目录

第 4 章　数据表 / 47

第 5 章　视图 / 63

第 6 章　SQL Server 2016 数据库管理 / 77

第 7 章　SQL Server 2016 系统维护 / 89

第 8 章 T-SQL / 103

第 9 章 SQL 数据库查询 / 131

第 10 章　SQL 数据操作 / 161

第 11 章　存储过程 / 183

第 12 章　触发器 / 195

第 13 章　索引 / 211

第 16 章 事务 / 267

第 17 章 数据库性能优化 / 283

第 1 章 认识数据库

数据库是依照某种数据模型组织起来并存放在二级存储器中的数据集合，可以将其视为电子化的文件柜。数据库具有不重复、以最优方式提供多种应用服务、数据结构独立于应用程序、对数据的操作由统一软件进行管理和控制等特点。从数据管理技术的发展历程来看，数据库是由文件管理系统发展起来的，是数据管理的高级阶段。本章重点内容：

- 了解数据库的发展与组成；
- 掌握数据库的体系结构；
- 掌握数据库的数据模型；
- 了解常见的数据库。

知识导图

笔记

1.1 数据库系统概述

数据库（DataBase，DB）是按照数据结构来组织、存储和管理数据的仓库，产生于距今 60 多年前随着信息技术和市场的发展，特别是 20 世纪 90 年代以后，数据管理不再仅仅用于存储和管理数据，还出现了用户所需要的各种数据管理的方式。从简单的存储各种数据的表格到能够进行海量数据存储的大型数据库系统都属于数据库的范畴，并在各个方面得到了广泛的应用。在信息化社会，充分有效地管理和利用各类信息资源是进行科学研究和决策管理的前提条件。数据库技术是管理信息系统、办公自动化系统、决策支持系统等各类信息系统的核心部分，是进行科学研究和决策管理的重要技术手段。

1.1.1 数据库技术的发展

使用计算机后，随着数据处理的增长，产生了数据管理技术。数据管理技术的发展与计算机硬件（主要是外部存储器）、系统软件及计算机应用的范围有着密切的联系。数据管理技术的发展经历了 4 个阶段：人工管理阶段、文件系统阶段、数据库阶段和高级数据库技术阶段。其中，数据库阶段和高级数据库技术阶段统称为系统阶段，即由数据库系统进行管理数据的阶段。

1. 人工管理阶段

20 世纪 50 年代中期之前，计算机的软硬件并不完善。硬件存储设备只有磁带、卡片和纸带，软件方面还没有操作系统，当时的计算机主要用于科学计算。人工管理阶段由于还没有软件系统对数据进行管理，程序员在程序中不仅要规定数据的逻辑结构，还要设计其物理结构，包括存储结构、存取方法、输入 / 输出方式等。当数据的物理结构或存储设备改变时，用户程序就必须重新编写。由于数据的组织面向应用，不同的计算程序之间不能共享数据，使得不同的应用之间存在大量的重复数据，很难维护应用程序之间数据的一致性。这一阶段的主要特征可归纳为以下几点：

（1）计算机中没有支持数据管理的软件。

（2）数据组织面向应用，数据不能共享，数据重复。

（3）在程序中要规定数据的逻辑结构和物理结构，数据与程序不独立。

（4）数据处理方式——批处理。

2. 文件系统阶段

这一阶段处于 20 世纪 50 年代中期到 60 年代中期，其主要标志是计算机中有了专门管理数据库的软件——操作系统。操作系统文件管理功能的出现标志着数据管理步入一个新的阶段。在文件系统阶段，数据以文件为单位存储在外存储器，由操作系统统一管理，而操作系统为用户使用文件提供友好界面。该阶段中的文件逻辑结构与物理结构脱钩，程序和数据分离，使数据与程序有了一定的独立性。用户的程序与数据可分别存放在外存储器上，各个应用程序可以共享同一组数据，实现了以文件为单位的数据共享。由于数据的组织仍然是面向程序的，因此仍存在大量的数据冗余。同时，由于数据的逻辑结构不能方便地修改和扩充，因此数据的逻辑结构的每一点微小改变都会影响应用程序。此外，由于文件之间互相独立，因此不能反映现实世界中事物之间的联系，而操作系统不负责维护文件之间的联系信息。如果文件之间有内容上的联系，那么只能由应用程序去处理，这就加

大了程序设计人员的工作量。

笔记

3. 系统阶段

20 世纪 60 年代后期，随着计算机在数据管理领域的普遍应用，人们对数据管理技术提出了更高的要求：希望面向企业或部门，以数据为中心组织数据，减少数据的冗余，提供更高的数据共享能力，同时要求程序和数据具有较高的独立性，当数据的逻辑结构改变时，不涉及数据的物理结构，也不影响应用程序，以降低应用程序研发与维护的费用。数据库技术正是在这样的应用需求基础上发展起来的。数据管理技术经历了人工管理阶段和文件系统阶段后，获得了大量的技术，这为数据库的诞生奠定了基础。具体来说，数据库技术有如下特点：

（1）面向企业或部门。数据库以数据为中心进行数据的组织，形成综合性的数据库，从而为各应用程序共享。

（2）采用一定的数据模型。数据模型不仅描述了数据本身的特点，而且描述了数据之间的联系。

（3）数据冗余小，易修改、易扩充。在数据库技术阶段中，不同的应用程序根据处理要求从数据库中获取需要的数据，这样就减少了数据的重复存储，也便于增加新的数据结构和维护数据的一致性。

（4）程序和数据具有较高的独立性。

（5）具有良好的用户接口，用户可以方便地开发和使用数据库。

（6）对数据进行统一管理和控制，保证了数据的安全性、完整性，实现了并发控制。

数据管理技术从文件系统发展到数据库系统，这在信息领域中具有里程碑的意义。在文件系统阶段，人们在信息处理中关注的中心问题是系统功能的设计，因此，程序设计占主导地位；而在数据库阶段，数据开始占据中心位置，数据的结构设计成为信息系统首要关心的问题，而应用程序则以既定的数据结构为基础进行设计。

4. 发展趋势

随着信息管理内容的不断扩展，出现了丰富多样的数据模型（如层次模型、网状模型、关系模型、面向对象模型、半结构化模型等），新技术也层出不穷（如数据流、Web 数据管理、数据挖掘等）。每隔几年，国际上一些资深的数据库专家就会聚集一堂，探讨数据库现状、研究存在的问题和未来需要关注的新技术焦点。数据库与学科技术的结合将会建立一系列新数据库，如分布式数据库、并行数据库、知识库、多媒体数据库等，这将是数据库技术重要的发展方向。未来数据库技术及市场发展的两大方向是数据库和电子商务，数据管理技术将在数据库技术以及与之相关的数据挖掘和知识发现领域持续发展。

