

基于流程的构件库管理系统及其实现

Process - based Component Library Management System

杜建伟 顾斌 (北京控制工程研究所 100080)

摘要:基于构件的软件复用被认为是解决软件危机的有效途径。软件构件库作为存储和管理软件构件的基础设施,成为联通构件开发者和使用者的中间桥梁。将介绍现有的构件库系统,分析此类通用构件库系统的弊端,阐述基于特定领域的构件库的优势所在,提出了基于流程的领域构件库系统及其在航天器控制领域中的实现。

关键词:软件构件库 软件复用 领域

1 构件库简介

基于构件的软件复用被认为是解决软件危机的有效途径。随着软件复用的不断深入,软件企业内部会积累大量自主开发的构件,同时企业之间也会通过购买等方式复用构件产品,从而在一定程度上节省开发时间,提高效率。构件库系统作为管理软件构件的基础设施,为软件企业共享,发现和获取可复用资产提供了方便,成为联结构件开发者和使用者的中间桥梁。

构件库是为构件的存储而开发,为了更好的让软件开发者发现和使用构件,构件库还必须提供相应的构件分类组织,检索下载以及反馈等功能。同时由于构件库系统的开放性,相应的权限管理和维护功能也是必不可少的。

国内外已经开发出了多个构件库系统,它们都基于某些共同的观点,大都给出了一定的构件分类以及相关的检索机制以方便使用,同时由于实现技术以及所处理的构件不同,相互之间也具有一定的差异。

RIG 组织 (Reuse Library Interoperability Group) 分析了软件构件库之间的可互操作性问题,并开发了一个数据模型 UDM (Uniform Date Model),其中定义了支持可互操作性的库之间交换软件构件所需的信息。

90 年代初欧盟建设了 REBOOT 构件库系统。REBOOT (Reuse Based on Object - Oriented Technology) 是欧盟信息技术计划 ESPRIT 中的一个项目,目前是研究、开发及推广复用驱动和面向对象的软件技术。它是国际上比较著名的构件库系统,包括一个存储可复用构件的构件库和一组产生、认证、插入、提取、评价和适配可复用构件的工具。它主要采用了剖面模型来分

类描述构件。该模型是基于已有软件构件的一种剖面分类和检索模型,从各个角度,即剖面 (Facet) 刻画软件构件属性。

90 年代中后期国际上出现了若干商业运作的构件库,其中以 ComponentSource、Flashline 等较为著名。前者成立于 1995 年,它拥有目前世界上最大的构件库,客户遍及 100 个国家,提供构件目录 114 类,近 6500 个构件产品,它还提供构件复用自动化技术 Component Find,通过 SOAP 和 XML,帮助客户检索构件,获得相关信息。

在国内,2005 年上海构件库构件开源平台正式开通并试运行。北大软件工程研究所开发了公共软件构件库管理系统。这两个构件库是比较成熟的系统,二者都是为软件企业提供软件复用的机制和软件资源管理的基础设施,使企业基于构件复用快速、高质量地开发新的应用系统,促进软件工业化生产技术的实现和推广。

由于这些通用的构件库系统要提供对多个领域软件开发的支持,使得库中的构件种类繁多,这对于构件库所采取的分类和组织方法提出了更高的要求,同时由于其开放性,构件库对于外部用户提交的构件缺乏完善的审核,对构件的提交过程,构件的属性信息,构件的下载和使用等环节缺乏管理,这在某种程度上造成了构件库的混乱。

基于此,本文针对航天器控制软件开发领域,提出了基于流程的构件库管理系统,针对现有构件库存在的问题提出了相应的改进方法,如下:

① 改变原有构件库简单的管理模式,提出了基于

流程的构件库管理过程:

② 针对航天器控制软件的特殊要求,增加构件库对构件使用的控制功能,对构件的使用进行跟踪;

③ 针对领域需求的特殊性和稳定性,采用方便灵活的枚举分类方法,配合多种检索方式,使得开发者更容易检索到恰当的构件。

2 特定领域构件库的优势

特定领域构件库指对于某一特定类型的应用,相对于一些通用的构件库而言,所涉及的构件数量较少便于管理和维护;构件库的使用者都是本领域的软件开发人员,对领域知识相对比较熟悉,在构件的应用中不会产生太多的问题,从而提高复用的效率。

特定领域中的软件复用更为容易实现。这是因为可复用信息的领域特定性和稳定性。可复用性不是信息的一种特定属性,它往往依赖于特定的问题以及对特定问题的特殊解决方法,而且对于某一特定领域,对某一问题的解决方法在一定时期内是不会发生太大的变化,这就是领域的稳定性。对航天器控制领域而言,控制软件所包含的功能,以及各个功能的具体实现方法表现出了极大的平台相似性,同一型号之间更是具有相同性。通过领域分析得到一系列构件,其应用范围更加集中,同时不会掺杂太多与本领域无关的构件,这使得构件库中存储的构件都能直接应用于软件的开发,可复用性更高。

