

第3章 数据库的设计



本章首先介绍数据库设计的方法和特点，然后重点介绍数据库设计的六个步骤，最后简单介绍数据库保护方面的知识。



- 了解数据库设计的基本方法
- 了解影响数据库设计的各因素及数据库设计的特点
- 能够根据项目需求分析进行数据库的概念模型设计
- 能运用关系模型的基本知识将概念模型转换为关系模型
- 能够用关系规范化方法对关系模型进行规范化和优化
- 能够根据完整性规则对关系模型进行完整性的设计
- 了解数据库系统安全的基础知识

3.1 数据库设计概述

数据库技术是对信息资源进行管理的最有效的手段。数据库设计是指对于一个给定的应用环境，构造最优的数据库模式，建立数据库及其应用系统，使之能有效的存储数据，满足用户的应用需求（信息要求和处理要求）。数据库设计是数据库应用系统开发的核心问题，是数据库在应用领域的主要研究课题，数据库设计的成败直接关系到数据库应用系统的可用性。

3.1.1 数据库设计方法

在数据库设计过程中，由于数据信息结构的复杂，应用环境的多样，在相当长的一段时间内，数据库设计主要采用手工试凑法。试凑法缺乏可靠的理论依据和科学的工程方法的支持，设计过程基本依赖于设计人员的经验和水平，从而难以保证数据库的设计质量，并且手动设计增加了系统维护的代价。

为解决试凑法的缺陷，在数据库技术发展过程中，数据库设计人员通过不断的努力探索，提出了很多数据库的设计方法。这些方法以软件工程的思想为基础，总结出了数据库设计中需遵守的准则和规程。具备一定的设计准则和规程的数据库设计方法都属于规范化设计方法。

规范化设计中著名的有新奥尔良法，它将数据库设计分为四个标准阶段：需求分析（分析用户需求）、概念设计（信息分析和定义）、逻辑设计（设计实现）和物理设计（物理数据库设计）。后来，很多设计者在此基础上对新奥尔良法进行了补充和丰富。S.B.Yao 法将数据库设计分为6个步骤：需求分析、模式构成、模式汇总、模式重构、模式分析和物理数据库设计；

I.R.Palmer 法主张将数据库设计当成一步步的过程并采用一些辅助手段实现每一过程；数据库生命周期法以软件生命周期（规划、设计、实施和运行维护）为主线对数据库进行设计。

基于 E-R 模型的数据库设计方法、基于 3NF（第三范式）的设计方法和基于抽象语法规范的设计方法，都是在数据库设计的不同阶段上支持实现的具体技术和方法。规范法设计从本质上看仍然属于手工设计方法，其基本思想是过程迭代和逐步求精，在遵从一定设计标准的基础上，设计出科学、合理的数据库系统。

3.1.2 数据库设计特点

数据库设计的基本任务是根据用户使用的硬件系统、操作系统与数据库管理系统等条件，设计出数据库模式，设计过程中受很多因素的影响。因此，数据库系统设计具有如下几个主要特点：

(1) 反复性。

通常数据库的设计并不是一次完成的。由于数据库设计是将现实世界中抽象的事物转换为数据库管理系统中的数据信息，因此在设计过程中通常存在许多不确定因素，如需求改变等。因此，数据库的设计往往需要多次反复和修正，通过多次反复和修正的数据库才可能真正接近用户的实际需求。

(2) 试探性。

在反复和修正数据库的过程中，需要设计者不断的对数据库进行修改和完善。由于很多修改会带来不确定的结果，因此，其设计过程具有一定的试探性。

(3) 多步性。

当前数据库设计过程都遵从一定的数据库设计方法，规范的设计方法为设计出规范的、可用性强的数据库提供了保障。规范化设计方法都将数据库设计过程分解为目标明确的多个设计步骤，因此，数据库设计具有多步性特点。

(4) 面向数据。

数据库设计过程中要以基础数据为依据进行设计，基础数据是所有数据库建立的基础。

3.2 数据库设计的步骤

3.2.1 SQL Server 数据库应用系统设计一般步骤

在数据库设计过程中，按照规范化进行设计是数据库开发成功的保证，考虑到数据库系统的开发过程，一般可将数据库设计分为以下 6 个阶段：

(1) 需求分析阶段。

进行数据库分析首先要收集资料，并对资料进行分析整理，画出数据流程图，然后建立数据字典，并把数据字典图集和数据字典的内容返回客户，进行用户确认，最后形成文档资料。需求分析是进行数据库设计的起点，需求分析的结果应能准确地反映客户的实际要求，该阶段直接影响到后面各个阶段的设计，并影响设计结果的合理性和实用性。

(2) 概念设计阶段。

根据需求分析的结果，建立独立于不同数据库管理系统的概念模型，该概念模型用 E-R 图来描述。

(3) 逻辑设计阶段。

将概念设计 E-R 图转换成具体数据库管理系统支持的数据模型（关系模型），形成数据库的模式，并对数据进行优化处理，形成数据库的外模式。

(4) 物理设计阶段。

根据数据库管理系统的特点和处理的需要，对逻辑设计的关系模型进行物理存储安排，形成数据库的内模式。

(5) 数据库实现阶段。

运用具体数据库管理系统提供的数据库语言、工具及宿主语言，根据逻辑设计和物理设计的结果建立数据库，编制与调试应用程序，组织数据入库，并进行试运行。

(6) 数据库运行和维护阶段。

经过试运行的数据库即可投入正式运行。在数据库系统运行过程中必须不断对其进行评价、调整与修改。

在实际工程中，能够按照以上步骤一次完成数据库设计的情况并不多见。通常上述过程需要经过多次反复，通过不断的反复和修正才能使系统逐步接近用户的真实需求。一个数据库的设计流程如图 3-1 所示。通常，每个项目都要有一个团队，包括项目经理、数据库设计人员、应用系统设计人员、程序员、技术文档编写人员和数据库管理员等。对于小的数据库设计项目，以上角色可由一个人承担。但要注意，对于每一方面一定要有专人负责以使项目的开发取得成功。

