库元组控制）和性能管理（参数设定与调整）。

(4) 了解对分布式应用的支持情况，特别是数据透明与网络透明程度。数据透明是指用户在应用中不需指出数据在网络中的什么节点上，数据库管理系统可以自动搜索网络，提取所需数据；网络透明是指用户在应用中无需指出网络所采用的协议，数据库管理系统自动将数据包转换成相应的协议数据。

(5) 了解并行处理能力，支持多 CPU 模式的系统（SMP、CLUSTER、MPP），负载的分配形式，并行处理的颗粒度、范围等。

(6) 了解可移植性和可扩展性情况。可移植性是指垂直扩展和水平扩展能力。垂直扩展要求新平台能够支持低版本的平台，数据库客户机/服务器支持集中式管理模式，以便保证用户以前的投资和系统；水平扩展要求满足硬件的扩展，支持从单 CPU 模式转换成多 CPU 并行机模式（SMP、CLUSTER、MPP）。

(7) 了解数据完整性约束情况。数据完整性是指数据的正确性和一致性保护，包括实体完整性、参照完整性、复杂的事务规则。

(8) 了解并发控制功能情况。对于分布式数据库管理系统，并发控制功能是必不可少的。因为它面临的是多任务分布环境，可能会有多个用户在同一时刻对同一数据进行读或写操作，为了保证数据的一致性，需要由数据库管理系统的并发控制功能来完成。并发控制至少应考虑：保证查询结果一致性方法，数据锁的颗粒度（数据锁的控制范围，表、页、元组等），数据锁的升级管理功能，死锁的检测和解决方法。

(9) 了解数据库管理系统的容错能力。异常情况下对数据的容错处理，硬件的容错，有无磁盘镜像处理功能软件的容错，有无软件方法异常情况的容错等。

(10) 了解安全性控制情况，包括安全保密的程度（账户管理、用户权限、网络安全控制、数据约束）等。

(11) 了解支持汉字处理能力，包括数据库描述语言的汉字处理能力（表名、域名、数据）和数据库开发工具对汉字的支持能力。

(12) 了解数据库恢复能力。当突然停电、出现硬件故障、软件失效、病毒或严重错误操作时，系统应提供恢复数据库的功能，如定期转存、恢复备份、回滚等，使系统有能力将数据库恢复到损坏以前的状态。

1.3 数据模型

数据库是模拟现实世界活动的信息集合，它所存储的数据，用来描述现实世界事物的特征及相互关系。在现实世界中，一个实际存在并且可以识别的事物称为个体，如一位学生。每个个体有其自身的特征，如学生的特征包括姓名、性别、年龄等。在信息世界中，每个个体称为实体（Entities），实体包括的特征用字段来表示。现实世界中的个体与个体之间存在着一些联系，反映到信息世界，是实体与实体之间的关系，由此构成了计算机世界中的数据模型(Data Models)。

简单地说，数据模型是数据结构的描述性表示。数据模型包括数据库数据的结构和操作、约束条件。数据模型中的数据结构描述数据的类型、内容、性质及数据间的关系，数据结构是数据模型的基础，数据操作和约束都建立在数据结构上；不同的数据结构具有不同的操作和约束。数据模型中，数据操作描述在相应的数据结构上的操作类型和操作方式；数据约束主要描述数据结构内数据间的语法、词义关系、制约和依存关系，以及数据动态变化的规则，以保证数据的正确、有效和相容。

数据模型按不同的应用层次可分成概念数据模型（Conceptual Data Model）、逻辑数据模型（Logical Data Model）及物理数据模型（Physical Data Model）三类。概念数据模型是面向用户的现实世界的模型，主要用来描述现实世界的概念化结构，使数据库设计人员在设计的初始阶段，摆脱计算机系统及 DBMS 的具体技术问题，集中精力分析数据与数据之间的联系等，用图的形式描述信息系统中的概念性结构，而与具体的数据库管理系统无关。概念数据模型需要转换成逻辑数据模型。逻辑数据模型是用户从数据库角度得到的模型，独立于具体的物理数据库的实现，帮助分析信息系统的结构。物理数据模型是面向计算机物理表示的模型，描述数据在存储介质上的组织结构，帮助分析数据库中的表、视图及其他数据库对象。

