

基于组织干部人事管理领域的构件库管理系统

唐文忠, 姚 铂

(北京航空航天大学 计算机学院, 北京 100083)

摘 要:近年来,我国各级政府部门对组织干部人事管理系统的需求与日俱增。为了应对大量的系统开发需求,希望通过组织干部人事管理领域的分析,设计生产出基于组织干部人事管理领域的软件构件,并进行构件的管理与组装,最终快速生成用户期望的组织干部人事管理系统。为了解决当前通用构件库无法适应组织干部人事管理领域需求的问题,设计建立了基于组织干部人事管理领域的构件库管理系统。文中分析了领域工程以及软件构件库的研究方法,根据组织干部人事管理领域的领域特性,建立起一套专有的构件库管理系统。

关键词:软件复用;软件构件;构件库管理系统

中图分类号:TP311

文献标识码:A

文章编号:1673-629X(2008)08-0237-05

Component Library Management System Based on Organization Cadre Management Domain Analysis

TANG Wen-zhong, YAO Bo

(School of Computer Science and Technology, Beijing University of Aeronautics
and Astronautics, Beijing 100083, China)

Abstract: In recent years, there is an ever-growing need for organization cadre management systems by various government agencies. For the purpose of meeting the great demands for system development, hope to design and build software components for the organization cadre management by analyzing this domain. Manage and assemble the components to build ultimately an organization cadre management system awaited by users in short term. At the same time, also design and build a component library management system for the organization cadre management in order to resolve the problem that the common component library can not meet the need of organization cadre management. Analyze at first the research methodology of domain engineering and software component library, and then do the domain analysis of organization cadre management. At last, a proprietary component library management system will be established.

Key words: software reuse; software component; component library management system

0 引言

近年来,随着计算机在人们日常生活和工作中承担的比重的逐步扩大,整个社会对于软件的需求也与日俱增。传统的软件开发模式因为存在着大量的重复劳动,其极低的开发效率已经不能适应当前软件开发的需要。因此,软件开发业界提出了软件复用的概念,软件复用是重复使用“为了复用目的而设计的软件”的过程^[1]。基于构件的软件开发方法是软件复用的有效手段。而构件库存储软件工程师的设计思想和开发成果,其本质就是软件工程师存储构件的工具^[2]。

当前的构件库管理系统大多都是通用构建库,面

向领域,尤其是面向我国组织干部人事管理领域的构件库管理系统还没有现成完好的实例。因此,基于我国组织干部人事管理领域,分析领域特征,在传统通用构件库的基础之上,建立起一套专有构建库管理系统,用以存储、管理软件构件,为最终实现面向组织干部人事管理领域的软件生产线提供最为基础的技术支撑。

1 领域分析与领域构件库

为了能快速高效地开发出用户满意的软件系统,软件开发业界提出了软件复用的概念。人们希望软件生产能像硬件生产的模块组建那样根据现有的系统或是软件产品迅速构架新的软件系统。为了避免对已存在的软件构件进行重复开发,软件工程师需要清楚地了解其他工程师的开发成果。因此,如果没有一个用于系统的记录软件和相关软件的数据库,在以后的软件项目开发中抽取可复用的软件开发成果就会变得异

收稿日期:2007-11-11

基金项目:国家“863”计划基金资助项目(2004AA113040)