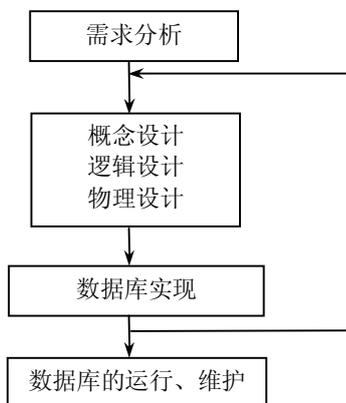


图 3-1 数据库设计流程

3.2.2 需求分析阶段

需求分析是整个数据库设计的基础，在进行数据库设计时，首先要了解与分析用户的应用需求，因为该阶段需要设计者与客户的沟通，因此也是最费时、最困难的一个阶段。

本章以学生成绩管理系统为例进行数据库设计。

任务 3-1 学生成绩管理系统需求分析。

任务分析：通过与教务处学生成绩管理职能部门的沟通，获得该部门的组织结构图，分析组织结构图后绘制该系统的流程图，分析学生成绩管理系统的功能需求，写出数据字典。

(1) 绘制学生成绩管理部门（教务处）组织结构图。组织结构是用户业务流程与信息的载体，对分析人员解释用户的业务、确定系统范围具有很好的帮助。取得用户的组织结构图，

是需求分析步骤中的基础工作之一。学生成绩管理部门的组织结构如图 3-2 所示。

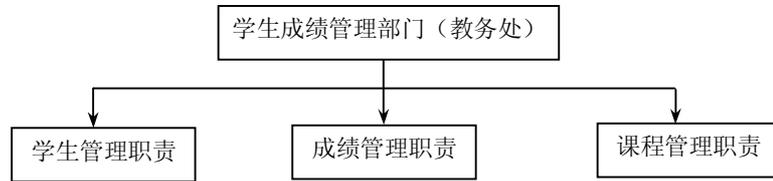


图 3-2 学生成绩管理部门的组织结构图

(2) 绘制系统数据流程图。通过收集资料，在对资料进行分析的基础上绘制出学生管理系统数据流程图，如图 3-3 所示。

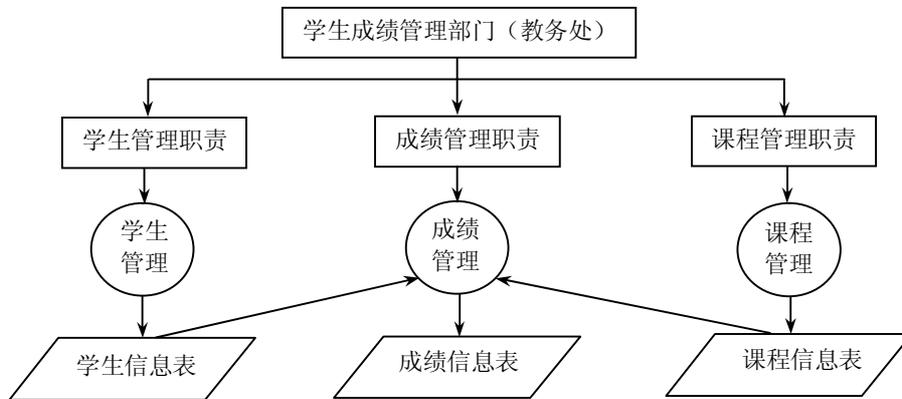


图 3-3 学生成绩管理系统数据流程图

(3) 了解系统功能需求。学生成绩管理系统需要完成如下功能：

- 学生管理：存储、检索、维护有关学生的信息。
- 课程管理：存储、检索、维护有关课程的信息。
- 成绩管理：存储、检索、维护有关学生成绩的信息。

(4) 细读数据字典。针对学生成绩管理系统的功能需求，通过分析、归纳，总结出需要的如下信息：

- 学生信息：学生编号、学生姓名、民族、性别、出生日期、班级专业系部信息、入学年份、联系电话、已修学分、家庭住址、密码、备注。
- 课程信息：课程编号、课程名称、课程类型、总课时、周课时、学分、备注。
- 成绩信息：学生编号、课程编号、学生成绩、学期。

需要注意的是，收集、分析需求中的每一步都可能出现问题，在收集需求时应注意以下几点：

- 注意与用户进行充分的交流。设计人员需重视需求分析的重要性，充分的需求交流是数据库设计的关键。
- 在交流中把握系统本质性的需求。通常在数据库设计中会遇到意见分歧的困难，在同一系统下，当不同的用户提出不同的需求时，设计者需能把握问题的实质，对合理的需求给予满足。

- 关注系统开发过程中需求的改变。当设计者正在为某一环境进行新系统的开发时，它的需求又改变了，这种情况会经常遇到。因此，设计者需及时关注系统开发中需求的改变，及时调整系统设计以满足实际需要。可以发现，使一个系统满足变化着的需求是极具挑战性的。

3.2.3 概念设计阶段

概念设计是将需求分析得到的用户需求抽象为数据库的概念结构，是对现实世界的抽象反映，它不依赖于具体的计算机系统，是现实世界到数据世界的一个中间层次，如图 3-4 所示。

