

# 管理信息系统中数据库的打开与关闭

王志龙 (合肥工业大学计算机网络研究所 230009)

**摘要:**本文针对直接利用 FOXPRO 数据库管理系统中数据库的打开和关闭操作命令的不足,提出了建立数据库的打开信息库、打开函数和关闭函数来打开和关闭数据库的方法,实践证明该方法行之有效。

**关键词:**数据库 打开信息库 打开函数 关闭函数

## 1. 引言

管理信息系统是一个以数据库为核心的系统,尤其是在基于 FOXPRO 开发的大型 MIS 中,成百上千的数据库分布在不同的物理设备上的不同路径之下,通过大量地应用程序对这些数据库进行着各种操作以满足特定的应用要求。在这些应用程序中,对所要操作的数据库,首先必须将其打开,最后将其关闭,如果直接利用 FOXPRO 数据库管理系统提供的打开/关闭数据库的命令---USE [< 数据库文件名>]...[ALIAS <工作区别名>][ AGAIN],将存在诸多不便之处,主要表现为:

(1) USE 命令打开数据库的基本要求是要指明数据库的名称及其存放路径,程序中由于加入了数据库的具体路径,一旦该数据库的存放路径发生改变,就必须对相关程序一一作相应地修改。

(2) 为了能在多个程序中同时打开同一个数据库,必须在每一个 USE 命令中加入 "ALIAS <工作区别名> AGAIN",并且要保证工作区别名不能重复。

(3) 当希望对同一数据库在打开后都指向相同的索引文件时,必须在 USE 命令中重复地加入 "CDX TAG <主索引标记>" 或在 USE 命令之后重复地写入 "SET ORDE TO <主索引标记>" 命令。

(4) 无论是打开还是关闭数据库,都必须在 USE 命令前用 SELE 命令逐个选定工作区。

为此,本文提出一种通过设计数据库的打开信息库、打开函数和关闭函数来打开和关闭数据库的方法,该方法不仅实现了应用程序中数据库的打开与其路径无关,而且使得程序设计简单化。

## 2. 设计思想

给系统中的数据库分别赋予不重名的数据库编码,设计和建立一个对这些数据库执行打开操作时所需信息进行集中管理的打开信息库、配对使用的一个数据库打

开函数---OPENDB() 和一个数据库关闭函数---CLOSEDB()。数据库的打开信息库在系统主控程序的开头部分打开,结尾部分关闭;OPENDB() 函数和 CLOSEDB() 函数挂接在系统主控程序中,以使得它们能在系统中的任一程序中直接调用,程序设计人员只需从当前程序自身的处理逻辑出发,在打开某个数据库时,只需定义为: <变量名> = opendb(' <数据库编码>'); 在关闭某个数据库时只需定义为: = closedb(' <数据库编码>', <变量名>)。

OPENDB() 函数根据数据库编码判定当前是否执行打开数据库的操作,并返回一个运行标志值,作为 CLOSEDB() 函数用来判定当前是否执行关闭该数据库操作的依据。若当前无需打开数据库,则保存好数据库的当前状态,以便在执行对应的 CLOSEDB() 函数时进行状态恢复。否则,将根据数据库编码到数据库的打开信息库中查找该数据库编码所对应的数据库名称、路径、主索引标记等打开数据库所必需的信息,用 "SELE 0" 选择工作区,在用 USE 命令打开该数据库的同时将数据库编码作为数据库所在工作区的别名,从而使得系统中所有程序对数据库的操作都统一用数据库编码表示,以提高程序的可读性和可理解性。

CLOSEDB() 函数根据数据库编码和 OPENDB() 函数返回的该数据库编码运行标志值判定当前是否执行关闭数据库的操作。若当前不关闭数据库,则将其恢复到执行 OPENDB() 函数之前的状态。

## 3. 设计方法

(1) 数据库编码的结构及长度设计。在进行数据库设计时,尤其是进行网络环境下的数据库设计时,往往存在将同一个数据库存放在多个路径之下情形。为便于对系统中的数据库进行有效地管理,必须寻求一种方法使得通过数据库编码就能从数据库的打开信息库中方便地确认出哪些是属于存放在多个路径下的同一个数据

库,为此,我们将数据库编码设计为由两大码段组成的结构,第一个码段表示数据库名称不同的数据库的编码,第二个码段表示数据库名称相同的数据库在各个路径下的编码或同一路径下有不同打开要求的编码。

