

第3章 数据库设计

合理的数据库结构是数据库应用系统性能良好的基础和保证,但数据库的设计和开发却是一项庞大而复杂的工程。从事数据库设计的人员,不仅要具备数据库知识和数据库设计技术,还要有程序开发的实际经验,掌握软件工程的原理和方法;数据库设计人员必须深入应用环境,了解用户具体的专业业务;在数据库设计的前期和后期,与应用单位人员密切联系,共同开发,可大大提高数据库设计的成功率。

本章主要讲述数据库设计过程中的需求分析、概念结构设计、逻辑结构设计、物理结构设计、数据库实施和运行、维护等内容,以及按照规范设计方法的步骤介绍 3 个数据库设计实例。



3.1 数据库设计概述

数据库设计是根据用户需求设计数据库结构的过程。具体来讲,数据库设计是对于给定的应用环境,在关系数据库理论的指导下,构造最优的数据库模式,在数据库管理系统上建立数据库及其应用系统,使之能有效地存储数据,满足用户各种需求的过程。

数据库设计方法有多种,概括起来分为 4 类:直观设计法、规范设计法、计算机辅助设计法和自动化设计法。按照规范设计的方法,考虑数据库及其应用系统开发全过程,数据库设计可分为 6 个阶段:需求分析阶段、概念结构设计阶段、逻辑结构设计阶段、物理结构设计阶段、数据库实施阶段以及数据库运行和维护阶段。



3.2 需求分析

需求分析是数据库设计的起点。需求分析就是数据库设计人员通过仔细地调查和向用户详细咨询,掌握用户的需求,理解用户的需求。需求分析的结果是否准确地反映了用户的实际要求,将直接影响到后面各阶段的设计,并影响到设计结果是否合理和实用。如果投入大量的人力、物力、财力和时间开发出的软件却没人要,那所有投入都是徒劳。如果费了很大的精力开发一个软件,最后却不满足用户的要求,从而要重新开发,这种返工是让人痛心的。

总之,需求分析对数据库的开发起到了决策的作用,提供了开发的方向,并指明了开发的策略,在数据库开发及维护中均起到了举足轻重的作用。可以说,在一个大型数据库系统

的开发中,它的作用要远远大于其他各阶段。永远别忘了,数据库设计得合理、可行、满足用户需求才是最重要的。

■ 3.2.1 需求分析的任务

需求分析的任务是通过详细调查现实世界要处理的对象(组织、部门、企业等),充分了解原系统(手工系统或计算机系统)工作概况,明确用户的各种需求,然后在此基础上确定新系统的功能。新系统必须充分考虑今后可能的扩充和改变,不能仅仅按当前应用需求设计数据库。

调查的重点是“数据”和“处理”,通过调查、收集与分析,获得用户对数据库的如下要求。

(1) 信息要求。

信息要求指用户需要从数据库中获得信息的内容与性质。由信息要求可以导出数据要求,即在数据库中需要存储哪些数据。

(2) 处理要求。

处理要求指用户要完成什么处理功能,对处理的响应时间有什么要求,处理方式是批处理还是联机处理等。

(3) 安全性与完整性要求。

安全性要求是指对数据库的用户、角色、权限、加密方法等安全保密措施的要求。完整性要求是指对数据取值范围、数据之间各种联系的要求等。

确定用户的最终需求往往是一件很困难的事,这是因为一方面用户缺少计算机知识,开始时无法确定计算机究竟能为自己做什么,不能做什么,往往不能准确地表达自己的需求,所提出的需求往往不断地变化;另一方面,设计人员缺少用户的专业知识,不易理解用户的真正需求,甚至误解用户的需求。因此,设计人员必须不断深入地与用户交流,才能逐步确定用户的实际需求。

■ 3.2.2 需求分析的方法

进行需求分析首先是调查清楚用户的实际要求,与用户达成共识,然后分析与表达这些需求。

1. 调查用户需求

调查用户需求的具体步骤如下。

(1) 调查组织机构情况,包括了解该组织的部门组成情况、各部门的职责等,为分析信息流程做准备。

(2) 调查各部门的业务活动情况,包括了解各部门输入和使用什么数据,如何加工处理这些数据,输出什么信息,输出到什么部门,输出结果的格式,这是调查的重点。

(3) 在熟悉业务活动的基础上,协助用户明确对新系统的各种要求,包括信息要求、处理要求、完全性与完整性要求,这是调查的又一个重点。

(4) 确定新系统的边界,对前面调查的结果进行初步分析,确定哪些功能由计算机完成或将来准备让计算机完成,哪些活动由人工完成。由计算机完成的功能就是新系统应该实

现的功能。在调查过程中,可以根据不同的问题和条件,使用不同的调查方法。

2. 常用的调查方法

(1) 跟班作业。通过亲身参加业务工作了解业务活动的情况。这种方法可以比较准确地理解用户的需求,但比较耗费时间。

(2) 开座谈会。通过与用户座谈了解业务活动情况及用户需求。座谈时,参加者之间可以相互启发,一般可按职能部门组织座谈会。

(3) 询问或请专人介绍。一般应包括领导、管理人员、操作员等。

(4) 设计调查表请用户填写需求。如果调查表设计得合理,这种方法是很有效,也易于被用户接受。

(5) 查阅记录。查阅与原系统有关的数据记录。

做需求调查时,往往需要同时采用上述多种方法。但无论使用何种调查方法,都必须有用户的积极参与和配合,最好能建立由双方人员参加的项目实施保障小组负责沟通联系。



扫一扫

视频讲解



3.3 概念结构设计

系统需求分析报告反映了用户的需求,但只是现实世界的具体要求,这是远远不够的。我们还要将其转换为信息(概念)世界的结构,这就是概念设计阶段所要完成的任务。

数据库概念结构设计是整个数据库设计的关键,此阶段要做的工作不是直接将需求分析得到的数据格式转换为 DBMS 能处理的数据模型,而是将需求分析得到的用户需求抽象为反映用户观点的概念模型,以此作为各种数据模型的共同基础,从而能更好、更准确地用某一 DBMS 实现这些需求。