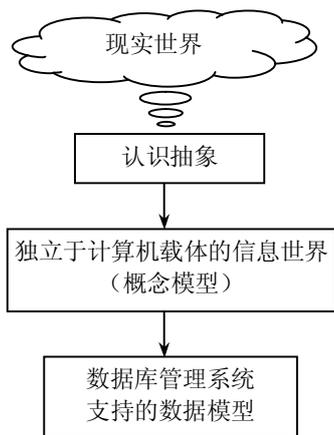


图 3-4 数据抽象过程

结合学生成绩管理系统的需求分析，对数据库系统进行概念设计。

(1) 定义实体。根据需求分析，找出数据实体。根据学生成绩管理系统的需求分析，可找出学生和课程两个数据实体。

(2) 定义联系。根据需求分析，找出实体与实体之间的联系。通过分析学生成绩管理系统可知，学生与课程之间存在选课考试并获得成绩的联系。一个学生可以选择多门课程并获得对应的多门课成绩，一门课程也可能被多个学生选择。

(3) 定义主码。根据需求分析，找出实体的主码。学生成绩管理系统中实体学生的主码为学生编号，课程的主码为课程编号。

(4) 定义属性。根据需求分析，找出实体的属性。根据需求分析中的数据字典可以得到学生和课程的属性。

(5) E-R 模型设计。综合以上分析进行 E-R 模型设计。

任务 3-2 根据学生成绩管理系统需求分析，绘制局部 E-R 图。

任务分析：E-R 模型设计过程中先绘制局部 E-R 图，即实体及属性 E-R 图。根据用户需求可知学生实体属性有学生编号、学生姓名、民族、性别、出生日期、班级专业系部信息、入学年份、联系电话、已修学分、家庭住址、密码、备注，主码为学生编号，其局部 E-R 图如图 3-5 所示。

课程实体属性有课程编号、课程名称、课程类型、总课时、周课时等，主码为课程编号，其局部 E-R 图如图 3-6 所示。

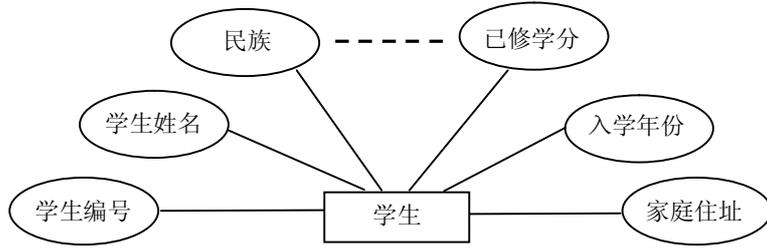


图 3-5 学生实体及属性局部 E-R 图

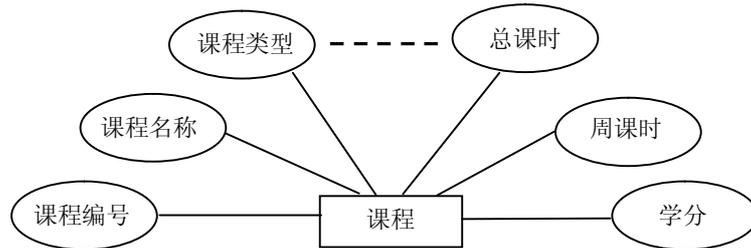


图 3-6 课程实体及属性局部 E-R 图

任务 3-3 根据学生成绩管理系统需求分析及局部 E-R 图，绘制综合 E-R 图。

任务分析：根据学生与课程之间的联系。一个学生可以选择多门课程进行考试，并获得对应的成绩，一门课程会有多个学生选择进行考试，学生与课程之间存在多对多的成绩联系。综合局部 E-R 图，可得综合 E-R 图如图 3-7 所示。

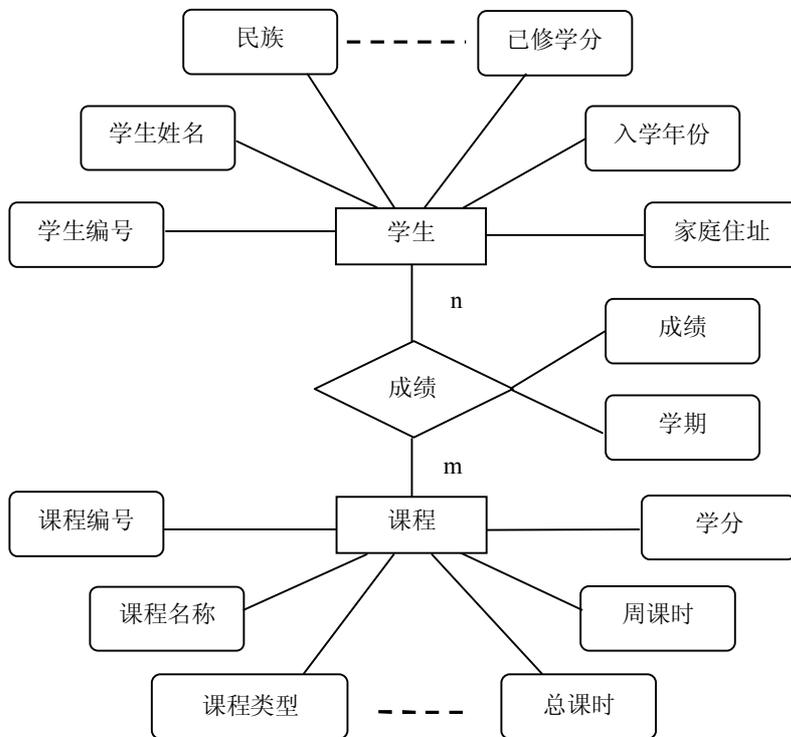


图 3-7 学生成绩管理系统概念设计 E-R 图

学生成绩管理系统的概念设计较为简单，而在实际的大、中型数据库应用中概念设计是非常复杂的，这就需要在工作中不断的积累经验。

3.2.4 逻辑设计阶段

在上节中知道独立于计算机载体的信息世界（概念模型）是通过对客观世界进行认知抽象后得到的，概念模型不依赖于具体的计算机，而数据库最终的实现都是以计算机为载体的，因此需对概念模型进行转化。

