

基于库存管理领域的软件复用技术

刘勇¹, 刘虹²

(1. 青岛科技大学信息学院, 青岛 266061; 2. 青岛大学国际学院, 青岛 266071)

摘要: 基于重用的软件复用技术可以减少软件开发时间和费用, 提高软件的灵活性、可维护性、可靠性, 缩短应用软件的开发周期。该文研究了软件复用技术, 提出了库存管理领域软件复用思想, 并以青岛高校软控公司“备品备件库存管理系统”为原型, 阐述了该软件模型的构建过程, 并证明了该技术的有效性。

关键词: 软件复用; 备品备件; 库存管理; 构件

Software Reuse Technology Based on Stock Management Domain

LIU Yong¹, LIU Hong²

(1. College of Information, Qingdao University of Science and Technology, Qingdao 266061;
2. International College, Qingdao University, Qingdao 266071)

【Abstract】 Reuse-based software technology reduces time and expenses for software development, enhances the flexibility, maintainability and reliability, and shortens the development cycle of software. This paper studies software reuse technology, proposes a software reuse idea in stock managing domain. Based on the spare parts stock management system produced by Qingdao MESNAC Co., Ltd., building process of the software is described. Experimental results show the technology is effective.

【Key words】 software reuse; spare parts; stock management; component

自从 McIlroy 在 1968 年的 NATO 软件工程会议上正式提出软件复用的概念以来, 软件复用已有了近 40 年的发展历程, 复用的对象也从早期的“代码复用”扩展到对软件开发过程中一切有价值的信息的复用, 包括需求、需求规约、设计、源代码、测试计划和测试案例等。软件复用可以有效地提高软件的质量和生产率。

软件界人士从不同的角度提出了一些软件重用的解决方法。特定领域思想的提出, 使大规模软件复用成为可能。在软件开发过程中, 经过该工程特定领域的分析, 将工程的特定需求变成该领域的典型需求、特殊需求, 特定领域的软件构架可以配置相应的可复用构件, 完成软件的开发过程^[1]。

库存管理领域软件复用模型的过程包括: 领域分析阶段, 领域体系结构 DSSA 的建立, 可复用构件的提取与实现等。

1 软件复用技术

(1) 软件复用的定义。软件复用是指重复使用“为了复用目的而设计的软件”的过程。软件复用是在软件开发过程中避免重复劳动的解决方案, 使应用程序的开发不再“从零开始”, 能够在现有的工作基础上, 充分利用过去应用系统开发中积累的知识和经验, 将开发的重点集中在应用的特有部分^[2]。

(2) 软件复用的范畴。依据复用的对象, 可以将软件复用分为: 1) 产品复用。复用已有的软件构件, 通过构件集成组装得到新系统。2) 过程复用。复用已有的软件开发过程, 使用可复用的应用生成器来自动或半自动地生成所需系统。

过程复用依赖于软件自动化技术的发展, 目前只适用于一些特殊的应用领域, 而产品复用则是目前主流的途径。

(3) 软件复用的方式。依据对可复用信息进行复用的方式, 可以将软件复用分为: 1) 黑盒复用。对已有构件不需做

任何修改, 直接复用。这是理想的方式。2) 白盒复用。已有构件并不能完全符合用户需求, 需要根据用户需求进行适应性修改后才可使用。而在大多数应用的组装过程中, 构件的适应性修改是必需的^[3]。

(4) 软件复用过程, 可以分为: 1) 领域分析。通过定义应用领域、分析应用领域来确定是否值得为该领域开发重用基础设施。2) 领域工程阶段。在这个阶段主要根据领域分析阶段得到的领域共性获得通用体系结构; 并规定资产如何匹配体系结构, 如何绑定可变点。3) 资产获取。包括针对重用的开发, 还可能包括某些可重用资产的外部采办。4) 资产分类。实际上是数据库管理任务, 包括分类和存储可重用资产。5) 资产维护。承担维护作用, 与配置管理和版本控制任务结合在一起^[4]。