1.4 数据库设计

数据库设计（Database Design）是管理信息系统开发中的核心，是根据用户的需求，在一个给定的应用环境下，设计数据库的结构和建立数据库以满足用户信息要求和处理要求的过程。在数据库设计需求分析阶段，需要综合每个用户现实世界的应用需求。在概念设计阶段形成独立于机器、独立于各数据库系统产品的、以实体-关系图（Entity Relationship Diagram，简称 E-R 图）表示的概念模型。在逻辑设计阶段，将 E-R 图转换成具体的数据库系统支持的数据模型，并根据用户处理的要求及安全性考虑，在基本表的基础上建立必要的视图，形成数据库逻辑模型。在物理设计阶段，根据数据库系统特点和处理的需要，进行物理存储安排，设计索引，

生成物理数据库。

数据库设计阶段有如下步骤。

1. 需求分析

调查组织机构情况、各部门的业务活动情况，协助用户明确新系统的边界。采取跟班作业、开调查会、请专人介绍、询问、设计调查表请用户填写、查阅记录等方法充分了解原有系统（手工系统或信息系统）的工作情况，弄清所用数据的种类、范围、数量，以及它们在业务活动中的使用情况，确定用户对数据库系统的使用要求和各种约束条件等，形成用户需求规约，获得用户对新系统的信息要求、处理要求、安全性与完整性要求。

结构化分析方法（Structured Analysis）从最上层的系统组织机构入手，采用逐层分解的方式分析系统，每一层都用数据流图和数据字典来表述处理过程和数据的关系。数据字典是关于数据库中数据的描述，即记录数据库中的数据结构信息，通常包括数据项描述、数据结构描述、数据流描述、数据存储描述和处理过程描述五个部分。

数据项描述 = {数据项名，含义说明，别名，数据类型，长度，取值范围，取值含义，与其他数据项的逻辑关系}

数据结构描述 = {数据结构名，含义说明，组成：{数据项或数据结构}}

数据流描述 = {数据流名，说明，数据流来源，数据流去向，组成：{数据结构}，平均流量，高峰期流量}

数据存储描述 = {数据存储名，说明，编号，流入的数据流，流出的数据流，组成：{数据结构}，数据量，存取方式}

处理过程描述 = {处理过程名，说明，输入：{数据流}，输出：{数据流}，处理：{简要说明}}

2. 概念设计

数据库概念设计是将需求分析得到的用户需求抽象为概念模型的过程。概念模型反映现实世界各部门的信息结构、信息流动、信息间的互相制约关系，以及各部门对信息存储、查询和加工的要求等。概念模型独立于数据库逻辑结构，避开数据库在计算机上的具体实现细节，用一种抽象的形式表示出来，易于理解，便于和不熟悉计算机的用户交流。

设计概念模型一般有四类方法：自顶向下、自底向上、逐步扩张与混合策略。自顶向下方法是首先定义全局概念模型的框架，然后逐步细化的过程；自底向上方法是首先定义各局部应用的概念模型，然后将它们集成起来，得到全局概念模型的方法，这种方法最常用到；逐步扩张方法是首先定义最重要的核心概念模型，然后向外扩充，以滚雪球的方式逐步生成，直到得到总体概念模型的方法；混合策略方法是结合自顶向下和自底向上的方法，用自顶向下方法设计一个全局概念模型框架，以它为骨架集成由自底向上方法设计的各局部概念模型。