本小节以学生成绩管理系统为例，介绍信息世界（概念模型）到机器世界（关系模型）的转换。

1. 实体（E）转换为关系模式的方法

实体转换为关系模式：实体的属性就是关系的属性，实体的主码就是关系的主码。由于逻辑设计是面向具体数据库管理系统，所以概念设计中实体、联系和属性名称在关系模型中最好设计为英文的标准命名标识符。

任务 3-4 将学生和课程实体转换为关系模式。

任务分析：将学生和课程实体对应的主码转换关系的主码，将实体的属性转换为关系的属性。

实体（E）：学生(学生编号,学生姓名,民族,性别,出生日期,入学年份,联系电话,已修学分,班级专业系部信息,家庭住址,密码,备注)

PK：学生编号

关系模式：Student(studentID,studentName,nation,sex,birthday,ru_date,telephone,credithour,class-speciality-department,address,pwd,remark)

PK：studentID

实体（E）：课程(课程编号,课程名称,课程类型,总课时,周课时,学分,备注)

PK：课程编号

关系模式：Course(courseID,coursename,coursetype,totalperiod,weekperiod,credithour,remark)

PK：courseID

2. 联系（R）转换为关系模式

实体间的联系存在一对一、一对多、多对多三种情况，在转换成关系模式的时候应分别遵从联系到关系的转换方法：

- 一对一：将联系与任意端实体所对应的关系模式合并，并加入另一端实体的主码和联系本身的属性。
- 一对多：将联系与多端实体所对应的关系模式合并，加入一端实体的主码和联系的属性。
- 多对多：将该联系相连的各实体的主码和联系本身的属性转换为关系的属性。

任务 3-5 将学生实体与课程实体之间的联系转换成关系模式。

任务分析：学生实体和课程实体之间的联系是多对多的。因此，将联系转换成一个关系模式，该联系相连的学生实体主码“学生编号”和课程实体的主码“课程编号”加上联系本身的属性“成绩”和“学期”转换为关系的属性，如图 3-8 所示。

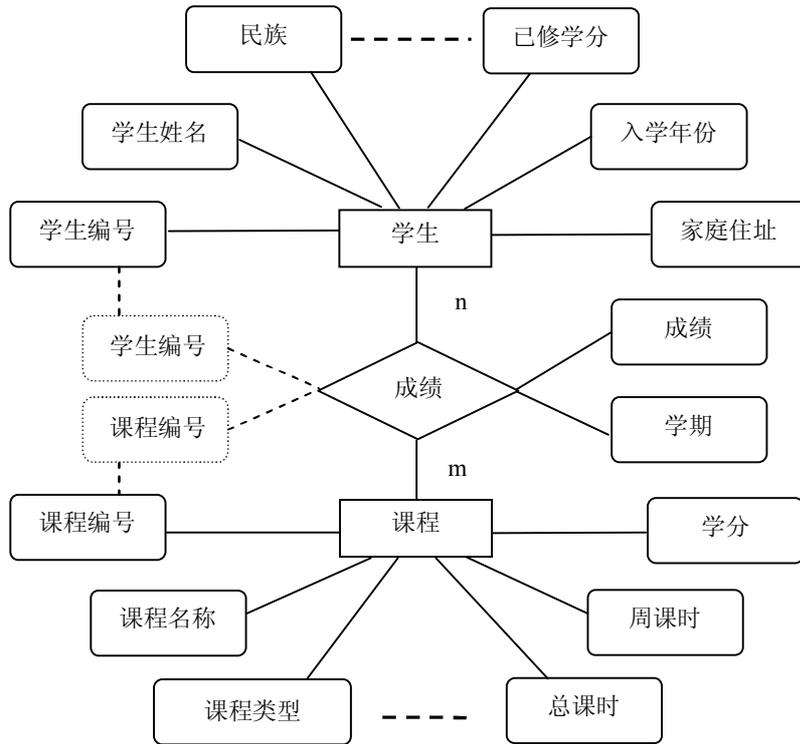


图 3-8 学生成绩管理系统概念设计 E-R 图

联系成绩的关系为:

Grade(studentID,courseID,Term,grade)

PK: studentID,courseID

FK: studentID 和 courseID

3. 关系规范化

数据库逻辑设计的好坏与关系模式的结构有很大关系, 关系模式的结构合理、规范化程度高, 可以确保所建立的数据库具有较高的数据密度、较好的数据共享度、较准确的数据一致性。关系规范化的相关理论在第二章已经介绍。在学生成绩管理系统中, 由于关系不规范, 可能存在大量的数据冗余, 此时需要进行关系规范化。

任务 3-6 分析课程关系的规范情况, 对其进行关系规范化。

任务分析: 课程关系的 coursetype 属性还包括课程类型编号 coursetypeID、课程类型名称 typename 属性。可知此时的课程关系模式为:

Course(courseID,coursename,coursetypeID,typename,totalperiod,weekperiod,credithour,remark)

PK: courseID

可以发现该关系属性之间存在传递函数依赖, 主码 courseID 决定 coursetypeID, 而 coursetypeID 决定非主属性 typename。即非主属性 typename 传递依赖主码 courseID。

