

第 1 章 数据库系统概述

本章要点

1. 了解数据管理技术的发展,其中重点是数据库系统阶段。理解数据库系统的主要特点,初步了解数据库系统。
2. 关系数据库系统是当前数据库系统的主流,也是本书的重点。初步理解关系、关系模型、关系数据库系统等基本概念。
3. 了解数据库管理系统的组成和各部分的基本功能。
4. 初步理解本章提到的有关数据库的专业术语,为后续各章的学习打下好的基础。

数据库技术是计算机应用领域中非常重要的技术,它产生于 20 世纪 60 年代末,是数据管理的最新技术,也是软件科学的一个重要分支。本章首先回顾数据管理技术的发展过程,然后介绍数据库技术的基本术语,并在此基础上介绍关系数据库系统和数据库系统结构以及数据库管理系统的体系结构。

1.1 数据管理技术的发展

数据管理指的是如何对数据进行分类、组织、储存、检索及维护。要注意,这里所说的数据,不仅是指数字,还包括文字、图形、图像、声音等。凡是计算机中用来描述事物的记录,统称为数据。

随着计算机软硬件的发展,数据管理技术不断地完善,经历了如下三个阶段:

- (1) 人工管理阶段;
- (2) 文件系统阶段;
- (3) 数据库系统阶段。

1.1.1 人工管理阶段

20 世纪 50 年代中期以前,计算机主要用于科学计算。那时的计算机硬件方面,外存只有卡片、纸带及磁带,没有磁盘等直接存取的存储设备;软件方面,只有汇编语言,没有操作系统和高级语言,更没有管理数据的软件;数据处理的方式是批处理。这些决定了当时的数据管理只能依赖人工来进行。

人工管理阶段的特点是:

- (1) 数据不进行保存。当时的计算机主要用于科学计算,一个程序对应一组数据。在计算某一问题时,把程序和对应的数据装入,计算完就退出,没有将数据长期保存的必要。
- (2) 没有专门的数据管理软件。数据需要由应用程序自己管理,因此应用程序的设计者不仅要考虑数据的逻辑结构,还要考虑数据的物理结构,比如存储结构、存取方法、输入输

出方式等等。一旦存储结构发生变化,应用程序也要做相应的修改,程序员的负担非常重,数据的独立性也很差。

(3) 数据面向应用。一组数据对应一个程序。倘若多个程序使用相同的数据,必须各自定义,不能共享。所以程序之间存在大量的数据冗余。

(4) 只有程序的概念,基本上没有文件的概念。
人工管理阶段的特征如图 1.1 所示。

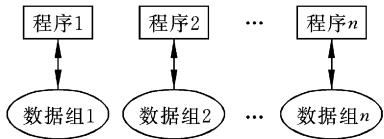


图 1.1 人工管理阶段的特征

1.1.2 文件系统阶段

20 世纪 50 年代末到 60 年代中期,随着科学技术的进步,计算机技术有了很大提高,计算机的应用范围也不断扩大,不仅用于科学计算,还大量用于管理。这时计算机硬件已经有了磁盘、磁鼓等直接存取的外存设备;软件则有了操作系统、高级语言,操作系统中的文件系统是专门用于数据管理的软件;处理方式不仅有批处理,还增加了联机实时处理。

文件系统阶段的特点如下:

(1) 数据可以长期保存在磁盘上。用户可以反复对文件进行查询、修改、插入和删除等操作。

(2) 文件系统提供了数据与程序之间的存取方法。应用程序和数据有了一定的独立性。数据物理结构的改变也不一定反映在程序上,大大减轻了程序员的负担。

(3) 数据冗余量大。文件系统中,文件仍然是面向应用的,一个文件基本上对应于一个应用程序。即使多个程序使用了一部分相同的数据,也必须建立各自的文件,不能对数据项进行共享,因此数据冗余大,存储空间浪费。由于数据可能有多个副本,对其中之一进行修改时还容易造成数据的不一致性。

