

基于软件总线的遗留 CAD 系统再开发

郝 昂, 史维峰, 袁 静, 赵 蓉

(西北大学信息科学与技术学院, 西安 710069)

摘要: 探讨从遗留的 CAD 系统中获得有效信息、组件的方法和过程, 结合当前 CAD 领域的新发展, 提出一种新的基于软件总线体系结构的 CAD 系统模型, 并分析了组成该模型各组件的功能特性。该系统具有较好的开放性与集成性, 使处于异地的设计人员能协同工作。文中也给出了该体系结构在博士 CAD 系统中的应用实例。

关键词: CAD 系统; 软件总线; 体系结构

Re-development of Legacy CAD System Based on Software Bus

HAO Ang, SHI Wei-feng, YUAN Jing, ZHAO Rong

(College of Information Science and Technology, Northwest University, Xi'an 710069)

【Abstract】 This article discusses the method and process to extract effective information and component form legacy CAD system. Combined with the new development domain of CAD, it proposes a new CAD system model based on software bus architecture and analyses the functions of each component composed the model. With better openness and integrity, the system enables designers at different places to work collaboratively. An example the architecture used in the DrCAD system is presented.

【Key words】 CAD; software bus; architecture

博士 CAD(DrCAD)系统是 20 世纪 90 年代中期推出的自主版权的二维 CAD 软件。现在版本的 DrCAD 系统由大量的模块构成。这些模块通过一些具有连接功能的模块, 以扁平的结构互连在一起。这种软件结构使对软件的改动非常困难, 而且易于出错。基于现实的考虑, 有必要在 DrCAD 的基础上, 利用构件技术及当前 CSCW 研究成果, 对其进行体系结构层面上的重新整合, 使其易于维护、升级和扩充。

1 软件组件与体系结构

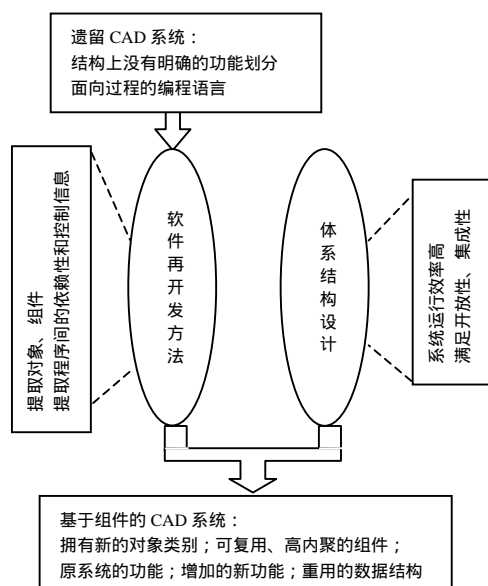
组件(component)是指语义完整、语法正确和有可重用价值的单位软件, 是软件重用过程中可以明确辨识的系统^[1]。其抽象程度比面向对象(object orientation)的抽象更高。它是对一组类的组合进行封装, 并代表完成一个或多个功能的特定服务, 也为用户提供了多个接口。设计模式(design patterns)的核心在于提供相关问题的解决方案。它提供了获得高层次的可复用组件的描述方式, 是复用成功设计的有效手段^[2]。

软件体系结构技术是基于构件的软件开发阶段以来软件开发所强调的一个重点^[3]。它描述的是组件及其关系, 定义了软件整体的结构、组成、协作及控制参数, 基于设计模式设计的软件组件可以是整个体系结构的一部分, 可以帮助体系结构框架获得更高层次的设计复用和代码复用, 有助于获得无须重新设计就可适用于多种应用的框架结构体。

2 组件及其需求规范的提取

软件重开发过程可以由图 1 来说明。遗留的 CAD 系统是以面向过程的语言编写的, 各个功能相互交织在一起。要从其中提炼对象和组件, 首先要对程序进行切片, 以实现功能的抽取。程序的切片过程也是按照各个功能的需求, 把程序分割成多个小片段, 以实现各功能独立的过程。每一个功能都是以一系列前置和后置条件给出的。可以用符号执行

(symbolic execution)技术提取程序语句执行的前置条件。新发现的条件与功能需求的前置和后置条件相比较, 结果相等时所对应的语句可以作为新功能块的出口或入口点的候选语句。一旦功能块划分的标准确定, 就能够用依赖图的方法把程序片段独立出来。符号执行和程序切片都需要用到数据流图和控制流图。



作者简介: 郝 昂(1982 -), 男, 硕士研究生, 主研方向: 计算机网络, CAD; 史维峰, 教授; 袁 静、赵 蓉, 硕士研究生
收稿日期: 2007-02-01 **E-mail:** anghao@tom.com

新构建的组件分离了数据流和控制流。组件上分别有数据和控制的输入输出接口。可以分为4个部分：数据输入，数据输出，控制输入和控制输出。图2是对组件接口的详细说明。



图2 组件接口

3 博士 CAD 系统的软件总线体系结构

Okii 定义了4个软件总线设计的原则^[4]：

- (1)最小的核心语义。把总线通信应用上的需求减少到最小，使其成为高效的通信设备。
- (2)自我描述的对象。对象的自省功能能够使对象自身从一种类型转变到另一种类型，减少应用间的通信量。自省也能使对象自己决定是否对接收到的信息感兴趣。
- (3)支持动态的操作。不必关闭总线就能够通过增加新的类或更改已存在的组件来增加新的功能或更改已有的功能
- (4)支持匿名的通信。数据对象的发送与接收可以在同步或异步的时间、空间模式下，不必考虑数据的发送者和接收者的身份。