由于存在传递依赖, 所以该关系存在如下问题:

- 数据冗余: 同一个类型的课程对应的课程类型信息存在大量重复。
- 插入异常: 在某课程类型没有对应课程的情况下, 不容许插入数据。
- 更新异常: 冗余带来更新不一致。

为解决上述问题，将该关系进行分解，分解如下：

Course(courseID,coursename,coursetypeID, totalperiod,weekperiod,credithour,remark)

PK: courseID

FK: coursetypeID

CT(courseID,coursetypeID)←该联系可通过增加外码省略

Coursetype(coursetypeID,typename)

此时，学生成绩管理系统的数据库模型规范化如下：

Course(courseID,coursename,coursetypeID, totalperiod,weekperiod,credithour,remark)

PK: courseID

FK:coursetypeID

Coursetype(coursetypeID,typename)

PK: coursetypeID

Grade(studentID,courseID,Term,grade)

PK: studentID,courseID

FK: studentID 和 courseID

Student(studentID,studentName,nation,sex,birthday,ru_date,telephone, credithour,class-speciality-department,address,pwd,remark)

PK: studentID

任务 3-7 分析学生关系的规范情况，对其进行关系规范化。

任务分析：学生关系的 class-speciality-department 属性还包括班级编号 classID、班级名称 className、专业编号 specialityID、专业名称 specialityName、入学年份 EntranceYear、班长编号 MonitorID、部门编号 departmentID、部门名称 DepartmentName、部门负责人 DepartmentHead 属性。

此时的学生关系模式为：

Student(studentID,studentName,nation,sex,birthday,ru_date,telephone, credithour,classID,className,specialityID,specialityName,EntranceYear,MonitorID,departmentID,DepartmentName,DepartmentHead,address,pwd,remark)

PK: studentID

可以发现该关系属性之间存在多重传递函数依赖，主码 studentID 决定 classID，而 classID 决定非主属性 className、specialityID、specialityName、EntranceYear、MonitorID、departmentID、DepartmentName、DepartmentHead，即以上非主属性通过 classID 传递依赖主码 studentID。同时 classID 决定属性 specialityID、specialityName、departmentID、DepartmentName、DepartmentHead；specialityID 决定属性 departmentID、DepartmentName、DepartmentHead。

根据以上分析，将该关系进行分解，分解如下：

Student(studentID,studentName,nation,sex,birthday,ru_date,telephone, credithour,classID,address,pwd,remark)

PK: studentID

SC(studentID,classID)←该联系可通过增加外码省略

Class(classID,specialityID,className,EntranceYear,MonitorID)

PK: classID

FK: specialityID 和 MonitorID

CS(specialityID,classID)←该联系可通过增加外码省略

Speciality(specialityID,specialityName,departmentID)

PK: specialityID

FK: DepartmentID

SD(departmentID,specialityID)←该联系可通过增加外码省略

Department(departmentID,DepartmentName,DepartmentHead)

PK: DepartmentID

此时, 学生成绩管理系统的数据模型规范化如下:

Student(studentID,studentName,nation,sex,birthday,ru_date,telephone,credithour,classID,address,pwd,remark)

PK: studentID

Class(classID,specialityID,className,EntranceYear,MonitorID)

PK: classID

FK: specialityID 和 MonitorID

Speciality(specialityID,specialityName,departmentID)

PK: specialityID

FK: DepartmentID

Department(departmentID,DepartmentName,DepartmentHead)

PK: DepartmentID

Course(courseID,coursename,coursetypeID, totalperiod,weekperiod,credithour,remark)

PK: courseID

FK: coursetypeID

Coursetype(coursetypeID,typename)

PK: coursetypeID

Grade(studentID,courseID,Term,grade)

PK: studentID 和 courseID

FK: studentID 和 courseID

3.2.5 物理设计阶段

对数据库进行设计后, 数据库最终是要存储在物理设备上的。为一个给定的逻辑数据模型选取一个最适合应用环境的物理结构(存储结构与存取方法)的过程, 就是数据库的物理设计。物理结构依赖于给定的数据库管理系统和硬件系统, 因此设计人员必须对所使用数据库管理系统的内部特征有充分了解, 特别是存储结构和存取方法; 充分了解该数据库的具体应用环境, 特别是对响应频率和响应速度的具体要求; 同时, 充分了解物理存储设备的使用特性。

数据库的物理设计通常分为两步:

- 确定数据库的物理结构。
- 对物理结构进行时间和空间效率评价。

1. 确定数据库的物理结构

(1) 确定数据的存储结构。确定数据库存储结构时要综合考虑存取时间、存储空间利用

率和数据库维护代价三方面的因素。这三个方面常常是相互矛盾的,例如消除冗余数据虽然能够节约存储空间,但往往会导致检索代价的增加,因此必须进行权衡,选择一个适合具体应用环境的折中方案,使存取时间、存储空间利用率和数据库维护代价三方面达到用户可以接受的平衡状态。

(2) 设计数据的存取路径。在关系数据库中,设计数据的存取路径主要是确定如何建立数据库索引。例如,应把哪些域作为次码建立次索引,建立简单索引还是组合索引,建立多少个为合适,是否建立聚集索引等问题。

(3) 确定数据的存放位置。在数据库应用过程中,为了提高系统在实际应用过程中的性能,数据应该根据应用情况将易变的数据部分与稳定的数据部分、经常存取的数据部分和存取频率较低的数据部分合理存放。