描述概念结构的模型应具有以下几个特点。

(1) 具有丰富的语义表达能力。能表达用户的各种需求,反映现实世界中各种数据及其复杂的联系,以及用户对数据的处理要求等。

(2) 易于人的理解和交流。概念模型是系统分析师、数据库设计人员和用户之间的主要交流工具。

(3) 易于修改。概念模型能灵活地加以改变,以反映用户需求和环境的变化。

(4) 易于向各种数据模型转换。设计概念模型的最终目的是向某种 DBMS 支持的数据模型转换,建立数据库应用系统。

人们提出了多种概念设计的表达工具,其中最常用、最有名的是 E-R 模型。

3.3.1 概念结构设计的方法

概括起来,设计概念模型的总体策略和方法可以归纳为 4 种。

1. 自顶向下法

首先认定用户关心的实体及实体间的联系,建立一个初步的概念模型框架,即全局 E-R

模型,然后再逐步细化,加上必要的描述属性,得到局部 E-R 模型。

2. 自底向上法

自底向上法有时又称为属性综合法,先将需求分析说明书中的数据元素作为基本输入,通过对这些数据元素的分析,把它们综合成相应的实体和联系,得到局部 E-R 模型,然后在此基础上再进一步综合成全局 E-R 模型。

3. 逐步扩张法

先定义最重要的核心概念 E-R 模型,然后向外扩充,以滚雪球的方式逐步生成其他概念 E-R 模型。

4. 混合策略

将单位的应用划分为不同的功能,每种功能相对独立,针对各个功能设计相应的局部 E-R 模型,最后通过归纳合并,消去冗余与不一致,形成全局 E-R 模型。

其中,最常用的策略是自底向上法,即先进行自顶向下的需求分析,再进行自底向上的概念设计。

■ 3.3.2 概念结构设计的步骤

在概念结构设计时,自底向上法可以分为两步:①进行数据抽象,设计局部概念模型,即设计局部 E-R 图;②集成各局部 E-R 图,形成全局 E-R 图,即 E-R 图的集成。

1. 设计局部 E-R 图

局部 E-R 图的设计步骤包括以下 4 步。

1) 确定局部 E-R 图描述的范围

根据需求分析所产生的文档,可以确定每个局部 E-R 图描述的范围。通常采用的方法是将功能划分为几个系统,每个系统又可分为几个子系统。设计局部 E-R 模型的第 1 步就是划分适当的系统或子系统,在划分时过细或过粗都不太合适。划分过细将造成大量的数据冗余和不一致,过粗有可能漏掉某些实体。

一般可以遵循以下两条原则进行功能划分。

(1) 独立性原则:划分在一个范围内的应用功能具有独立性与完整性,与其他范围内的应用有最少的联系。

(2) 规模适度原则:局部 E-R 图规模应适度,一般以 6 个左右实体为宜。

2) 确定局部 E-R 图的实体

根据需求分析说明书,将用户的数据需求和处理需求中涉及的数据对象进行归类,指明对象的身份,是实体、联系还是属性。

3) 定义实体的属性

根据上一步确定的实体的描述信息确定其属性。

4) 定义实体间的联系

确定了实体及其属性后,就可以定义实体间的联系了。实体间的联系按其特点可分为3种:存在性联系(如学生有所属的班级)、功能性联系(如教师要教学生)、事件性联系(如学生借阅书籍)。实体间的联系方式分为一对一、一对多、多对多3种。

设计完成某一局部结构的 E-R 图后,再看还有没有其他的局部,如果有则转到第 2 步继续,直到所有局部 E-R 图都设计完为止。

2. 局部 E-R 图的集成

由于局部 E-R 图反映的只是单位局部子功能对应的数据视图,可能存在不一致的地方,还不能作为逻辑设计的依据,这时可以去掉不一致和重复的地方,将各个局部 E-R 图合并为全局 E-R 图,即局部 E-R 图的集成。

一般来说,局部 E-R 图的集成可以有两种方式:一种是多个分 E-R 图一次集成;另一种是逐步集成,用累加的方式一次集成两个分 E-R 图。第 1 种方式比较复杂,做起来难度较大。第 2 种方式每次只集成两个分 E-R 图,可以降低复杂度。

无论采用哪种集成法,每次集成都分为两个阶段:第一步是合并,以消除各局部 E-R 图之间的不一致情况,生成初步全局 E-R 图;第二步是优化,消除不必要的冗余,包括冗余的数据和实体间冗余的联系,生成最终全局 E-R 图。

【例 3-1】 图 3-1 所示的教学信息数据库系统 E-R 图中,学生和教师之间不需要直接的联系,因为不常用,偶尔在用户程序中需要用到二者之间的连接查询时,可以通过课程实体间接实现,所以学生和教师之间为多余的联系;同样,学生和学院之间也为多余的联系;平均成绩为多余的属性,因为它完全可以通过其他属性计算得到,除非应用程序有特殊需要,否则不需要设置此属性。