1.1.2 数据库系统的组成

数据库系统（DataBase System，DBS）是指一个具体的数据库管理系统软件和用它建立起来的数据库，通常由系统软件、数据库和数据管理员组成。系统软件主要包括操作系统、各种宿主语言、实用程序以及数据库管理系统（DBMS）；数据库由数据库管理系统统一管理，数据的插入、修改和检索均要通过数据库管理系统进行；数据管理员（DBA）负责创建、监控和维护整个数据库，使数据能被任何有权使用的人有效使用，数据库管理员一般由业务水平较高、资历较深的人员担任。数据库系统是软件研究领域的一个重要分

笔记

支，常称为数据库领域。数据库系统是为适应数据处理的需要而发展起来的一种较为理想的数据处理的核心机构，具体来说由如下部分组成：

（1）数据库：长期存储在计算机内，有组织、可共享的数据集合。数据库中的数据按一定的数学模型组织、描述和存储，具有较小的冗余、较高的数据独立性和易扩展性，并可为各种用户共享。

（2）硬件：构成计算机系统的各种物理设备，包括存储所需的外部设备，如物理硬盘、光盘等媒介。

（3）系统软件：包括操作系统、数据库管理系统及应用程序。数据库管理系统（DataBase Management System，DBMS）是数据库系统的核心软件，在操作系统的支持下工作，是科学地组织和存储数据、高效地获取和维护数据的系统软件。其主要功能包括数据定义、数据操纵、数据库的运行管理和数据库的建立与维护。

（4）人员：主要包括如下 4 类。

第一类为系统分析员和数据库设计人员。系统分析员负责应用系统的需求分析和规范说明，他们和用户及数据库管理员一起确定系统的硬件配置，并参与数据库系统的概要设计。数据库设计人员负责数据库中数据的确定、数据库各级模式的设计。

第二类为应用程序员。他们负责编写使用数据库的应用程序。这些应用程序可对数据进行检索、建立、删除或修改。

第三类为最终用户。他们利用系统的接口或查询语言访问数据库。

第四类是数据库管理员（DataBase Administrator，DBA），他们负责数据库的总体信息控制。DBA 的具体职责包括确定数据库中的信息内容和结构，决定数据库的存储结构和存取策略，定义数据库的安全性要求和完整性约束条件，监控数据库的使用和运行，负责数据库的性能改进、数据库的重组和重构，以提高系统的性能。

1.2 数据库的体系结构

人们为数据库设计了一个严谨的体系结构——数据库领域公认的标准结构——三级模式结构，包括外模式、概念模式和内模式。数据库体系结构能够有效地组织、管理数据，提高数据库的逻辑独立性和物理独立性。

1.2.1 什么是模式

虽然实际的数据库管理系统产品种类很多，支持不同的数据模式，使用不同的数据库语言，建立在不同的操作系统之上，数据的存储结构也各不相同，但它们在体系结构上通常具有相同的特征，即采用 3 级模式结构并提供两级映像功能。模式是数据库中全体数据的逻辑结构和特征的描述，仅仅涉及型的描述而不涉及具体的值。模式的一个具体值称为一个实例，同一个模式可以有很多实例。模式是相对稳定的，而实例是相对变动的，因为数据库中的数据是在不断更新的。模式反映的是数据的结构及其联系，而实例反映的是数据库某一时刻的状态。

1.2.2 三级模式结构

美国国家标准协会（American National Standards Institute，ANSI）的数据库管理系统

研究小组于 1978 年提出了标准化的建议，将数据库结构分为 3 级：面向用户或应用程序员的用户级、面向建立和维护数据库人员的概念级、面向系统程序员的物理级。其中，用户级对应外模式，概念级对应概念模式，物理级对应内模式。不同级别的用户对数据库形成不同的视图。所谓视图，就是指观察、认识和理解数据的范围、角度和方法，是数据库在用户眼中的反映。显然，不同层次（级别）的用户所看到的数据库是不同的。数据库系统结构层次如图 1-1 所示。

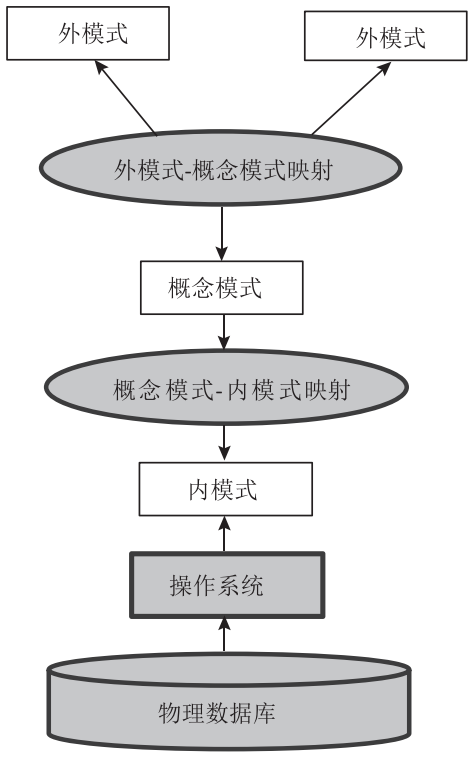


图 1-1　数据库系统结构层次图

1. 分类

（1）外模式。外模式又称子模式或用户模式，对应用户级。它是某个或某几个用户所看到的数据库的数据视图，是与某一应用有关的数据的逻辑表示。外模式是从模式导出的一个子集，包含模式中允许特定用户使用的那部分数据。用户可以通过外模式描述语言来描述、定义对应用户的数据记录（外模式）. 也可以利用数据操纵语言（Data Manipulation Language，DML）对这些数据记录进行操作。总的来说，外模式反映了数据库的用户观。

（2）概念模式。概念模式又称逻辑模式，对应概念级。它是由数据库设计者综合所有用户的数据，按照统一的观点构造的全局逻辑结构，是对数据库中全部数据的逻辑结构和特征的总体描述，是所有用户的公共数据视图（全局视图）。它是由数据库管理系统提供的数据模式描述语言（Data Description Language，DDL）描述、定义的，体现、反映了数据库系统的整体观。