DrCAD 系统体系结构的设计是以上4个原则为基础的。

王云鹏在CAXA电子图板的设计中，提出了双总线的体系结构^[5]。该结构有很强的开放性和灵活性，使软件容易对底层库更改和扩充。但是系统中具有两级总线，增加了系统的复杂性。同时，系统的核心组件要与上下两级总线相连，增加了组件开发的复杂性和流程控制的复杂性，从另一个侧面又制约了系统的开放性。陆薇提出了一种应用总线+应用代理的体系结构^[6]。该结构有很强的扩充性，支持用不同程序语言开发的构件。以上两种系统都是应用在单机系统中，没有考虑网络环境下的计算机辅助设计。张秋余提出了分布式环境下的软件总线体系结构^[7]。该结构妥善地解决了分布式计算机的通信问题，但它对软件总线分布式功能上的扩充，限制了总线的通用性，进而约束了其开放性。

在以上研究的基础上，本文提出了一种单总线的体系结构，并应用于博士 CAD 系统中。如图3所示。

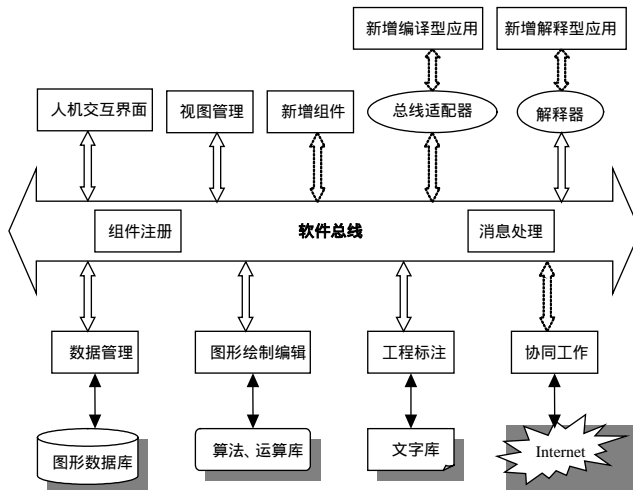


图3 博士 CAD 系统体系结构

总线是借鉴硬件系统中总线的概念。在硬件系统中，总线只不过是各个功能部件数据传输的通路，仅仅为各部件提供统一的接口，便于系统的集成。应用在软件系统中，总线相当于中间件，便于新的功能组件加入到系统中。软件总线的内核功能越少，其通用性越高，进而开放性也越好。因此在本文的设计中，总线只具有基本的组件注册与消息处理两项功能。考虑到核心组件及其用到的底层库变动相对较小，为减少组件的复杂度和开发的困难度，核心组件直接操作地层库，而不在它们之间再加一条总线。把系统协同工作的功能也独立出来，用一个组件来实现，保持了软件总线的简洁性。

3.1 软件总线

软件总线有组件注册与消息处理两种功能。每个插入到总线中的组件必须先注册模块中注册。注册包括组件的逻辑名称、物理地址、提供给其他组件的功能接口等。注册功能具有动态加载与注销组件的能力，同时它保证加载或卸载多个组件时不影响其他组件的正常使用。消息处理模块根据各组件的注册信息，把系统中数据与控制信息传送到相应的一个或一组组件。消息处理并不负责具体功能的流程控制，流程控制的任务分布到各相应的组件中。这样的设计不仅可以简化总线的设计，还可以使组件具有自省功能，减少组件间的通信量，提高系统效率。

3.2 核心功能组件

从博士 CAD 系统中提取的核心功能组件包括如下几个部分：

- (1)视图管理组件。负责对视图的缩放、平移以及显示样式的更改。
- (2)数据管理组件。提供对图形数据库的访问操作，实现对图形对象的存取。还具有不同数据格式间相互转换的功能。
- (3)图形绘制编辑组件。与算法支持库交互，负责图形的绘制与编辑。具有对图形绘制流程的完全控制。
- (4)工程标注组件。与底层文字库交互，提供对图形的标注功能。
- (5)人机交互界面。以鼠标和命令行方式提供人机交互的接口。图形的绘制、组件的加载等功能都能在界面中找到相应的接口。

3.3 协同工作组件

协同工作组件是对原 DrCAD 系统功能的增加。它提供了网络环境下，多个设计者共同设计的功能。该组件不属于 CAD 系统核心，在系统开始运行时并不加载。当需要协同工作时再动态地加载到总线。该总线可以根据需要充当服务器或客户的角色。当某主机创建或打开了一个图形文件，等待其他机器加入进行协同设计时，该主机的协同工作组件就充当服务器的角色，其他机器的就充当客户端角色。

3.4 对开放性的支持

软件总线对外提供了统一而简洁的接口，只要符合该接口的组件都能集成到系统中。对于异构的组件，提供了相应的解决方法。异构编译型构件通过总线适配器和软件总线相连，其执行过程需要该种语言解释器的支持，因此由解释器统一管理，再通过总线和其他构件集成^[6]。

4 结束语

笔者从旧版 DrCAD 系统中提取组件，再按照本文中提出的基于软件总线的体系结构思路，构建出新版 DrCAD 系统。图4是 DrCAD 的运行界面。（下转第269页）