作者简介:唐文忠(1968-),男,河北涿鹿人,研究员,博士研究生,研究领域为信息安全、电子政务、决策支持系统、网络系统集成。

常困难。软件复用技术必须以构件库为其最基本的技术支撑。

目前,较为流行的是青鸟构件和 REBOOT 的构件库管理系统。它们采用基于刻面的分类模式,并主要针对源代码级别的构件^[3]。它们不适合描述粒度较大、结构较为复杂的模式化、框架化的大粒度构件。针对该问题,软件业界提出了建立面向领域的软件构件库,相对于已有的构件库管理系统,在构件分类和构件检索方面进行改进,通过对特定领域的分析和建模,建立具有领域特征的软件构件库和构件库管理系统,最终形成了规范、可靠的应用系统。领域是指一组具有相似或相近软件需求的应用系统所覆盖的功能区域^[4]。领域工程是为一组相似或相近的应用工程建立基本能力和必备基础的过程,它覆盖了建立可复用软件构件的所有活动^[5]。领域工程师可以通过进行领域分析,并与“领域专家”进行交流,以获得关于领域的知识。领域工程的实施是为了能准确地描述出同一领域中的系统的需求和功能显著的共性,依据已获取的领域需求则可以建立起领域模型。领域模型的建立就是用来描述领域中系统之间的共同需求^[4]。领域构件是某一领域中的可复用软件构件,领域构件是领域工程的成果。领域工程支持的构件开发过程如图 1 所示。

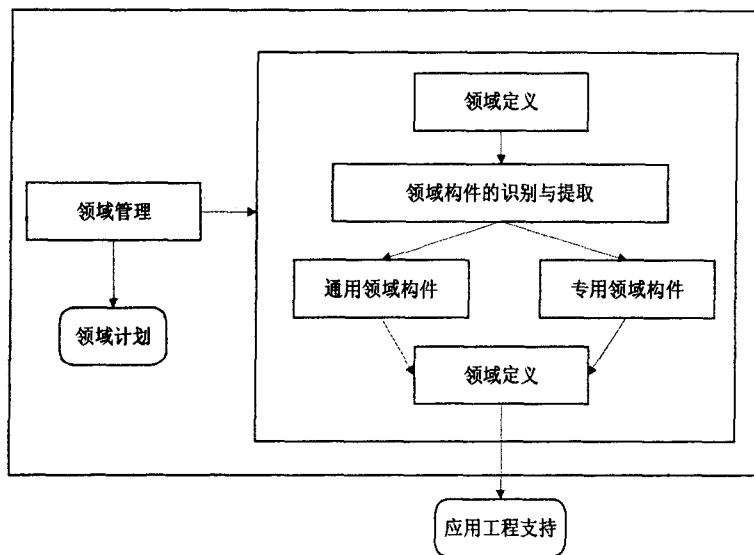


图 1 领域工程支持的构件开发过程

由于领域构件具有领域的特征,相对通用构件而言,领域构件的适用性往往更强。相比通用构件库管理系统而言,领域构件存储管理系统的研究起步较晚,技术还未成熟。通用构件库管理系统和领域构件存储管理系统之间存在着相似之处,但是技术上存在许多差异。主要区别如下:

(1)存储的构件范围不同,通用构件库存储的存储构件的数量庞大,而且范围不受任何限制,构件库的结

构复杂。领域构件存储管理系统仅限于特定领域。

(2)通用构件库中构件的通用性较高但是构件的适用性较差,领域构件存储管理系统中的构件大部分是特定领域下的构件,针对性和适用性都比较好。

(3)通用构件库管理系统为了存储来自不同领域的构件及其描述信息,术语空间的维护工作量较大,若采用刻面描述法进行构件表示,当新领域的构件加入构件库时需要扩展刻面分类模式。而领域构件库管理系统中的构件局限于某一特定领域,分类模式无需扩展。

2 基于组织干部人事管理领域的构件库管理系统的设计方案

2.1 构件分类

面向人事管理领域的软件构件是指对软件构件按照人事管理领域应用的业务逻辑进行细分,主要分为三类构件:

(1)基础构件:在进行构件的划分与建立时,为了提高构件的重用性,抽取不同领域应用中与业务无关的基础架构和通用功能部分,构成基础构件库。基础构件是与人事领域无关的,但它是组装人事管理系统所必需的构件。指特定于 D2G 平台开发出的人事管理系统的构成成分,如基本的页面资源构件,用以 Web 展示的各类多媒体资源构件等,它们可以存在于各种应用系统中。

(2)领域构件:领域构件是指封装了人事管理领域共性需求和变化性需求的构件,它能够表示领域应用系统的各个子业务逻辑功能。而领域应用系统则将各个子业务逻辑联系起来,领域应用系统就是这样由一系列基础构件和领域构件通过筛选,组装而成。基于组织干部人事管理的面向对象的思想,对于组织干部人事管理不同应用领域进行构件划分、设计并最后完成组织干部人事管理系统的组装。针对人事管理领域应用系统的特点,封装领域