通常，在概念设计阶段，确定建模目标，开发建模计划，组织建模队伍，收集材料，制定约束和规范，形成基本数据资料表，再从基本数据资料表中直接或间接得出大部分实体以形成初步实体表。根据实际的业务需求和规则，使用实体关系矩阵来标识实体间的二元关系，然后根据实际情况确定连接关系的关系名、关系类型和说明。如果子实体的每个实例都需要通过和父实体的关系来标定，则为标定关系；否则为非标定关系。在非标定关系中，如果每个子实体的实例都与而且只与一个父实体关联，则为强制标定关系；否则为非强制标定关系。选出候选键，以便唯一识别每个实体的实例，再从候选键中确定主键，为了确定主键和关系的有效性，可以通过非空和非多值规则来保证，即一个实体实例的主键不能是空值，也不能在同一个时刻有一个以上的值。从基本数据资料表中的说明性文字获得属性表，确定属性的所有者，定义非主键属性，检查属性的非空及非多值性。检查完全依赖规则和非传递依赖规则，保证非主键属性必须依赖且仅依赖于主键，以符合关系理论第三范式。之后，定义属性的数据类型、长度、精度、非空性、缺省值、约束规则等。定义触发器、存储过程、视图、角色、同义词、序列等对象信息。

3. 逻辑设计

数据库逻辑设计是将现实世界的概念数据模型设计成某种特定数据库管理系统支持的逻辑数据模型，产生逻辑数据库并对其进行优化，以便在功能、性能、完整性和一致性约束及数据库可扩充性等方面均满足用户提出的要求。为提高数据库系统性能，应以规范化理论为指导，适当地调整数据模型结构。

首先要形成初始模型，把 E-R 图的实体和关系转换成指定的 DBMS 支持的类型，再根据定量分析和性能测算对逻辑模型进行评价，然后利用 DBMS 索引及其他措施，对模型进行优化。定量分析是处理频率和数据容量及其增长情况。性能测算是逻辑记录访问数目、应用程序传输总字节数和数据库的总字节数等。

4. 物理设计

数据库的物理设计是根据特定数据库管理系统所提供的存储结构和存取方法及具体计算机的各项物理设计指标，对具体的应用任务选定最适合应用系统的物理存储结构、存取方法和存取路径的过程，这一过程产生物理数据库。设计人员需要充分了解给定的硬件系统、操作系统、数据库管理系统等环境情况，保证响应时间及性能达到设计的要求。

5. 验证设计与实施

在上述设计的基础上用 DBMS 提供的 DDL 来建立一个具体的数据库，收集或产生足够的数据后再向数据库中装载，运行一些典型的应用任务来验证数据库设计的正确性和合理性。在进行数据库设计时应考虑到未来设计修改的可能性、方便性及可扩展性。大型数据库的设计过程往往需要经过多次循环反复，当设计的某步发现问题时，可能就需要返回到前面去进行修改。

6. 运行与维护

数据库系统投入正式运行，意味着数据库的设计与开发阶段的基本结束，运行与维护阶段的开始。数据库的运行和维护是个长期的工作，是数据库设计工作的延续和提高。在数据库运行阶段，需要掌握 DBMS 的存储、控制和数据恢复等基本操作，而且要经常性地涉及物理数据库，甚至逻辑数据库的再设计。数据库的运行和维护阶段的主要工作包括对数据库性能的监测、分析和改善；数据库的转存和恢复；维护数据库的安全性和完整性；数据库的重组和重构等。

第 2 章　关系数据库

关系数据库，是建立在关系模型基础上的数据库，现实世界中的各种实体及实体之间的各种关系均用关系模型来表示。关系模型是以关系代数为基础的集合论的产物，是一个二维表格模型，因而一个关系型数据库就是由二维表及其之间的联系组成的一个数据组织。实体关系模型（Entity-relationship Model）是一套数据库的设计工具，运用现实世界中事物与关系的观念，来解释数据库中抽象的数据架构。实体关系模型利用图形的方式（E-R 图）来表示数据库的概念设计，有助于设计过程中的构思及沟通。