（3）内模式。内模式又称存储模式，对应物理级。它是数据库中全体数据的内部表示或底层描述，是数据库最低一级的逻辑描述，它描述了数据在存储介质上的存储方式和物理结构，对应着实际存储在外存储介质上的数据库。在一个数据库系统中只有唯一的数据库，因而作为定义、描述数据库存储结构的内模式和定义、描述数据库逻辑结构的模式也是唯一的，但建立在数据库系统之上的应用则是非常广泛、多样的，所以对应的外模式不是唯一的，也不可能是唯一的。

2. 工作原理

数据库的三级模式是数据库在 3 个级别（层次）上的抽象，使用户能够逻辑地、抽象地处理数据而不必关心数据在计算机中的物理表示和存储。实际上，对于一个数据库系统而言，物理级数据库是客观存在的，是进行数据库操作的基础；概念级数据库不过是物理级数据库的一种逻辑、抽象的描述（模式）；用户级数据库则是用户与数据库的接口，是概念级数据库的一个子集（外模式）。用户应用程序根据外模式进行数据操作，通过外模式 - 概念模式映射定义和建立某个外模式与概念模式间的对应关系，将外模式与概念模式联系起来，当模式发生改变时，只要改变其映射，就可以使外模式保持不变，对应的应用程序也保持不变；另一方面，通过概念模式 - 内模式映射定义建立数据的逻辑结构（模

笔记

式）与存储结构（内模式）间的对应关系，当数据的存储结构发生变化时，只需改变概念模式 - 内模式映射，就能保持模式不变，因此应用程序也可以保持不变。

1.3 数据模型

数据模型（Data Model）是数据特征的抽象，是数据库管理的教学形式框架，也是数据库系统中用以提供信息表示和操作手段的形式结构。数据模型包括数据库数据的结构部分、数据库数据的操作部分和数据库数据的约束条件。数据模型描述了在数据库中结构化和操纵数据的方法，模型的结构部分规定了如何描述数据。

1.3.1 数据模型的分类

1. 组成部分

数据模型所描述的内容包括 3 部分：数据结构、数据操作和数据约束。

（1）数据结构：数据模型中的数据结构主要描述数据的类型、内容、性质以及数据间的联系等。数据结构是数据模型的基础，数据操作和约束都基本建立在数据结构上。

（2）数据操作：数据模型中的数据操作主要描述在相应数据结构上的操作类型和操作方式。

（3）数据约束：数据模型中的数据约束主要描述数据结构内数据间的语法、词义联系、它们之间的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、有效和相融。

2. 分类

数据模型的研究包括以下 3 方面。

（1）概念数据模型。这是面向数据库用户的现实世界的数据模型，主要用来描述现实世界的概念化结构，可以使数据库的设计人员在设计的初始阶段摆脱计算机系统及数据库管理系统的具体技术问题，集中精力分析数据以及数据之间的联系等。概念数据模型与具体的数据库管理系统无关。需要注意的是，概念数据模型必须换成逻辑数据模型才能在数据库管理系统中实现。

（2）逻辑数据模型。这是用户在数据库中看到的数据模型，是具体的数据库管理系统所支持的数据模型，主要有网状数据模型、层次数据模型和关系数据模型 3 种类型。此模型既要面向用户，又要面向系统，主要用于数据库管理系统的实现。

（3）物理数据模型。这是描述数据在存储介质上的组织结构的数据模型，不仅与具体的数据库管理系统有关，还与操作系统和硬件有关。每一种逻辑数据模型在实现时都有与其相对应的物理数据模型。数据库管理系统为了保证其独立性与可移植性，将大部分物理数据模型的实现工作交由系统自动完成，而设计者只设计索引、聚集等特殊结构。数据库的类型是根据数据模型来划分的，而任何一个 DBMS 也是根据数据模型有针对性地设计出来的，这就意味着必须把数据库组织成符合 DBMS 规定的数据模型。目前应用在数据库系统中的数据模型有层次模型、网状模型和关系模型。它们之间的根本区别在于数据之间联系的表示方式不同（记录模型之间的联系方式不同）。层次模型以“树结构”表示数据之间的联系；网状模型以“图结构”来表示数据之间的联系；关系模型是用“二维表”（或称为关系）来表示数据之间的联系的。

1.3.2 E-R 模型

笔记

E-R 方法是“实体·联系方法”（Entity-Relationship Approach）的简称，是描述现实世界概念结构模型的有效方法。E-R 方法是表示概念模型的一种方式，用矩形表示实体型，在矩形框内写明实体名；用椭圆表示实体的属性，并用无向边将其与相应的实体型连接起来；用菱形表示实体型之间的联系，在菱形框内写明联系名，并用无向边分别与有关实体型连接起来，同时在无向边旁标上联系的类型（*l*:*l*、*l*:*n* 或 *m*:*n*）。用 E-R 方法描述的数据模型即为 E-R 模型，也称为 E-R 图。如图 1-2 所示为一个简单学生管理系统的数据库 E-R 模型图。