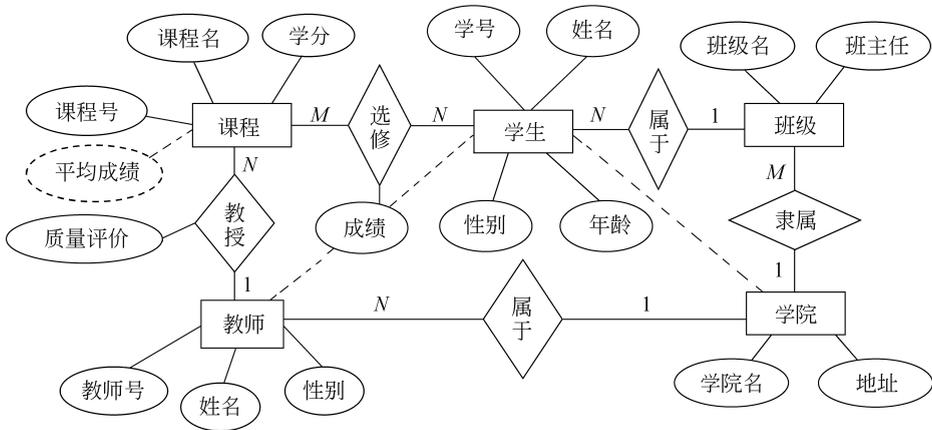


图 3-1 教学信息数据库系统 E-R 图

3.4 逻辑结构设计

数据库概念设计阶段得到的数据模式是用户需求的形式化,它独立于具体的计算机系统和 DBMS。为了建立用户所要求的数据库,必须把上述数据模式转换为某个具体的

DBMS 所支持的数据模式,并以此为基础建立相应的外模式,这是数据库逻辑设计的任务,是数据库结构设计的重要阶段。

逻辑设计的主要目标是产生一个 DBMS 可处理的数据模型,该模型必须满足数据库的存取、一致性及运行等各方面的用户需求。逻辑结构设计阶段一般要分为 3 步进行:将 E-R 图转换为关系数据模型,关系模式的优化,设计用户外模式。

■ 3.4.1 将 E-R 图转换为关系数据模型

关系数据模型是一组关系模式的集合,而 E-R 图是由实体、属性和实体之间的联系三要素组成的。所以,将 E-R 图转换为关系数据模型实际上是要将实体、属性和实体之间的联系转换为关系模式。

转换过程中要遵循如下原则。

1. 实体的转换

一个实体转换为一个关系模式,实体的属性就是该关系模式的属性,实体的主码就是该关系模式的主码。

2. 联系的转换

(1) 两实体集间 1:1 联系可以转换为一个独立的关系模式,也可以与任意一端对应的关系模式合并。

方法 1: 转换为一个独立的关系模式。

转换后的关系模式中关系的属性包括与该联系相连的各实体的主码以及联系本身的属性(如果有),关系的主码为两个实体的主码的组合。

【例 3-2】 将图 3-2 所示的 E-R 图按方法 1 转换为关系模式。

首先,将其中的两个实体按实体的转换原则转换为两个关系模式:班级和班长。然后,将班级和班长之间的一对一联系转换为一个独立的关系模式:班级-班长。3 个关系模式如下。

班级(班号,系别,班主任,入学时间)

班长(学号,姓名,性别,年龄)

班级-班长(班号,学号,任期)

方法 2: 与某一端对应的关系模式合并。

合并后关系模式的属性包括自身关系模式的属性和另一关系模式的主码及联系本身的属性,合并后关系的主码不变。

【例 3-3】 将图 3-2 所示的 E-R 图按方法 2 转换为关系模式。

首先,将其中的两个实体按实体的转换原则转换为两个关系模式:班级和班长。然后,将班级和班长之间的一对一联系合并到班长或班级实体中。两个关系模式如下。

班级(班号,系别,班主任,入学时间)

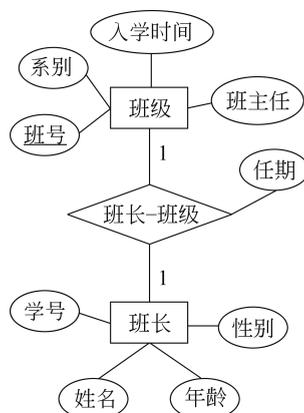


图 3-2 1:1 联系的 E-R 图



视频讲解

班长(学号,姓名,性别,年龄,班号,任期)

或

班级(班号,系别,班主任,入学时间,班长学号,班长任期)

班长(学号,姓名,性别,年龄)

(2) 两实体集间 1 : N 联系可以转换为一个独立的关系模式,也可以与 N 端对应的关系模式合并。

方法 1: 转换为一个独立的关系模式。

关系的属性包括与该联系相连的各实体的主码以及联系本身的属性,关系的主码为 N 端实体的主码。

【例 3-4】 将图 3-3 所示的 E-R 图按方法 1 转换为关系模式。

首先,将其中的两个实体按实体的转换原则转换为两个关系模式:系和教师。然后,将系和教师之间的一对多联系转换为一个独立的关系模式:工作。3 个关系模式如下。

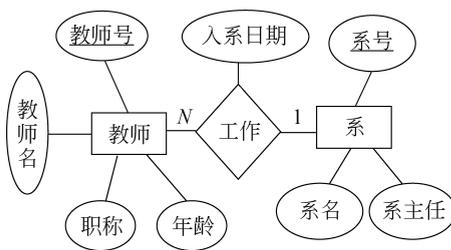


图 3-3 1 : N 联系的 E-R 图

系(系号,系名,系主任)

