

# 通用管理信息系统的实现原理及技术

肖午光 吴锡琪 (武汉工业大学)

**摘要:** 本文根据当前管理应用软件存在的问题,具体介绍已开发的通用管理信息系统的设计原理、实现技术及系统特点。

## 一、软件重用技术及程序生成技术

软件重用是指一个实体应用于不包括最初使用它的任何上下文中,随着子程序的出现以及子程序的建立,这些思想得到了发展。软件重用技术不仅能够提高软件生产的速度和效率,而且还能够使软件的可靠性得到进一步的保证,因为组成软件的构件是使用工程原型法编制并经过反复调试和测试的。

根据被重用软件成份的形式,人们把软件重用技术分为合成技术和生成技术。子程序库法是合成技术的代表性方法,而程序生成器则是生成技术的代表性方法。

程序生成器是这样一个特殊软件系统,它接收一些特殊的描述信息作为规范说明,产生符合要求的具体程序。由于程序生成器采用了可重用软件工程新技术以及第四代语言的新概念,运用它进行软件开发主要有以下优点:

- (1) 系统生成速度快,软件开发周期短,成本低。
- (2) 生成的程序质量高、运行速度快、冗余度小;软件维护费用低。
- (3) 生成的系统适应性强,易于修改、扩充。

程序生成器共有两种形式:目标程序方式和表驱动方式。在目标程序方式中,程序生成器在一个专用的程序代码段或程序代码模式库的支持下工作,依据输入的需求信息,程序生成器自动生成符合要求的具体程序或程序段,并按照目标程序描述语言的语法要求,生成必要的语句,使之与库中的程序段一起组成可执行的目标程序。不同语言之间的转换多采用这种方式,但这种方式有某些缺点,如生成的目标程序的功能完全依赖于目标程序描述语言,相同功能的模块之间仍存在冗余现象

等。

表驱动方式的程序生成器采用的是一种独特的软件设计方式,它把程序中的控制信息从程序中分离出来,形成抽象程序与控制信息表两部分。该抽象程序是可重用的,在具体的应用中不必改变。而控制信息表则是可变的,控制信息表的内容不同,程序的运行结果就会不同。控制信息表和抽象程序是建立在一个统一的标准之上的,但控制信息表的结构一般比较繁琐,用手工编制难以完成,通常用一种工具表生成程序来产生,这种表生成程序也称之为“程序生成器”。

表驱动方式的程序生成器比目标程序方式的程序生成器更具有优越性。首先,程序生成器只生成控制信息表中的一些信息数据,系统生成速度更快,修改维护也更方便;其次,抽象程序是可重用的,一套程序可适用于所有的数据库,程序的冗余度更小,几乎为零;再其次,由于这种形式只生成信息数据而不生成程序,使这些抽象程序也可以用多种语言联合编写实现,不仅可以完善和加强系统软件的总体功能,而且还可以提高其运行效率。

本系统的设计采用后一种方式,并对其进行了某些改进和扩充,从程序中分离出来的不仅仅是一些控制信息,而且还有一些特征参数和数据,这样就可以在保持其原有的优点的前提下,进一步提高程序的通用性和灵活性。

## 二、程序抽象技术

在表驱动方式的程序生成器的实现中,有一个至关重要的问题,即程序抽象问题。

在 dBASE III 中,它的每一条命令(操作)都是针对抽象数据库而不是针对具体的逻辑数据库设计的,通过当

前分区上打开的数据库库结构信息,它实现了操作与具体数据库之间的映射(如图1所示)。

而采用两层式结构,两层之间数据依赖性很强,一旦其中的数据库个数、数据库库结构等特性参数发生变化,这种管理应用软件将会由于所组成的模块涉及到过多的具体细节而无法灵活地在新环境中重用。因此,有必要修改一下一般管理软件的操作模型。