(4) 文件之间缺乏联系,相互孤立,仍然不能反映现实世界各种事物之间错综复杂的联系。

文件系统阶段的特征如图 1.2 所示。

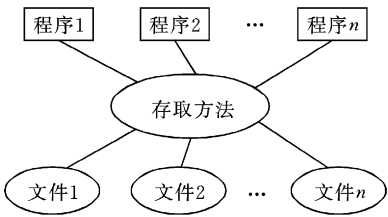


图 1.2 文件系统阶段的特征

1.1.3 数据库系统阶段

20 世纪 60 年代末以来,计算机的应用更为广泛,用于数据管理的规模也更为庞大,由此带来数据量的急剧膨胀。计算机磁盘技术有了很大发展,出现了大容量的磁盘。在处理方式上,联机实时处理的要求更多。这种种变化都促进了数据管理手段的进步,数据库技术应运而生。

数据库系统的特点如下:

(1) 数据的结构化

在文件系统阶段,只考虑了同一文件记录内部数据项之间的联系,而不同文件的记录之间是没有联系的,也就是说,从整体上看数据是无结构的。如上一节所述,这样的文件是有局限性的,不能反映现实世界各种事物之间错综复杂的联系。在数据库系统中,实现了整体

数据的结构化,把文件系统中简单的记录结构变成了记录 and 记录之间的联系所构成的结构化数据。在描述数据的时候,不仅要描述数据本身,还要描述数据之间的联系。数据之间的联系通过存取路径来实现,把相关的数据有机地组织在一起。

例如在学校的管理系统中,不同的部门有不同的要求,人事、医疗、教务等部门分别了解学生的人事情况、医疗保健情况、选课情况等等。传统的文件系统中,不同的应用要使用不同的文件。比较简单的文件形式是等长、同格式记录的集合。比如学生的人事记录文件,可以采用图 1.3 所示的记录格式。

学号	姓名	性别	出生年月	系别	政治面貌	籍贯	家庭成员	简历
----	----	----	------	----	------	----	------	----

图 1.3 学生人事记录

而学生的选课记录文件,则可以采用图 1.4 所示的记录格式。

学号	姓名	性别	出生年月	系别	课程号	课程名	成绩
----	----	----	------	----	-----	-----	----

图 1.4 学生选课记录

由图 1.3 和图 1.4 可见,首先,每个学生的情况不同,其家庭成员、简历、选课的数据量有多有少,如果用等长记录格式存储学生数据,只能按数据量最大的学生记录来安排存储,这样会造成极大的浪费,如果用变长记录来存储,又不便于数据管理;其次,无论是人事记录文件还是选课记录文件,每个文件记录的数据项都包括了学号、姓名、性别和出生年月等,这造成了大量的重复存储。

在数据库系统中,从整体的角度来组织数据,综合考虑各种应用,有效地解决了上述问题。数据组织方式如图 1.5 所示。

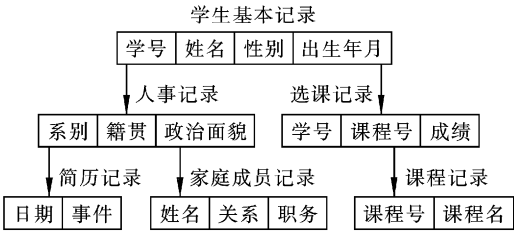


图 1.5 结构化的学生记录

(2) 数据共享性好

由图 1.5 设计的数据结构可见,人事部门可以据此了解学生的人事情况,教务部门也可以据此了解学生的选课情况,这些数据可以供多个部门使用,实现了数据的共享。各个部门的数据基本上没有重复的存储,数据的冗余量较小。