教师(教师号,教师名,年龄,职称)

工作(教师号,系号,入系日期)

方法 2: 与 N 端对应的关系模式合并。

合并后关系的属性包括在 N 端关系中加入 1 端关系的主码和联系本身的属性,合并后关系的主码不变。

【例 3-5】 将图 3-3 所示的 E-R 图按方法 2 转换为关系模式。

首先,将其中的两个实体按实体的转换原则转换为两个关系模式:系和教师。然后,将系和教师之间的一对多联系合并到多端教师实体中。两个关系模式如下。

系(系号,系名,系主任)

教师(教师号,教师名,年龄,职称,系号,入系日期)

注意: 实际使用中两实体间 1 : 1 和 1 : N 联系通常采用方法 2,以减少数据库系统中的关系模式,因为多一个关系模式就意味着查询过程中可能要进行连接运算,而连接运算会大大降低查询的效率。

(3) 同一实体集内实体间的 1 : N 联系,可在这个实体集所对应的关系模式中多设一个属性,用来作为与该实体相联系的另一个实体的主码。

例如,学生实体集中有部分学生是班长,可以在学生关系模式中加入一个班长学号属性,表示这个学生所在班的班长。

(4) 两实体集间 M : N 联系,必须为联系产生一个新的关系模式。该关系模式中至少包含被它所联系的双方实体的主码,若联系中有属性,也要并入该关系模式中。如果没有指定另外的属性(如此联系的 ID)作为该关系的主码,则该关系的主码一般为双方实体的主码的组合,也可能为双方实体的主码再加联系中某些属性的组合。

【例 3-6】 将图 3-4 所示的 E-R 图转换为对应的关系模式。

首先,将其中的两个实体按实体的转换原则转换为两个关系模式:职工和项目。然后,将职工和项目之间的多对多联系转换为一个独立的关系模式:参加。3 个关系模式如下。

职工(职工号,姓名,性别,年龄,职务)
项目(项目号,项目名,起始日期,鉴定日期)

参加(职工号,项目号,薪酬)

(5) 同一实体集内实体间 $M:N$ 联系,必须为联系产生一个新的关系模式。该关系模式中至少包含被它所联系的双方实体的主码,若联系有属性,也要并入该关系模式中。

例如,学生实体集中有部分学生是社团的负责人,而一名学生可以参加多个社团,一个社团也可以有多名学生,所以学生实体集中学生和社团负责人就存在 $M:N$ 联系。学生实体集转换为一个学生关系,学生实体集内实体间 $M:N$ 联系就生成一个新的关系模式,该关系中至少包含学号和社团负责人编号两个属性。

(6) 两个以上实体集之间 $M:N$ 的联系,必须为联系产生一个新的关系模式,该关系模式中至少包含被它所联系的所有实体的主码,若联系有属性,也要并入该关系模式中,关系的主码可以指定一个单独的属性(如此联系 ID),否则一般为它所联系的所有实体的主码的组合,也可能为所有实体的主码再加联系中某些属性的组合。

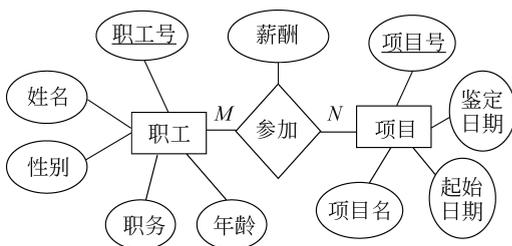


图 3-4 两实体间 $M:N$ 联系的 E-R 图

3.4.2 关系模式的优化

通常情况下,数据库逻辑设计的结果不是唯一的。为了进一步提高数据库应用系统的性能,还应努力减少关系模式中存在的各种异常,改善完整性、一致性和存储效率。规范化理论是数据库逻辑设计的重要理论基础和进行关系模式优化的有力工具,规范化的具体过程详见第 2 章。

为了提高数据库应用系统的性能,规范化后的关系模式还需要进行修改,调整结构,这就是关系模式的进一步优化,通常采用合并或分解的方法。

关系模式的优化方法如下。

- (1) 确定函数依赖。
- (2) 对于各关系模式之间的函数依赖进行极小化处理,消除冗余的函数依赖。
- (3) 按照函数依赖的理论对关系模式逐一进行分析,考查是否存在部分函数依赖、传递函数依赖等,确定各关系模式分别属于第几范式,对不符合规范的关系模式规范化。
- (4) 按照需求分析阶段得到的各种应用对数据处理的要求,分析对于这样的应用环境这些模式是否合适,确定是否要对它们进行合并或分解。
- (5) 对关系模式进行必要的合并或分解。

规范化理论为数据库设计人员判断关系模式优劣提供了理论标准,可用来预测模式可能出现的问题,使数据库设计工作有了严格的理论基础。

3.4.3 设计用户外模式

外模式也叫作子模式或视图,是用户可直接访问的数据模式。同一系统中,不同用户可

扫一扫



视频讲解

扫一扫



视频讲解

有不同的外模式。外模式来自逻辑模式,但在结构和形式上可以不同于逻辑模式,所以它不是逻辑模式简单的子集。

外模式的作用主要包括:通过外模式对逻辑模式的屏蔽,为应用程序提供了一定的逻辑独立性;可以更好地适应不同用户对数据的需求;为用户划定了访问数据的范围,有利于数据的保密等。

