

数据库概览——实例、

概念与认识

数据库技术是现代信息科学与技术的重要组成部分，是计算机数据处理与信息管理的核心。随着数据容量的急剧增长和内容的迅速变化，建立满足信息处理要求的行之有效的数据管理系统已成为各行各业生存和发展的重要条件。因此，数据库技术得到了越来越广泛的应用，从小型的单项事务处理到大型复杂的信息系统都采用数据库来存储和管理信息资源，以保证数据的有效性、完整性和共享性。本章介绍数据库系统的一些基本概念和常用术语，作为后面各章节的准备和基础。

1.1 数据管理技术的进展

数据处理是对数据进行收集、存储、加工、传播等一系列活动的总和，其目的是从大量复杂的甚至难以理解的数据中抽取并导出对于应用有价值的、有意义的数据，作为决策的依据。数据管理指对数据的收集、整理、组织、存储、维护、检索和传输等操作。数据管理是数据处理的中心活动，直接影响数据处理的效率，两者密不可分。

随着计算机科学与技术的发展，利用计算机进行数据管理经历了三个阶段，即人工管理阶段、文件系统阶段和数据库系统阶段。

1.1.1 人工管理阶段

20 世纪 50 年代以前，计算机主要用于科学计算。从硬件看，外存储器只有纸带、卡片和磁带，没有磁盘等直接存取的设备；从软件看，没有操作系统，没有管理数据的专门软件。数据处理方式是批处理，数据管理由程序员设计和安排。程序员将数据处理纳入程序设计的过程中，编制程序中需要考虑数据的逻辑结构和物理结构，包括存储结构和存取方法等。人工管理阶段应用程序与数据之间的对应关系如图 1.1 所示，其特点如下：

- (1) 数据不长期保存在计算机中。
- (2) 应用程序管理数据，数据与程序结合在一起；若数据的逻辑结构或物理结构发生变化，则必须对程序进行修改；这种特性称为数据与程序不具有独立性。
- (3) 数据是面向应用的，一组数据对应一个程序，数据不共享。当多个应用程序涉及相同数据时，必须各自定义。

1.1.2 文件系统阶段

20 世纪 50 年代后期至 60 年代中期，计算机开始用于数据处理。从硬件看，外存储器有了磁盘、磁鼓等直接存取设备；从软件看，有了操作系统，且操作系统中有了专门的数据管理软件，

即文件系统。采用文件系统进行数据管理，其基本思想是由应用程序利用文件系统提供的功能将数据按一定的格式组织成独立的数据文件，然后以文件名访问相应的数据。文件系统阶段应用程序与数据之间的对应关系如图 1.2 所示，其优点如下：

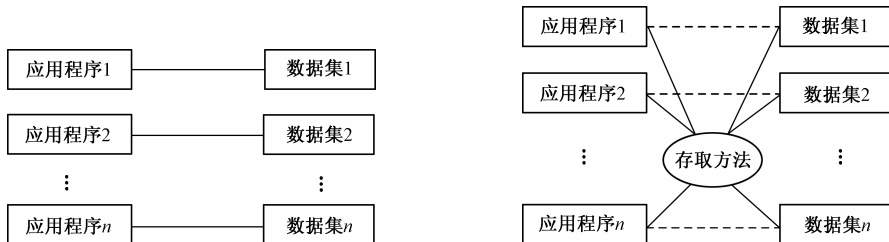


图 1.1 人工管理阶段应用程序与数据间对应关系 图 1.2 文件系统阶段应用程序与数据间对应关系

(1) 数据能够长期保存，可以反复对其进行查询、修改等操作。

(2) 由专门软件对数据进行管理，应用程序与数据之间由文件系统所提供的存取方法进行转换，程序与数据之间有了一定的独立性。程序员可不必过多考虑文件的存储细节，并且数据在存储上的改变不一定影响程序，从而减少了程序维护的工作量。

但是，文件系统仍存在以下缺点：

(1) 数据共享性差，冗余度大。数据文件是面向应用的，当不同应用程序具有部分相同数据时，也必须建立各自的文件，导致同一数据项可能重复出现在多个文件中，因此数据冗余度大，会导致数据冲突，数据一致性维护困难等问题。

(2) 数据独立性差。由于数据的组织和管理直接依赖于应用程序，如果数据的逻辑结构发生改变就需要相应地修改应用程序。

由此可见，虽然文件系统记录内有结构，但文件之间是孤立的，整体仍然是一个无结构的数据集合，因此不能反映现实世界实体之间的联系。

1.1.3 数据库系统阶段

20 世纪 60 年代后期，数据处理成为计算机应用的主要领域，数据量急剧增长，数据关系更加复杂，对数据管理提出了更高的要求。为了满足多用户、多应用程序共享数据的需求，实现数据的统一管理，人们开始了对数据建模和组织、以及对数据进行统一管理和控制的研究，形成了数据库这一计算机科学与技术的重要分支。

数据库系统的主要特征是数据的统一管理和数据共享，即数据采用统一的数据模型进行组织和存储，由专门的管理软件——数据库管理系统 (Data Base Management System, DBMS) 进行统一管理和控制；应用程序在 DBMS 的控制下，采用统一的方式对数据库中的数据进行操作和访问。数据库系统阶段应用程序与数据之间的对应关系如图 1.3 所示，其特点如下。

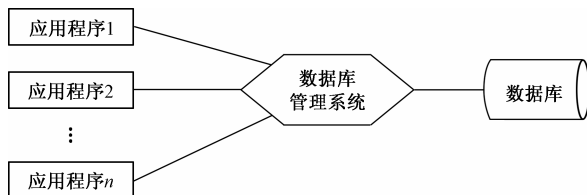


图 1.3 数据库系统阶段应用程序与数据之间的对应关系

1. 数据结构化

数据结构化是数据库系统与文件系统的根本区别，也是数据库系统的主要特征之一。在文件系统中，数据是面向特定应用的，文件的记录内部是有结构的，但各文件间没有联系，不能反映现实世界中各实体间的联系。在数据库系统中，采用统一的数据模型，将数据组织为一个整体；数据不再仅面向特定应用，而是面向全组织的；数据内部不仅是结构化的，而且整体也是结构化的，能较好地反映现实世界中各实体间的联系。这种整体结构化有利于实现数据共享，保证数据和应用程序之间的独立性。