2.1　关系数据模型

关系数据模型由关系数据结构、关系操作集合、关系完整性约束组成。关系数据结构用二维表来表示实体及实体之间的关系。在关系数据模型中，把数据看成二维表中的元素，操作的对象和结果都是二维表，一张二维表就是一个关系。如第 1 章的图 1-1 中的客户信息表就是一张二维表。关系操作的对象和结果都是集合，常用操作包括选择（Select）、插入（Insert）、删除（Delete）、修改（Update）、连接（Join）、并（Union）等。关系数据模型包括实体完整性、参照完整性和用户定义的完整性等三类完整性约束，其中实体完整性和参照完整性是关系数据模型必须满足的完整性约束条件，由关系数据库系统自动支持。

在一张二维表表示的一个关系的逻辑结构中，每个关系有一个关系名，对应一个表名，每个字段是表的一列，称为属性（Attribute），每一列有一个属性名，一行是一个元组，用以描述某个实体的完整信息。第 1 章中图 1-1 的客户信息表，在不考虑具体内容时，把属性名用逗号分开，放在括号内，表示为客户信息表（客户 ID，客户名，客户地址），所有属性不可再分（称为原子性），并在表中有唯一的属性名。属性的取值范围称为域（Domain），即不同元组对同一个属性的取值所限定的范围，如性别属性的取值范围只能为“男”或“女”。

2.2　关键字

在一个关系（二维表）中，如果一个属性或一些属性组合的值能唯一标识一个关系的元组而又不含有多余的属性，则该属性（或该属性组合）称为候选关键字

（Candidate Key），有时也简称为候选键。有时一个关系中有多个候选关键字，被选用的候选关键字称为主关键字（Primary Key），简称为主键，每一个关系都有一个并且只有一个主键。如图 1-1 中客户信息表的客户 ID 一般用来作为其主键，而主键一般用下画线来表明，如客户信息表（客户 ID，客户名，客户地址）。

2.3　关系运算

关系运算（Relational Data Manipulations）可分为传统的集合运算（并、差、交等）及专门的关系运算（选择、投影、连接等），前者把关系看成元组的集合，以元组作为集合的元素来进行运算，即运算是从行的角度来进行的；后者主要是从列的角度来进行的。

选择（Select）：按照一定条件在给定的关系中选取若干元组，即在关系的水平方向选取符合条件的子集，表示为 SELECT　<关系名>　WHERE〈条件〉，如图 2-1 所示。

投影（Project）：在关系的垂直方向去掉其他属性，选取含有给定属性的子集。关系中不存在重复的属性，因此投影运算结果集中，一般也不包括重复的值，如图 2-2 所示。

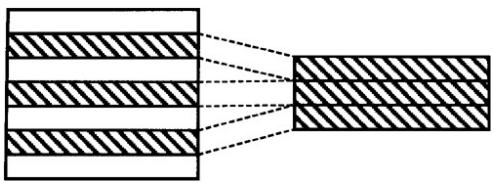


图 2-1　选择

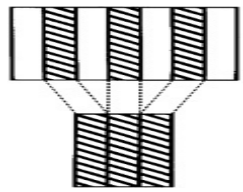


图 2-2　投影

连接（Join）：从两个关系的广义笛卡儿积中，选取满足给定属性间一定条件的元组，即水平合并两个关系，如图 2-3 所示。

并（Union）：垂直合并两个关系，产生包括每个表中的任何元组，但参与合并的关系中的属性必须相同，如图 2-4 所示。

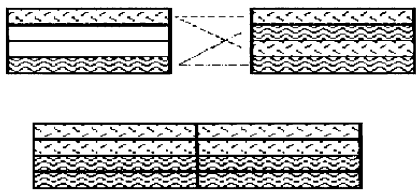


图 2-3　连接

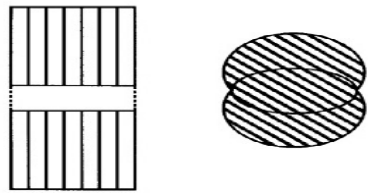


图 2-4　并