(3) 数据独立性好

数据库系统有三层结构:用户(局部)数据的逻辑结构、整体数据的逻辑结构和数据的物理结构。在这三层结构之间数据库系统提供了两层映像功能。首先是用户数据逻辑结构和整体数据逻辑结构之间的映像,这一映像保证了数据的逻辑独立性:当数据库的整体逻辑结构发生变化时,通过修改这层映像可使局部的逻辑结构不受影响,因此不必修改应用程

序。另外一层映像是整体数据逻辑结构和数据物理结构之间的映像,它保证了数据的物理独立性;当数据的存储结构发生变化时,通过修改这层映像可使数据的逻辑结构不受影响,因此应用程序同样不必修改。

(4) 数据存取粒度小

文件系统中,数据存取的最小单位是记录;而在数据库系统中,数据存取的粒度可以小到记录中的一个数据项。因此数据库中数据存取的方式非常灵活,便于对数据的管理。

(5) 数据库管理系统(database management system,DBMS)对数据进行统一的管理和控制

DBMS 不仅要有最基本的数据管理功能,还要有如下的控制功能:

① 数据的完整性。保证数据的正确性,要求数据在一定的取值范围内或相互之间满足一定的关系。比如规定考试的成绩在 0 分到 100 分之间,血型只能是 A 型、B 型、AB 型、O 型中的一种等等。

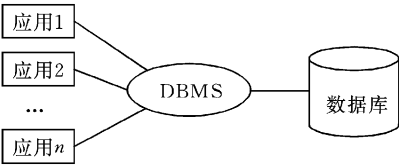
② 数据的安全性。让每个用户只能按指定的权限访问数据,防止不合法地使用数据,造成数据的破坏和丢失。比如学生对于课程的成绩只能进行查询,不能修改。

③ 并发控制。对多用户的并发操作加以协调和控制,防止多个进程同时存取、修改数据库中的数据时发生冲突、造成错误。比如在学生选课系统中,某门课只剩下最后一个名额,但有两个学生在两台选课终端上同时发出了选这门课的请求,必须采取某种措施,确保两名学生不能同时拥有这最后的一个名额。

④ 数据库的恢复。当数据库系统出现硬件软件的故障或者遇上误操作时,DBMS 应该有能力把数据库恢复到最近某个时刻的正确状态上来。

(6) 为用户提供了友好的接口

用户可以使用交互式的命令语言,如将在第 4 章介绍的 SQL(structured query language,结构化查询语言)语言,对数据库进行操作;也可以把普通的高级语言(如 C++ 语言等)和 SQL 语言结合起来,从而把对数据库的访问和对数据的处理有机地结合在一起。总而言之,用户可以很方便地对数据进行管理。



注: 这里所说的应用,既包括应用程序,也包括用户联机交互式操作

图 1.6 数据库系统阶段的特征

数据库系统阶段的特征如图 1.6 所示。

1.2 有关数据库的基本术语

现在回顾一下 1.1 节提到过的与数据库技术密切相关的几个最基本的术语,并进一步加深对它们的理解。

1.2.1 数据

在计算机领域内,数据(data)这个概念已经不局限于普通意义上的数字,还包括文字、图形、图像、声音等等。凡是计算机中用来描述事物的记录,都可以统称为数据。比如用学

号、姓名、年龄、系别这几个特征来描述学生时,(9900001,王红,19,计算机系)这一记录就是一个学生的数据。

1.2.2 数据模型

为了用计算机处理现实世界中的具体事物,往往要对客观事物加以抽象,提取主要特征,归纳形成一个简单清晰的轮廓,从而使复杂的问题变得易于处理,这就是“建模”——建立模型的概念。数据模型就是一种对客观事物抽象化的表现形式。数据模型,首先,要真实地反映现实世界,否则就没有实际意义了;其次,要易于理解,和人们对外部事物的认识相一致;最后,要便于实现,因为最终是要由计算机来处理。

