

财政应用系统中的代码管理问题

李奇云 (财政部计算中心)

摘要:本文介绍了财政应用系统涉及的代码管理中的一系列问题,包括代码的设计、代码库结构的设计和代码管理功能的实现等问题。

一、引言

财政计算机应用系统所管理的对象是大量的、错综复杂的、多变的财政信息。为了方便、有效地管理这些信息,有必要对所管理的基本信息分门别类编以代码。所谓代码是指用来表示特定事物的一个或一组字符。可以说,代码是任何一个大的计算机应用系统的基础。其作用主要表现在以下几个方面:

- (1) 简化和规范信息;
- (2) 节省信息的存储空间;
- (3) 便于信息的使用和控制;
- (4) 容易建立信息之间的联系;
- (5) 为实现数据共享奠定了基础;
- (6) 使财政口径调整成为可能。

代码管理就是对应用系统中所涉及的各种不同类别的代码有组织地进行统一的、规范化的管理,建立一个完备的代码体系,并且能随时根据代码的变化情况进行维护。

二、代码设计

对于现有的代码标准,应坚持代码标准和实际情况相结合的观点。在代码标准符合实际情况、满足实际需要的前提下,尽可能采用已有的代码标准;若代码标准与实际情况不完全相符,则对它们进行适当的改进后加以利用。

若没有现有的代码标准可利用,则自行设计。所设计的代码应合情合理,能准确、有效、充分反映对象信息,确保信息检索、分类、访问、判断和控制简单容易,提高信息的处理效率和信息的利用价值。为此,在设计编码规则和进行具体编码时,应遵循以下原则:

- (1) 完备性,每一代码集应充分包括该集所有的对象,没有任何遗漏;
- (2) 唯一性,每一代码集中的每一代码表示且只能表示一个对象;
- (3) 类属性,将代码集中的所有代码按某种属性分成若干互不相交的类,使每一个代码且只属于其中一类;

(4) 层次性, 将代码划分为若干层次等级, 以此反映代码的从属关系;

(5) 可扩充性, 在确保完整性的前提下, 代码的长度和段数应留有余地, 以便今后根据需要进行扩充。

代码的基本类型, 按码的组成成分有: 数字型代码、字母型代码、字母数字型代码等; 按码的结构分有: 顺序码、层次码、复合码等。

财政应用系统可能涉及到的代码主要有以下两大类:

(1) 指标科目类, 包括预算科目代码、会计科目代码、预算外科目代码、企财指标代码、税务代码和国民经济综合指标代码等。

(2) 单位类, 包括地区代码、行业代码、企业代码和部门代码等。

下面各举一例说明编制代码的方法。

例一、预算科目代码。它是由表码、类码、款码、项码和目码所组成的数字型代码, 共十二位。其结构如下:

表 码 类 码 款 码 项 码 目 码

例二: 地区代码。它是由省码、市码和县码所组成的数字型代码, 共六位。其结构如下:

省 码 市 码 县 码

三、代码库结构设计

代码库是用来存储各种代码信息的数据库, 其结构设计得是否合理直接影响代码存储、代码检索、代码管理和基于其上的应用系统的效率。根据代码的不同性质和不同作用, 可分以下几种情形来考虑代码库的结构。

1. 不含口径调整的代码库的结构

如果待设计的代码库仅用来存储每一年所使用对象名称与其代码的对应关系, 并不涉及口径调整问题, 那可以按年份设计代码字典。其基本结构如下:

dm	CHAR / INTEGER	代码
mc	CHAR	名称
bm	CHAR	别名

2. 含口径调整的代码库的结构

由于财政业务自身的复杂性及改革中的财政的多变性, 财政指标科目也经常随之变化, 不同年份相应的指标科目所包含内涵不一致, 因而导致不同内涵的相应指标

科目在进行对比分析时不具有可比性。为了使不同内涵的相应的指标科目能进行对比分析, 在对比分析之前, 需要将相应的指标科目进行口径调整。

为了便于进行口径调整, 在设计代码库时, 除了按年份建立代码字典之外, 我们还建立一个代码档案, 用代码档案来记载、存放代码建立以来代码的所有变化过程及变化情况。这些情况包括每一个具体科目代码何时建立、何时发生变化及变化关系如何、何时被删除等。代码档案的基本结构如下:

dm	CHAR / INTEGER	代码
mc	CHAR	名称
bm	CHAR	别名
rq	DATE	日期
bzgzmx	CHAR	变更模型
bglx	CHAR	变更类型

