

<<数据库系统设计与开发>>

图书基本信息

书名：<<数据库系统设计与开发>>

13位ISBN编号：9787811231311

10位ISBN编号：781123131X

出版时间：2007-8

出版时间：北京交大

作者：朱扬勇

页数：244

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## &lt;&lt;数据库系统设计与开发&gt;&gt;

## 前言

从20世纪80年代开始,计算机得到普及,被应用到了各行各业各领域。

其中有两样东西功不可没,一是微型计算机,二是数据库。

微型计算机让每个人都有能力、有机会使用计算机,数据库让每个人都有需要、有必要使用计算机。

进入21世纪后,大众可能更加熟悉“Intel Inside; Windows Outside”,但人们真正需要的是其中的数据库。

所谓的国民经济和社会的信息化其实就是将现实世界中的事物以数据的形式存储到计算机系统中,即信息化是一个生产数据的过程。

因此,如何在计算机系统中存储、管理和使用数据应该是一个长期的问题。

数据库技术已经经历了近四十年之久的发展和应用,先后出现过层次数据库、网状数据库、关系数据库、演绎数据库(逻辑数据库,知识库)、面向对象数据库、工程数据库、时态数据库、地理数据库、多媒体数据库,等等。

其中,关系数据库从20世纪80年代以来获得了广泛的应用,到目前为止仍然是主流数据库。

这是因为关系数据库是建立在良好数学基础之上的,包括以关系代数为基础的数据库管理系统和以规范化理论为基础的关系数据库设计技术。

长期以来,数据库系统是由“数据库和访问数据库的软件系统”两部分组成。

因此,需要先设计好一个数据库,然后在这个数据库上开发应用程序,即所谓的数据驱动。

由于数据库设计技术和应用程序设计技术分别来自于数据库研究领域和软件工程研究领域,使得在实际系统开发时,常常有一个数据库小组和一个应用程序小组各行其是,在技术和文化上都存在严重的“阻抗不匹配”问题,难以融合,从而导致数据库系统开发危机重重。

虽然两个领域的专家都在努力解决这个问题,也提出了面向对象技术和持久程序设计技术等,但到目前为止并没有取得里程碑式的进展。

“一个位于对象空间的应用系统”这样的表述听起来很美,但由于没有好用的面向对象数据库管理系统,所以对象要在关系数据库管理系统上做持久化,持久化为关系表。

因此,系统开发在对象空间上进行,并最终在对象空间上建立数据库系统,数据库只是持久化对象的一个存放地,即所谓的对象驱动。

但是,开发出来的系统本质上仍然是“一个数据库和访问数据库的软件系统”。

数据库系统开发仍然需要良好的数据库设计(如规范化),数据库性能调优也常常在数据库逻辑模式上进行。

这对数据库系统开发工程师而言是一个巨大的挑战。

## <<数据库系统设计与开发>>

### 内容概要

《数据库系统设计与开发》全面地介绍当前数据库系统开发的各个方面，内容包括数据库的基本知识、数据库设计的基本原理、数据库概念设计的ER方法和语义对象方法、关系数据库规范化原理和方法、数据持久化技术、数据库保护技术、数据库性能调优、遗留数据库重建技术等等。

《数据库系统设计与开发》还介绍了数据资源开发利用技术和数据库系统的开发方法和项目管理等内容。

《数据库系统设计与开发》主要作为计算机软件专业硕士研究生教材。

《数据库系统设计与开发》在介绍各种原理、方法和技术时，强调它们的实际应用，以满足数据库系统开发高级工程师的需要。

阅读《数据库系统设计与开发》需要读者具有计算机及相关专业的基础知识，尤其需要具备数据库和软件工程知识以及一定的程序设计经验。

《数据库系统设计与开发》亦可作为计算机软件专业大学本科高年级教材或教学参考书，以及软件产业高级技术管理人员的参考书。

## 作者简介

朱扬勇，1963年生，浙江武义人。

1994年于复旦大学获计算机软件专业理学博士学位，现为复旦大学计算机与信息技术系教授、博士生导师；上海市政府信息化专家；上海生物信息研究中心学术委员会委员；上海市计算机学会理事；上海市生物信息学会理事等。

当前主要研究数据挖掘、生物信息学、数据库技术及其应用等。

在国内外主要刊物上已发表论文90多篇，其中权威刊物40余篇，建立了著名的数据挖掘讨论组网站。作为项目负责人先后从事863计划、国家自然科学基金等10多项科研课题的研究工作，获相关软件著作权6项。