数据模型通常由数据结构、数据操作和完整性约束三要素组成。

数据结构描述的是系统的静态特性,是所研究对象的类型的集合。由于数据结构反映了数据模型最基本的特征,因此,人们通常都按照数据结构的类型来命名数据模型。传统的数据模型有层次模型、网状模型和关系模型。近年来,对象模型得到广泛应用。

数据操作描述的是系统的动态特性,是对各种对象实例允许执行的操作的集合。数据操作主要分更新和检索两大类,更新包括插入、删除、修改。两类统称增、删、改、查。

完整性约束的目的是保证数据的正确性、有效性和相容性。例如,在关系模型中,任何关系都必须满足实体完整性和引用完整性这两个条件。

1.2.3 数据库

数据库(database,DB),实际上,就是按照一定的数据模型组织的、长期储存在计算机内、可为多个用户共享的数据的聚集。在引入了数据库管理系统(DBMS)这个概念之后,可以认为,数据库就是由 DBMS 统一管理和控制的数据的聚集。

1.2.4 数据库管理系统

数据库管理系统(database management system,DBMS),是专门用于建立和管理数据库的一套软件,介于应用程序和操作系统之间。DBMS 不仅具有最基本的数据管理功能,还能保证数据的完整性、安全性,提供多用户的并发控制,当数据库出现故障时对系统进行恢复。

1.2.5 数据库系统

数据库系统(database system,DBS),它包括和数据库有关的整个系统:数据库、DBMS、应用程序以及数据库管理员和用户等等。当然,人们也常把除人以外与数据库有关的硬件和软件系统称为数据库系统。一个数据库系统应该是:

(1) 允许用户用一种叫做“数据定义语言”的专用语言来建立新的数据库。

(2) 允许用户用一种叫做“数据操作语言”或者“查询语言”的专用语言来对数据库中的数据进行查询和更新。

(3) 支持存储大量的数据,保证对数据的正确及安全使用。

(4) 控制多用户的并发访问,保证并发访问不相互影响,不损坏数据。

由于数据模型是数据库系统的基础,因此人们就按数据模型来命名数据库系统,如数据

模型为层次模型、网状模型或关系模型，则相应的数据库系统就称为层次数据库系统、网状数据库系统或关系数据库系统。

1.3 关系数据库系统

1970 年,美国 IBM 公司的 Ted Codd 发表了一篇著名的论文,文中首次提出了关系数据库的概念。在此以前,先后出现过层次数据库系统和网状数据库系统,这两种数据库目前仍有少量应用,但关系数据库早已占据了主导地位。

1.3.1 什么是关系

所谓关系,就是一张表。表的各列以属性开始,属性是列的入口。

例:表 1. 1 是一个名为 Course(课程)的关系,记录的是课程的相关信息,包括 CourseNo(课程号)、CourseName(课程名)、Teacher(任课老师)。

表 1.1 Course(课程)关系

CourseNo(课程号)	CourseName(课程名)	Teacher(任课老师)
1234	C++ 程序设计	张明
5678	数据库原理	张明
.....		

表头是三个属性: CourseNo, CourseName 和 Teacher。属性下面的每一行,称为一个元组。这个例子给出了两个具体的元组,其他的元组都省略了,每个元组对应了一门课程。第 1 个元组对应了课程号为 1234 的课程,课程名是“C++ 程序设计”,任课老师是张明;第 2 个元组对应了课程号为 5678 的课程,课程名是“数据库原理”,任课老师也是张明。

如果根据关系中的某个属性或属性集能惟一确定一个元组,则把这个属性(集)称为关系的键码。在上面的例子中规定,在为课程编排课程号时,各门课程的课程号互不相同,那么根据课程号 CourseNo 这个属性就能惟一确定一个元组;例如给出课程号“1234”,就能确定这是张明老师任教的“C++ 程序设计”课,因此 CourseNo 是 Course 关系的键码。反之,如果有两门课由同一位老师任教,比如上例,那么给出任课老师“张明”,仍不能确定这到底是“C++ 程序设计”课,还是“数据库原理”课,因此 Teacher 就不是 Course 关系的键码。