共性需求和变化性需求,分析、设计、开发和领域相关的领域构件库,并通过软件构件的复用和组装形成领域应用系统。领域构件可以继续按照组织干部人事管理领域中的子业务领域进行划分,其中每个子领域都代表一个大类,涉及领域知识的复用构件放入相应的领域子库中。例如干部任免子领域构件、人员分类子领域等等。

(3)框架构件:框架构件完成构件组装,用户重新配置和完成构件间交互的构件。

2.2 系统的开发方案

D2G 平台的重点是研发面向政府组织干部人事管理领域的软件生产线。基于组织干部人事管理领域的软件产品线的开发过程借鉴了“产品线的双周期模型”。

根据系统的设计方案,将整个构件库管理系统结构设计为三个层次结构,分别为:客户端表示层,功能逻辑层,数据层。其中,在客户端的复用者可以斯用 IE 通过 Internet 来访问本系统的各个 Web 服务页面,而 Web 服务将用户的提交请求交给功能逻辑层,由逻辑层进行业务分析,并与数据层交互,得到构件的描述信息或者实体信息,最后将结果数据返回给客户端用户。人事领域构件存储管理系统的核心是维护一个存储构件及构件描述信息的数据库,各种领域构件存储管理系统采用关系数据库来存储构件和构件描述信息以及其它的系统信息。为了提高复用者对构件库中的可复用构件的理解,更准确地检索到所需要的构件,针对人事管理领域的特性,分析人事管理领域模型,定义人事管理领域范围,并且使用刻画描述模型来对人事管理领域的标准构件进行分类、表示,以使用户重用。美国军方发起的 STARS 项目主要考虑开放体系结构的构件库之间资源共享和无缝互操作问题,同时考虑了关键字 (Keyword) 和刻画 (Facet) 的概念^[6]。参照 NATO 标准、3C 模型以及北大青鸟构件模型,设计了人事管理领域的构件的刻画以及它的术语。软件产品线的开发过程如图 2 所示。

构件经过审核和认证,要登记构件的基本属性,并确定构件的分类刻画,分别填写每个刻画,并建立和确定该构件与构件库中已有构件的关系。最后将构件入库。组织干部人事管理领域构件库管理系统的功能模型如图 3 所示。

2.3 构件的分类编码

构件分类编码是将人事管理领域构件按照上述的分层综合刻画表示法进行分类,并通过构件的编码将一组分类术语与构件相关联,从而支持构件的分类和快速查找。构件的编码加上流水号就成为构件的唯一标识。人事管理领域的构件描述模型 (Personnel Component Description) 的刻画描述定义为以下几个方面:环境 (Environment)、类型 (Classification)、处理对象 (Worked Object)、服务 (Service)、表示