由于在打开数据库时用数据库编码作为其所对应的数据库所在工作区的别名,因此,它必须满足 FOXPRO 数据库管理系统自身对工作区别名不超过 10 个字符的长度限制,考虑到一个系统中可能拥有的数据库个数、所确定的数据库编码结构及数据库编码应具有一定的可扩展性等因素,我们将码长设计成八位,前六位用于第一个码段,后两位用于第二个码段。

(2) 打开信息库的设计。打开信息库是 OPENDB() 函数实现按数据库编码打开其所对应的数据库的依据,因此,在打开信息库中必须记录打开数据库所需的数据库的编码、名称、存放路径、打开方式和主索引标记这几项基本信息,此外,还可记录一些为便于管理系统中的数据库而需要的辅助信息,如数据库的中文简称等。一般地打开信息库的结构设计如下:

字段名	类型	长度	注释
SJKBM	C	8	记录数据库的编码
WLLJ	C	20	记录数据库的存放路径
SJKMC	C	10	记录数据库的名称
ZWMC	C	24	记录数据库的中文名称
ZSYBZ	C	10	记录在数据库打开时就指向的主索引标记
GXX	C	10	记录数据库的打开方式(独享/共享)

为了提高根据数据库编码对数据库的查询速度,我们将打开信息库按数据库编码为关键字进行索引,索引标识记为-SJKBM。

(3) 打开信息库所在工作区别名的设计。虽然对于一个具体的 MIS 系统而言,其打开信息库的名称是确定的,但是各个 MIS 系统的打开信息库的名称不一定相同,为了消除不同系统打开信息库名称的差异对 OPENDB() 函数的影响,提高 OPENDB() 函数的通用性,使其能直接挂载在任一系统的主控程序中,我们在任一系统的主控程序中打开数据库的打开信息库的同时,赋予一个统一的固定不变的工作区别名,在 OPENDB() 函数中引用同样的工作区别名就可实现对打开信息库的操作。

(4) 数据库打开函数(OPENDB())的设计。

①参数设计。OPENDB()函数运行的依据是数据库编码和打开信息库,在(3)中已经阐明,打开信息库因其在被打开之时就赋给了一个固定不变的工作区别名,在

OPENDB()函数中可以直接引用而无需将其作为一个参数传入。由于系统中要打开的数据库是多变的,因此,在 OPENDB()函数中必须设有一个能接收数据库编码的参数,以确定在 OPENDB()函数所要操作的数据库。

②数据库虚拟打开时的返回值设计。OPENDB()函数最重要的一点就是在判断出数据库编码所表示的数据库已经打开、当前只是虚拟打开(并不真的执行打开数据库操作)的情况下,返回一个什么样的函数值,使得 CLOSEDB()函数能正确地对该数据库的当前主索引和当前记录等状态信息进行恢复。经过分析之后,我们选取由数据库的当前记录号和当前主索引标记共同组成数据库虚拟打开时的返回值。

此外,我们选取'MUSTCLOSE'作为当前执行了打开数据库操作的函数返回值,以告之在执行对应的 CLOSEDB()函数时必须关闭该数据库。

(5) 数据库关闭函数(CLOSEDB())的设计。

①参数设计。由于设计人员在设计每一个程序时都是从当前程序自身的处理逻辑出发,确定是否需要调用 OPENDB()/CLOSEDB()函数来打开/关闭数据库的,这样在系统运行过程中,就可能存在对同一个数据库进行多次调用 OPENDB()/CLOSEDB()函数的问题。因此,在 CLOSEDB()函数中必须带有两个参数,分别接收当前所操作的数据库编码和 OPENDB()函数对应的操作该数据库编码的返回值,以确定当前是执行关闭数据库操作,还是进行数据库虚拟打开时的状态恢复。

②数据库虚拟打开时的状态恢复。在 OPENDB()函数中由于选取了由数据库的当前记录号和当前主索引标记共同组成数据库虚拟打开状态下的返回值,这样我们只需要简单地对 CLOSEDB()函数中接收该返回值的参数进行分解就能获取并恢复执行 OPENDB()函数前数据库的主索引,但是在恢复数据库的记录指针时,必须考虑到程序对数据库进行的记录删除操作,因此应将参数分解所得到的记录号与数据库的当前记录数进行比较,若记录数大于记录号,则将指针指向记录号所示的记录,否则将指针指向最后一条记录。

#### 4. 结束语

本文提出的方法已成功的应用到我们承担的《安庆石油化工总厂计算机辅助管理及生产过程监控系统》(安徽省科技进步一等奖)、《永信通用财务软件开发》等大、中型项目的开发中,实践证明该方法效果很好。

(来稿时间:1997年3月)