2. 数据共享性高、冗余度低、易于扩充

由于数据库是面向整个系统，而不是面向某个特定应用的，因此数据能够被多个用户、多个应用程序共享。数据库中相同的数据不会多次重复出现，数据冗余度降低，并可避免由于数据冗余度大而带来的数据冲突问题。同时，当应用需求发生改变或增加时，只需重新选择不同的子集，或增加数据即可满足要求。

3. 数据独立性高

数据独立性是指数据的组织和存储与应用程序之间互不依赖、彼此独立的特性，它是数据库领域的一个重要概念。数据独立性包括物理独立性和逻辑独立性。物理独立性是指应用程序与存储于外存储器上的数据是相互独立的，即数据在外存上的存储结构是由 DBMS 管理的，应用程序无须了解；当数据的物理结构发生变化时，应用程序不需改变。逻辑独立性是指应用程序与数据库的逻辑结构是相互独立的，当数据的逻辑结构发生变化时，应用程序可以不改变。

数据独立性是由 DBMS 的二级映像功能来保证的。数据独立于应用程序，降低了应用程序的维护成本。

4. 数据统一管理与控制

数据库中的数据由数据库管理系统（DBMS）统一管理与控制，应用程序对数据的访问均需经由 DBMS。DBMS 必须提供以下 4 个方面的数据控制功能。

（1）并发（Concurrency）访问控制。数据库的共享是并发共享的，多个用户可同时存取数据库中的数据。当多个用户同时存取或修改数据库中的数据时，可能发生相互干扰，导致得到错误结果或破坏数据的完整性。因此 DBMS 必须对多用户的并发操作加以控制。

（2）数据完整性（Integrity）检查。数据完整性是指数据的正确性、有效性和相容性。数据完整性检查的目的是保证数据是有效的，或保证数据之间满足一定的约束关系。

（3）数据安全性（Security）保护。数据库安全性是指保证数据不被非法访问，保证数据不会因非法使用而被泄密、更改和破坏。

（4）数据库恢复（Recovery）。当计算机系统出现硬件、软件故障，或操作员失误以及他人故意破坏时，均可能会影响到数据库的正确性，还有可能造成数据的丢失。当数据库出现故障后，DBMS 应能将其恢复到之前的某一正常状态，这就是数据库的恢复功能。

数据库系统克服了文件系统的缺陷，自 20 世纪 70 年代以来，得到了迅速发展，涌现了许多新产品，得到了广泛应用，成为现代数据管理的主要技术。可以毫不夸张地说，有计算机的地方就有数据库。

1.2 理解数据库系统

本节将从一个简化的“商品订购管理系统”入手，简要介绍对商品订购管理数据库中数据的

访问过程,使读者对数据库系统有直观的认识,然后再介绍什么是数据库系统。

1.2.1 实例——商品订购管理系统

完整的商品订购系统是比较复杂的,本书设计了一个简化的“商品订购管理系统”作为全书的主线实例。

该简易商品订购管理系统的主界面如图 1.4 所示,主界面包含了系统功能的导航菜单。主要功能包括客户数据维护(包括增、删、改)、商品数据维护(包括增、删、改)、订单数据(包括录入、修改、删除、查询)和用户管理等。商品、客户、订单和用户数据均被存储于数据库系统中。

单击“订单数据”“订单数据查询”菜单命令,则应用程序向数据库管理系统发出数据查询请求,由数据库管理系统从商品订购数据库中检索出符合条件的数据,并返回给应用程序,应用程序再以特定的形式显示给用户,如图 1.5 所示。



图 1.4 商品订购管理系统主界面

客户编号	商品编号	订购时间	数量	需票日期	付款方式	送货方式
100001	10010001	2013/2/18 12:20	2	2013/2/20	现金	客户自取
100001	30010001	2013/2/10 12:30	10	2013/2/20	现金	客户自取
100001	30010002	2013/2/10 14:20	1	2013/2/15	信用卡	送货上门
100002	10010001	2013/2/18 13:00	1	2013/2/21	现金	客户自取
100002	50020001	2013/2/18 13:20	1	2013/2/21	现金	客户自取
100004	20180002	2013/2/19 10:00	1	2013/2/28	信用卡	送货上门
100004	30010002	2013/2/19 11:00	10	2013/2/28	信用卡	送货上门
100004	50020002	2013/2/19 10:40	2	2013/2/28	信用卡	送货上门
100005	40010001	2013/2/20 8:00	2	2013/2/27	现金	送货上门
100005	40010002	2013/2/20 8:20	3	2013/2/27	现金	送货上门
100006	10020001	2013/2/23 9:00	5	2013/2/26	信用卡	送货上门

图 1.5 商品订单数据查询

若用户要向系统中添加数据(如添加客户数据),则单击“客户数据维护”命令菜单,出现如图 1.6 所示的客户数据维护界面。

客户编号	客户姓名	出生日期	性别	所在省市	联系电话	是否VIP?	备注
100001	张小林	1979/2/1	男	江苏南京	02561234678	<input checked="" type="checkbox"/>	金牌客户
100002	李虹红	1982/3/22	女	江苏苏州	13908199120	<input checked="" type="checkbox"/>	银牌客户
100003	王晓美	1978/8/20	女	上海市	02186552101	<input type="checkbox"/>	新客户
100004	赵明	1972/3/28	男	河南郑州	13809900118	<input type="checkbox"/>	新客户
100005	张帆一	1980/9/10	男	山东烟台	13809633201	<input type="checkbox"/>	
100006	王芳芳	1986/5/1	女	江苏南京	13709082011	<input type="checkbox"/>	

