

① 面向对象数据库管理系统的设计及其实现

115-119

TP311.13

刘 晓 霞

(西北大学计算机科学系, 710069, 西安, 30岁, 女, 讲师)

A 摘 要 依据面向对象数据库管理系统设计, 提供了一个面向对象数据库管理系统 EMS 的设计和实现的全过程。

关键词 对象; 类; 继承; 封装; 模式生成器; 浏览器; STEP 标准; 聚合

分类号 TP311.56

设计
数据库管理系统, 面向对象

面向对象的概念在软件技术的各领域中正得到越来越多的使用。今天使用的面向对象技术包括面向对象程序设计语言(如 C++ 和 smalltalk), 面向对象数据库系统(如 versant 和 objectstore), 面向对象用户界面(如 microsoft 窗口系统, frame 桌面印刷系统)等。为了满足航空工业对于工程数据管理的需要, 我们采用面向对象技术结合数据库研制开发了一套面向对象数据库管理系统。

1 面向对象的提出

术语“对象”意味着一个“数据”和“程序”的联合体。它是表示某个现实世界的实体。该对象的程序部分是一组程序, 而数据部分由各种类型的数据组成, 包括基本数据类型和用户自定义的数据类型。一个对象“封装”了数据和程序就是说用户不能看见对象“封闭壳”的内部, 但可以调用对象的程序部分来使用对象。总之, 对象概括地讲有 3 点: ①“对象”是真实世界数据的直接表现, 它同数据模型的概念相对应; ②“对象”具有永久性; ③“对象”能够共享和同时执行。

面向对象简单说就是对象封装和继承的组合。继承的意思是一个新对象可以通过扩充一个已存在的对象来创建。当继承和封装一起发挥作用时, 面向对象概念的威力就体现出来了。其原因是: ①继承使得不同类共享一组相同的属性和方法成为可能, 同一段程序就可运行在属于不同类的对象上; ②若有多个类, 并且每个类有很多属性和方法, 那么共享属性和程序的好处将是极大的, 即减少了编程量, 又降低了错误引入的机会。

由此可见“面向对象”的完整含义应当是: 从面向对象的观点去认识客观世界, 用面向对象的方法去模拟客观世界。

2 面向对象数据库管理系统

面向对象数据库管理系统, 是把面向对象的设计方法同数据库技术相结合, 而产生的用以支持非常规应用领域的新一代数据库管理系统。

2.1 产生背景

以前的关系数据库主要用于事务处理领域,其特点为:①数据量大;②只有少量计算;③进行较高频率的事务处理。而一个事务处理,则由 a. 检索少量数据;b. 对检索的数据进行简单计算;c. 把结果变更了的数据再次放入数据库等一系列简单操作所组成。以后,数据库的应用开始扩大到事务数据处理以外的领域。新的领域同以前有 3 点不同:①需要处理的数据构造复杂;②多媒质;③计算复杂。基于新的需求及面向对象技术的迅速发展和面向对象程序设计语言的启示与推动,便应运而产生了面向对象数据库管理系统。

2.2 面向对象数据库的数据模型

一个数据模型就是现实世界中的对象,以及对象的约束和对象之间的逻辑组织。数据库语言则是数据模型的具体语法,而一个数据库系统则是一个数据模型的具体实现。正如关系数据库是一种数据库系统,它实现关系型数据模型一样,面向对象数据库系统也是一种数据库系统,它直接支持面向对象数据模型。在当前缺乏统一的面向对象数据模型标准的情况下,我们完成了面向对象数据模型所包含的面向对象的核心概念。包括以下几个方面。

(1)对象和对象标识符(oid):在面向对象的系统和语言里,任何现实世界的实体都同样被模拟为一个对象,而每个对象同唯一的一个对象标识相对应。

(2)属性和方法:每个对象都有其状态和行为,对象的状态是对象属性值的集合,而对象的行为则是作用在对象状态上的方法(程序代码)的集合。状态和行为被封装在一个对象里,从对象外对它的存取或调用只能通过明确的传递消息来进行。

(3)类:类被定义为共用相同属性集和方法集的所有对象的集合。一个类相当于一个抽象的数据类型。

(4)继承:面向对象系统允许用户从现有类中导出新类,称之为原有类的子类,新类继承原有类的所有属性和方法。一个类可以有任意数量的子类,同样由基类导出的子类,也可以有任意的基类来导出,叫做多重继承。

2.3 面向对象数据库管理系统的设计方法

设计面向对象数据库管理系统的方法主要有以下 4 种方法:①在现有的关系数据库,面向记录的数据模型的上层,增加一层面向对象的数据模型;②修改现存关系数据库,使之支持面向对象的数据模型;③在面向对象的程序设计语言中嵌入数据库功能形成面向对象数据库;④独立开发全新的系统,使之支持面向对象的数据模型。