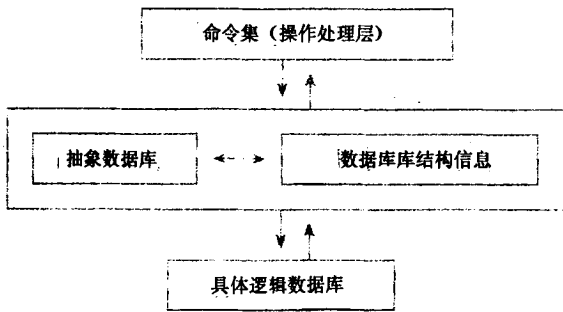


图1 DBMS的操作模型

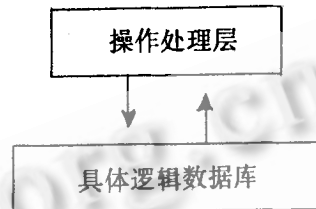


图3 一般管理软件的操作模型

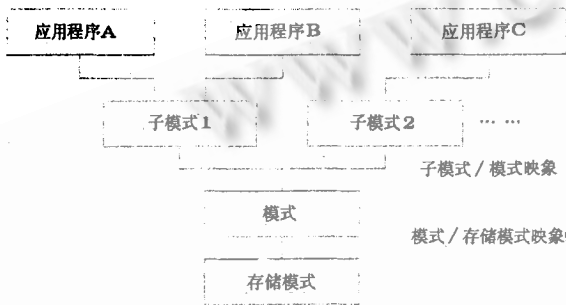


图2 应用程序和子模式及各级模式间的联系图

参照一下应用程序和子模式以及各级模式之间的联系图(如图2)不难发现:由于模式与存储模式之间有模式/存储模式映像,如果存储模式变化,有可能修改此映像使模式保持不变,从而不必修改程序,这称为程序与数据的物理独立性;如果模式改变,同理可以修改子模式/模式映像而使子模式保持不变,从而不必修改程序,这就是程序与数据的逻辑独立性。受此启发,可以设想在应用程序和子模式之间再加入一层映像,使得在子模式变化的情况下,可以通过修改这层映像而保证应用程序不必修改,从而形成程序与数据的更高一级的独立性。

目前国内大多数管理应用软件都是自行设计开发的,其可移植性差的主要原因就在于这些管理软件所采用的系统结构都是以逻辑数据库为核心的,建立一个信息处理层之后,再外加一层用户界面。具有这种系统结构的管理软件的操作模型都是以具体操作对具体数据库的方式建立的(如图3所示),系统设计时抛弃了中间层

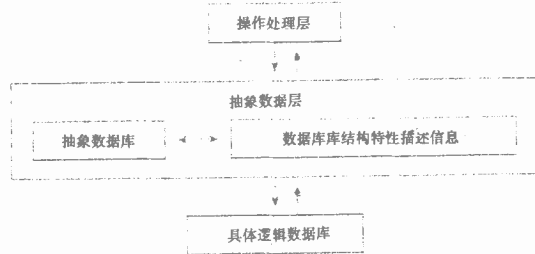


图4 通用管理应用系统的操作模型

本系统采用结构重用方法,在逻辑数据库与操作处理层之间增加了一个抽象数据层(如图4所示),使得本系统的操作模型变成了三层结构。此抽象数据层的功能相当于映像的功能,它主要包括以下两个内容:面向操作处理层的抽象数据库,以及面向具体数据库的库结构特性描述信息。通过实现数据库库结构的参数化,使得对操作处理层来说不同的数据库仅是抽象数据库的一个实例而已。这样,数据的操作处理都针对抽象数据层进行,而其具体实现则通过抽象数据层中的数据库库结构特性描述的映射对具体数据库进行,从而达到在数据库发生变化时,其特性参数变化仅通过映射反映到特性描述块,而不会影响到系统操作处理层的效果,于是就实现了具体逻辑数据库的抽象性、封闭性及隐藏性,而这正是本系统程序抽象实现的关键所在。

显然,如果用这种方法进行管理信息系统的设计,那么一套程序就可以完成对任何具体数据库的基本操作,

大大提高程序的通用性。

### 三、数据字典技术

在抽象数据层中,数据库库结构特性描述是实现程序抽象、完成从抽象数据库到具体逻辑数据库的有效映射所必不可少的,因此,必须为数据库库结构的特性描述建立辅助数据库,即数据字典。这样,当操作处理层对抽象数据层中的抽象数据库执行数据操作时,抽象数据层会自动提供当前数据库的库结构特性描述信息,然后将其与抽象数据库复合后映射到当前数据库上。

数据字典在数据库中被定义为关于数据描述信息的集中的库。当然在本系统中,数据字典的含义和应用得到了进一步扩充,它不仅用于组成抽象数据层和控制程序的流程,还广泛用于菜单设计、文档管理、数据录入/编辑格式设计、数据查询、参数设置以及系统资源管理等各方面。数据字典的使用不仅使程序通用性得以实现,而且使各功能模块通过它进行信息传递,并以此为标准结合成一个整体。

### 四、软件集成技术

集成环境是通用型软件所必不可少的,因为通用型软件必然包含多种功能,每种功能中又有若干选择项,以满足不同用户的不同功能需求。为了使用户能方便地调用这些功能,最好的办法当然是将它们置于一个集成环境中,由用户根据需要去选择,因此本系统的各功能模块通过数据字典和抽象数据库实现了软件的集成。本系统软件不仅具有一般管理应用软件的数据录入、编辑、检索、统计以及表格输出等功能,而且还有管理应用软件的生成修改、统计图形、文档管理、数据保护、系统扩充等功能。所以将这些众多的功能置于集成环境之中,用菜单对它们进行分类编排,然后用一个统一的菜单管理模块来驱动。