图 1.6 客户数据维护界面

在各输入框中录入相应的数据项,单击“增加”按钮,则应用系统向数据库管理系统发出数据插入请求,由数据库管理系统向数据库中提交商品数据表格字段的数据值,DBMS 成功执行数据添加操作后,返回正常状态,应用程序再以对话框的形式提示用户操作成功。此时,在客户数据维护界面中便可查看到新增的客户信息,如图 1.7 所示。

从这个实例可知,以数据库为核心的应用系统(也称数据密集型应用),用户对数据库中数据的访问路径为:用户操作命令 应用程序 DBMS 数据库,如图 1.8 所示。

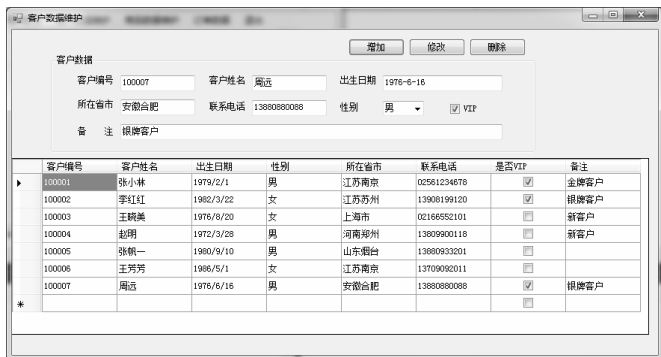


图 1.7 新增加了客户信息

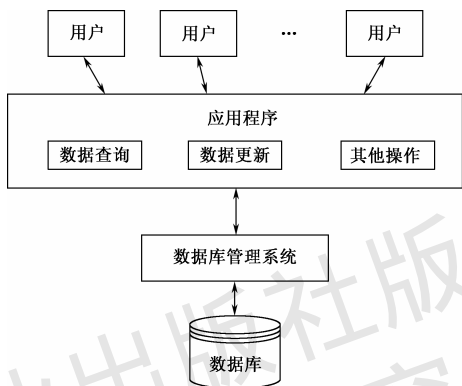


图 1.8 数据访问路径

要设计以数据库为核心的应用系统，必须进行数据库的设计和应用系统的设计。有关数据库设计相关的理论与方法，将在第 4 章讨论。

1.2.2 什么是数据库系统

数据库系统 (DataBase System , DBS) 是指带有数据库并利用数据库技术进行数据管理的计算机系统。数据库系统一般由数据库、数据库管理系统 (及其开发工具) 、数据库管理员 (DataBase Administrator , DBA) 、数据库应用系统和用户组成，如图 1.9 所示。

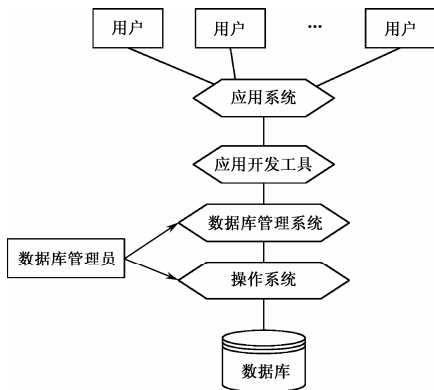


图 1.9 数据库系统组成

数据库 (DataBase, DB) 是长期存储在计算机内的、有组织的、可共享的数据集合。可将数据库看成一个数据高度集成和共享的、基于计算机系统的持久性数据“容器”。

注意：数据库、数据库管理系统、数据库系统三个不同的概念。数据库强调的是相互关联的数据；数据库管理系统强调的是管理数据库的系统软件；而数据库系统强调的是基于数据库技术的计算机系统。

1.3 数据库系统结构

数据库系统产品很多，虽然它们建立于不同的操作系统之上，支持不同的数据模型，采用不同的数据库语言，但它们在体系结构上通常都具有相同的特征，即采用三级模式结构，并提供两级映像功能。

1.3.1 数据库系统的三级模式结构

数据库系统的三级模式结构是指数据库系统是由外模式、模式和内模式三级构成，如图 1.10 所示。

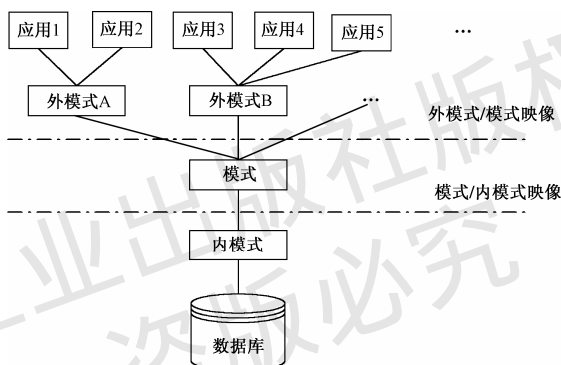


图 1.10 数据库系统的三级模式结构

1. 模式 (Schema)

模式也称逻辑模式或概念模式，它是数据库中全体数据的逻辑结构和特征的描述。模式是面向所有用户的公共数据视图，是数据库的全局视图。一个数据库只有一个模式，它既不涉及物理存储细节、也不涉及应用程序和程序设计语言。定义模式时，不仅要定义数据的逻辑结构，而且还要定义数据之间的联系，以及与数据有关的安全性、完整性要求。

2. 外模式 (External Schema)

外模式也称子模式或用户模式，它是模式的子集。外模式是具体面向应用的，是数据库用户（包括应用程序员和最终用户）所能使用的局部数据的逻辑结构和特征的描述，是数据库用户的数据视图。由于不同的应用有不同的外模式，因此一个数据库可以有多个外模式。

3. 内模式 (Internal Schema)