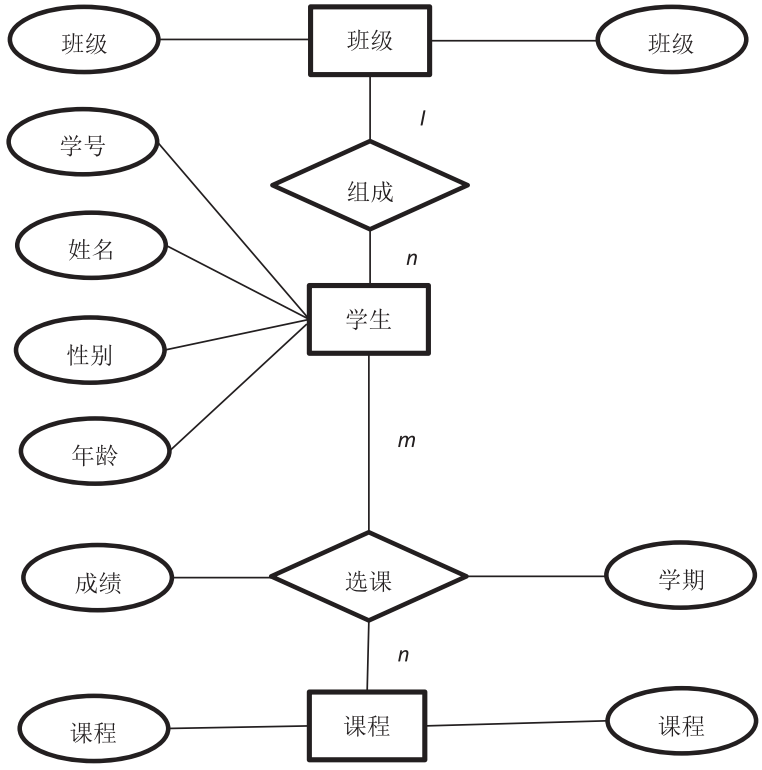


图 1-2　简单学生管理系统的数据库 E-R 图

1. E-R 的成分

在 E-R 图中，有如下 4 个成分。

（1）矩形框：表示实体，在框中记入实体名。

（2）菱形框：表示联系，在框中记入联系名。

（3）椭圆形框：表示实体或联系的属性，将属性名记入框中。对于主属性名，则在其名称下加一下画线。

（4）连线：实体与属性之间、实体与联系之间、联系与属性之间用直线相连，并在直线上标注联系的类型。

2. 相图要素

构成 E-R 图的基本要素是实体型、属性和联系，其表示方法如下：

（1）实体型（Entity）：有相同属性的实体具有相同的特征和性质，用实体名及其属性名集合来抽象和刻画同类实体，在 E-R 图中用矩形表示，在矩形框内写明实体名。例如，

笔记

学生“张三丰”、学生“李寻欢”都是实体。

（2）属性（Attribute）：实体所具有的某一特性，一个实体可由若干个属性来刻画。属性在 E-R 图中用椭圆形表示，并用无向边将其与相应的实体连接起来。例如，学生的姓名、学号、性别都是属性。

（3）联系（Relationship）：也称关系，用于在信息世界中反映实体内部或实体之间的联系。联系包括实体内的联系和实体间的联系两种。实体内部的联系通常是指组成实体的各属性之间的联系；实体之间的联系通常是指不同实体集之间的联系。联系在 E-R 图中用菱形表示，在菱形框内写明联系名，并用无向边分别与有关实体连接起来，同时在无向边旁标上联系的类型（1:1、1:*n* 或 *m*:*n*）。例如，老师给学生授课存在授课关系，学生选课存在选课关系。

需要注意的是，联系也可能有属性。例如，学生“学”某门课程所取得的成绩，既不是学生的属性也不是课程的属性。由于“成绩”既依赖于某名特定的学生又依赖于某门特定的课程，因此它是学生与课程之间的联系“学”的属性。一般来说，联系可分为以下 3 种类型。

①一对一联系（1:1）：例如，一个部门有一个经理，而每个经理只在一个部门任职，则部门与经理的联系是一对一的。

②一对多联系（1:*n*）：例如，某校教师与课程之间存在一对多的联系“教”，即每个教师可以教多门课程，但是每门课程只能由一个教师来教。

③多对多联系（*m*:*n*）：例如，如图 1-2 所示学生与课程间的联系“学”是多对多的，即一个学生可以学多门课程，每门课程也可以有多个学生来学。

3. 设计步骤

一般来说，用户在设计数据库之前需要先设计 E-R 模型，而 E-R 模型用 E-R 图来表示，其设计分为 3 个步骤：调查分析、合并生成和修改重构。

（1）调查分析。在需求分析阶段，通过对应用环境和要求进行详尽的调查分析，用多层数据流图和数据字典描述整个系统，逐一设计分析 E-R 图每个局部应用对应的数据流图，同时将局部应用涉及的数据都收集在数据字典中。

（2）合并生成。由于实体之间的联系在不同局部视图中呈现不同的类型，因此用户需要设计多个针对局部应用的 E-R 图。合并生成步骤是将多个局部 E-R 图的实体、属性和联系合并，从而生成整体的 E-R 图。

（3）修改重构。经合并生成后的基本 E-R 图可能存在冗余的数据和冗余的实体间联系，即存在可由基本数据导出的数据和由其他联系导出的联系。冗余数据和冗余联系容易破坏数据库的完整性，给数据库的维护增加困难。因此，得到基本 E-R 图后，还应当进一步检查 E-R 图中是否存在冗余。如果存在，应设法予以消除。修改重构步骤主要采用分析的方法来消除基本 E-R 图中的冗余，也可以用规范化理论来消除冗余。

1.3.3 层次模型

当前数据库领域常用的数据模型主要有 3 种：层次模型、网状模型和关系模型。其中，层次模型和网状模型统称非关系模型，如图 1-3 所示。

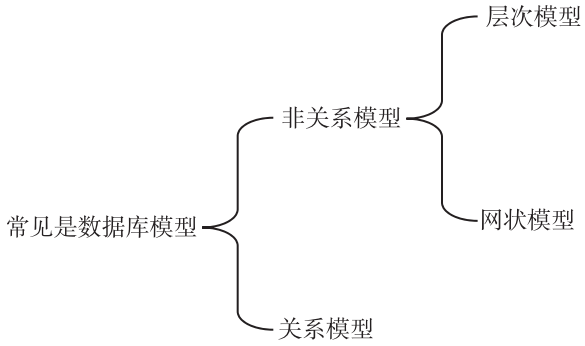


图 1-3　常见的 3 种数据模型

1. 层次模型的定义

现实世界中许多实体之间的联系本来就呈现出一种很自然的层次关系，如家族关系、军队编制、行政机构等，这就需要用层次结构来描述。层次模型是按照层次结构的形式组织数据库数据的数据模型，用树形结构来表示各类实体以及实体间的联系。层次模型是在数据结构中满足下面两个条件的基本层次联系的集合：

（1）有且只有一个节点且没有双亲节点，这个节点称为根节点。

（2）除根节点之外的其他节点有且只有一个双亲节点。

在层次模型中，使用节点表示记录。记录之间的联系用节点之间的连线表示，这种联系是父子之间的一对多的实体联系。层次模型中的同一双亲的子女节点称为兄弟节点，没有子女节点的节点称为叶节点。层次模型如图 1-4 所示。