定义数据库全局模式主要是从系统的时间效率、空间效率、易维护等角度出发。由于用户外模式与模式是相对独立的,因此在定义用户外模式时可以注重考虑用户的习惯与方便。这些习惯与方便包括:

- (1) 使用符合用户习惯的别名。
- (2) 可以对不同级别的用户定义不同的视图,以保证系统的安全性。
- (3) 简化用户对系统的使用。

如果某些局部应用中经常要使用某些很复杂的查询,为了方便用户,可以将这些复杂查询定义为视图,用户每次只对定义好的视图进行查询,大大简化了用户的使用。



3.5 物理结构设计

数据库最终要存储在物理设备上。将逻辑设计中产生的数据库逻辑模型结合指定的 DBMS,设计出最适合应用环境的物理结构的过程,称为数据库的物理结构设计。

数据库的物理结构设计分为以下两个步骤。

- (1) 确定数据库的物理结构。
- (2) 对所设计的物理结构进行评价。

如果所设计的物理结构的评价结果满足原设计要求,则可进入物理实施阶段;否则,就需要重新设计或修改物理结构,有时甚至要返回逻辑设计阶段修改数据模型。

3.5.1 确定数据库的物理结构

数据库物理设计内容包括确定数据的存储结构、设计数据的存取路径、确定数据的存放位置和确定系统配置。

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、存储空间利用率和维护代价三方面的因素。这三方面通常是相互矛盾的。例如,消除一切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,选择一个折中方案。确定数据的存储结构包括为各行记录分配连续或不连续的物理块等。

2. 设计数据的存取路径

DBMS 常用存取方法有 B+树索引方法、聚簇(Cluster)方法和 Hash(哈希)索引方法。

1) B+树索引方法

在关系数据库中,选择存取路径主要是指确定如何建立索引。例如,建立单列索引还是复合索引,应把哪些列作为主关键字、哪些列作为次关键字建立索引,建立多少个索引合适,是否建立聚集索引等。

2) 聚簇方法

为了提高某个属性(或属性组)的查询速度,把这个或这些属性(称为聚簇码)上具有相同值的元组集中存放在连续的物理块称为聚簇。

聚簇可以大大提高按聚簇属性进行查询的效率。聚簇功能不但适用于单个关系,也适用于多个关系。假设用户经常要按系别查询学生成绩单,这一查询涉及学生关系和选修关系的连接操作,即需要按学号连接这两个关系,为提高连接操作的效率,可以把具有相同学号的学生元组和选修元组在物理上聚簇在一起。这就相当于把多个关系按“预连接”的形式存放,从而大大提高连接操作的效率。

3) Hash 索引方法

有些数据库管理系统提供了 Hash 索引方法。由于 Hash 索引比较的是进行 Hash 运算之后的 Hash 值,所以,满足下列两个条件之一,此关系才可以选择 Hash 索引方法。

(1) 该关系的属性主要出现在等值连接条件中或主要出现在相等、不等或属于比较选择条件中。Hash 索引在这几种情况中的查询效率要远高于 B+树索引。

(2) 该关系的大小可预知且关系的大小不变或该关系的大小动态改变但所选用的 DBMS 提供了动态 Hash 索引方法。

3. 确定数据的存放位置

为了提高系统性能,数据应该根据应用情况将易变部分与稳定部分分磁盘存放、经常存取部分和存取频率较低部分分磁盘存放或数据表和索引分磁盘存放、数据和日志分磁盘存放等。

4. 确定系统配置

DBMS 产品一般都提供了一些存储分配参数,供设计人员和 DBA 对数据库进行物理优化。初始情况下,系统都为这些变量赋予了合理的默认值。但是这些值不一定适合每种应用环境。

对系统配置的变量,如同时使用数据库的用户数、同时打开的数据库对象数、缓冲区分配参数、物理块装填因子、数据库的大小、锁的数目等,在物理设计时应根据应用环境确定这些值,以使系统性能最佳。

■ 3.5.2 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要求进行权衡,其结果可以产生多种方案,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方案作为数据库的物理结构。

评价物理结构的方法完全依赖于所选用的 DBMS,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,选择一个较优的合理的物理结

构。如果该结构不符合用户需求,则需要修改设计。

3.6 数据库实施和运行、维护

在数据库正式投入运行之前,还需要完成很多工作。例如,在模式和子模式中加入数据库安全性、完整性的描述,完成应用程序和加载程序的设计,数据库系统试运行,并在试运行中对系统进行评价。如果评价结果不能满足要求,还需要对数据库进行修正设计,直到满意为止。数据库正式投入使用,并不意味着数据库设计生命周期的结束,而是数据库维护阶段的开始。

3.6.1 数据库实施

根据逻辑和物理设计的结果,在计算机上建立起实际的数据库结构,并装入数据,进行试运行和评价的过程,叫作数据库的实施(或实现)。

1. 建立实际的数据库结构

用 DBMS 提供的数据库定义语言(DDL)编写描述逻辑设计和物理设计结果的程序(一般称为数据库脚本程序),经计算机编译处理和执行后,就生成了实际的数据库结构。

2. 数据加载

数据库应用程序的设计应该与数据库设计同时进行。一般来讲,应用程序的设计应该包括数据库加载程序的设计。在数据加载前,必须对数据进行整理。由于用户缺乏计算机应用背景的知识,常常不了解数据的准确性对数据库系统正常运行的重要性,因而未对提供的数据进行严格的检查。所以,数据加载前,要建立严格的数据登录、录入和校验规范,设计完善的数据校验与校正程序,排除不合格数据。