内模式也称存储模式，它是数据库的物理结构，是数据库在存储介质上的存储结构。内模式主要描述数据的物理结构和存储方式，例如，记录是按 B 树结构还是按 Hash 方式存储，索引如何组织、数据是否加密等。一个数据库只有一个内模式。

数据库系统的三级模式是对数据的三个抽象层次。外模式是面向用户的，反映了不同用户对

所涉及的局部数据的逻辑要求；模式处于中间层，它反映了数据库设计者通过综合所有用户的数据需求并考虑数据库管理系统支持的逻辑数据模型而设计出的数据的全局逻辑结构。内模式处于最低层，它反映了数据在计算机辅助存储器上的存储结构。

数据库系统的这种分层结构把数据的具体组织留给 DBMS 管理，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具体表示方式与存储结构。

1.3.2 数据库系统的二级映像

由图 1.8 所描述的数据访问路径和图 1.10 所描述的数据库系统的三级模式结构可知，当通过应用系统访问数据库中的数据时，应用系统调用外模式，去查找模式中的某一数据；而模式是逻辑上的，对它的访问最终要反映到对外存上数据的操作。要能够顺利地访问数据，必须在外模式与模式之间、模式与内模式之间建立映像关系，这就是数据库系统的二级映像（Mapping）：外模式/模式映像、模式/内模式映像。

（1）外模式/模式映像。对于每一个外模式，数据库系统都有一个外模式/模式映像，它定义了该外模式与模式的对应关系。外模式/模式映像的描述通常包含在外模式中。外模式/模式映像保证了数据的逻辑独立性。当模式发生改变时（如增加新的数据类型或数据项），只要对各外模式/模式映像作相应修改，就可以使外模式保持不变，从而不必修改应用程序。

（2）模式/内模式映像。数据库系统的模式/内模式映像是唯一的，它定义了数据库全局逻辑结构与存储结构之间的对应关系，其描述通常包含在模式定义中。模式/内模式映像保证了数据库的物理独立性。当数据库的存储结构发生改变时，对模式/内模式映像作相应的修改，就可以使模式保持不变，从而应用程序也不必修改。

在数据库系统的三级模式和二级映像结构中，模式是数据库的核心和关键，它独立于数据库的其他层次。因此设计数据库模式是数据库设计的核心任务。内模式不需要数据库设计人员设计，它是由 DBMS 定义好的。对设计好的数据库模式，DBMS 会自动按其定义的内模式进行存储。因此，数据库的内模式依赖于其模式，而独立于其外模式，也独立于具体的存储介质。数据库的外模式是面向具体的应用程序的，需要根据用户需求进行设计。它定义在模式之上，但独立于内模式和存储介质。当用户需求发生变化，相应外模式不能满足应用要求时，该外模式就必须作相应修改，所以设计外模式时应充分考虑到应用的扩展性。应用程序依赖于特定的外模式，不同的应用程序可以共用一个外模式。

数据库系统的三级模式与二级映像具有以下优点：

保证数据独立性。将外模式与模式分开，保证了数据的逻辑独立性；将内模式与模式分开，保证了数据的物理独立性。

有利于数据共享，减少数据冗余。

有利于数据的安全性。不同的用户在各自的外模式下根据要求操作数据，只能对限定的数据进行操作。

简化了用户接口。用户按照外模式编写应用程序或输入命令，而无须了解数据库全局逻辑结构和内部存储结构，方便用户使用。

1.3.3 数据库管理系统

数据库中的数据可以具有海量级别，并且结构复杂，需要进行科学的组织与管理。数据库管理系统（DBMS）就是对数据进行统一管理与控制的专门系统软件。它是位于用户与操作系统之间的一个十分重要的系统软件，其在计算机系统中的地位如图 1.11 所示。对数据库的所有管理，

包括定义、查询、更新等各种操作都需要通过 DBMS 实现。DBMS 是数据库管理的中枢机构，是数据库系统具有数据共享、并发访问和数据独立性的根本保证。

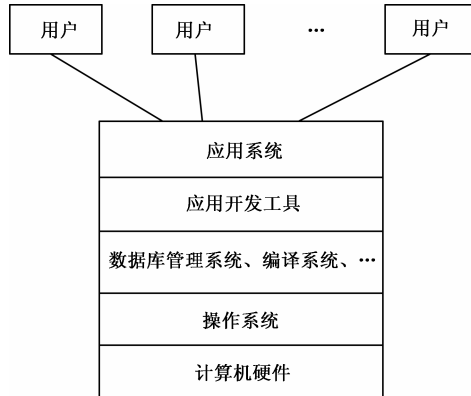


图 1.11 数据库管理系统在计算机系统中的地位

数据库管理系统的主要功能包括以下 6 个方面：

有效地组织、存取和维护数据。

数据定义功能。DBMS 通过数据定义语言（Data Definition Language，DDL）定义数据库的各类数据对象，包括数据的结构、数据约束条件等。

数据操纵功能。DBMS 提供数据操纵语言（Data Manipulation Language，DML），用户使用 DML 实现对数据库中的数据进行查询、增加、删除和修改等操作。

数据库的事务管理和运行管理。DBMS 提供数据控制语言（Data Control Language，DCL），数据库管理员使用 DCL 实现对数据库的安全性保护、完整性检查、并发控制、数据库恢复等数据库控制功能。

数据库的建立和维护功能。

其他功能。包括：数据库初始数据输入与转换、数据库转储、数据库重组、数据库性能监视与分析、数据通信等，这些功能通常由 DBMS 提供的实用程序或管理工具完成。

DDL、DML 和 DCL 统称为数据库子语言（Data Sublanguage）。它们都是非过程性语言，具有两种表现形式：

交互型命令语言。这种方式的语言结构简单，可以在终端上实时操作。又称为自含型（或自主型）语言。

宿主型语言。这种方式通常是将数据库子语言嵌入在某些宿主语言（Host Language）中，如嵌入 C、C++、Java 语言中。

1.3.4 数据库系统所需人员

开发、管理和使用数据库系统的人员主要包括：数据库管理员、系统分析员和数据库设计人员、应用程序员和最终用户。他们各自的职责如下。

1. 数据库管理员（DBA）

DBA 是指对数据库和 DBMS 进行管理的一个或一组人员，负责全面管理和控制数据库系统。其具体职责包括：

参与数据库设计。DBA 参与数据库设计的全过程，与用户、系统分析员和应用程序员共同协商，决定数据库的信息内容、逻辑结构和存取策略等；并优化数据存储结构和存取策略，以