3 基于流程的构件库管理过程

现有构件库系统多是基于角色开发的 B/S 系统,用户通过浏览器访问构件库,上传,检索或下载构件。构件管理者对系统进行维护,但是管理者并不对用户提交的构件进行细致的管理。首先,多数构件库没有对用户提交的构件进行审核的机制。这导致了构件的正确性难以得到保证。其次,用户提交中往往忽略构件的基本信息和组成,对构件的功能,接口等没有细致的描述,缺乏必需的文档等资料,这样的构件也给复用者带来使用上的困难。再次,构件库缺乏对构件使用的跟踪机制,很难确定某一构件的流向和应用情况。

根据航天器控制软件构件开发和应用的特殊性,提出基于流程的构件库管理过程,开发完整恰当的流程,规定各种角色的职能,完善构件库的管理功能,来弥补通用构件库系统的不足。这里的流程是构件库管

理过程中可运转的部分,包括任务的执行者,执行顺序等,通过一系列规则,使得构件信息,实体,文档等在不同执行者之间传递。流程管理是建立在对业务过程的分析基础之上的,能够对已定义的业务过程的具体实施提供自动化的支持。

通过对本领域构件库系统需求的分析,整个系统以构件为核心,所有进入构件库的构件都要经过从提交,审核到入库的一个流程。构件在整个流程中表现出多种状态。每个状态对应不同的角色,包括:构件提交人员,构件审核人员,构件库管理员等。每种角色都负责对相应状态的构件进行处理,以此保证入库的构件都正确可用。构件库的状态流程图如图 1 所示。

可见,系统包含了构件从开发者提交,到管理员入库,再到使用者下载使用的全部过程。同时在相应的状态转换中包含对应的数据和信息表单,用以记录和保存重要的信息,如在构件提交过程中,提交人员需要填写构件相关的属性信息,这部分信息中包含构件的功能描述,接口描述等,是构件进行有效复用的重要信息。

4 构件的跟踪和控制

由于航天器控制软件特殊的安全性要求,每个软件开发者对构件的检索下载和使用都要进行相关的记录,以保持对构件使用的跟踪。如下表所示:

表 1 构件下载记录表单

记录项	注释
构件名称	下载构件名称
构件版本	下载的构件版本信息
下载人员	构件下载的人员
下载日期	构件下载的日期
应用项目	构件应用到的项目
实现功能	构件参与实现的功能

同时考虑到构件在实际应用中可能出现不适应性的问题,增加了构件使用的反馈机制,使用者可以对构件的应用情况提交相应的反馈意见,对构件的正确性,可靠性,实时性等进行评价,从而给构件的升级和修改提供依据。

5 构件的分类与检索

构件的分类技术解决用何种分类方法存放构件的

问题。依据领域的特征和构件的特点选择合适的构件分类方法可以降低构件的复用成本,使整个构件库中的构件易检索、易修改、易使用。目前多数构件库都采用刻面分类模式和关键词分类模式等分类方法。