3. 数据库试运行和评价

当加载了部分必需的数据和应用程序后,就可以开始对数据库系统进行联合调试,称为数据库的试运行。一般将数据库的试运行和评价结合起来,目的是:①测试应用程序的功能;②测试数据库的运行效率是否达到设计目标,是否为用户所容忍。

测试的目的是发现问题,而不是为了说明能达到哪些功能。所以,测试中一定要有非设计人员的参与。

对于数据库系统的评价比较困难。需要估算不同存取方法的 CPU 服务时间及 I/O 服务时间。因此,一般还是从实际试运行中进行估价,确认其功能和性能是否满足设计要求,对空间占用率和时间响应是否满意等。

3.6.2 数据库的运行与维护

数据库试运行结果符合设计目标后,就可以真正投入运行了。数据库投入运行标志着

开发任务的基本完成和维护工作的开始。

对数据库设计进行评价、调整、修改等维护工作是一个长期的任务,也是设计工作的继续和提高。

概括起来,维护工作包括:数据库的转储和恢复,数据库的安全性和完整性控制,数据库性能的监督、分析和改造,数据库的重组和重构造。

3.7 数据库设计实例

3.7.1 银行卡管理系统数据库设计

1. 需求分析

与用户协商,了解用户的需求,需要哪些数据和操作(主要是查询),确定系统中应包含以下实体:储户、账户、交易记录。

储户的属性确定为:身份证号、姓名、电话、VIP。账户的属性确定为:账号、开户日期、类型、币种、密码、余额、有效日期。交易记录的属性确定为:交易时间、账号、支出、收入、对方账号、交易地点、摘要。其中,每个储户可以拥有多个账户,每个账户拥有多个交易记录。

2. 概念结构设计

画出银行卡管理系统的 E-R 图,如图 3-5 所示。

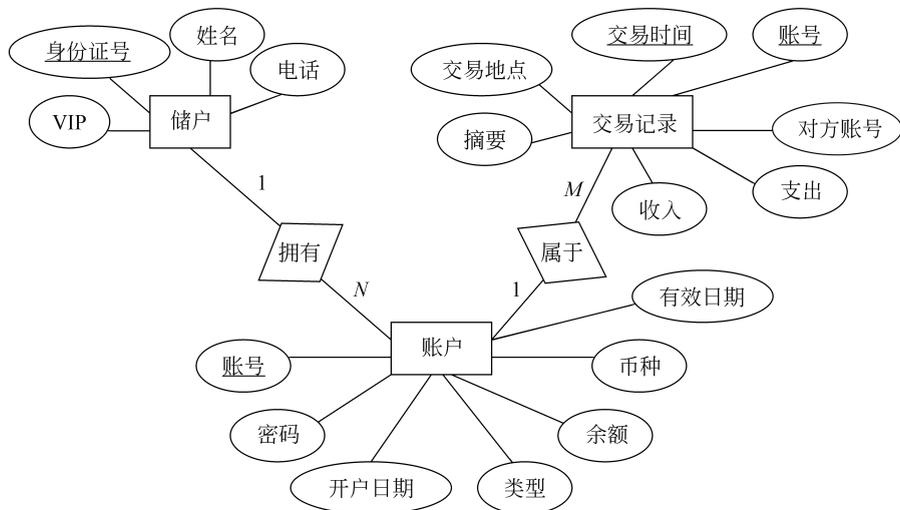


图 3-5 银行卡管理系统的 E-R 图

3. 逻辑结构设计

根据前面的转换原则,银行卡管理系统的关系模式设计如下。

储户(身份证号,姓名,电话,VIP)

账户(账号,身份证号,密码,开户日期,类型,币种,余额,有效日期)
交易记录(交易时间,账号,支出,收入,对方账号,交易地点,摘要)
表结构设计如表 3-1~表 3-3 所示。

表 3-1 储户表

属性名	类型	宽度	主键	取值范围	属性名	类型	宽度	主键	取值范围
身份证号	字符型	18	是		电话	字符型	11		数字字符
姓名	字符型	10			VIP	字符型	1		是或否

表 3-2 账户表

属性名	类型	宽度	主键	取值范围	属性名	类型	宽度	主键	取值范围
账号	字符型	20	是	数字字符	类型	字符型	3		信用卡、借记卡等
身份证号	字符型	18		参考储户表主键	币种	字符型	3		人民币、美元等
密码	字符型	6		数字字符	余额	货币型			
开户日期	日期型			默认值:当前日期	有效日期	日期型			大于开户日期

表 3-3 交易记录表

属性名	类型	宽度	主键	取值范围
账号	字符型	20	是	数字字符,参考账户表主键
交易时间	日期型		是	
支出	货币型			
收入	货币型			
对方账号	字符型	3		数字字符
交易地点	字符型	30		
摘要	字符型	20		转账、消费、工资、劳务等

4. 物理结构设计

根据查询需求设计每个关系的 B+ 树索引文件。在每张表上都按主键建立聚集索引;在账户表上分别按身份证号和开户日期建立非聚集索引;在交易记录表上分别按账号和交易时间建立非聚集索引。