获得较高的存取效率和空间利用率。

数据完整性和安全性管理。包括数据不被破坏的安全策略的制定，数据完整性约束管理等。DBA 负责定义对数据库的存取权限、数据安全级别和完整性约束条件。

数据库运行维护和性能评价。DBA 要维护数据库正常运行，及时处理运行过程中出现的问题。制定数据库维护计划，实施数据库备份和恢复策略，遇故障要及时恢复数据库，并尽可能不影响或减少影响计算机系统其他部分的正常运行。DBA 还要负责监控系统的运行情况，监视系统处理效率、空间利用率等性能指标。

数据库改进和重构。DBA 应对运行情况进行统计分析，并对数据库性能进行评价，提出数据库改进方案。当用户需求增加或改变时，DBA 还要参与数据库的重构。

2. 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格说明，要和用户及 DBA 协商，确定系统的软/硬件配置，并参与数据库系统的概要设计。

数据库设计人员是数据库设计的核心人员，负责数据库中数据内容及结构的确定、数据库各级模式的设计。数据库设计人员必须参加用户需求调研和系统分析，然后进行数据库设计。通常情况下，数据库设计人员是由 DBA 或系统分析员担任的。

3. 应用程序员

应用程序员负责设计和开发数据库应用程序，并负责进行调试和安装。

4. 最终用户 (End User)

最终用户通过应用程序的用户接口使用数据库。常用的接口方式有菜单驱动、表格操作、图形显示等。

1.4 数据模型

1.4.1 数据模型的概念

模型是现实世界特征的抽象与模拟。模型可分为实物模型和抽象模型。建筑模型、汽车模型等都是实物模型，它们是客观事物的某些外观特征或内在功能的模拟与刻画。而数学模型是一种抽象模型，揭示了客观事物的固有规律。如公式 $s = \pi r^2$ ，抽象了圆面积与半径之间的数量关系。

数据模型 (Data Model) 是一种抽象模型，是对现实世界数据特征的抽象。数据模型为数据库系统的信息表示和操作提供必须的抽象框架。计算机上实现的各种数据库管理系统都必须基于某种确定的数据模型。因此，数据模型是数据库系统的灵魂，理解和掌握数据模型是学习数据库技术与理论的基础。

数据模型的选择应满足三方面要求：能较真实地模拟现实世界；易于理解；便于在计算机上实现。然而用一种模型同时满足上述要求是较困难的，因此，在数据库系统中一般是针对不同对象和应用目的采用不同的数据模型。通常，根据实际问题的需要和应用目的的不同，有三种层面上的数据模型，如图 1.12 所示。

(1) 概念数据模型 (Conceptual Data Model)，也称概念模型或信息模型。它是面向用户的模型，是现实世界到

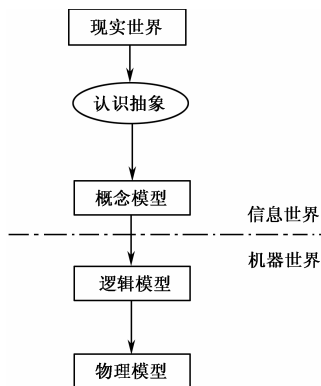


图 1.12 数据抽象过程

机器世界的一个中间层次。其基本特征是按用户观点对信息进行建模,与具体 DBMS 无关。概念模型的作用和意义在于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能够摆脱计算机系统和 DBMS 具体技术的约束,集中精力分析数据及其联系。最常用的概念模型是实体-联系 (E-R) 模型。

(2) 逻辑数据模型 (Logical Data Model), 也称结构数据模型,其特征是按计算机系统的观点对数据建模,服务于 DBMS 的应用实现。结构化数据模型包括:层次模型、网状模型、关系模型等。

(3) 物理数据模型 (Physical Data Model), 用于描述数据在存储介质上的组织结构,它与具体 DBMS 有关,也与操作系统和硬件有关,是物理层次上的数据模型。

1.4.2 概念数据模型

概念数据模型是现实世界到信息世界的抽象,是数据库设计人员与用户进行交流的工具。因此概念数据模型的选择应具有较强的语义表达能力,同时还应简单、清晰、用户易于理解。目前使用较多的概念数据模型描述工具有 E-R 模型、UML 等。在此以 E-R 模型为工具介绍概念数据模型。

P. P. S. Chen 于 1976 年提出了实体-联系方法 (Entity Relationship Approach), 简称 E-R 方法。它简单实用,因而得到了广泛应用。E-R 方法使用的工具是 E-R 图,它所描述的现实世界的信息结构称为 E-R 模型。

1. E-R 模型的三要素

(1) 实体 (Entity)。实体是指客观存在并可相互区别的事物。实体可以是人、事或物,也可以是抽象的概念。如一件商品、一个客户、一份订单等都是实体。

(2) 属性 (Attribute)。实体通常有若干特征,每个特征称为实体的一个属性。属性刻画了实体在某方面的特性。例如,商品实体的属性可以有商品编号、商品类别、商品名称、生产商等。

(3) 联系 (Relationship)。现实世界中事物之间的联系反映在 E-R 模型中就是实体间的联系。例如,订单就是客户和商品之间的联系。

2. 实体型和实体值

在数据库系统中,引入的基本对象通常都有“型 (Type)”和“值 (Value)”之分。“型”是对象特性的抽象描述,“值”是对象的具体内容。

实体型 (Entity Type) 是指对某一类数据结构和特征的描述。通常实体型由实体名和属性名的集合来抽象和刻画同类实体。例如,商品(商品编号,商品类别,商品名称,生产商,单价,库存量,保质期)是一个实体型。