2 领域分析阶段

在面向领域软件的开发过程中, 领域分析是第 1 阶段的工作。主要对属于这个领域的应用系统进行定性、广泛、深入地分析, 然后对领域内应用系统的共性和变性进行分析, 从而形成领域边界模型, 形成领域模型, 并撰写新应用的需求规约说明书。

领域工作流程(图 1)如下:

(1) 由车间提出年度采购计划和月度采购计划, 提交给设备管理部门。

(2) 由该部门进行汇总平衡, 然后生成采购计划, 提交给采购业务员。

作者简介: 刘勇(1971-), 女, 副教授, 主研方向: 计算机工程, 软件复用技术; 刘虹, 讲师

收稿日期: 2007-01-02 **E-mail:** liuyong0202@sohu.com

(3)进行采购,可以分为:1)国内采购:流程为向供应商发送询价单,根据供应商发回的价格单进行比质比价,确定供应商;2)国外采购:流程为寻找供应商,签定单,进行采购。

- (4)供应商发货。
- (5)质检部门进行质检。
- (6)仓库管理部门收货、入库。;
- (7)财务部门付款。
- (8)车间人员到仓库领取备品、备件。

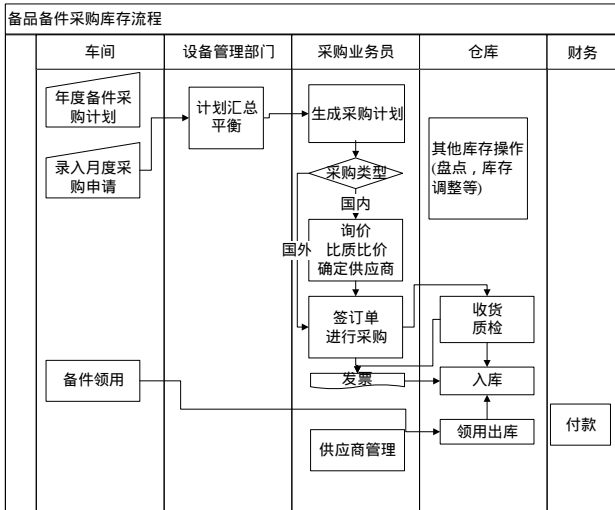


图1 领域工作流程

通过对流程的分析,可以形成领域的数据模型(图2)。

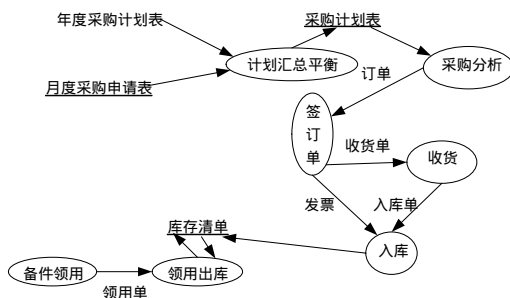


图2 领域数据模型

3 软件体系结构的搭建

软件体系结构是阐述整体软件系统高层计算部件的组织、交互、设计和调整的指导原则。在本质上,它是一个高层次上的抽象,提供了一种自顶向下实现基于构件的软件复用的途径。将体系结构作为系统构造和演化的基础,可以实现大规模、系统化的软件复用。

梅宏教授提出基于软件体系结构的构件组装(architecture-based component composition)的ABC方法。该方法将软件体系结构和基于构件的软件开发相结合,并把软件体系结构作为系统开发的蓝图,在软件开发的各个生命周期指导系统的分析、设计、实现和维护,最终实现了基于构件的软件复用。在ABC方法中,软件体系结构是贯穿整个软件生存周期的重要的软件产品,软件系统的构造将围绕软件体系结构来进行^[5]。基于ABC方法的指导思想,比较了目前各种主流的架构模式,本文决定采用分层模式作为库存管理领域系统的架构模式。

如图3所示,基于ABC的设计原则可以将这个系统分为

4个模块,按照层次架构对其进行了组织。下面就每一层的功能进行说明:

- (1)基础数据层:该层包括了整个系统的公共基础部分,即基础数据和系统服务部分。这一层是其他各层的基础。
- (2)数据访问层:该层为业务逻辑层提供了统一的数据访问接口。业务逻辑层通过该层读/写数据。
- (3)业务逻辑层:该层是整个系统的核心。整个系统所有的业务逻辑和业务规则的判断与实施都是在这一层完成的。在这一层重点为:业务逻辑和业务规则的实现。
- (4)表示层:该层是用户与系统交互的界面,通过该层调用业务逻辑层的相关功能为用户提供服务。
- (5)环境层:主要是系统所要求的基本环境,例如COM+, Internet以及通信协议等。
- (6)集成机制:是系统复用的关键。控制器主要是用来订制符合客户要求的各种用户界面,而过程引擎则是客户的业务逻辑,业务流程的实现。
- (7)业务过程层:该层可以定制出的一些有关业务的模块。
- (8)应用层:该层对应用模板的组合形成的具体的实例。



图3 体系结构

4 可复用构件的提取与实现

通过以上分析,可以提取出构件,见表1。

表1 构件列表

序号	构件名	构件编号	类型(领域构件/基础构件)
1	入库	GJ001	领域构件
2	出库	GJ002	领域构件
3	计价	GJ003	基础构件
4	货位管理	GJ004	基础构件
5	仓库管理	GJ005	基础构件
6	备件基础信息	GJ006	基础构件
7	库存调拨	GJ007	领域构件
8	盘点	GJ008	领域构件
9	库存调整	GJ009	领域构件
10	库存月末结账	GJ010	领域构件

下面以入库构件为例,说明构件的设计与实现。

4.1 构件描述

(1)功能描述。有收货、记账和查询入库信息功能。根据订单送的备件进行实物入库处理;收到采购发票后,财务对采购收货单进行财务确认。

(2)应用场景。

1)入库

操作员选择仓库,填写入库日期和入库人等信息
For 每一条记录

```

IF 指定货位
THEN 输入货位信息
END IF
    修改入库数量和价格等信息
End for
保存

```

2) 记账

```

找到入库单据
填写会计期间, 记账日期, 发票号等信息
For 每条明细
填写价格信息
End for
修改单据状态为账实同立
保存
(3)使用约束

```

依赖于库存账目表、入库单表、入库单明细表、供应商信息表、备件基础信息表、仓库管理表、货位信息表、备件货位关系表等几个基础构件。

4.2 接口设计

由于提供的记账服务, 只是修改入库单和明细的部分信息, 并将入库单的状态变更为“账实同立”, 因此可以归到收货接口中, 基于本构件, 本文设计了 2 个接口: 收货(IReceiveSP), 入库信息查询(IInStockQuery)。

构件对外需求的功能函数列表如下: (1)查询仓库信息 from 仓库管理构件。(2)查询指定备件基础信息 from 备件基础信息构件。(3)查询特定供应商 from 供应商基础信息构件。(4)生成入库单 from 入库单构件。(5)账务入库 from 入库单构件。(6)生成入库单明细项 from 入库单构件。(7)更新库

存量 from 库存账目表。

构件对外提供的功能函数列表如下: (1)接口收货: 1)新增入库; 2)更新; 3)删除; 4)记账入库。(2)接口入库信息查询: 1)通过入库单号查询; 2)通过供应商编码查询; 3)其他构件。

接口设计见图 4。

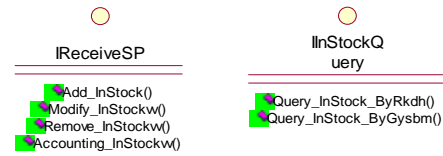


图 4 接口设计

5 结束语

在软件开发中, 本文引入了软件复用技术, 该技术具有良好的应用前景, 今后的工作将进一步完善该技术的开发思想。

参考文献

- 张秋雨, 张冬冬. 特定领域软件复用技术的研究与应用[J]. 计算机工程与应用, 2004, 40(14): 213-215.
- 李基鸿. 基于可复用构件的软件复用技术[J]. 山西煤炭管理干部学院学报, 2004, 17(3): 109-110.
- 杨芙清. 软件复用及相关技术[J]. 计算机科学, 1999, 26(5)
- Hafedh M. Reused-based Software Engineering——Techniques, Organization, and Controls[M]. 北京: 电子工业出版社, 2004.
- 梅 宏, 陈 峰, 冯耀东. ABC: 基于软件体系结构、面向构件的软件开发方法[J]. 软件学报, 2003, 14(4): 721-723.