扫一扫

视频讲解

3.7.2 图书借阅管理系统数据库设计

1. 需求分析

与用户协商,了解用户的需求,需要哪些数据和操作(主要是查询),确定系统中应包含以下实体:书籍、员工、部门和出版社。

书籍的属性确定为:图书号、分类、书名、作者、单价、数量。员工的属性确定为:工号、

姓名、性别、出生年月。部门的属性确定为：部门号、部门名称、电话。出版社的属性确定为：出版社名、地址、电话、联系人。

其中，每个员工可以借阅多本书，每本书也可以由多个员工借阅，每个员工每借一本书都有一个借阅日期、应还日期和实际还书日期；每个员工只属于一个部门；每本图书只能由一个出版社出版。

2. 概念结构设计

画出图书借阅管理系统的 E-R 图，如图 3-6 所示。

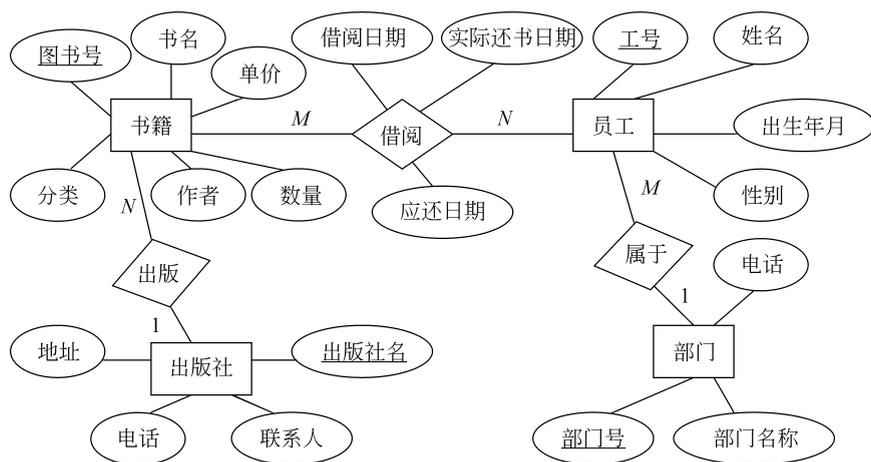


图 3-6 图书借阅管理系统的 E-R 图

3. 逻辑结构设计

根据前面的转换原则，图书借阅管理系统的关系模式设计如下。

书籍(图书号,分类,书名,作者,出版社名,单价,数量)

员工(工号,部门号,姓名,性别,出生年月)

部门(部门号,部门名称,电话)

出版社(出版社名,地址,电话,联系人)

借阅(工号,图书号,借阅日期,应还日期,实际还书日期)

表结构设计如表 3-4~表 3-8 所示。

表 3-4 书籍表

属性名	类型	宽度	主键	取值范围	属性名	类型	宽度	主键	取值范围
图书号	字符型	15	是		出版社名	字符型	20		参考出版社表主键
分类	字符型	10			单价	实型			0.00~999.99
书名	字符型	30			数量	整型			
作者	字符型	10							

表 3-5 员工表

属性名	类型	宽度	主键	取值范围	属性名	类型	宽度	主键	取值范围
工号	字符型	5	是		性别	字符型	2		(男,女)
部门号	字符型	4		参考部门表主键	出生年月	日期型			1956-1-1—2050-1-1
姓名	字符型	10							

表 3-6 部门表

属性名	类型	宽度	主键	取值范围
部门号	字符型	4	是	
部门名	字符型	20		
电话	字符型	11		数字字符

表 3-7 出版社表

属性名	类型	宽度	主键	取值范围
出版社名	字符型	20	是	
地址	字符型	40		
电话	字符型	11		数字字符
联系人	字符型	10		

表 3-8 借阅表

属性名	类型	宽度	主键	取值范围
工号	字符型	5	是	参考员工表主键
图书号	字符型	15	是	参考书籍表主键
借阅日期	日期型			2000-1-1—2030-1-1
应还日期	日期型			2000-1-1—2030-2-1
实际还书日期	日期型			2000-1-1—2030-2-1

4. 物理结构设计

根据查询需求设计每个关系的 B+ 树索引文件。在每张表上都按主键建立聚集索引；在书籍表上分别按分类、书名、作者和出版社名建立非聚集索引；在员工表上分别按姓名、部门名建立非聚集索引；在部门表上按姓名建立非聚集索引；在借阅表上分别按工号、图书号、借阅日期和应还日期建立非聚集索引。

■ 3.7.3 设备仓库管理系统设计

1. 需求分析

与用户协商,了解用户的需求,需要哪些数据和操作(主要是查询),确定系统中应包含以下实体:职工、仓库、设备和供应商。

职工的属性确定为:工号、姓名、性别、出生年月、工种(销售员、采购员、仓库管理员)。仓库的属性确定为:仓库编号、仓库名称、地址、联系电话、容量。设备的属性确定为:设备号、设备名、品种、规格。供应商的属性确定为:供应商编号、供应商名称、地址、电话、联系人。

其中,一种设备可以存放于多个仓库内,一个仓库也可以存放多种设备;一个供应商可

以供应多种设备,一种设备也可以由多个供应商提供,每个供应商供应一种设备有一个报价;设备、仓库与销售员之间有销售关系,它们是多对多的,每个销售员销售每个仓库的每种设备都有一个出库单号、出库数量和出库日期;采购员、设备与仓库之间有采购关系,它们是多对多的,每个采购员采购每种设备都有一个入库单号、入库数量和入库日期;每个仓库有多名管理员,每个管理员只能管理一个仓库。