实体值 (Entity Value) 是实体型的内容,由描述实体的各个属性值组成。例如,(50020005,体育用品,足球,美好体育用品公司,120,20,2012-1-1)是实体值。

实体集 (Entity Set) 是指具有相同实体型的若干实体构成的集合。例如,全部商品构成一个实体集。实体集包含了实体的“值”,也隐含了实体的“型”。

通常为了叙述方便,在不引起混淆的情况下,也可不仔细区分实体“型”和“值”,而都称为“实体”。

3. 联系的分类型

从联系的不同层面看,存在实体内部的联系和实体之间的联系。实体内部的联系是指实体集

内部各个实体间的联系。例如，职工实体型内部有领导与被领导的关系。实体之间的联系是指一个实体集中的实体与另一实体集中实体之间的联系。

从联系的表现形式看，联系又分为存在性联系、功能性（也称事件性）联系。

存在性联系：如学校有教师，工厂有车间。

功能性联系：如教师授课，学生选课，客户订购商品。

两个实体型之间的联系有三种：一对一联系、一对多联系、多对多联系。

(1) 一对一联系 (1:1)。如果对于实体集 A 中的任一实体，在实体集 B 中至多有一个实体与之联系；反之亦然，则称实体集 A 与实体集 B 具有一对一联系，记为 1:1。例如，在公司中，一个部门只有一个经理，而一个经理只在一个部门任职，则部门与经理之间具有一对一联系。

(2) 一对多联系 (1:n)。如果对于实体集 A 中的任一实体，在实体集 B 中有 n ($n > 1$) 个实体与之联系；而对于实体集 B 中的每一个实体，实体集 A 中至多有一个实体与之联系，则称实体集 A 与实体集 B 具有一对多联系，记为 1:n。例如，在公司中，一个部门可有多名职工，而一名职工只在一个部门任职，则部门与职工之间具有一对多联系。

(3) 多对多联系 ($m:n$)。如果对于实体集 A 中的任一实体，在实体集 B 中有 n ($n > 1$) 个实体与之联系；而对于实体集 B 中的每一个实体，实体集 A 中有 m ($m > 1$) 个实体与之联系，则称实体集 A 与实体集 B 具有多对多联系，记为 $m:n$ 。例如，在商品订购中，一个客户可订购多种商品，而一种商品也可被多个客户订购，则客户与商品之间具有多对多联系。

由定义可知，一对一联系是一对多联系的特例，一对多联系是多对多联系的特例。

通常，两个以上的实体型之间也存在一对一、一对多、多对多联系。例如，对于课程、教师、参考书三个实体型，如果一门课程可以由多个教师讲授，而一个教师只能讲授一门课程，每本参考书仅为门课程参考，则课程与教师、参考书之间的联系是一对多的。

4. E-R 模型表示——E-R 图

E-R 模型中使用 E-R 图描述实体型之间的联系。在 E-R 图中，用矩形框表示实体型，矩形框内标明实体名；用椭圆框表示实体型的属性；用无向线段连接实体与属性。

【例 1.1】商品实体具有商品编号、商品类别、商品名称、生产商、单价、库存量、保质期等属性，用 E-R 图表示如图 1.13 所示。

E-R 图中用菱形框表示联系，在菱形内写出联系名，用无向边分别与有关实体连接，同时在无向边旁标注联系的类型 (1:1, 1:n, $m:n$)。图 1.14 是两个实体型间三类联系的 E-R 图表示。

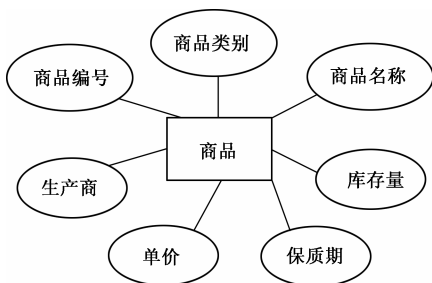


图 1.13 实体及属性表示法

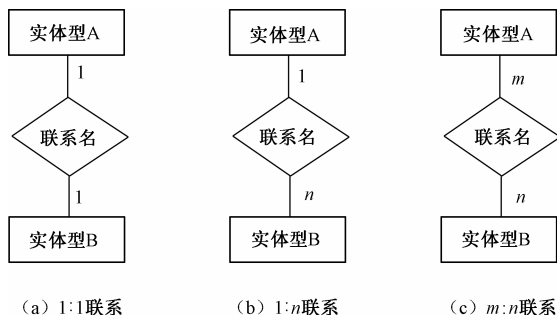


图 1.14 实体间三种联系表示法

若一个联系具有属性，则这些属性也要用无向边与联系连接起来。

【例 1.2】客户订购某类商品均有数量，则实体型“客户”与实体型“商品”之间的联系就具有属性“数量”，其 E-R 图表示如图 1.15 所示。

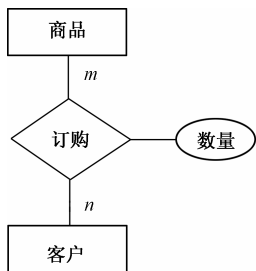


图 1.15 关系的属性

这里只介绍 E-R 图的要点,有关如何认识和分析现实世界,从中抽取实体、实体间联系,建立概念模型等将在第 4 章讲述。

1.4.3 逻辑数据模型

1.4.2 节所讨论的概念数据模型是独立于计算机系统的,表示一种特定组织或机构所关心的“概念数据结构”,完全不涉及信息在计算机中的表示。而逻辑数据模型是数据库管理系统呈现给用户的数据模型,即用户从数据库中看到的数据组织形式,它与 DBMS 直接相关。用概念数据模型描述的数据,必须用逻辑数据模型表示才能由 DBMS 管理。为叙述简便,在下面的讨论中,除非特殊说明,术语“逻辑数据模型”均以“数据模型”表示。