(上接第 269 页)

理就是把电压值转换成二进制数据, 并统计 1 的总数和标识 0 突变成 1 和 1 突变成 0 的位置, 计算出这 2 个位置的中间位置, 然后计算出与 CCD 像元的中间位置的偏离值, 并判断是否超出允许的偏离误差, 以作为纠偏的依据。图 6 为算法实现的示意图。

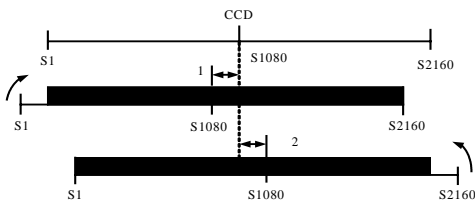


图 6 算法实现示意图

4 试验结果

试验是在室内环境中进行的, 试验场地面积为 12m × 12m, 铺设深绿色地板, 地板上面贴有白色导引线, 速度在 1m/s 以下, 由于机械结构设计时充分考虑了整个机器人系统的重心位置, 因此不会出现打滑现象。经过多次调试和运行, 机器人巡线稳定, 巡线过程中没有出现纠偏不足、纠偏过度及纠偏过于频繁的现象。通过设计程序和控制算法, 机器人根据导引线指示的路径很容易地实现了自主移动, 完成前进、转弯、后退等动作, 并最终巡线到达了指定的位置。由此可以看出, 本文的自主巡线机器人系统达到了很好的效果。

5 结束语

上述实践结果表明, 本方法能够很好地实现机器人巡线

行走, 由于采用 CCD 采集图像, 消除了环境光、场地等的影响, 极大地提高了引导线和场地颜色的分辨率; 在数据处理上采用 ARM7 核的 S3C44B0X 处理器, 数据处理快、实时性强、稳定可靠、效率高。与传统的巡线技术相比, 本系统实时性、稳定性和控制精度更高。目前, 在自主机器人的研究中还常使用光源、声音、无线电等多种导航与定位方式, 本文所设计的自主巡线系统是其中一种有效的方法, 对智能机器人的研究有一定的参考意义。该技术也可以应用于机器人比赛、自动化无人工厂、仓库、服务机器人等领域。

参考文献

- 吴明晖. 基于 ARM 的嵌入式系统开发与应用[M]. 北京: 人民邮电出版社, 2004: 1-2.
- 王庆有. 图像传感器应用技术[M]. 北京: 电子工业出版社, 2003.
- 白井良明. 机器人工程[M]. 北京: 科学工业出版社, 2001: 95-111.
- 龚 箭, 李晋惠. 基于 DSP 的条码标尺识别系统研究[J]. 西安工业学院学报, 2006, 26(1): 41-44.
- 赵凯生, 刘 爽. 基于 ARM 的线阵 CCD 测量系统应用分析[J]. 光电技术应用, 2006, 21(1): 31-34.
- 许松清, 吴海彬. 基于 DSP 的半自主移动机器人远程控制系统[J]. 单片机与嵌入式系统应用, 2005, 56(8): 58-60.
- 黄燕平. μC/OS ARM 移植要点详解[M]. 北京: 北京航空航天大学出版社, 2005.
- Labrosse J J. 嵌入式实时操作系统 μC/OS- [M]. 2 版. 邵贝贝, 译. 北京: 北京航空航天大学出版社, 2005.
- 章 晋. 图像工程(中册)——图像分析[M]. 北京: 清华大学出版社, 2005.