“日期”用来记载科目代码建立及发生变化时的具体日期。

“变更模型”用来描述某一年的科目是由上一年中哪些相应科目代码演变过来的, 其具体形式为上一年相应科目代码的数学表达式。它是进行口径调整的基础和依据。

“变更类型”用来区分每一科目代码是新建立的, 还是发生变化后修改的, 还是已被删除而不存在的等情况。

通过设计代码档案和变更模型, 可以记载每一代码的变化过程及变化关系, 从而可以实现口径调整。

3. 用户外码库的结构

所谓用户外码是指用户根据不同业务的需要或使用的方便性而自行定义的代码, 这些代码有别于应用系统内部使用的代码(内码), 但它们又密切相关。为了存储用户外码, 并能和系统内部使用的代码(内码)建立对应关系, 我们创建了用户外码库。其基本结构如下:

wm	CHAR / INTEGER	外码
mc	CHAR	名称
rm	CHAR / INTEGER	内码

四、代码管理功能

1. 功能结构

功能结构如图 1 所示:

2. 功能说明

(1) 内码管理。对代码字典和代码档案同时进行管理。

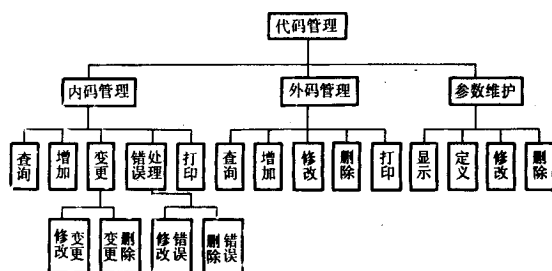


图 1

①查询:根据具体查询要求,选择某一查询方式,查询不同年份的代码及其有关信息。从代码字典中,可以获取当年代码信息;从代码档案中,可以查询历史代码信息以及代码的变更情况。查询方式有:按代码查询、按名称查询、模糊查询、跟踪查询和逐级查询等。

②增加:向代码字典和代码档案中同时增加新的代码,并用变更类型“建立”标识该代码为新建立的。

③变更:对发生变化的代码进行维护,并在代码档案中记载代码的变化情况及变化关系,它对代码字典和代码档案的维护不一样。

④变更修改:当代码发生变化时,对代码字典的操作是直接修改相应的代码;而对代码档案的操作是,在保留原代码记录不变的前提下,向代码档案中再增加变更后的代码记录,并用变更类型“修改”标识该代码记录被修改过,用变更模型反映变化前后的对应关系。

⑤变更删除:当某代码现已不存在时,对代码字典的操作是直接删除该代码;而对代码档案的操作是,在保留原代码记录不变的前提下,向代码档案中再增加一代码记录,并用变更类型“删除”标识该代码记录已被删除。

⑥错误处理:对输入错误的代码记录进行更正处理,包括修改错误和删除错误,它对代码字典和代码档案的维护完全一样,都是直接对错误的代码记录进行修改或删除。

⑦打印:根据某种要求和一定的条件,打印所需要的不同年份的代码信息。

(2) 外码管理。对用户外码库进行管理,建立和维护用户自己所定义的各类外码及其与系统内码的对应关

系。

(3) 参数维护。配置和维护通用代码管理构件所需具体实际应用的参数。由于代码管理构件具有通用性,其本身不包含任何实际参数,因此在应用之前,必须根据情况给它配置所需参数。

3. 采用的技术和措施

为了使该代码管理具有可维护性、可移植性和一定的通用性,我们按结构化设计方法划分功能模块,并对每一功能模块采用抽象和局部化技术;我们还采用构件技术,代码管理构件由若干互相独立的功能模块所组成,与之对应有一个参数维护构件,通过参数维护构件来定义和维护代码管理构件所需的参数模式。

同时,为了确保代码管理的实用性和有效性,还采用了以下措施:

(1)自动校验。包括阻止非法录入、保持一致性、检验唯一性。

(2)利用选择。包括减少输入、避免失误、保持一致性。

4. 运行环境

. UNIX 操作系统;

. INFORMIX 数据库管理系统,包括 4GL、SQL、ONLINE。

五、结束语

代码管理通过代码档案和变更模型记载的代码变化情况和变化关系,来建立口径调整机制,从而,使对比分析成为可能;又通过建立用户外码和系统内码的映射机制,来解决用户外码的灵活性和系统内码的相对稳定性的矛盾。该代码管理已用于《中等城市财政管理信息系统》中,它所建立的代码体系使得汇总统计、对比分析能适应业务系统的变化,并使信息共享、数据转换得以实现。由于它采用了构件技术,因此能很方便地移植到具有同样环境的其它财政应用系统中去。

参考文献:

[1] 许舒人、李奇云,《中等城市财政管理信息系统开发方法初探》,载《计算机系统应用》,1994年第4期。

[2] 王勇领,《计算机数据处理系统分析和设计》,清华大学。