

中小企业管理软件架构设计

陈 述, 赵广利

(大连海事大学 计算机科学与技术学院, 辽宁 大连 116026)

摘 要:文中对中小企业进行了深入的分析;然后通过面向服务的设计思想,提出了一个新型的中小企业管理软件架构。目标是提高中小企业管理软件的规范性、通用性、可扩展性以及开发效率;并且也为希望能为以后的中小企业管理软件架构提供新的设计思路。

关键词:中小企业;软件架构;面向服务

中图分类号: TP311

文献标识码: A

文章编号: 1673-629X(2006)05-0059-02

Design of Software Architecture for Medium - Small Enterprise

CHEN Shu, ZHAO Guang-li

(Computer Science and Technology College, Dalian Maritime University, Dalian 116026, China)

Abstract: Bring forward a new services-oriented software architecture aiming at medium-small enterprise by analysing of medium-small enterprise and based on the services-oriented design pattern. Its object is to improve the criterion, generality, expansibility and development efficiency of medium-small enterprise management software, and it also provides a new thought for medium-small enterprise management software architecture.

Key words: medium-small enterprise; software architecture; services-oriented

0 引 言

中小企业在中国的国民经济中占有重要地位。据统计,中国中小企业约1000万家,占全国企业总数的99%。伴随着中国经济活动的市场化、社会化、城市化的进程,快速发展的中小企业,已成为推动中国经济发展的生力军。然而,长期以来,由于发展环境和体制的制约,中小企业普遍存在资金不足、信息不畅、人才缺乏、管理水平落后等问题。为了解决这些问题,加强信息化建设是最重要的举措之一。信息化建设可以降低企业生产和经营成本,提高管理水平和客户满意度,进而提高企业的核心竞争力,因此,得到了越来越多中小企业的重视。但多年企业信息化建设的实践表明:管理软件质量不高,设计思想和技术落后,实施、维护困难对企业的发展也产生了很大的负面作用。

经过几十年,特别是近年来的研究,软件界已经把解决软件设计问题的出路寄托在从宏观的层面把握和控制软件的复杂性的方法上了^[1],这就是反映软件总体结构和关系的“软件架构(software Architecture)”。软件架构的合理设计有助于全面提高软件的质量^[2]。因此,研究中小企业管理软件的架构设计和实现方法,对提高中国管理软件的质量和开发水平,进而提高中小企业信息化水平和管

理水平,具有重要的理论价值和现实意义。

1 中小商业企业的特点分析

通过对相关资料的分析与研究,对中国中小企业的特点归纳如下:

(1)机制灵活。中国中小企业多是非国有企业,其生产、经营、分配机制灵活,决策权集中,利益分配明确,这使得中小企业相对大型企业来说,更具有活力。

(2)对市场信息反应灵敏,业务流程多变。中小企业因自身生存、发展的需要,对市场的信息反应比较灵敏,为了更好地适应市场需求,其业务流程会经常调整 and 变化。

(3)抗风险能力差。中小企业在资金、技术、信息、人才方面处于劣势。同时,由于缺乏政府的政策和融资方面的扶持,其抵御风险的能力明显不如大型企业。

(4)管理不规范。中国中小企业多是通过私营、合资、股份、民营等形式发展起来的,其企业管理受个人和家庭的影响比较大,管理很不规范。所以,引入先进的管理思想对中小企业的发展壮大至关重要。

按照行业特点,又可以把中小企业分成3种类型:商业企业;生产企业;服务性企业。商业企业是生产企业和用户的中介,它通过商品流通这一环节赚取价差,获得利润。从管理的角度来讲,生产企业以生产的计划与控制为基点,以产品供需链管理、生产管理、成本控制为核心;商业企业以分销计划与控制为起点,以产品的进销调存和财

收稿日期:2005-09-02

作者简介:陈 述(1981-),男,吉林蛟河人,硕士研究生,研究方向为计算机网络技术;赵广利,副教授,硕士,研究方向为计算机网络技术与数据库安全。

务管理为核心;服务业以服务业务过程为起点,以客户关系、人员管理为核心。中小商业企业的管理软件应该以物流和资金流管理为核心包含先进管理思想,实施简单,价格合理,支持流程定制及电子商务。

2 架构总体设计

为了保证软件产品的质量,提高可维护性、可重用性,产品的架构设计是至关重要的^[3]。在架构的参考模型选择上,笔者认为支持 Internet 和以流程为中心的开发模式是未来管理软件的发展趋势,所以最终决定采用面向服务的架构作为架构设计的参考模型。

在商业逻辑方面,考虑到最终的产品主要针对中小商业企业,所以,架构的商业逻辑将从商业企业管理软件的通用模块中抽取核心流程,同时提供扩展机制,以方便具体开发中对业务流程进行定制。中小企业管理软件的通用模块如图 1 所示。