例如,数据库数据备份、日志文件备份等由于只在故障恢复时才使用,而且数据量很大,可以考虑存放在磁带上长期保存。目前应用数据库的计算机都有多个磁盘,因此进行数据库物理设计时可以考虑将数据库的表和索引分别放在不同的磁盘上,在查询时,由于多个磁盘驱动器分别同时工作,从而保证物理读写速度较快。同样,也可以将比较大的表分别放在两个磁盘上,从而可以实现并行读写,以加快存取速度,这在多用户环境下特别有效。此外还可以将日志文件与数据库对象放在不同的磁盘以改进系统的性能。

(4) 确定系统配置。数据库管理系统一般都提供一些存储分配参数,供设计人员和数据库管理员对数据库进行物理优化。例如,同时使用数据库的用户数,同时打开的数据库对象数,使用的缓冲区长度、个数,时间片大小、数据库的大小等,这些参数值直接影响存取时间和存储空间的分配,在物理设计时就要根据应用环境确定这些参数值,以使系统性能最优。

一般数据库管理系统会提供一些默认参数,但是这些默认参数值不一定适合每一种具体的应用环境,在根据具体应用环境进行物理设计时,需要重新根据具体的应用环境对这些参数进行赋值,以改善系统的性能,适应数据库具体应用的需求。

同时,前面我们已经提到,数据库的设计是一个反复的过程,随着具体应用的改变,很多设计也要根据实际情况进行调整。此时的物理设计对系统配置参数的调整只是初步的,在系统运行时还要根据系统实际运行情况对其参数配置做进一步的调整,通过符合实际应用的调整,切实改进系统性能。

任务 3-8 根据学生成绩管理系统数据库的具体应用情况和 SQL Server 2005 关系数据库系统特点,对学生成绩管理系统数据库进行物理设计。

任务分析:在学生成绩管理系统逻辑设计的基础上,根据其系统特点,根据实际应用情况对学生成绩管理系统进行物理设计,设计结果如表 3-1 至表 3-7 所示。

表 3-1 学生信息表 (Student)

序号	名称	数据类型	可否为空	说明	备注
1	studentID	char (10)	否	学生编号, 主键	
2	studentName	varchar (10)	否	学生姓名	
3	nation	char (10)	是	民族	
4	sex	char (2)	是	性别	
5	birthday	datetime	是	出生日期	
6	classID	char (7)	是	班级编号	

续表

序号	名称	数据类型	可否为空	说明	备注
7	telephone	varchar (16)	是	联系电话	
8	credithour	tinyint	否	已修学分	
9	ru_date	char (4)	是	入学年份	
10	address	varchar (50)	是	家庭住址	
11	pwd	varchar (16)	是	密码	
12	remark	varchar (200)	是	备注	

表 3-2 班级信息表 (Class)

序号	名称	数据类型	可否为空	说明	备注
1	classID	char (7)	否	班级编号, 主键	
2	className	varchar (12)	否	班级名称	
3	specialityID	char (5)	是	专业编号	外键约束
4	EntranceYear	char (4)	是	入学年份	
5	MonitorID	char (10)	是	班长编号	外键约束

表 3-3 专业信息表 (Speciality)

序号	名称	数据类型	可否为空	说明	备注
1	specialityID	char (5)	否	专业编号, 主键	
2	specialityName	varchar (30)	否	专业名称	
3	departmentID	char (3)	是	专业所在系部 ID	外键约束

表 3-4 部门信息表 (Department)

序号	名称	数据类型	可否为空	说明	备注
1	DepartmentID	char (3)	否	系部编号, 主键	
2	DepartmentName	varchar 30	否	系部名称	
3	DepartmentHead	char 8	是	部门负责人	

表 3-5 课程信息表 (Course)

序号	名称	数据类型	可否为空	说明	备注
1	courseID	char (8)	否	课程编号, 主键	
2	coursename	varchar (20)	否	课程名称	
3	coursetypeID	char (3)	是	课程类型编号	外键约束
4	totalperiod	tinyint	是	总课时	
5	weekperiod	tinyint	是	周课时	
6	credithour	tinyint	是	学分	
7	remark	varchar (50)	是	备注	

表 3-6 课程类型信息表 (CourseType)

序号	名称	数据类型	可否为空	说明	备注
1	courseTypeID	char (3)	否	课程类型编号, 主键	
2	typename	varchar (18)	否	课程类型名称	

表 3-7 成绩信息表 (Grade)

序号	名称	数据类型	可否为空	说明	备注
1	studentID	char (10)	否	学生编号, 主键	外键约束
2	courseID	char (8)	否	课程编号, 主键	外键约束
3	Term	nvarchar(20)	否	学期	
4	grade	Tinyint	是	学生成绩	

2. 评价物理结构

数据库物理设计过程中需要对时间效率、空间效率、维护代价和各种用户的实际要求进行权衡,在权衡的过程中便会产生很多可选的物理设计方案,而最终使用的设计方案只有一种,因此,数据库设计人员必须对这些方案进行细致的评价,从中选择一个较优的设计方案作为数据库的物理结构。

评价物理数据库的方法完全依赖于所选用的数据库管理系统,主要是从定量估算各种方案的存储空间、存取时间和维护代价入手,对估算结果进行权衡、比较,选择出一个较优的、合理的物理结构。

如果没有符合用户需求的设计结构,则需要修改设计。因此,物理结构的设计与物理结构的评价往往也需要一个反复的过程,通过不断的完善才能达到适应具体应用环境的物理结构设计。

3.2.6 数据库实施阶段

数据库实施主要包括以下工作:

- 用 DDL 定义数据库结构
- 组织数据导入数据库
- 编制与调试相关应用程序
- 数据库试运行

1. 定义数据库结构

确定了数据库的逻辑结构与物理结构后,就可以用所选用的数据库管理体系所提供的数据库定义语言 (DDL) 来定义数据库的结构。例如创建数据库、创建数据库的表、创建数据库的视图等。

2. 数据装载

数据库的结构定义好以后,就可以向数据库中装载数据了。数据是数据库的基础,合理有效的组织数据录入数据库是数据库实施阶段最主要的工作。对于数据量不是很大的小型系统,可以用人工的录入方法完成数据的录入,其步骤为:

- (1) 数据选择。需要录入数据库中的数据通常都是原始数据,这些数据都存放在相应的

原始文件中,如纸制文件等。即便文件存放在计算机存储介质中,也可能是分散而不规律的,因此,首先必须对需要录入数据库的数据信息进行选择。

(2) 数据格式转换。选择出来的需要录入数据库的数据,其格式一般不能满足具体数据库管理系统的要求,还需要进行转换。这种转换涉及数据的具体格式,因此往往非常复杂。

(3) 录入数据。将转换好的数据输入计算机中,原始数据的录入过程较简单,但由于数据量较大,往往需要消耗较多的人工。

(4) 数据校验。对录入的数据进行检查,检查输入的数据是否有误。

对于中大型系统,由于数据量十分庞大,用人工方式对数据进行录入将会耗费大量的人力、物力,最重要的是人工录入很难保证数据的正确性。因此,在大中型应用中,可以通过设计一个数据输入子系统,由计算机来辅助数据的录入工作,以保障数据录入的效率和录入的准确率。

3. 编制与调试应用程序

在数据库应用中还需要一些具体的应用设计,即一些应用程序的设计。在数据库实施阶段,当数据库结构建立好后,就可以开始编制与调试数据库的应用程序,也就是说,编制与调试应用程序应与组织数据录入数据库同步进行。调试应用程序时由于数据录入尚未完成,因此可先使用模拟数据进行应用程序的测试。

4. 数据库试运行

数据库的应用程序测试完成,并且已有一小部分数据入库后,就可以开始数据库的试运行,该阶段也称为数据库联合测试阶段。数据库试运行的主要工作包括功能测试和性能测试两部分。

功能测试:实际运行数据库应用程序,执行对数据库的各种应用操作,在实际环境下测试应用程序的各种功能。

性能测试:测试数据库系统的性能指标,分析是否符合设计目标。

3.2.7 运行和维护阶段

当数据库试运行结果符合数据库设计目标后,就可以真正投入运行了。数据库的运行标志着数据库开发任务的基本结束和数据库维护工作的开始,但这并不意味着数据库设计过程的终结,由于在具体的应用环境下数据库使用的具体需求也在不断变化,随着数据库的运行,其物理存储也会不断变化,因此对数据库设计进行调整、修改、评价等工作是一个长期反复的任务,也是设计工作的继续和对数据库设计质量的提高。

在数据库运行阶段,对数据库的维护工作主要是由数据库管理员完成的,其工作内容包
括以下几点:

(1) 数据库的备份和恢复。

数据库的运行过程中可能会发生很多意想不到的事情,从而影响数据库的安全和数据信息的完整。因此需要定期对数据库和数据库日志文件进行备份,以保证一旦数据库系统发生故障,数据库管理员能利用数据库备份及日志文件备份,在尽可能减少数据库数据损失的前提下,将数据库恢复到某种无错状态。

(2) 数据库的安全性、完整性控制。

由于应用环境是不断变化的,因此数据库管理员必须对数据库的安全性和完整性控制负起责任。例如根据用户的实际需要授予不同的数据库操作权限,并根据数据库完整性约束条件

的不断变化对完整性约束进行修正，以满足用户要求。

(3) 对数据库性能的监督、分析和改进。

许多数据库管理系统都提供了监测系统性能参数的监测工具，数据库管理员可以利用这些工具方便地得到系统运行过程中的一系列性能参数。数据库管理员通过仔细分析这些数据，可以调整某些数据库设置参数来进一步改进数据库的可用性。

(4) 数据库的重组和重构造。

数据库运行过程中，由于数据信息的不断增、删、改，会使数据库的物理存储性能下降，从而降低数据库存储空间的利用率和数据的存取效率，使数据库的存储性能下降。这时数据库管理员就要对数据库进行重组。数据库的重组不会改变原有数据库设计的数据逻辑结构和物理结构，只是按原设计要求重新安排存储位置，例如：回收垃圾、减少指针链等，以提高系统性能。数据库管理系统一般都提供了供重组数据库使用的实用程序，帮助数据库管理员重新组织数据库。

在数据库应用环境发生变化时，往往会导致实体及实体间的联系也发生相应的变化，使原有的数据库设计不能很好地满足新环境下应用的需求，从而不得不适当调整数据库的模式和内模式，这就是数据库的重构造。数据库管理系统都提供了修改数据库结构的功能。

重构造数据库的程度往往是有限的。如本书中对介绍的学生成绩管理系统，可以根据实际应用情况增加教师信息表 and 用户信息表，这些都是根据实际需求对数据库进行的简单重构造。

但很多情况下，由于应用环境变化太大，在新的应用环境下已无法通过重构造数据库来满足新的需求，或重构造数据库的代价太大，则表明此时数据库应用系统的生命周期已经结束，应该重新设计新的数据库系统，开始新数据库应用系统的生命周期，以满足新应用环境的应用需求。