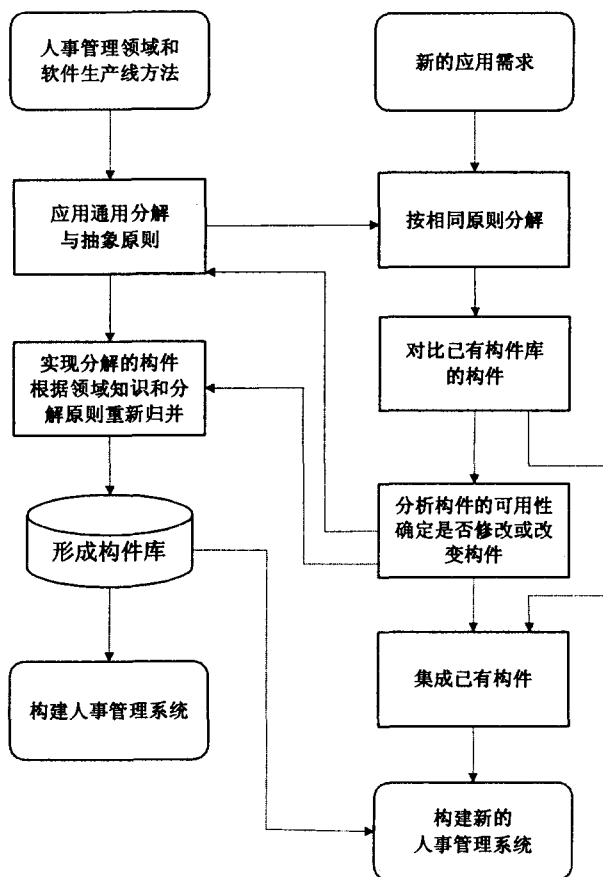


图 2 基于组织干部人事管理领域的软件产品线的开发过程

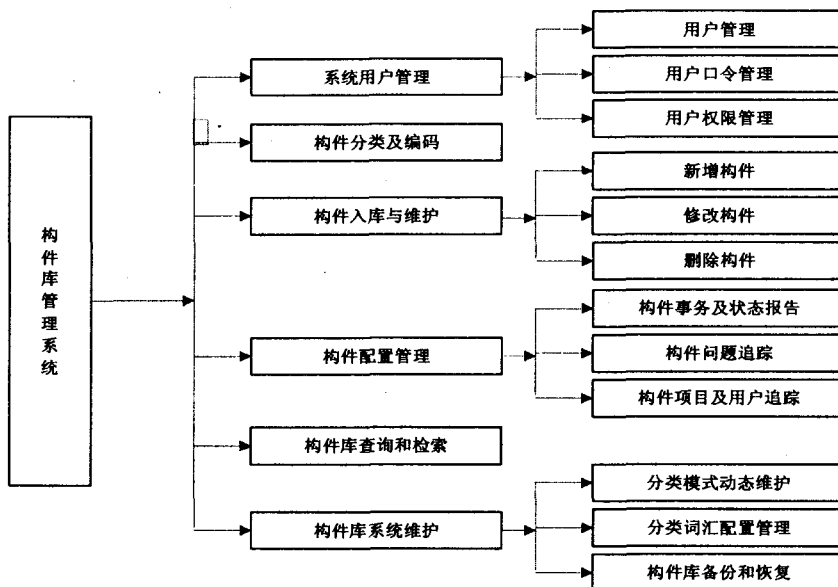


图 3 组织干部人事管理领域构件库管理系统的功能模块图

方法 (Representation)。

(1) 环境:是指适用该构件时必须提供的硬件或软件平台;

(2) 类型:指相对于软件开发过程阶段的抽象层

次;

(3)处理对象:指人事管理业务应用的有关人事管理的信息;

(4)服务:指构件提供的软件功能集。构件库中任何构件都必须提供一种或多种功能,针对人事管理领域,这个服务就是指该构件能够完成的人事管理的某项工作任务,或是某项工作任务的某个部分;

(5)表示方法:用来描述构件内容的语言形式或媒体,如源代码构件所用的编程语言等。

构件库中任何构件都以一定的形式存在,因而必然有其外在的表现形式。这种表现形式与构件的上下文环境(context)直接相关。

2.4 构件入库与维护

构件入库与维护完成构件的登记、审核、刻面分类、入库,同时对构件库中的构件进行维护管理,包括构件新增入库、构件修改、更新和删除等。人事管理领域工程中设计并实现的领域构件满足人事管理应用工程的实际需求,按照一定的分类方式,这些构件需要加入到人事管理领域构件库中,等待人事管理应用系统软件工程师来检索。构件在进入构件库之前,构件的制作者或提供者必须填写构件的登记表格,登记表格中列出了构件与复用相关的所有属性,构件的登记表格中有下列属性:

(1)名称:每个构件都必须有一个确定的名称,该名称必须完整地标识该构件的本质。

(2)作者:制作或提供该构件的单位或个人的名称,以及联络方式等相关信息。

(3)制作日期:构件制作的完成日期。

(4)构件入库日期:构件进入构件库的日期。

(5)版本号:该构件在一组构件演化系列中所对应的版本号。

(6)使用环境:使用(包括理解/组装/修改)该构件时必须提供的硬件和软件平台。如所需的特定的硬件环境、操作系统、数据库平台和网络环境等。

(7)应用领域:该构件原来或可能被使用到的应用领域(及其子领域)的名称。

(8)用途:该构件在被应用的领域中所发挥的作用。

(9)功能:该构件在原有或可能的软件系统所提供的软件功能集。

(10)表示方法:用来描述该构件内容的语言形式或媒体。如源代码构件所用的编程语言等。

(11)形态:该构件的组成成分及其相互关系。如类、类树、框架、模块等。

(12)层次:该构件相对于软件开发过程阶段的抽

象层次。如分析、设计、编码等。

(13)上下文环境:该构件在组装时系统所必须提供的程序级的上下文环境。

(14)尺寸(Size):该构件的粒度大小。

(15)业务信息:该构件实现的人事管理业务的业务信息。

(16)其他说明和资料:包括有关构件特性、安装、验证及操作的完整指令构件的摘要信息,构件的测试计划、目标、脚本及预期的结果等。

与以前的通用构件库管理系统中构件的刻面描述不同的是这里增加了一个处理业务对象的刻面,它的内容是人事管理业务的一个业务信息,是和人事管理领域密切相关的一个方面,当使用这样的方案来描述构件,构件的专业方面的属性得到了体现。用户在查找时目的就更加明确,构件的描述增加了专业的刻面,在查找构件时可以用处理对象刻面中的术语作为检索词,可以更准确地找到所需要的构件。

考虑到构件库管理及其它需要,实际的构件在入库前它的描述还需要添加如下元素:

1)构件校检状态:构件库管理系统允许用户上传构件,这些构件只有在通过系统校检后才会被设为已校验。使用用户可自己决定是否使用未校验的构件。

2)构件授权形式:注明构件使用的授权形式。

3)入库时间:由构件库管理系统自动设置。

4)被检索次数、下载次数:由系统自动管理。根据这两个状态参数,可以了解构件被检索和下载情况。

5)反馈信息:用于了解构件使用情况反馈,便于交流和沟通。

构件经过登记后,构件库管理员应对每个推荐入库的可复用构件进行定性和定量的评价。理想的构件应该能满足用户的特定需求,并具有广泛的适用性和高度的可见性,同时有较好的兼容性和较低的复杂性。

2.5 构件的配置管理

构件的配置管理中构件事务及状态包括构件的使用情况(谁提取构件、提取时间以及使用的项目)、构件用户信息(包括用户的名称、电话、通讯地址以及使用的项目)、构件分类模式、构件的历史(包括构件的修改日志、反馈日志、与其它构件的继承依赖关系)等;构件的问题报告与追踪不仅记录构件中存在的突出问题,还考虑到一个问题的解决对其它相关构件的影响(尤其是具有继承关系的构件);构件用户以及项目追踪用于发布构件的新版本和问题报告,获取使用反馈。从构件进入构件库,并可以从项目工程师获得该构件的时时刻刻起,它就被作为一个产品来管理,构件的配置管理很大程度上类似于一个产品的配置管理。构件的配置

管理应该记录下复用者对构件每次提取的情况,包括构件的使用、问题报告、失败的查询和其它构件配置管理所需的数据,并以此来改进构件库的查询机制。构件的配置管理还应追踪以各种方式从构件库中提取过构件的复用者,以获取构件的使用意见以及复用的经验反馈。在构建库中登记入库的构件,为了向开发小组提供了一个统一框架或是规范模型,在登记构件其各种属性时需要采取本体描述,使得不同背景、不同观点、不同工作任务的人员之间能顺畅准确地交流,使人事管理领域内不同子业务系统、不同模型之间能够实现互操作和集成。

2.6 构件库系统维护

构件库系统维护包括构件分类模式动态维护、分类词汇的配置和管理以及系统的备份和恢复。构件分类模式动态维护保证刻面分类术语将不断动态地增加和删除;构件分类词汇的配置和管理用以避免对库中构件未经授权的修改,记录构件的版本变化,用户针对构件的问题报告,为了增强功能而对构件进行的修改以及对于分类构件的术语的修改的严格控制。系统的备份和恢复用于对构件库进行备份保存。

2.7 构件检索

构件检索是构件库的重要的、关键的技术,构件检索要求提供多样化的查询手段和灵活方便的查询界面,以减轻构件复用者的查询负担,减少查询的时间和代价,提高检索效率,最大程度地支持软件复用。