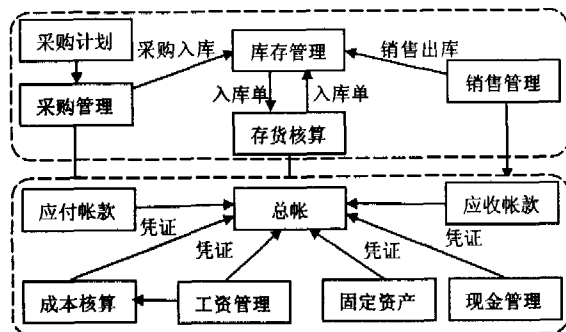


图 1 中小企业管理软件的通用模块

根据上述的架构设计,从中可提取如下核心流程:

·总帐。总帐包括科目设置、凭证处理、帐簿管理、月末处理等功能。

·应收/应付。应收/应付实现企业对应收/应付帐款的核算与管理,可以与总帐、销售/采购集成使用。

·仓库管理。仓库管理主要处理出、入库业务和企业存货控制。

·订单管理。订单管理包括各种采购订单和销售订单的处理,将其单独提出是为了更好地支持电子商务。

通过上述分析,笔者总结出对架构的需求描述如下:

(1)为以财务为核心的管理软件开发提供基础平台和核心商业逻辑。

(2)提供业务数据与财务数据的集成和对企业内部审计的支持。

(3)商业逻辑独立于底层数据库和用户界面。

(4)支持商业流程的定制。

(5)封装数据访问逻辑,使基于架构的产品可支持多种数据库。

(6)支持 Internet 和电子商务。

经过对中小企业运作流程的分析,设计的软件架构如图 2 所示。

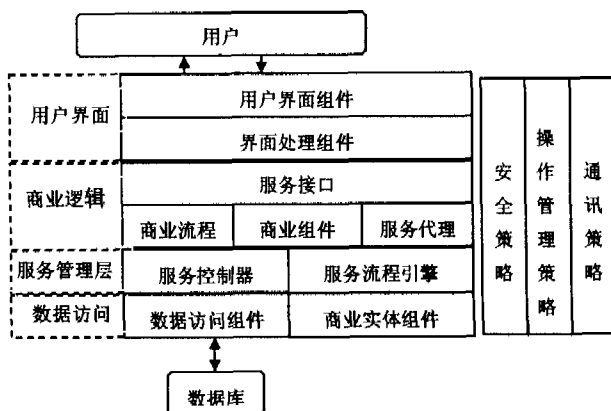


图 2 面向服务的中小企业管理软件架构设计

3 架构分析

架构包括数据访问、服务管理、商业逻辑、用户界面 4 层及贯穿于各层的安全、管理和通讯策略。各部分的功能定义如下。

(1)数据访问层。

提供对数据访问逻辑的封装,包括商业实体组件和数据访问逻辑组件。其中商业实体组件用 XML 文件实现对商业对象的封装(如订单);数据访问逻辑组件包括数据访问组件和实体控制组件。数据访问组件实现数据库的连接、SQL 命令的执行等功能;实体控制组件则负责完成对商业实体的插入、删除、修改、查找等操作。

(2)服务管理层。

主要负责对服务的管理和业务流程的定义和执行,包括服务控制器和服务流程引擎。服务控制器实现服务的实例化、管理和定位等;服务流程引擎完成对流程的解释、执行等工作。

(3)商业逻辑层。

定义系统商业逻辑,主要包括通用商业实体和核心商业流程两部分。通用商业实体定义系统各流程都需要的公共对象,如货币、财务日历、帐套、合作伙伴、银行等。核心商业流程包括服务和流程。服务实现各种简单商业操作,如开发票等;而流程实现一些完整的业务,如订货、产品入库等。一般服务的实现由商业实体、数据访问组件和通用商业对象相互协作完成,而流程由若干服务组成。为了支持系统与其他系统的集成,该层还定义了服务代理,主要用来调用外部服务。

(4)用户界面层。

提供用户接口,实现与用户的交互。为了使系统更加灵活,在这层定义了两类组件:用户界面组件和界面处理组件。用户界面组件负责用户数据的输入、输出及验证。界面处理组件完成界面与商业逻辑的协调。

(5)安全、管理与通讯策略。

定义系统的安全(包括权限管理)机制、异常/配置管理,以及组件间的通讯协议。

(下转第 64 页)

实现这个操作(不需要从点、线开始编码实现结点的绘制);开发人员也不需要考虑各种数据结构可视化的布局,JVDSCL 本身就提供了基本的布局方法,也可以根据自己的需要,重写这些方法达到个性化的实现。