属性是有一定的取值范围的。上例中可以规定课程号的取值范围是 0000~9999 之间的整数;课程名和任课老师的取值只能是长度不超过 20 的字符串等。属性的取值范围又叫做属性的域。

例:一个名为 StudentCourse(学生选课)的关系,记录的是学生的选课信息,包括三个属性: StudentNo(学号)、CourseNo(课程号)、Score(成绩)。表 1. 2 给出几个具体的数据。

第 1 个元组表示学号为 9900011 的学生选修了课程号为 1234 的课程,考试成绩为 89;第 2 个元组表示学号为 9900548 的学生选修了课程号为 5678 的课程,考试成绩为 73。当然还有很多未列出的元组。

表 1.2 StudentCourse(学生选课)关系

StudentNo(学号)	CourseNo(课程号)	Score(成绩)
9900011	1234	89
9900548	5678	73
.....		

说明：关系不仅可以用来表示某个实体(如课程、学生),也可以用来表示实体和实体之间的联系(如学生选课)。

1.3.2 关系模型

数据以“关系”的形式表示,也就是以二维表的形式表示,其数据模型就是所谓的关系模型。在关系模型中,无论是从客观事物中抽象出的实体,还是实体之间的联系,都用单一的结构类型——关系来表示。在对关系进行各种处理之后,得到的还是关系——一张新的二维表。由于关系模型中数据结构简单清晰,概念单一,易学易用,深受用户喜爱。

1.3.3 关系数据库系统

关系数据库系统就是以关系模型为基础的数据库系统。由于关系模型有严格的数学基础,而且简单清晰,便于理解和使用。因此,关系数据库系统推出以后,迅速得到推广应用,目前已在数据库系统中占统治地位。国外大公司推出的关系数据库产品主要有:微软公司的 MS SQL Server,Oracle 公司的 Oracle,IBM 公司的 DB2 Universal Database,Informix 公司的 IDS (Informix Dynamic Server),Sybase 公司的 ASE (Adaptive Server Enterprise)等。

1.4 数据库系统的体系结构

数据库系统的体系结构从不同的角度可有不同的划分方式。从数据库管理系统的角度可分为三层,从外到内依次为外模式、模式和内模式。

数据库的三层结构是数据的三个抽象级别,用户只要抽象地处理数据,而不必关心数据在计算机中如何表示和存储。

为了实现三个抽象级别的联系和转换,数据库管理系统在三层结构之间提供了两层映像:外模式/模式映像和模式/内模式映像。

下面分别介绍三层模式结构和两层映像功能。

1.4.1 三层模式结构

1. 外模式

外模式(external schema)又称为用户模式,是数据库用户和数据库系统的接口,是数据库用户的数据视图(view),是数据库用户可以看见和使用的局部数据的逻辑结构和特征的描述,是与某一应用有关的数据的逻辑表示。

一个数据库通常都有多个外模式。当不同用户在应用需求、保密级别等方面存在差异时,其外模式描述就会有所不同。一个应用程序只能使用一个外模式,但同一外模式可为多个应用程序所使用。

外模式是保证数据库安全的重要措施。每个用户只能看见和访问所对应的外模式中的数据,而数据库中的其他数据均不可见。

2. 模式

模式(schema)又可细分为概念模式(conceptual schema)和逻辑模式(logical schema),是所有数据库用户的公共数据视图,是数据库中全部数据的逻辑结构和特征的描述。

一个数据库只有一个模式。其中概念模式可用实体-联系模型来描述,逻辑模式以某种数据模型(比如关系模型)为基础,综合考虑所有用户的需求,并将其形成全局逻辑结构。模式不但要描述数据的逻辑结构,比如数据记录的组成,各数据项的名称、类型、取值范围,而且要描述数据之间的联系、数据的完整性、安全性要求。