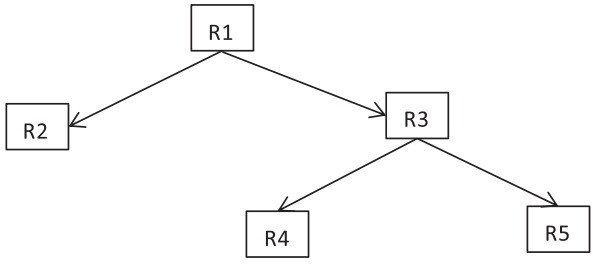


图 1-4　层次模型示例

层次模型像一棵倒立的树，只有一个根节点，有若干个叶节点，节点的双亲是唯一的。图 1-5 是一个教学院系的数据结构模型，该层次数据结构中有 4 个记录。

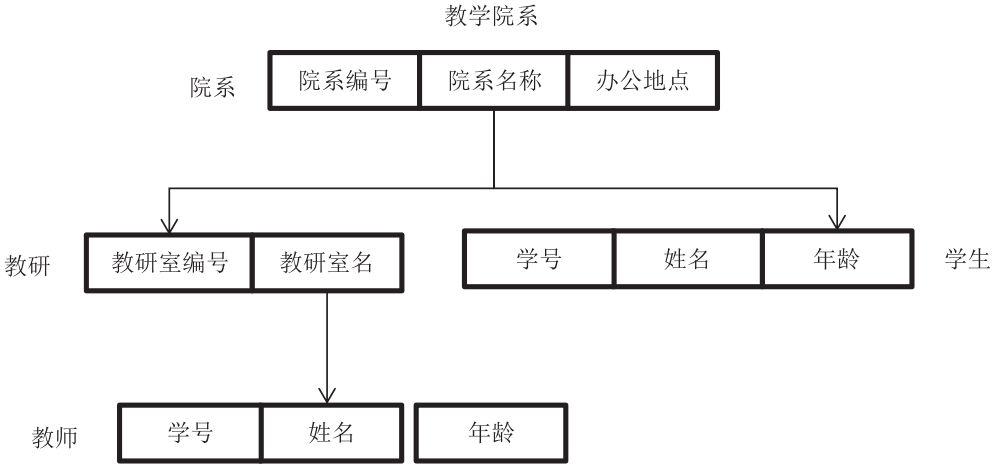


图 1-5　教学院系的数据结构模型

笔记

2. 层次模型的数据操作

层次模型的数据操作主要有查询、插入、删除和更新。需要注意的是，进行插入、删除、更新操作时，要满足层次模型的完整性约束条件。层次模型必须满足的完整性约束条件如下：

（1）在进行插入记录值操作时，如果没有指明相应的双亲记录值，就不能插入子女记录值。

（2）进行删除记录值操作时，如果删除双亲记录值，相应的子女记录值也同时被删除。

（3）进行修改记录值操作时，应修改所有相应记录值，以保证数据的一致性。

3. 层次模型的优、缺点

层次模型能够描述自然界的一些基本关系，是其他数据模型所不能代替的，其主要优点如下：

（1）层次模型的数据结构比较简单。

（2）对于实体间联系是固定的且预先定义好的应用系统，采用层次模型实现，其性能低于关系模型，不低于网状模型。

（3）层次数据模型提供了良好的完整性支持。

需要注意的是，层次模型中的任何一个给定的记录值只有按其路径查看时才能显示它的全部意义，没有一个子记录值能够脱离其双亲记录值而独立存在。因此，层次模型对具有一对多的层次关系的描述非常直观、自然、容易理解。同样地，由于层次模型是较为单一的模型，因此能描述的基本关系较少。该模型存在的主要缺点如下：

（1）现实世界中很多联系是非层次性的，如多对多联系、一个节点具有多个双亲等。

（2）对插入和删除操作的限制比较多。

（3）查询子节点必须通过双亲节点。

（4）由于结构严密，层次命令趋于程序化。

1.3.4 网状模型

在现实世界中，事物之间的联系更多是非层次关系，用层次模型表示非树形结构很不直观，而网状模型则可以克服这一缺陷。网状数据模型的典型代表是 DBTG 系统，这是 20 世纪 70 年代数据系统语言研究会（Conference On DataSystem Language，CODASYL）下属的数据库任务组（DataBase Task Group，DBTG）提出的一个系统方案。DBTG 系统虽然不是实际的软件系统，但是它提出的基本概念、方法和技术具有普遍意义，对于网状数据库系统的研制和发展起了重大的影响。后来许多系统都采用 DBTG 模型或者简化的 DBTG 模型，如 CuUinet Software 公司的 IDMS 等。

1. 网状模型的定义

网状模型是指满足下面两个条件的基本层次联系的集合：

（1）有一个以上的节点没有双亲。

（2）节点可以有多于一个的双亲。

网状模型如图 1-6 所示。

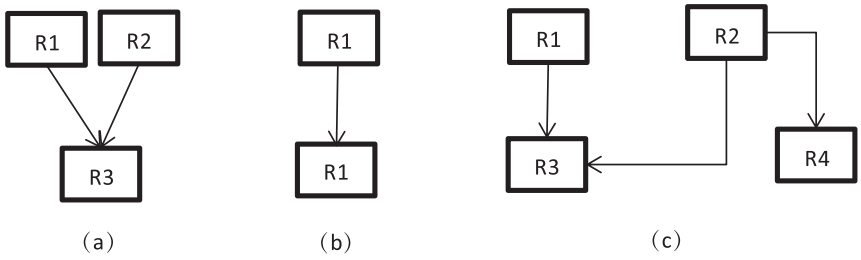


图 1-6　网状模型示例

网状模型是一种比层次模型更具普遍性的结构，去掉了层次模型的两个限制，允许多个节点没有双亲节点，允许节点有多个双亲节点，此外还允许两个节点之间有多种联系。因此，网状模型可以更直观地描述现实世界，而层次模型实际上是网状模型的一个特例。与层次模型一样，网状模型也使用记录和记录值表示实体集和实体，每个节点也表示一个记录，每个记录可包含若干个字段。

2. 网状模型的数据操作与完整性约束

与层次模型相似，网状模型的数据操作主要包括查询、插入、删除和更新。进行插入操作时，允许插入尚未确定双亲节点值的子节点值。进行删除操作时，只允许删除双亲节点值。进行更新操作时，只需更新指定记录即可。因此，一般来说，网状模型没有层次模型那样严格的完整性约束条件，但具体的网状数据库系统（如 DBTG）对数据操作都加了一些限制，提供了一定的完整性约束。DBTG 在模式 DDL 中提供了定义 DBTG 数据库完整性的若干概念和语句，主要有以下几种。