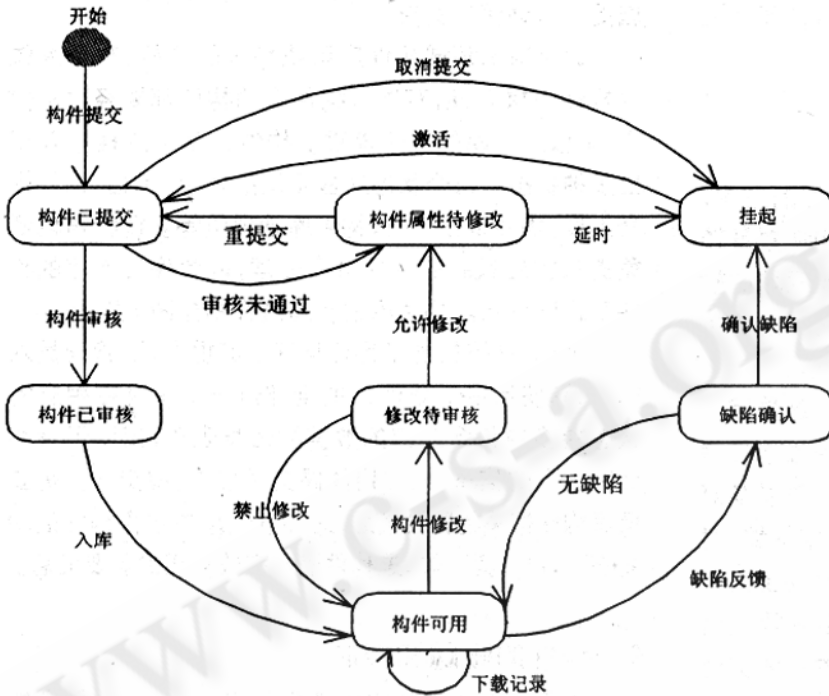


图 1 构件库流程图

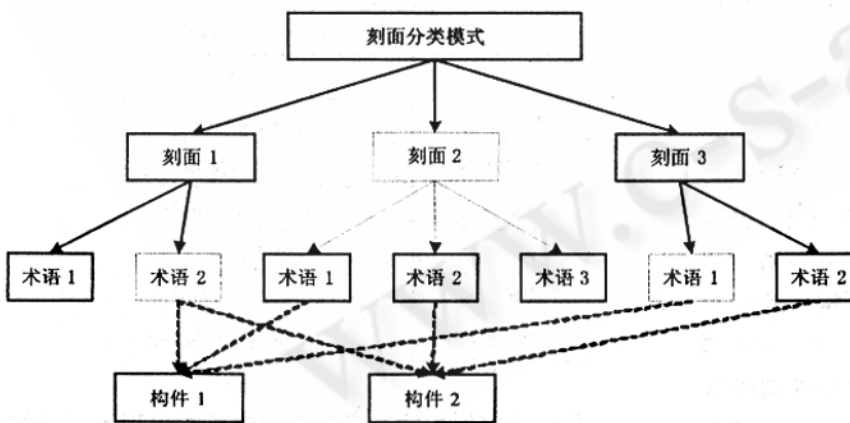


图 2 构件的刻面分类模式

刻面分类模式对构件的分类使用一组{刻面, 刻面术语}对,刻面分类模式由一组描述构件本质特征的刻面所组成,每个刻面从不同的侧面对构件库中的构件进行分类。例如,这组刻面可以是构件的应用领域,

功能,操作对象,使用环境等。每个刻面是由一组基本的术语所构成的,称为术语空间。刻面的术语空间是一个有限的集合。一个构件可以被每个刻面中的一个术语所刻画。各个刻面的术语组合起来形成构件的描述语,用来描述构件库中的特定构件。

针对控制软件特点,选取最能够体现构件特征的刻面作为构件库的一种分类标准。这几个刻面分别为:开发语言,开发工具,卫星平台和硬件环境。

航天器控制软件构件库作为一个特定领域的内部构件库,所涉及的构件数量相对较少,加上开发人员对领域相对比较熟悉,所以构件库可以通过枚举的方式,将构件库中构件的整体存储结构展示给开发者,使得构件检索查询的效率更高。同时考虑系统的扩展,类别的层次可以进行定义和维护。枚举分类如图 3 所示。

枚举分类是和领域分析密不可分的。通过领域分析,确定领域的边界以及各个子领域之间的关系,将整个构件库划分为若干不相交的领域和其子领域,确定构件所属的子领域,采用树形结构对构件逐步细化,对控制软件进行领域划分的结果如图 4 所示。

6 总结

本文解决了在航天器控制软件中实施构件复用所面临的构件管理问题,开发了基于航天控制领域的构件库管理系统,通过严格的流程保证构件入库的合理性和可用性,同时也解决了对构件应用的问题,加强了构件应用的可控性。随着软件复用的不断深入以及航天器控制软件的发展,构件库的功能将会不断

完善。

参考文献

- 1 NATO Standard for Management of A Reusable Soft

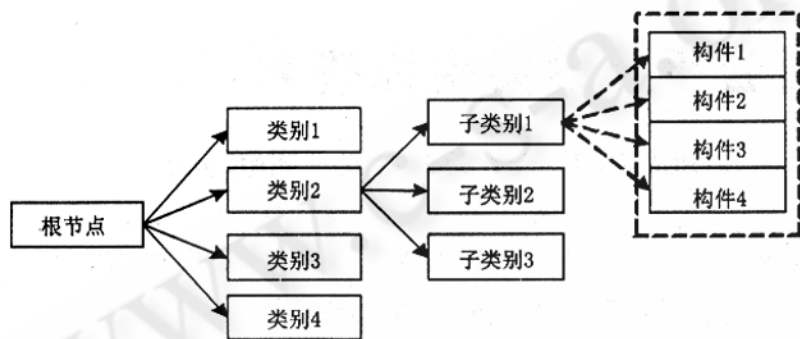


图 3 枚举分类方法

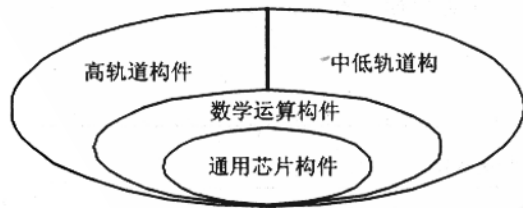


图 4 构件库中的构件划分

ware Component Library, issued and maintained by NATO Communication and Information System Agency.

2 王渊峰、张涌、任洪敏等,基于剖面描述的构件检索,软件学报,2002.

3 Nada, N. and Rine, D. C., A Validated Software Reuse Reference Model Supporting Component - Based Management, 20th ICSE International Workshop on Component - Based Software Engineering (Kyoto, Japan), 1998.

4 Milli, R., Mili, A. and Mittermeir, R., Finding and Retrieving Software Components: A Refinement Based System, IEEE Transactions on Software Engineering, Vol. 23, No. 7, pp. 445 - 460, July 1997.

5 Frakes, W. B. and Pole, T. P., Proteus: A Reuse Library System that Supports Multiple Representation Methods, ACM SIGIR Forum, Vol. 24, 1990.