## &lt;&lt;数据库系统设计与开发&gt;&gt;

## 书籍目录

第1章 数据库系统基础知识1.1 数据库概念1.1.1 从文件系统到数据库系统1.1.2 数据库系统1.1.3 数据模型1.1.4 数据库系统的体系结构1.1.5 与数据库相关的人员1.2 数据库发展1.2.1 层次数据库和网状数据库1.2.2 关系数据库1.2.3 阻抗不匹配与后关系数据库1.2.4 数据库现状1.3 数据库分类1.3.1 根据数据模型分类1.3.2 根据体系结构分类1.3.3 根据数据类型分类1.4 数据库系统类型1.5 数据建模与数据库系统开发1.5.1 人与计算机的交流1.5.2 建模的困难与方法的局限1.5.3 数据库系统开发内容1.5.4 数据建模与软件工程1.6 运行数据库的计算机系统1.6.1 集中式系统1.6.2 文件服务器系统1.6.3 分布式系统1.6.4 客户-服务器数据库系统1.6.5 浏览器系统1.6.6 说明1.7 小结思考题第2章 设计原理2.1 为什么要进行数据库设计2.1.1 现实数据表格的转换2.1.2 数据操作的异常现象2.1.3 关系表的意义2.1.4 错误的数据库2.1.5 对象持久化2.2 数据库概念设计2.2.1 为什么要进行概念设计2.2.2 ER方法2.2.3 语义对象方法2.2.4 UML方法2.2.5 XML方法2.2.6 概念设计讨论2.3 数据库设计的框架2.3.1 数据库设计的目标2.3.2 数据库生命周期2.3.3 数据库设计的内容2.3.4 数据库设计运用的方法2.4 数据库的等价性2.4.1 等价性的意义2.4.2 等价性的定义2.4.3 等价性的实现2.4.4 对象持久化的等价性2.5 其他问题2.5.1 数据库保护2.5.2 数据库性能2.5.3 遗留数据库2.5.4 数据字典2.6 小结思考题第3章 ER方法3.1 基本ER构造3.1.1 实体3.1.2 联系3.1.3 属性3.2 高级ER构造3.2.1 泛化3.2.2 汇集3.2.3 弱实体、存在依赖和标识依赖3.3 建立ER模型3.3.1 建立ER模型的基本问题3.3.2 区分实体和属性3.3.3 找出汇集层次3.3.4 找出泛化层次3.3.5 找出弱实体3.3.6 定义联系3.3.7 建立ER模型的几点原则3.4 ER模型的集成3.4.1 集成的基本问题3.4.2 简单的ER模型集成3.4.3 ER模型集成的基本步骤3.4.4 举例3.5 实际应用中的注意事项3.5.1 ER方法和ER工具软件的差异3.5.2 ER图的复杂性3.6 小结 思考题第4章 语义对象模型及其建模4.1 语义对象的概念4.1.1 语义对象4.1.2 属性4.1.3 语义对象标识4.1.4 语义对象实例4.1.5 语义对象视图4.2 语义对象的种类4.2.1 简单语义对象4.2.2 组合语义对象4.2.3 复合语义对象4.2.4 混合语义对象4.2.5 关联语义对象4.2.6 父 / 子类型语义对象4.2.7 原型 / 版本语义对象4.3 语义对象的建模4.3.1 某大学数据库4.3.2 语义对象模型4.3.3 语义对象建模的步骤4.4 与相关概念的比较4.4.1 面向对象程序设计与语义对象4.4.2 语义对象模型和ER模型的比较4.5 小结 思考题第5章 关系数据库模式规范化5.1 规范化的意义和基本假设5.1.1 关系数据库5.1.2 规范化的意义5.1.3 泛关系假设5.2 基本概念5.2.1 投影与联结5.2.2 函数依赖5.2.3 Armstrong公理系统5.2.4 闭包和投影5.2.5 覆盖5.2.6 范式5.3 模式规范化5.3.1 规范化的框架5.3.2 无损联结分解5.3.3 保持函数依赖的分解5.4 规范化算法5.4.1 计算属性闭包5.4.2 计算投影5.4.3 计算最小覆盖5.4.4 通用分解方法5.4.5 3NF分解5.4.6 BCNF分解5.5 小结 思考题第6章 概念模型转换到关系模式6.1 基本问题6.1.1 正确的概念模型转换6.1.2 数据库实现的难易程度6.1.3 转换方法问题6.2 将ER模型转换成关系模式6.2.1 转换实体6.2.2 转换弱实体6.2.3 转换汇集实体6.2.4 转换泛化层次6.2.5 转换联系6.2.6 从ER模型中产生函数依赖6.3 将语义对象模型转换成关系模式6.3.1 转换简单语义对象6.3.2 转换组合语义对象6.3.3 转换复合语义对象6.3.4 转换混合语义对象6.3.5 转换关联语义对象6.3.6 转换父 / 子类语义对象.....第7章 数据库保护第8章 数据库系统性能调优第9章 遗留系统重建第10章 数据利用技术第11章 数据库系统开发参考文献

## 章节摘录

语义对象有七种类型。

简单对象没有多值属性和对象属性；组合对象有多值属性，但没有语义对象属性；复合对象有语义对象属性；混合对象结合了组合语义对象和复合语义对象；关联对象关联两个或多个语义对象；子类型对象用来表示对象的特化；最后，原型 / 版本对象用来为包含随多个变量或版本变化的基本数据的对象建模。

应用开发通过用户视图来处理对象，一个对象的视图包含该对象的名字和所有从该视图可见的属性，视图定义和对象定义通常是递归的过程。

设计语义对象模型图集合的过程是递归的。

首先考察报表或表格，文档化一个初始的对象集合。

然后检查新的报表和表格来揭示新的对象和对现存对象的改变。

这一个过程一直持续到所有的表格和报表都考察完毕。

语义对象和OOP有一个共同的思想方向，但它们在方法方面有区别，而且是用来解决不同问题领域的。

语义对象模型和实体关系模型都是用来理解和文档化用户的数据需求的工具，主要的区别在于：ER模型把实体看作基本出发点，而语义对象模型则把语义对象看作基本出发点，而且语义对象模型所包含的关于数据意义的信息比实体关系模型多。

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>