（1）支持记录码的概念。码是唯一标识记录的数据项的集合。在数据库中不允许出现重复值。

（2）保证一个联系中双亲记录和子记录之间是一对多的联系。

（3）可以支持双亲记录和子记录之间的某些约束条件。例如，有些子记录要求双亲记录存在才能插入，双亲记录删除时也连同删除。

3. 网状数据模型的优、缺点

相对于层次模型，网状数据模型所能描述的自然关系更多，主要优点如下：

（1）能够更为直接地描述现实世界。例如，一个节点可以有多个双亲，节点之间可以有多种联系。

（2）具有良好的性能，存取效率较高。

网状数据模型也存在不少缺点，主要表现在结构比较复杂，而且应用环境越大，数据库的结构就变得越复杂，不利于最终用户掌握。模型的数据定义语言（DDL）、数据操作语言（DML）复杂，用户不容易使用。此外，网状模型中由于记录之间的联系是通过存取路径实现的，应用程序在访问数据时必须选择适当的存取路径。因此，用户必须了解系统结构的细节，这就加重了编写应用程序的负担。

1.3.5 关系模型

关系模型是当前最重要的、应用最广泛的一种数据模型。目前，主流的数据库系统大部分都是基于关系模型的关系数据库系统（Relational DataBase System，RDBS）的。1970 年，美国 IBM 公司 San Jose 研究室的研究员 E.F.Codd 首次提出数据库系统的关系模型，

笔记

笔记

开创了数据库关系方法和关系数据理论的研究，为数据库技术的发展奠定了理论基础。20世纪80年代以来，计算机厂商新推出的DBMS几乎都支持关系模型，非关系模型的产品也大都添加了关系接口，数据库领域当前的研究工作也都是以关系方法为基础的。

1. 关系模型的数据结构

关系数据模型是建立在严格的数学概念基础上的。在关系模型中，数据的逻辑结构是一张二维表，由行和列组成。关系模型中的主要术语如下：

（1）关系。一个关系对应通常所说的一张二维表。

（2）元组。表中的一行称为一个元组，许多系统中把元组称为记录。

（3）属性。表中的一列称为一个属性。一个表中往往会有多个属性，为了区分属性，要给每一列起一个属性名。

（4）码。表中的某个属性或属性组的值可以唯一地确定一个元组，且属性组中不含多余的属性，这样的属性或属性组称为关系的码。

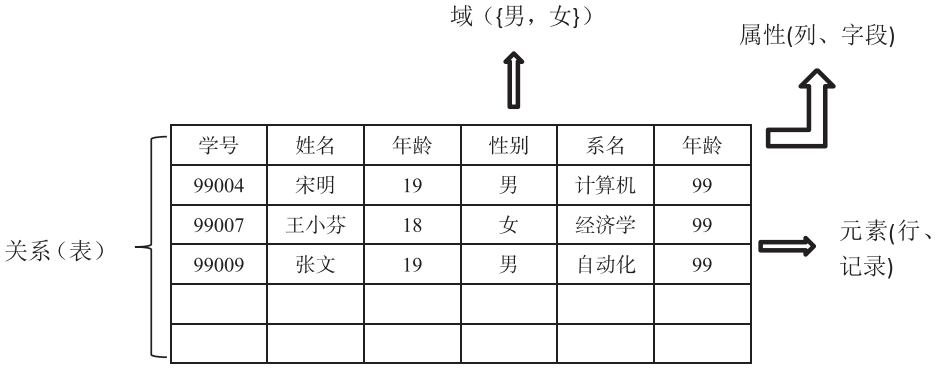
（5）域。域指属性的取值范围。例如，大学生年龄属性的域是（18~30），性别的域是（男，女）。

（6）分量。分量指元组中的属性值。

（7）关系模式。关系的型称为关系模式，是对关系的描述。关系模式的一般表示如下：

关系名（属性1，属性2，…，属性n）

在关系模型中，实体集以及实体间的联系都是用关系来表示的。关系模型要求关系必须是规范化的，即要求关系必须满足一定的规范条件，这些规范条件中最基本的一条就是：关系的每一个分量必须是一个不可分的数据项，也就是说，不允许表中还有表。关系模型如图1-7所示。



学号	姓名	年龄	性别	系名	年龄
99004	宋明	19	男	计算机	99
99007	王小芬	18	女	经济学	99
99009	张文	19	男	自动化	99

图1-7　关系模型示例

2. 关系模型的数据操作与完整性约束

关系数据模型的操作主要包括查询、插入、删除和修改数据，这些操作必须满足关系的完整性约束条件。关系模型中数据操作的特点是集合操作方式，即操作对象和操作结果都是集合，这种操作方式也称为一次一集合的方式。相应地，非关系数据模型的操作方式是一次一记录的方式。关系的完整性约束条件包括三大类：实体完整性、参照完整性和用户定义的完整性。实体完整性定义数据库中每一个基本关系的主码应满足的条件，能够保证元组的唯一性。参照完整性定义表之间的引用关系，即参照与被参照关系。用户定义完整性是用户针对具体的应用环境制定的数据规则，反映某一具体应用所涉及的数据必须满足的语义要求。

笔记

3. 关系模型的优缺点

关系模型是当前使用最为广泛的一类模型，目前的主流数据库系统如 Oracle、SQL Server 等都采用关系模型。关系数据模型的优点主要体现在以下几点：

（1）关系模型与非关系模型不同，它是建立在严格的数学理论基础上的。

（2）关系模型的概念单一，实体与实体间的联系都用关系表示，对数据的检索结果也是关系（即表），所以其数据结构简单、清晰，用户易懂易用。

（3）关系模型的物理存储和存取路径对用户透明，从而具有更高的数据独立性、更好的安全保密性，简化了程序员的数据库开发工作。

需要注意的是，虽然关系模型是现在的主流，但该模型也存在一定的缺陷，主要表现在如下两方面：

（1）由于存取路径对用户透明，查询效率往往不如非关系数据模型高，因此为了提高性能，必须对用户的查询请求进行优化，这就增加了开发数据库管理系统的难度和负担。

（2）关系数据模型不能以自然的方式表示实体集间的联系，存在语义信息不足、数据类型过少等弱点。

1.4 常见数据库

目前，商品化的数据库管理系统以关系型数据库为主导产品，技术比较成熟。面向对象的数据库管理系统虽然技术先进，数据库易于开发、维护，但尚未有成熟的产品。目前主流关系型数据库管理系统有 Oracle、Access 和 SQL Server 等。本节根据选择数据库管理系统的依据比较分析这几种主流数据库管理系统的优势和不足。

1.4.1 Access

Microsoft Access 是由微软（Microsoft）公司发布的一款关系数据库管理系统。它结合了 Microsoft Jet Database Engine 和图形用户界面两个特点，是 Microsoft Office 的系统程序之一。

1. 优势

Microsoft Access 提供了一个丰富的开发环境。这个开发环境给了用户足够的灵活性和对 Microsoft Windows 应用程序接口的控制，同时保护用户免遭用高级或低级语言开发环境时所碰到的各种麻烦。Microsoft Access 数据库的主界面如图 1-8 所示。