开发人员只需调用数据结构类中的 draw() 方法,就可以在现有的画图区域实现数据结构的可视化。图 6 演示了后序遍历二叉树。

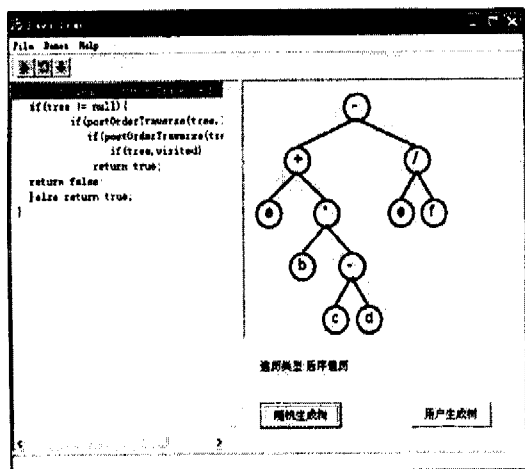


图 6 后序遍历二叉树

3 结束语

实践表明,构造可视数据结构是实现数据结构可视化的一种有效的方法。JVDSCL 为设计相关可视化程序提供了一个可复用的类库,提高了软件开发的效率。现阶段,JVDSCL 只是初步的实现,功能还不够完善。进一步

的工作是完善 JVDSCL 的功能,提高软件的重用性和扩展性。

参考文献:

- [1] Price B A, Baecker R M, Small I S. A principled taxonomy of software visualization[J]. Visual Languages and Computing, 1993, 3(3): 211 - 264.
- [2] Kerren A, Stasko J T. Algorithm animation[EB/OL]. www.ads.tuwien.ac.at/people/kerren/pubs/22690001.pdf, 2002.
- [3] Ossling G R, Freisleben B. ANIMAL: A System for Supporting Multiple Roles in Algorithm Animation[J]. Journal of Visual Languages and Computing, 2002, 13(3): 341 - 354.
- [4] Demetrescu C, Finocchi I. Smooth Animation of Algorithms in a Declarative Framework[J]. Journal of Visual Languages and Computing, 2001, 12(3): 253 - 281.
- [5] Ben-Ari M, Myller N, Sutinen E, et al. Perspectives on Program Animation with Jeliot[A]. International Seminar[C]. Berlin: Springer, 2002. 31 - 45.
- [6] Kamada T. Visualizing Abstract Objects And Relations[M]. Hong Kong: World Scientific Publishing Co Pte Ltd, 1989. 69 - 104.
- [7] Bertault F. A force-directed algorithm that preserves edge-crossing properties[J]. Information Processing Letters, 2000, 74(1-2): 7 - 13.
- [8] 吴伟民, 樊敏, 曹勇锋. 动态图的实时三维可视化稳定性算法[J]. 计算机工程与设计, 2005, 26(3): 801 - 802.
- [9] 孙炜, 吴伟民, 陈志峰. 基于遗传模拟退火算法的图的三维可视化[J]. 广东工业大学学报, 2002, 19(1): 37 - 41.

(上接第 60 页)

4 基于本架构开发的优点

与传统的开发方法相比,采用基于架构的方法可以使开发人员有共同的参考基础——架构。使得各成员之间的交流变得非常容易^[4]。采用基于架构的方法也具有其他显著的优点:

(1) 编码时间明显缩短。因为大量使用重用机制,可以使编码数量大为减少。

(2) 产品的测试和维护更加容易。由于框架定义了代码规范和大量的程序模板,所以开发很有规律。而且测试的时候可以先对某一模块进行测试,其他模块则根据测试结果进行类似的修改、调整,大大节约了测试时间。

(3) 产品实施的灵活性提高。基于服务的软件部署和配置方案,使管理软件的实施更加灵活。比如:很容易就可以同时支持局域网应用和基于互联网的远程下订单、综合查询等应用。同时,面向服务的架构对界面的多样性支持得很好。

5 结束语

软件架构对中小企业管理软件的功能和质量有着重

要影响,良好的产品架构设计可以从整体上保证管理软件质量,提高软件的灵活性、可扩展性和可重用性^[5]。面向服务的软件架构具有协议开放、易于集成、支持业务流程持续改进等特点,可以作为基于 Internet 的下一代中小企业管理软件架构设计的参考模型。

参考文献:

- [1] Perry D E, Wolf A L. Foundations for the Study of Software Architecture Software Engineering Notes[J]. ACM SIGN-SOFT, 1992, 17(4): 40 - 52.
- [2] Kruchten P B. The 4 + 1 View Model of Architecture[J]. IEEE Software, 1995, 12(6): 42 - 50.
- [3] 柴晓路. Web Services 技术、架构和应用[M]. 北京: 电子工业出版社, 2003.
- [4] Binns P, Vestal S. Formal real-time architecture specification and analysis[A]. In Tenth Workshop on Real-Time Operating Systems and Software[C]. New York: [s. n.], 1993. 125 - 133.
- [5] 柴晓路. Web Services 架构[EB/OL]. http://www.900.cn.ibm.com/developerWorks/cn/webservices/ws-wsar/index.shtml, 2001 - 12 - 03.