3 面向对象数据库管理系统的设计与实现

3.1 设计方法

采用在面向对象的程序设计语言中嵌入数据库功能,形成面向对象数据库的方法来实现。这种方法使得面向对象数据库的一个很重要特征体现出来,即将数据定义语言(DDL)与数据操作语言(DML)融为一体,集成为数据库程序设计语言(DBPL)。

3.2 系统结构

我们设计的数据库管理系统简称 EMS(Engineering Data Management System for Integrated Software)。EMS 将数据的操作和管理与面向对象的程序设计语言 C++“无缝”地结合在一起。这样,采用 C++ 语言中增加数据库技术,来实现 EMS 系统的核心部分。系统组成如图 1。

EMS 系统是由 6 个部分组成,从用户角度看这 6 个部分实现后的逻辑关系如图 2。

由 EMS 的逻辑结构图中可以看出:

(1)数据库的模式定义要进行两步处理。第一步用户自己使用 C++ 语言来进行数据库的模式定义,建立数据库框架;第二步用户使用模式生成器来实现对数据库的模式源文件的解释处理,将模式信息存入应用模式数据库。

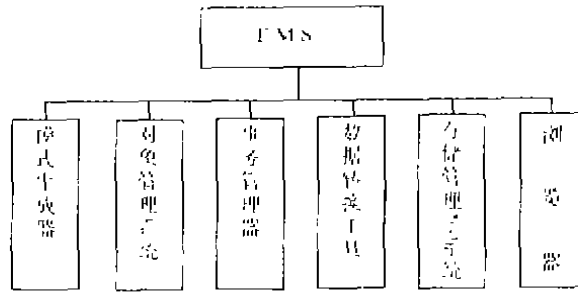


图 1 EMS 系统组成

Fig. 1 The Structure of EMS System

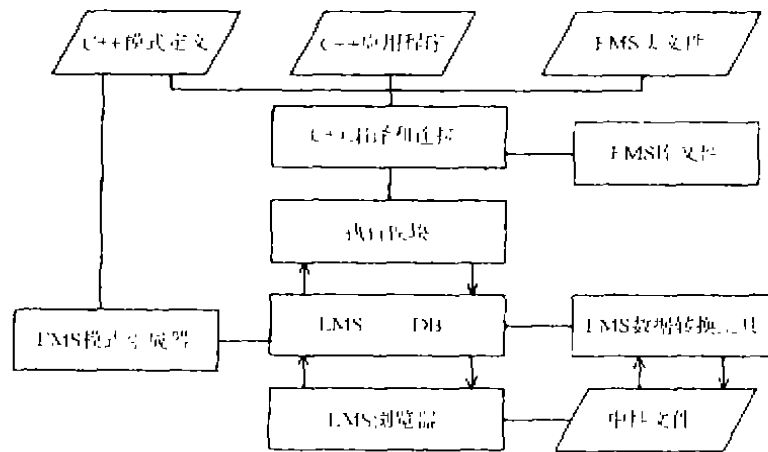


图 2 EMS 的逻辑结构图

Fig. 2 The Logic Structure of EMS System

EMS 系统的模式定义完全使用 C++ 语言的类来定义,可用 class 类,也可以用 struct 类。其语法规则与 C++ 语言相一致,这样,由全部永久类的定义形成一个模式文件。

(2)EMS 的浏览器可以使用户很方便地看到数据库中存放的数据。

(3)EMS 的数据转移工具是以产品数据表示与交换的国际标准 STEP (ISO 10303) 作为标准的一种转换工具。该工具提供了异质数据库间进行数据交换的一种手段。

(4)EMS 系统的执行模块的生成,是把 EMS 系统的操纵函数以库文件的方式来使用,这样大大方便了用户的使用。

3.3 系统实现

EMS 系统的核心部分完全是采用 C++ 语言来实现的。而在提供辅助工具中,也采用 WINDOWS 操作系统提供的便利的窗口管理函数。以下对 EMS 系统 6 个组成部分的实现进行说明。

3.3.1 模式生成器 模式生成器解释处理模式源文件,将模式信息存入应用模式数据库,如图 3 所示。

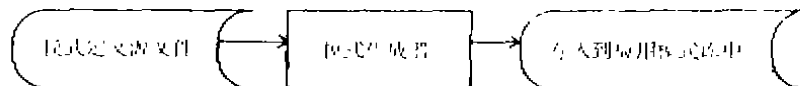


图 3 模式生成数据流图

Fig. 3 The Data Flow Diagram of Scheme Generator

将一个模式定义源文件(我们设计是以 .DDL 为扩展名的源程序文件)输入到模式生成器时,通过解释执行,进而产生应用程序的头文件(我们称之为 .HPP 扩展名的文件)和应用程序的字典文件(我们

称之为, DIC 扩展名的文件), 建立起数据库的框架。

3.3.2 对象管理子系统 该子系统主要使对象永久化, 以及使对象之间发生联系, 由两部分组成。

(1) 对象管理。为了进行对象管理, 我们为用户提供如下若干操纵函数:

- a. obj-def 函数, 定义一个新的永久对象, 即将永久类在内存中的一个临时对象定义为永久对象。
- b. obj-redef 函数, 用一个已定义的对象标识定义一个永久对象。
- c. obj-undef 函数, 取消永久对象的定义。
- d. obj-get 函数, 从数据库中取出一个永久对象的值。
- e. obj-put 函数, 向数据库存入一永久对象的值。
- f. obj-retrieve 函数, 在数据库中检索一个永久类的所有永久对象。

(2) 聚合管理。聚合管理是在对象管理的基础上, 来建立各个对象之间关系, 它可以方便地描述对象之间 1:m 的关系, 从而可以实现对象之间的多对多关系。并且能进行相应的查询功能。它不同于关系数据库中表的关系, 它是一种更灵活、更能反映现实世界真实情况的一种技术。一个聚合是由一个主对象和若干成员对象组成的。我们为用户提供的聚合管理操纵函数主要有:

- a. col-create 函数, 创建一个聚合。
- b. col-delete 函数, 删除一个聚合。
- c. col-insert 函数, 在聚合中插入永久对象。
- d. col-remover 函数, 在聚合中删除永久对象。
- e. col-findoidbyowner 函数, 按主对象标识寻找聚合。
- f. col-findoidbymember 函数, 按成员对象标识寻找聚合。
- g. col-oidismember 函数, 判断一个对象是否为聚合的成员对象。
- h. col-number of member 函数, 获取聚合中成员对象的数目。
- i. col-findowner 函数, 寻找聚合的主对象。
- j. col-findmember 函数, 查找聚合的一个成员对象。
- k. col-findallmembers 函数, 查找聚合的所有成员对象。

3.3.3 事务管理器 EMS 系统允许多个事务处理同时进行, 这就需要进行并发控制。我们采用事务管理器来完成并发控制, 为用户提供如下操纵函数:

- a. transaction-begin 函数, 表示一个事务处理的开始。
- b. transaction-abort 函数, 说明一个事务处理被中断, 不改变数据库中的内容。
- c. transaction-end 函数, 表示一个事务处理成功的结束, 并将新的内容写入数据库中。

3.3.4 数据转换工具 数据转换工具是为了解决异质数据库之间进行数据交换而提供的, 它采用产品数据表示与交换的国际标准 STEP 作为标准。处理流程如图 4。

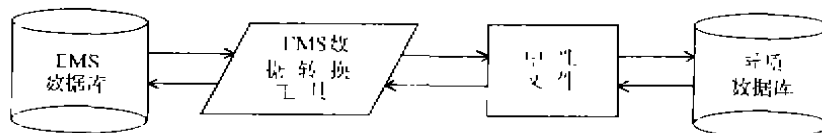


图 4 数据转换工具的处理过程

Fig. 4 The Handling Process of Data Transform Tool

3.3.5 存储管理子系统 该子系统是整个系统的最底层。正如许多数据库管理系统的存储管理一样, EMS 系统的最底层不是面向对象, 而是传统的面向记录。对于对象的存储, 是采用 B+ 树的方法来组织的。数据库中的对象和内存中的对象有不同的格式, 并且放在不同的缓冲区中。当一个对象创建后, 存储在数据库, 它首先创建一个对象缓冲区, 然后转换为一个记录, 再移到由 DBMS 管理的一个记录缓冲区。在该子系统中, 为用户提供如下操纵函数:

- a. db-open 函数, 数据库的打开。
- b. db-destory 函数, 数据库的毁坏。

c. db-close 函数,数据库关闭。

3.3.6 浏览器 EMS 系统的浏览器是一个用户可以显示一个应用数据库内容的图形工具。它是一个在 WINDOWS 系统支持下的具有良好用户界面的窗口管理系统。通过浏览器,用户可以进行如下的操作:①显示一个数据库模式的信息;②显示存储在数据库中各个对象的信息;③创建对数据库中对象聚合的查询;④运行对数据库中对象聚合的查询。

3.4 系统的技术优势

本系统的技术优势在于以下几个方面:

(1)采用了完全的面向对象思想和方法来设计,充分使用 C++ 语言所给的类库和 WINDOWS 操作系统的类库。

(2)采用国际标准的产品数据表示与交换 STEP 来解决异质数据库之间数据交换的问题。

(3)解决了长数据处理的问题,特别对当今多媒质的数据处理有着特殊的意义。

参 考 文 献

- 1 汪成为,郑小敏,彭木昌. 面向对象分析、设计及应用. 北京:国防工业出版社,1993
- 2 Won Kim. 面向对象数据库系统:诺言、现实、前景. 计算机科学,1994,21(5):51~56

责任编辑 张素敏

Design and Implementation of an Object-oriented Database System

Liu Xiaoxia

(Department of Computer Science, Northwest University, 710069, Xi'an)

Abstract The design process of an object-oriented database management system is discussed; A completed object-oriented database management system EMS is presented.

Key words object; class; inheritance; encapsulation; Scheme generator; browser; STEP standard; collection