Microsoft Access 是一个把数据库引擎的图形用户界面和软件开发工具结合在一起的数据库管理系统，其主要优势表现在如下几个方面。

（1）存储方式单一。Access 管理的对象有表、查询、窗体、报表、页、宏和模块，以上对象都存放在后缀为（.mdb）的数据库文件中，便于用户的操作和管理。

（2）面向对象。Access 是一个面向对象的开发工具，利用面向对象的方式将数据库系统中的各种功能对象化，将数据库管理的各种功能封装在各类对象中。它将一个应用系统当作是由一系列对象组成的，对每个对象都定义一组方法和属性。通过对象的方法、属性完成数据库的操作和管理，极大地简化了用户的开发工作。同时，这种面向对象的开发方式，使得开发应用程序更为简便。

笔记

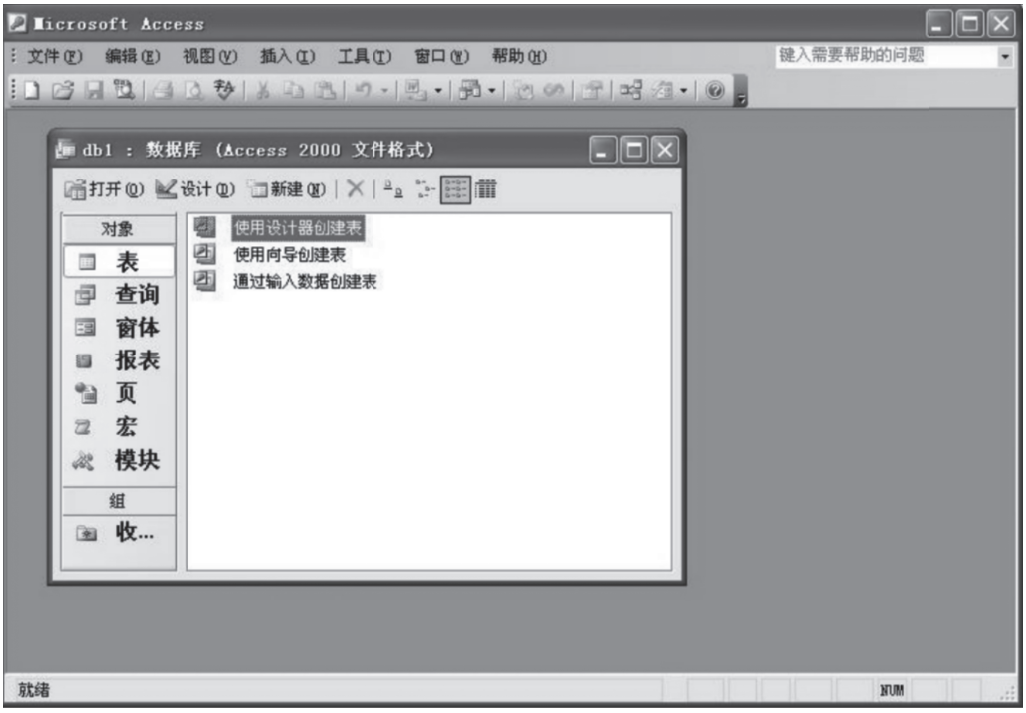


图 1-8　Microsoft Access 数据库的主界面

（3）界面友好、易操作。Access 是一个可视化工具，风格与 Windows 完全一样，用户想要生成对象并应用，只要使用鼠标进行拖放即可，非常直观方便。系统还提供了表生成器、查询生成器、报表设计器以及数据库向导、表向导、查询向导、窗体向导、报表向导等工具，使得操作简便，容易使用和掌握。

（4）集成环境、处理多种数据信息。Access 是基于 Windows 操作系统下的集成开发环境，该环境集成了各种向导和生成器工具，极大地提高了开发人员的工作效率，使得建立数据库、创建表、设计用户界面、设计数据查询、报表打印等可以方便有序地进行。

（5）Access 支持 ODBC（开放数据库连接），利用 Access 强大的 DDE（动态数据交换）和 OLE（对象的连接和嵌入）特性，可以在一个数据表中嵌入位图、声音、Excel 表格、Word 文档，还可以建立动态的数据库报表和窗体等。Access 还可以将程序应用于网络，并与网络上的动态数据相连接。利用数据库访问页对象生成 DML 文件，轻松构建 Internetl Intranet 的应用。

2. 缺陷

尽管 Microsoft Access 具有许多优点，但它毕竟是一个小型数据库，不可避免地存在一些缺陷，主要表现在以下几个方面。

（1）数据库过大时性能下降明显。一般来说，当 Access 数据库达到 100MB 的时候，数据库性能会显著下降。例如，当访问使用 Access 作为数据库的网站时，人数过多时容易造成 IIS 假死，过多消耗服务器资源。

（2）容易出现各种因数据库刷新频率过快而引起的数据库问题。

（3）Access 数据库安全性比不上其他类型的数据库。

1.4.2 SQL Server

笔记

SQL Server 也是 Microsoft 公司推出的关系型数据库管理系统，具有使用方便、伸缩性好与相关软件集成程度高等优点，可跨越从运行 Microsoft Windows 98 的 PC 到运行 Microsoft Windows 2012 的服务器等多种平台使用。图 1-9 为 Microsoft SQL Server 数据库的 Management Studio 主界面。