基于刻面的分类检索机制是构件检索方法中最重要的一种检索方法。刻面分类模式由一组描述构件本质特征的刻面组成,每个刻面从不同的侧面对构件库中的构件进行说明。每个刻面由一组基本的术语所构成,构件可以用刻面中的一个和多个术语刻画(刻面术语是一个确定的集合),每个刻面则反映了对库中构件的一种划分^[7]。对于每个刻面细节,我们可以获取该刻面细节的属性,以及相对应的属性值。属性值即可以是单个的值,也可以是枚举类型或者是一个范围,还有一些值可以通过配置构件后动态计算出来的。因此用户可以直观地从不同的角度指明待检索的构,利于用户对构件的理解。对于描述构件的大量刻面,有许多是用户不感兴趣的,为了提高检索效率,经过反复筛选,把构件的功能、对象、使用环境和应用领域作为D2G平台下基于组织干部人事管理领域构件库中构件的刻面。

(1)构件的功能:是构件在原有或可能的软件系统中所提供的人事管理业务功能集,构件库中任何构件都必须提供一种或者多种功能。

(2)构件的对象:用于描述构件操作和最终产生的

对象。

(3)构件的使用环境:使用(包括理解/组装/修改)该构件时必须提供的硬件和软件平台。如所需的特定硬件环境、操作系统、数据库平台、网络环境和编译系统等。构件库中构件必须依赖于一定的使用环境才能得到复用,即使是源代码级的某些通用构件,也必须依赖于特定的编译系统。

(4)构件的应用领域:是该构件原来或可能被使用到的人事管理领域(及其子领域)的名称。这里的领域,指共享某种功能性(functionality)的系统或应用程序的集合。人事管理领域构件库中任何构件都有一个适用的业务子领域,一般的通用构件适用领域为General。

利用构件的刻面细节和细节属性值,以及构件所使用的应用环境,帮助开发人员和终端用户进行构件的检索,其步骤如下:

(1)根据目标系统的需要的构件类别进行初始检索,从构件库中检索满足该类别的构件,在此基础上从构件库中检索合适的构件,生成初始构件集,根据这些构件获得构件的刻面细节和刻面细节属性值,从而生成构件检索的初始查询参数。

(2)修改上一步得到的查询参数,根据对系统的分析选择所需要的刻面细节添加到用户的查询中,只有满足要求的构件被检索出。

(3)对于选择的刻面细节,用户可以有选择地为刻面属性赋予属性值,可以是单个值,也可以是值范围。

(4)执行检索表达式,符合检索要求的构件被检索。

用户可以利用构件的刻面信息来帮助用户理解构件的目的和对构件进行配置。如果检索的构件太多,可以重新修改检索表达式继续查询。

3 结束语

作为对人事管理领域构建库管理系统,在对于组织干部人事管理领域分析和构建库管理系统的设计上可能还有许多不够全面细致的地方,某些设计可能还不是最优方案。这些问题,都等待在今后的基于人事管理领域的软件生产线的设计实施中去逐步完善。

参考文献:

- [1] Tracz W. Confessions of a Used program Salesman - Institutionalizing Software Reuse[M]. New York, NY: Addison - Wesley Publishing Co., 1995.
- [2] 潘颖,赵俊峰,谢冰. 构件库技术的研究与发展[J]. 计

(下转第245页)

```
HRESULT ExceptionUndefinedHandler(void); //未定义指令
异常处理回调
```

2.4 时间管理模块

定时管理接口类有8个接口,分别是初始化定时器接口;创建定时器接口(返回3种情况:操作成功返回定时器编号,定时器号无效,定时时间无效);释放抽象定时器接口(返回2种情况:成功,定时器号无效);启动抽象定时器接口(返回2种情况:成功,定时器号无效);获取抽象定时器接口(返回2种情况:抽象定时器状态,定时器号无效);设置抽象定时器接口(返回3种情况:成功,定时器号无效,定时器模式错误);获取抽象定时器周期接口(返回周期数);设置抽象定时器周期接口(返回3种情况:成功,定时器号无效,定时时间无效)。