3. 内模式

内模式(internal schema)又称为存储模式(storage schema),是数据库物理结构和存储方式的描述,是数据在数据库内部的表示方式。

一个数据库只有一个内模式。内模式描述记录的存储方式、索引的组织方式、数据是否压缩、是否加密等。但内模式并不涉及物理记录,也不涉及硬件设备,比如,对硬盘的读写操作是由操作系统(其中的文件系统)来完成的。

在三层模式结构中,数据库模式是数据库的核心与关键,外模式通常是模式的子集。数据按外模式的描述提供给用户,按内模式的描述存储在硬盘上,而模式介于外、内模式之间,既不涉及外部的访问,也不涉及内部的存储,从而起到隔离作用,有利于保持数据的独立性。内模式依赖于全局逻辑结构,但可以独立于具体的存储设备。

1.4.2 两层映像功能

所谓映像(mapping)就是一种对应规则,说明映像双方如何进行转换。

1. 外模式/模式映像

通过外模式与模式之间的映像把描述局部逻辑结构的外模式与描述全局逻辑结构的模式联系起来。由于一个模式与多个外模式对应,因此,对于每个外模式都有一个外模式/模式映像用于描述该外模式与模式之间的对应关系。外模式/模式映像通常放在外模式中描述。

有了外模式/模式映像,当模式改变时,比如增加新的属性、修改属性的类型,只要对外模式/模式映像做相应的改变,使外模式保持不变,则以外模式为依据的应用程序就不受影响,从而保证了数据与程序之间的逻辑独立性,也就是数据的逻辑独立性。

2. 模式/内模式映像

通过模式与内模式之间的映像把描述全局逻辑结构的模式与描述物理结构的内模式联

系起来。由于数据库只有一个模式,也只有一个内模式,因此,模式/内模式映像也只有一个,通常就放在内模式中描述。

有了模式/内模式映像,当内模式改变时,比如存储设备或存储方式有所改变,只要对模式/内模式映像做相应的改变,使模式保持不变,则应用程序就不受影响,从而保证了数据与程序之间的物理独立性,也就是数据的**物理独立性**。

从上面的介绍可以看出,由于有两层映像,在内模式发生变化,甚至模式发生变化时,都可以使外模式在最大限度上保持不变。由于应用程序是在外模式所描述的数据结构的基础上编写的,外模式的稳定性就保证了应用程序的稳定性。而这正是数据库结构采用三层模式、两层映像为系统提供了高度的数据独立性所得到的结果。

数据库系统的三层模式结构如图 1.7 所示。

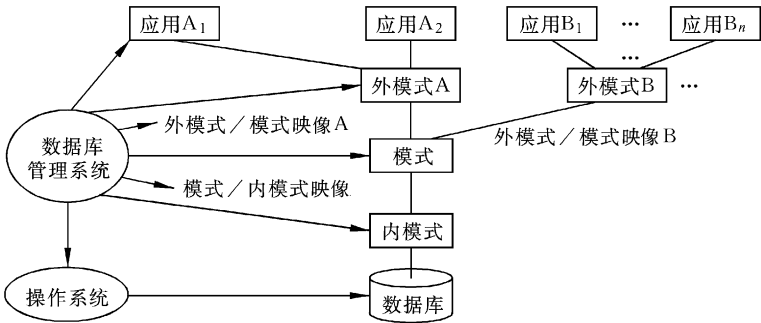


图 1.7 数据库系统的三层模式结构

1.5 DBMS 的体系结构

本节将简要介绍典型的 DBMS 的结构,并且说明 DBMS 是如何处理用户的查询及其他数据库操作的。

1.5.1 DBMS 的组成概述

DBMS 的主要组成部分如图 1.8 所示。图 1.8 的最上方,是三种类型的 DBMS 输入。

(1) 查询

对数据的查询有两种生成方式。一是通过通用的查询接口,比如关系数据库管理系统允许用户输入 SQL 查询语句,然后将查询传给查询处理程序,并给出回答;二是通过应用程序的接口,典型的 DBMS 允许程序员通过应用程序调用 DBMS 来查询数据库。