2. 概念结构设计

画出设备仓库管理系统的 E-R 图,如图 3-7 所示。

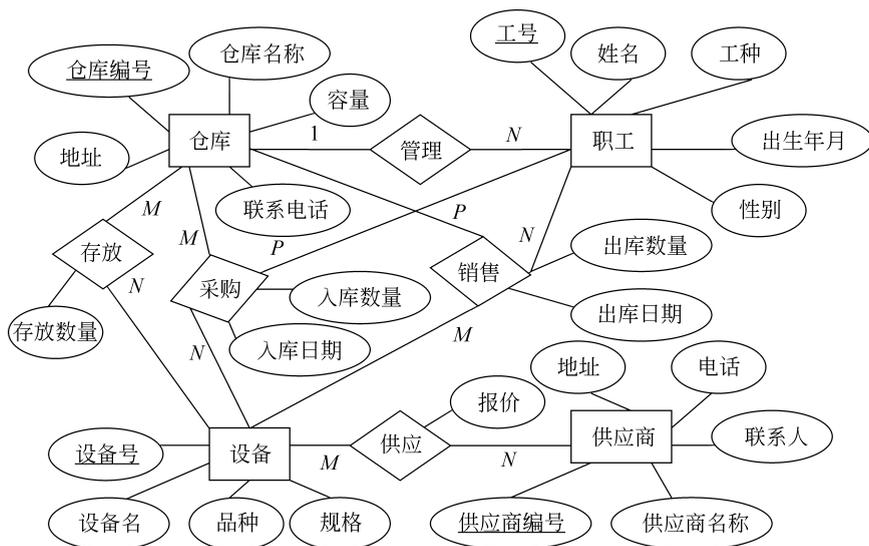


图 3-7 设备仓库管理系统的 E-R 图

3. 逻辑结构设计

根据前面的转换原则,设备仓库管理系统的关系模式设计如下。

职工(工号,姓名,性别,出生年月,工种,仓库编号)

仓库(仓库编号,仓库名称,地址,联系电话,容量)

设备(设备号,设备名,品种,规格)

供应商(供应商编号,供应商名称,地址,电话,联系人)

存放(仓库编号,设备号,存放数量)

供应(供应商编号,设备号,报价)

销售(出库单号,设备号,仓库编号,销售员工号,出库数量,出库日期)

采购(入库单号,设备号,仓库编号,采购员工号,入库数量,入库日期)

其中,职工关系中如果职工的工种为仓库管理员,则仓库编号属性的取值为仓库关系中仓库编号属性的某个值,否则为空值。

表结构设计如表 3-9~表 3-16 所示。

表 3-9 职工表

属性名	类型	宽度	主键	取值范围
工号	字符型	5	是	
姓名	字符型	10		
性别	字符型	2		(男,女)
出生年月	日期型			1956-1-1—2050-1-1
工种	字符型	10		(销售员,采购员,仓库管理员)
仓库编号	字符型	4		参考仓库表主键

表 3-10 仓库表

属性名	类型	宽度	主键	取值范围
仓库编号	字符型	4	是	
仓库名称	字符型	20		
地址	字符型	20		
联系电话	字符型	10		数字字符
容量	整型			大于或等于总存放数量

表 3-11 设备表

属性名	类型	宽度	主键
设备号	字符型	4	是
设备名	字符型	20	
品种	字符型	10	
规格	字符型	10	

表 3-12 供应商表

属性名	类型	宽度	主键	取值范围	属性名	类型	宽度	主键	取值范围
供应商编号	字符型	5	是		电话	字符型	11		数字字符
供应商名称	字符型	30			联系人	字符型	10		
地址	字符型	40							

表 3-13 存放表

属性名	类型	宽度	主键	取值范围
仓库编号	字符型	4	是	参考仓库表主键
设备号	字符型	4	是	参考设备表主键
存放数量	整型			

表 3-14 供应表

属性名	类型	宽度	主键	取值范围
供应商编号	字符型	5	是	参考供应商表主键
设备号	字符型	4	是	参考设备表主键
报价	实型			0.00~9999.99

表 3-15 销售表

属性名	类型	宽度	主键	取值范围
出库单号	字符型	10	是	
仓库编号	字符型	4		参考仓库表主键
销售员工号	字符型	5		参考员工表主键
设备号	字符型	4		参考设备表主键
出库数量	整型			小于或等于存放数量
出库日期	日期型			2000-1-1—2050-1-1

表 3-16 采购表

属性名	类型	宽度	主键	取值范围
入库单号	字符型	10	是	
仓库编号	字符型	4		参考仓库表主键
采购员工号	字符型	5		参考员工表主键
设备号	字符型	4		参考设备表主键
入库数量	整型			
入库日期	日期型			2000-1-1—2050-1-1

4. 物理结构设计

根据查询需求设计每个关系的 B+ 树索引文件。在每张表上都按主键建立聚集索引；在职工表上按姓名建立非聚集索引；在仓库表上按仓库名称建立非聚集索引；在设备表上分别按设备名、品种和规格建立非聚集索引；在供应商表上按供应商名称建立非聚集索引；在供应表上按设备号和报价的组合建立非聚集复合索引；在销售表上分别按仓库编号、销售员工号、设备号和出库日期建立非聚集索引；在采购表上分别按仓库编号、采购员工号、设备号和入库日期建立非聚集索引。



习题 3