1. 数据模型三要素

逻辑数据模型是严格定义的一组概念的集合,主要由数据结构、数据操作和数据完整性约束三部分组成,通常称为逻辑数据模型的三要素。

(1) 数据结构。数据结构是对系统静态特性的描述,主要描述数据库组成对象以及对象之间的联系。数据结构是刻画数据模型最重要的方面。因此在数据库系统中,通常按照其数据结构的类型来命名数据模型。主要的数据模型有层次模型、网状模型和关系模型。

(2) 数据操作。数据操作指对数据库中各种对象(型)的实例(值)允许执行的操作及有关的操作规则,它是对数据库动态特性的描述。数据库中的数据操作主要分为查询、更新两大类,其中数据更新主要是指对数据记录的增、删、改。数据模型需要定义这些操作的语义、操作符号、操作规则及实现操作的相关语句。

(3) 数据完整性约束。完整性约束是数据的一组完整性规则的集合。完整性规则是给定的数据模型中数据及其联系所具有的制约和存储规则,用以限定符合数据模型的数据库状态以及状态的变化,保证数据的正确、有效、相容。

2. 数据模型的类型

数据模型的发展带动了数据库系统的更新换代。自 20 世纪 60 年代末数据库技术产生以来,先后出现了层次模型(Hierarchical Model)、网状模型(Network Model)、关系模型(Relational Model)、面向对象模型(Object Oriented Model)、对象关系模型(Object Relational Model)等逻辑数据模型。其中层次模型和网状模型统称为格式化模型。

由于格式化模型现已不是主导,本书不再讨论。下面简要介绍关系数据模型,而本书后面章节讨论的重点将是关系数据库系统。

3. 关系数据模型

关系模型源于数学,由于关系模型具有完备的关系理论作为基础,因此自 20 世纪 80 年代以来,关系数据模型被普遍采用,关系模型的数据库系统得到了广泛应用。

(1) 数据结构。关系模型建立在严格的数学概念之上,有关定义将在第 2 章给出,这里仅非形式化地讨论。关系模型的基本数据结构是表格,它使用二维表来表示实体及其联系。

(2) 数据操作。关系模型的数据操作主要包括查询和更新。关系数据操作具有两个显著特点:一是关系操作是集合操作,即操作的对象和结果均为集合;二是关系模型将操作中的存取路径向用户屏蔽起来,用户只要说明做什么,而不必指出怎样做。

(3) 数据完整性约束。对关系模型中的数据操作必须满足关系完整性约束规则。关系完整性

约束规则包括三类：实体完整性、参照完整性和用户定义完整性。关系完整性的内容将在后面章节详细讨论。

关系模型主要有以下特点：

- (1) 关系模型建立在严格的数学理论基础之上。
- (2) 数据结构简单清晰，用户易懂易用。关系模型的数据结构虽然简单，但却能表达丰富的语义，能够较好地描述现实世界的实体以及实体间的各种联系。
- (3) 数据物理存取路径对用户是透明的，有更高的数据独立性、更好的数据安全性。

关系数据库在各个领域的广泛应用推动了数据库技术的发展。但关系模型描述能力也存在不足，如难以直接描述超文本、图像、声音等复杂对象；难以表达工程、地理、测绘等领域一些非格式化的数据语义；不能提供用户定义复杂类型及数据抽象的功能等。因此人们又提出了面向对象模型和对象关系模型等新的数据模型，并得到了一定的支持。

1.5 数据库系统的发展

1.5.1 数据库系统发展的阶段

数据库技术是计算机科学与技术中发展最快的分支之一。从 20 世纪 60 年代末数据库技术诞生至今，按照数据模型发展的阶段划分，数据库系统的发展大致可划分为三代。

1. 第一代数据库系统

采用格式化模型的数据库系统属于第一代数据库系统。格式化模型的数据库系统在 20 世纪 70 年代至 80 年代初非常流行，后逐渐被关系数据库系统所取代。层次模型数据库系统的典型代表是 IBM 公司的 IMS (Information Management System)；网状模型的典型代表是 DBTG 系统，它是 20 世纪 70 年代数据库系统语言研究会 CODASYL (Conference On Data Systems Language) 下属的数据库任务组 (Data Base Task Group) 提出的一个系统方案。

第一代数据库系统的主要特点有：

- (1) 支持三级模式体系结构。
- (2) 用存取路径表示实体间的联系。层次模型用有向树结构表示实体以及实体间的联系；网状模型用有向图结构表示实体以及实体间的联系。
- (3) 导航式的数据操纵语言。导航指应用程序需一步步地按照数据库中预先定义的存取路径来访问数据库，最终到达要访问的数据目标。访问数据库时，每次只能存取一条记录。

2. 第二代数据库系统

第二代数据库系统是指支持关系模型的关系数据库系统。1970 年，美国 IBM 公司 San Jose 研究室的研究员 E.F.Codd 提出数据库的关系模型，之后，又提出了关系代数和关系演算以及范式的概念，开始了数据库关系方法和关系理论的研究，这是对数据库技术的一个重大突破。E.F.Codd 的工作奠定了关系数据库的理论基础，为此他获得了 1981 年的 ACM 图灵奖。在 20 世纪 70 年代末，关系数据库的软件系统研制也取得了成果，最具代表性的实验系统有 IBM 公司研制的 System R 和美国加州大学伯克利分校研制的 INGRES，商用系统则有由 System R 发展而来的 SQL/DS 以及由 INGRES 实验系统发展而来的 INGRES 关系数据库软件产品。自 20 世纪 80 年代以来，数据库管理系统产品几乎都是支持关系模型的，数据库领域当前的很多研究工作也都是以关系方法为基础的。经过 30 多年的历程，关系数据库系统的研究和开发取得了辉煌的成就。关系数据库成为最重要、应用最广泛的数据库系统，如广泛使用的小型数据库系统 Foxpro、Access，大型数据库