### 五、系统总体结构设计

通用管理信息系统的系统结构复杂,不仅功能模块众多,而且各功能模块之间信息传递频繁,软件结构可以用一个可生长的多级树表示。为了使系统结构清晰,本系统在设计时首先将多级树中的枝与叶分开,枝完成用户程序接口、菜单驱动和模块驱动三种功能,由总控模块

实现;而叶只完成某特定的功能,因而可以用模块法分别单独进行开发,这样就一个多层次的结构分解成若干个独立的部分,将一个复杂的问题简化了。

模块化方法为软件系统的结构设计提供了最佳手段,它通过模块分解、抽象、信息隐蔽等原理构造一个结构清晰、容易理解、容易阅读的系统。这样的系统在分工编程、调试、工程组织和管理方面提供了极大的方便,大大提高了 MIS 的可靠性和可维护性。

本系统的设计采用了模块化设计中的结构设计法(SD法)。由于传统的 SD 法把整个系统的结构画在一张图上,对于大型系统很不方便,因此,本系统按照低耦合高内聚的原则,在将本系统进行合理的模块分解后,对各功能模块再实行 SD 法,简化了结构图的设计过程,设计文档的管理工作也变得简单而有条理。此外,动态结构的实现使用户可以方便地通过程序接口扩充系统的功能,增加系统的灵活性和自适应性。

本系统软件充分利用数据字典,实现了动态结构,主要包括以下四个方面:

#### 1. 动态菜单

动态菜单的实现,使得功能模块的增减只影响菜单数据字典中的数据记录的数目,用户通过外部程序接口和菜单的扩展就可方便地将自己开发的功能模块纳入到集成环境中,反之亦然。

#### 2. 动态库结构

动态库结构的实现,使得本系统不仅允许用户随时可根据环境变化来修改系统,而且还可以使系统生成任何类型的管理应用系统。此外,它也能让多种不同性质的事务管理或信息处理子系统(如财务管理,学籍管理,设备管理等等)处于同一系统之内。

#### 3. 动态数据录入/编辑格式

动态的数据库库结构,必然要求动态数据录入/编辑格式与之相对应。此外,为了方便用户,按用户习惯的格式进行数据录入/编辑,用户也可能自定义屏幕格式。

#### 4. 动态报表格式

本系统软件能生成和输出任意格式的三维报表,同时支持长字段的分行打印。动态报表格式的实现使用户可以用不同的表格形式输出不同(或相同)的内容,同时修改表格也更加方便,彻底改变了一般管理应用程序设

计所采用的一个表格对应一个专用程序的格局。

## 六、系统主要特点

### 1.生成速度快

全屏幕操作方式和热键求助,使系统生成和修改操作简单易学,再加上系统生成的只是参数数据,所以系统的生成速度很快。如果已有现成的数据库库结构,生成一个典型的中等规模的 MIS 仅需要几个小时,大大提高了 MIS 软件的开发效率。

### 2.生成系统功能强

它既能生成各类型的事务处理数据库应用系统,又可生成由各种异类事务处理数据库应用系统组成的多子系统的综合系统。

### 3.系统程序质量高,冗余度极小,维护费用很低

### 4.生成的应用系统用户界面良好和操作简便

精炼的人机对话使操作简单而容易;下拉式菜单和热键求助使用户操作时刻都可以得到提示和帮助。

### 5.生成的应用系统功能强

系统有用户自定义屏幕格式、表格式操纵自由组合条件查询、统计图形、数据保护、系统修改和扩充等这些传统的 MIS 软件所没有的功能。

### 6.生成的应用系统适应性强

由于系统设计采用了模块化结构,系统容易扩充;同时动态结构的实现,使用户随时都可以根据环境变化对系统进行修改。

### 7.生成的应用系统运行效率高

本系统主要采用编译 dBASE III 实现,一般编译 dBASE III 比解释 dBASE III 快 5 倍左右,再加上利用汇编语言解决了汉字信息处理中的速度瓶颈问题(如半个汉字的判断处理问题等),使系统运行效率进一步提高。

### 8.程序和数据的保密性和安全性好

编译后的程序很难破译其中的加密部分和口令代码,同时程序可自动对系统数据字典和用户数据库文件进行加密,使其它用户很难得到系统或其他用户的明文数据。

### 9.支持“快速原型”以及“用户早期验证”等新的软件开发模式

快速建立原型和快速执行原型,使软件开发信息反馈速度快,反复修改周期短,用户参与彻底,从而大大提高了软件开发的效率和有效性,降低了软件的开发成本和维护费用。

### 主要参考文献:

[1] 萨师煊、王珊:《数据库系统概论》,高等教育出版社,1983年8月

[2] 金淳兆、余江:软件重用技术,《计算机科学》,1989年第5期

[3] 吴锡琪、肖午光:信息管理应用软件生成器——通用管理信息系统,《微机发展》,1991年第1期

[4] 肖午光:表格操纵式的自由组合查询,《现代电子技术》,1992年第2期