3.3 数据库保护

在数据库运行过程中，需要对数据库进行保护，以保障数据库安全正常的运行，数据库保护是计算机安全中的一个重要部分。计算机系统安全是指为计算机系统建立和采取的各种安全保护措施，以保护计算机系统硬件、软件及数据，防止其因偶然或恶意的原因使系统遭到破坏、数据遭到更改或泄露等。计算机系统的安全性问题可分为技术安全类、管理安全类和政策法律三大类。

1. 技术安全类

指计算机系统中采用具有一定安全性的硬件、软件来实现对计算机系统及其所存数据的安全保护，当计算机系统受到无意或恶意的攻击时仍能保证系统的正常运行，保证系统内的数据安全。

2. 管理安全类

指技术安全之外的，诸如软硬件意外故障、事故和管理不善导致的计算机设备的物理破坏等安全问题。

3. 政策法律类

指相关管理部门建立的有关计算机犯罪、数据安全保密的法律、道德准则和政策法规、法令。

本书只介绍技术安全保护。根据计算机安全的基本理论和数据库系统的特点，在一般计算机系统中，数据库的保护措施是分层设置的。具体包括四个级别的安全性：

- 操作系统级别的安全性
客户要访问数据库, 首先要获得计算机操作系统的使用权。
- 服务器级别的安全性
SQL Server 的服务器级安全性建立在控制服务器登录账号和密码的基础上, 用户名和密码保证了用户在使用数据库前能获得 SQL Server 的访问权限。
- 数据库级别的安全性
用户获得服务器安全性验证后, 将直接面对不同的数据库入口, 这是用户接受的第三层安全性验证。
- 表和列级的安全性
该层是对某一数据库中对象使用权限的验证。

本小节只对数据库保护做简单介绍, 数据库保护的相关知识将在第 12 章进行详细介绍。



习题 3

一、选择题

1. 数据库系统设计特点 ()。

A. 反复性	B. 试探性
C. 单步性	D. 面向数据
2. SQL Server 数据库应用系统设计的一般步骤 ()。

A. 需求分析阶段	B. 概念设计阶段
C. 逻辑设计阶段	D. 物理设计阶段
E. 数据库实现阶段	F. 数据库运行和维护阶段
3. 在数据库需求分析阶段, 收集、分析需求中的每一步都可能出现的问题, 在收集需求时应注意以下几点 ()。

A. 注意与用户进行充分的交流	B. 在交流中把握系统本质性的需求
C. 根据主观意识完全可以完成一个小型数据库的需求分析	D. 关注系统开发过程中需求的改变
4. 需求分析阶段和逻辑结构设计阶段得到的结果分别是 ()、()。

A. 数据字典描述的数据需求	B. E-R 图表示的概念模型
C. 某个 DBMS 所支持的数据模型	D. 包括存储结构和存取方法的物理结构
5. 对数据库系统进行概念设计包括 ()。

A. 根据需求分析, 找出数据实体	B. 根据需求分析, 找出实体与实体之间的联系
C. 根据需求分析, 找出实体的主码	D. 根据需求分析, 找出实体的属性
E. 进行 E-R 模型设计	

6. 数据库的实施主要包括以下哪些工作 ()。
- 用 DDL 定义数据库对象
 - 用 DML 操作数据库对象
 - 组织数据导入数据库
 - 编制与调试相关应用程序
 - 数据库试运行
7. 数据库在运行阶段, 数据库管理员的工作内容包括 ()。
- 数据库的备份和恢复
 - 数据库的安全性、完整性控制
 - 对数据库性能的监督、分析和改进
 - 数据库的重组和重构造
8. 数据库的保护措施是分层设置的, 具体包括四个级别的安全性 ()。
- 操作系统级别的安全性
 - 服务器级别的安全性
 - 数据库级别的安全性
 - 表和列级的安全性
 - 视图级的安全性

二、填空题

1. 数据库的物理设计通常分为确定数据库的物理结构、对物理结构进行评价两步, 其中确定数据库的物理结构包括_____、_____、_____、_____。
2. 对于数据量不是很大的小型系统, 可以用人工方法完成数据的录入, 其步骤为_____、_____、_____、_____。
3. 在一般计算机系统中, 数据库的保护措施是分层设置的。具体包括四个级别的安全性_____、_____、_____、_____。

三、简答题

- 需求分析阶段的设计目标是什么? 调查的内容是什么?
- 试述数据库概念结构设计的重要性和设计步骤。
- 试述数据库设计过程各个阶段上的设计描述。
- 简述实体转换成关系模式时遵从的转换方法。
- 试述数据库设计过程中结构设计部分形成的数据库模式。

上机实验

一、实验目的和要求

- 了解数据库设计的基本方法。
- 能够对数据库开发项目进行需求分析。
- 能够根据项目需求分析进行数据库的概念模型设计。

4. 能将概念模型转换为关系模型。
5. 能对关系模型进行规范化。

二、实验内容

根据实际情况完成一个实际部门的数据库系统设计。内容包括：数据库系统需求分析、数据库概念设计、数据库逻辑设计阶段、数据库物理设计。

1. 分组实验，每组 4~5 人。合理分工，每人担任不同的角色，包括需求分析人员、数据库概念设计人员、数据库逻辑设计人员、数据库物理设计人员等。
2. 选择一个实际部门的数据库系统进行项目需求分析，写出需求调查报告。
3. 在需求分析的基础上，进行数据库概念设计，并用 E-R 图来描述该概念模型。
4. 进行数据库逻辑设计，将概念模型转化成关系模型，并对该模型进行关系规范化。
5. 根据数据库管理系统的处理和需要的需要，对逻辑设计的关系模型进行物理存储安排。
6. 将各组的设计结果进行集体讨论、互相学习，指出各自的特点和不足，交流数据库设计过程中的收获和体会。