系统 Oracle、SQL Server、Sybase、DB2、Informix 等都是关系数据库系统。

第二代数据库系统的主要特点有：

- (1) 概念单一，实体以及实体之间的联系都用关系表示。
- (2) 以关系代数为基础，形式化基础好。
- (3) 数据独立性强，数据的物理存取路径对用户屏蔽。

(4) 关系数据语言实现了标准化，即创建了结构化查询语言 SQL (Structured Query Language)。关系数据语言是非过程化的，它将用户从数据库记录的导航式检索中解脱出来，大大降低了编程的难度。

通常，将第一代数据库系统和第二代数据库系统称为传统数据库系统。由于传统数据库系统特别是关系数据库系统具有许多优点，它们被广泛用于数据管理，并被应用到许多新领域。如计算机辅助设计/计算机辅助制造 (CAD/CAM)、计算机辅助工程 (CASE)、地理信息处理、智能信息处理等，这些新领域的应用不仅需要传统数据库所具有的快速检索和修改数据的特点，而且提出了一些新的数据管理需求，如要求数据库能够处理声音、图像、视频等多媒体数据。因此传统数据库在这些新领域中暴露了其局限性，已经不能完全满足应用的需要。在这种情况下，新一代数据库技术应运而生。

3. 第三代数据库系统

第三代数据库系统是指以更丰富的数据模型、更强大的数据管理能力为特征，满足更广泛更复杂的新应用需求的各类数据库系统的大家族。这些新的数据库系统包括：面向对象数据库、分布式数据库、并行数据库、工程数据库、统计数据库、空间数据库等。

1990 年，高级 DBMS 功能委员会发表了《第三代数据库系统宣言》，提出了第三代数据库管理系统的三个基本特征：

- (1) 第三代数据库系统应支持数据管理、对象管理和知识管理。
- (2) 第三代数据库系统必须保持或继承第二代数据库系统的技术。即必须保持第二代数据库系统的非过程化数据存取方式和数据独立性等特性。
- (3) 第三代数据库系统必须对其他系统开放。数据库系统的开放性表现在：支持数据库语言标准，支持网络标准，系统具有良好的可移植性、可连接性、可扩展性和互操作性等。

1.5.2 数据库系统主要研究领域

随着计算机软/硬件和计算机网络的发展，数据库技术仍需不断向前发展。数据库系统的研究范围很广泛，概括起来主要包括以下三个方面。

(1) DBMS 软件研制。DBMS 是数据库系统的基础与核心，开发可靠性好、效率高、功能齐全的 DBMS 始终是数据库领域研究的重要内容。并且为了充分发挥数据库的应用功能，还需开发一些必须能在 DBMS 上运行的软件系统，包括数据通信软件、报表书写系统、表格系统和图形系统等。因此研制以 DBMS 为核心的一组相互关联的软件系统或工具包也是当前数据库软件产品的发展方向。

(2) 数据库应用系统设计与开发。在 DBMS 支持下，设计与开发满足用户要求的数据库应用系统，这是数据库领域研究的另一个重要内容。其目标是：按照用户需求，为某一部门或组织设计和开发功能强大、效率高、使用方便和结构优良的数据库及其配套的应用系统。数据库应用系统设计与开发的主要研究课题有：数据库设计方法、自动化设计工具和设计理论的研究、数据建模的研究、计算机辅助设计方法及其软件系统的研究、数据库设计规范和标准等。

(3) 数据库基础理论。自关系数据模型提出以来，很长一段时间内，数据库理论研究主要集

中在关系数据理论上,包括关系数据模型、规范化理论等。随着面向对象模型等新的数据模型的提出,近年来也开始了面向对象数据库、空间数据库、多媒体数据库等方面的理论研究。随着人工智能与数据库技术的结合,演绎数据库和知识库系统的研究成为数据库理论新的研究热点。从数据库向知识库发展将是一个新的发展趋势。

本章小结

本章简要地介绍了数据管理技术发展的三个阶段,使读者对数据库技术的产生背景以及所要解决的问题有了初步认识。以对一个商品订购数据库中内容的访问过程为例,讲解数据库系统的构成和处理过程,使读者对数据库系统有直观的认识。简要介绍了数据库系统的组成,使读者了解数据库系统不仅是一个计算机系统,而且是一个“人-机”系统。

阐述了数据库系统的一些基本但重要的概念和术语,包括:数据库系统的三级模式结构、数据库系统、数据库、数据库管理系统、模式、内模式、外模式、二级映像、数据独立性、视图、用户、DBA等。使读者初步认识数据库领域的常用概念和术语。

数据模型是数据库系统的核心和基础。本章介绍了数据模型的概念,突出数据模型在数据库技术中的重要作用;阐述概念数据模型、逻辑数据模型,重点讲解了E-R模型和关系模型的特点。使读者对数据结构化这一数据库系统的基本特征有所认识。

最后简单介绍了数据库系统的发展历程,给出数据库领域研究的主要方面,包括:数据库理论、数据库管理系统研制、数据库应用系统设计与开发等,使读者对数据库领域的研究内容有较为全面的了解。

习题 1

1. 简述数据库系统的特点。
2. 什么是数据库系统?
3. 简述数据库系统的组成。
4. 试述数据库系统的三级模式结构。这种结构的优点是什么?
5. 什么是数据的物理独立性与逻辑独立性?并说明其重要性。
6. 数据库管理系统的功能主要有哪几方面?
7. 数据库系统的人员主要包括哪些?
8. 什么是数据模型?
9. 什么是概念数据模型?E-R模型的三要素是什么?
10. 举例说明联系的三种类型。
11. 什么是逻辑数据模型?逻辑数据模型的三要素是什么?
12. 简述关系模型的特点。