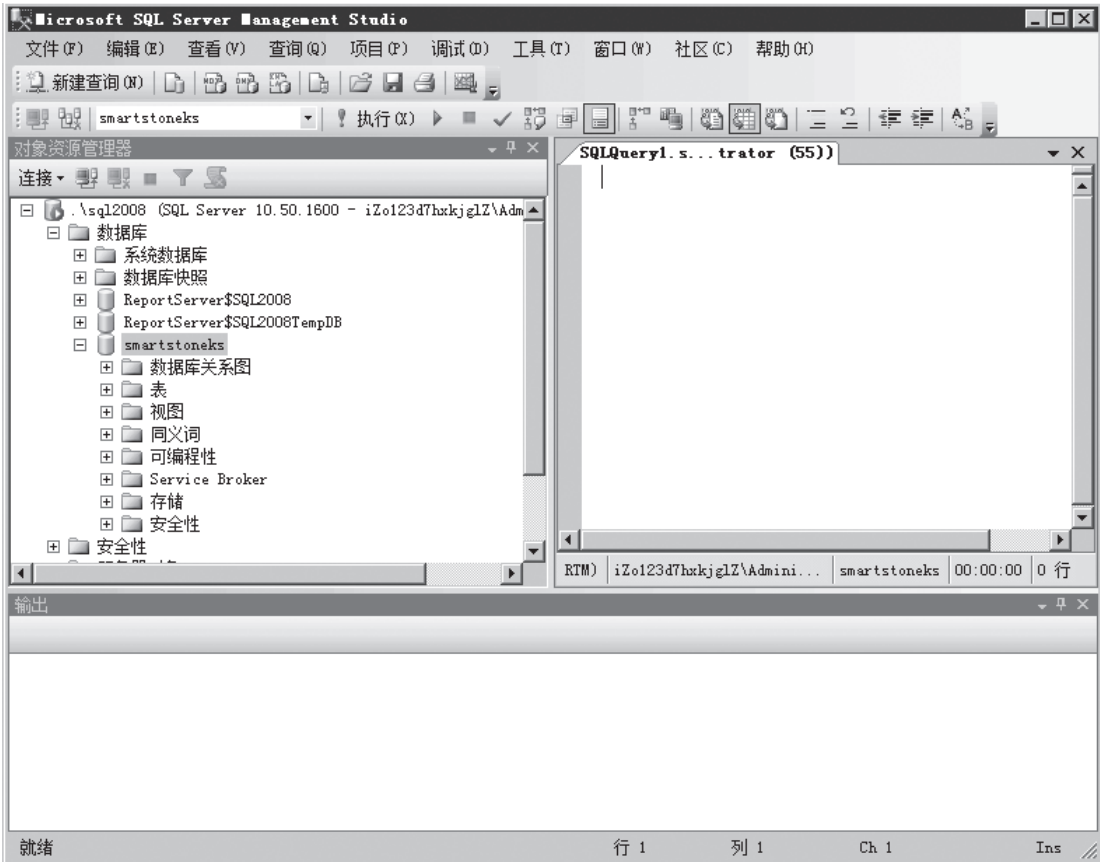


图 1-9　Microsoft SQL Server 数据库的 Management Studio 主界面

Microsoft SQL Server 是一个全面的数据库平台，使用集成的商业智能（BI）工具提供了企业级的数据管理。Microsoft SQL Server 数据库引擎为关系型数据和结构化数据提供了更安全可靠的存储功能，使用户可以构建和管理用于业务的高可用和高性能的数据应用程序。SQL Server 的主要特点如下：

（1）真正的客户机 / 服务器体系结构。

（2）图形化用户界面，使系统管理和数据库管理更加直观、简单。

（3）丰富的编程接口工具，为用户进行程序设计提供了更大的选择余地。

（4）SQL Server 与 Windows NT 完全集成，利用了 NT 的许多功能，如发送和接收消息、管理登录安全性等，SQL Server 也可以很好地与 Microsoft Office 产品集成。

（5）具有很好的伸缩性，可跨越多种平台使用。

（6）对 Web 技术的支持度高，使用户能够很容易地将数据库中的数据发布到 Web 页面上。

（7）SQL Server 新版本提供数据库功能，这个功能只在 Oracle 和其他更昂贵的

笔记

DBMS 中才有。

（8）内存在线事务处理（OLTP）引擎，内存 OLTP 整合到 SQL Server 的核心数据库管理组件中，它不需要特殊的硬件和软件就能够无缝整合现有的事务过程，允许将 SQL Server 内存缓冲池扩展到固态硬盘（SSD）或 SSD 阵列上。这一点对于支持繁重读负载的 OLTP 操作特别好，能够降低延迟、提高吞吐量和可靠性，消除 IO 瓶颈。

（9）云整合，引入了智能备份（Smart Backups）概念，能自动决定要执行完全备份还是差异备份，以及何时执行备份。还允许将本地数据库的数据和日志文件存储到 Azure 上。此外，SQL Server Management Studio 提供了一个部署向导，它可以帮助用户轻松地将现有本地数据库迁移到 Azure 虚拟机上。

1.4.3 Oracle

Oracle 数据库系统是美国甲骨文（Oracle）公司提供的以分布式数据库为核心的一组软件产品，是目前流行的客户 / 服务器（CLIENT/SERVER）或 B/S 体系结构的数据库之一。

图 1-10 为 Oracle 12c 数据库的 Developer 主界面。

图 1-10　Oracle 数据库的主界面

Oracle 数据库是目前世界上使用最为广泛的数据库管理系统，作为一个通用的数据库系统，它具有完整的数据管理功能；作为一个关系数据库，它是一个完备关系的产

品；作为分布式数据库，它实现了分布式处理功能。只要在一种机型上学习了 Oracle 知识，便能在各种类型的机器上使用。编写本书时，Oracle 数据库的最新版本为 Oracle Database12c。Oracle 12c 引入了一个新的多承租方架构，使用该架构可轻松部署和管理数据库云。此外，一些创新特性可最大限度地提高资源使用率和灵活性，如 Oracle Multitenant 可快速整合多个数据库，而 Automatic Data Optimization 和 Heat Map 能以更高的密度压缩数据和对数据分层。这些独一无二的技术进步再加上在可用性、安全性和大数据支持方面的增强，使得 Oracle12c 成为私有云和公有云部署的理想平台。

笔记

Oracle 的特点如下：

（1）名副其实的大型数据库：由 Oracle 建立的数据库，最大数据量可达几百吉字节。

（2）共享 SQL 和多线索服务器体系结构：这两个特性的结合可减少 Oracle 的资源占用，增强处理能力，支持成百甚至上千用户。

（3）跨平台能力：Oracle 数据库管理系统可以运行在 100 多个硬件和软件平台上。这一点是其他 PC 平台上的数据库产品所不能及的。

（4）分布式数据库：可以把物理分布不同的多个数据库上的数据看成一个完整的逻辑数据库。尽管数据操纵的单个事务可能要运行于多个地点，但这对应用程序却是透明的，就好像所有的数据都是物理地存储在本地数据库中。

（5）卓越的安全机制：包括对数据库的存取控制、决定可以执行的命令、限制单一进程可用的资源数量以及定义数据库中数据的访问级别等。

（6）支持客户机服务器方式，支持多种网络协议。

除上面讲解的 Microsoft Office Access、SQL Server 和 Oracle 3 个典型数据库外，还有许多关系型数据库也较为常见，如 IBMDB2、Informix、Sybase、MySQL 等，有兴趣的读者可自行了解，此处不再赘述。