(2) 更新

对数据的插入、修改和删除等操作统称为更新。对数据的更新和对数据的查询一样,也可以通

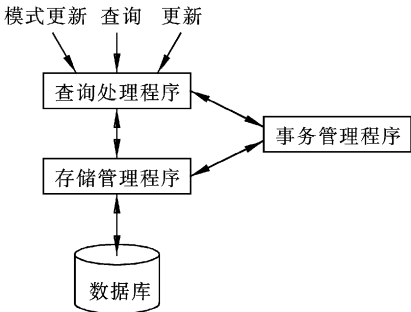


图 1.8 DBMS 的主要组成部分

过通用接口或应用程序接口来提出。

(3) 模式更新

所谓数据库的模式,就是指数据的逻辑结构。模式更新命令一般只能由数据库管理员使用。比如学生选课系统要求能提供课程的上课地点,就要在课程关系中加入一个新的属性——上课地点(address),这就是对模式的更新。

接下来是查询处理程序,其实这个名称不甚贴切,因为它不仅负责查询,也负责发出更新数据或模式的请求。查询处理程序的功能是接受到一个操作的请求后,找到一个最优的执行方式,然后向存储管理程序发出命令,使其执行。存储管理程序的功能是从数据库中获得上层想要查询的数据,并根据上层的更新请求更新相应的信息。

事务管理程序负责系统的完整性。它必须保证同时运行的若干个数据库操作不互相冲突,保证系统在出现故障时不丢失数据。事务管理程序要和查询处理程序互相配合,因为它必须知道当前将要操作的数据,以免出现冲突。为了避免发生冲突,还可能需要延迟某些操作。事务管理程序也要和存储管理程序互相配合,因为数据库恢复一般需要一个日志文件,记录每一次数据的更新,这样即使系统出现故障,也能有效而可靠地进行恢复。

1.5.2 查询处理程序

查询处理程序的任务是把用较高级的语言所表示的数据库操作(包括查询、更新等)转换成一系列对数据库的请求。查询处理最复杂和最重要的部分是查询优化,也就是选择一个好的查询计划,从而尽可能地减少开销,使用户的操作尽快完成,得到结果。

例如我们现在想要查询张明老师所教的课程中,有哪些同学成绩在 60 分以下。

一种可能的查询方案是先按 Score(成绩)属性检索 StudentCourse 关系,找出所有成绩低于 60 分的元组,再用这些元组的 CourseNo(课程号)信息去检索 Course 关系,从中找出 Teacher(任课老师)为张明的那些学生选课元组。

还有一种方案是先按 Teacher 属性检索 Course 关系,找出所有任课老师为张明的课程的课程号,再根据这些课程号检索 StudentCourse 关系,找出成绩低于 60 分的学生选课元组。

既然有多种查询方案,就要判断采用哪一种方案其时间和空间开销比较小,从而决定选择哪一种方案,这就是查询优化所要做的事情。

一般说来,上例中第二种方案比第一种方案的时间和空间开销都要小,因此比第一种方案优越。

1.5.3 存储管理程序

在简单的数据库系统中,存储管理程序可能就是底层操作系统的文件系统;但有时为了提高效率,DBMS 往往直接控制磁盘存储器。

存储管理程序包括两个部分——文件管理程序和缓冲区管理程序。

文件管理程序跟踪文件在磁盘上的位置,并负责取出一个或几个数据块,数据块中含有缓冲区管理程序所要求的文件。磁盘通常划分成一个个连续存储的数据块,每个数据块大小从 4KB 到 16KB 不等。

缓冲区管理程序控制着主存的使用。它通过文件管理系统从磁盘取得数据块,并选择