2.5 I/O 管理模块

I/O 管理接口类有3个接口:驱动程序地址表管理接口(有3个方法:初始化表,登录所有驱动程序,返回表头指针;根据主设备号定位驱动程序地址,返回驱动程序地址;更新表,当安装新的驱动时,将其登录在表中);设备名表管理接口(有3个方法:初始化设备名表,返回表头指针;搜索设备名表 返回对应搜索设备的表项;更新表);文件描述符表管理接口(有3种方法:初始化表,返回表头指针;搜索表;更新表)^[6]。

2.6 组件管理器模块

该硬件抽象层框架应用组件(COM)技术,把操作系统内核的各个功能抽象成功能模块,封装成组件。组件管理器用于管理组件,由组件登录表和组件管理表两部分组成。

凡在组件登录表中登录的组件,即在系统中注册,系统可以调用这些组件;并控制其生命周期,实现目标系统的动态性,当新的组件加入系统,只需在表中为其分配一个表项,并填写相关信息;当要剥离一个组件时,只需将对应表项删除即可。

组件登录表登录了系统中的所有组件,但并非每个组件都是活动的。当不需要某个模块时,就将其设

为“休眠”,节省电能。组件管理表中的每一项对应系统中一个处于“活动”状态的组件,并附有活动组件和对活动组件进行调用所需的信息。在系统中激活一个组件,就把组件登录表中的对应项写到组件管理表中,并补充完整信息。当要把某个组件设为“休眠”时,只要取消组件管理表中对应项,但保留组件登录表中的项。

3 结束语

针对嵌入式操作系统开发现在面临的诸多挑战,介绍了一种基于组件技术的通用硬件抽象层的框架,为嵌入式操作系统内核的设计开发屏蔽硬件平台的特性,提供统一的硬件相关的服务接口,使其设计开发可不依赖特定的硬件平台,并且具有更强的可移植性。

在设计通用硬件抽象层时使用组件的技术,将操作系统内核的功能抽象为各个功能模块,并封装为组件,这无疑又为快速、动态地搭建嵌入式操作系统提供了一条可行之路。

参考文献:

- [1] Labrosse J. 嵌入式系统构件[M]. 第2版. 袁勤勇, 黄绍金, 唐青译. 北京:机械工业出版社, 2001.
- [2] Massa A. Embedded Software Development with eCos[M]. [s.l.]: Prentice Hall, 2002.
- [3] Cai Xia, Lyu M R. Component - Based Embedded Software Engineering: Development Framework, Quality Assurance and A Generic Assessment Environment [J]. International Journal of Software Engineering and Knowledge Engineering, 2003, 12(2): 107 - 133.
- [4] Ford B. The Flux OSKit. A Substrate for Kernel and Language Research[C]//Proc. 16th ACM Symp. Operating Systems Principles. New York: ACM Press, 1997: 38 - 51.
- [5] Friedrich, Stankovic. A Survey of Configurable, Component - Based Operating - systems for Embedded Applications[J]. IEEE Micro, 2001, 21(3): 54 - 68.
- [6] Tanenbaum A S. 现代操作系统[M]. 陈向群译. 北京:机械工业出版社, 2000.

(上接第241页)

算机科学, 2003, 30(5): 90 - 93.

- [3] 杨美清, 梅宏, 李克勤, 等. 支持构件复用的青岛 III 型系统概述[J]. 计算机科学, 1999, 26(5): 50 - 55.
- [4] Krut R, Zalman N. Domain Analysis Workshop Report for the Automated Prompt Response System Domain[R]. [s.l.]: Carnegie Mellon University, 1996.
- [5] 张荣. 面向领域的 CASE 与 MIS 开发[J]. 计算机应用研

究, 1998, 4: 71 - 74.

- [6] STARS Technical Committee. Asset library open architecture framework version 1.2[M]//Software Technology for Adaptable, Reliable Systems (STARS). [s.l.]: [s.n.], 1992.
- [7] 李景峰, 李琰, 陈平. 一种特定领域的软件开发模型[J]. 西安电子科技大学学报: 自然科学版, 2000, 27(5): 602 - 606.