

文章编号:1001-9081(2006)01-0246-03

一种 CAD 系统组件模型的组合技术及其实现

王 珉¹, 吴广茂², 田 林², 罗志清¹, 汤幼宁²

(1. 西北工业大学 机电学院, 陕西 西安 710072;

2. 中国航空计算技术研究所, 陕西 西安 710068)

(wangm_china@163.com)

摘 要:当前 CAD 系统中采用的组件模型主要有两类:标准组件模型和自定义组件模型。尽管这两种情况都有成功的例子,但是不能完全照搬成功经验。在分析了当前 CAD 开发中面临的挑战后,讨论了组件模型的选择策略,并给出了一个较好的解决方案——在标准组件模型的基础上增加一些新的特征和技术。最后,提出和实现了一个在 COM 组件模型基础上组件复用和扩展的新技术。

关键词:组件模型; CAD; 基于组件开发; COM

中图分类号: TP391; TP311 **文献标识码:** A

Composition techniques of CAD component model and its implementation

WANG Min¹, WU Guang-mao², TIAN Lin², LUO Zhi-qing¹, TANG You-ning²

(1. College of Mechano-Electronics Engineering, Northwestern Polytechnic University, Xi'an Shaanxi 710072, China;

2. Aeronautics Computing Technique Research Institute, Xi'an Shaanxi 710068, China)

Abstract: The component models used in CAD are mainly classified into two categories: standard component model and in-house component model. The successful experience should not be copied blindly, although there are some successful instances in both cases. After the analysis of the challenge in development of CAD, the selecting strategy of component model was discussed and an appropriate solution that enhanced some new features and techniques based on standard component model was given. At last, a new composition technique of component reused and extension based on COM was represented and implemented. This technique had two advantageous properties. It didn't change either the identity of the component being extended, or the Code of the component being extended.

Key words: component model; CAD; CBD; COM

组件模型是目前基于组件开发(CBD)领域研究和应用的热点问题之一,但是“目前组件模型的理论研究与它在大规模的工业开发中的实际使用之间存在巨大的鸿沟。这是特别令人忧虑的,因为组件模型是一个决定系统开发的成功(或失败)的关键因素”^[1]。本文针对 CAD 这类特定的软件系统,讨论其组件模型的选择。并在 COM 模型的基础上提出并实现了一个组件组合和扩展的新技术。

1 当前 CAD 软件采用的组件模型

从目前的发展现状来看,开发一个新的 CAD 产品应尽可能利用市场上可以获得的现有资源,“凡是市场上能够买到的,只要定价合理,外购一定比自制合算”^[2]。例如 Solidworks 和 SolidEdge 都是立足于商业几何造型引擎 parasolid,专注于开发几何造型功能,其他专业模块,一般交由合作联盟开发。这种基于组件的开发模式要处理好以下几种组件:从市场上获取的成熟的组件,例如几何造型引擎等;需要二次开发的组件,如需要进一步完善的不成熟模块(如原型系统)或功能还不全面的原有组件;外包开发的组件,如一些专业模块外包给在该领域更有实力的公司;自行研制的组件。

因此组件模型的选择不仅仅要考虑技术上的问题,还要受到市场上商业组件能否与之很好的互操作等其他因素的制约。

1.1 采用自行定义和实现的组件模型

CAD 软件的高端产品 DS 公司的 CATIA V5,采用的就是自行定义和实现的组件模型,称为 OM(Object Modeler)^[1],取得很大成功。OM 有四个重要的组合与扩展机制:扩展、组件继承、条件接口和代理。这些机制为 CATIA 产品提供了丰富的组件扩展能力。

OM 模型较好地满足了 CATIA 的需求,也取得了巨大的成功,这并不意味着完全照搬 DS 经验而采用自行开发组件模型一定会获取成功。

首先,采用自定义组件模型风险很大。尽管 OM 模型是成功的组件模型,但是 DS 在总结经验时依然觉得“自行设计和开发一个组件模型是非常危险的”^[1]。OM 模型要比 COM 模型提供了更多的机制,但是也复杂得多,需对员工进行长期培训,并为避免重复出现的概念误用而建立了许多反模式。这些是不熟悉 OM 模型的其他工程师很难获得的。

其次,创建自己的组件模型限制了与第三方组件实现互操作的可能性。DS 公司具有多年开发经验,并且具有很多当今世界上顶尖的模块,从几何造型引擎、几何约束求解,到航空特色模块(如航空钣金模块),因此绝大多数基础性、平台性的组件由 DS 公司开发。这样 DS 公司内部可以大量采用

收稿日期:2005-07-03;修订日期:2005-09-08 基金项目:国家 863 计划资助项目(2001AA415120)

作者简介:王珉(1975-),男,山东蓬莱人,博士研究生,主要研究方向:软件工程、CAD、CIMS、CBSE; 吴广茂(1941-),男,河北人,研究员,博士生导师,主要研究方向:CAD、网络工程和软件工程; 田林(1977-),男,陕西宝鸡人,工程师,主要研究方向:软件工程、软件测试、网络安全; 罗志清(1971-),男,甘肃白银人,博士研究生,主要研究方向:先进制造技术及优化; 汤幼宁(1966-),男(汉族),研究员,博士,主要研究方向:软件工程、计算机图形学、网络工程。

这种组件模型。而在这方面积累较少的公司,一般要借鉴 Solidworks 的成功经验,与其他专业公司合作,并采用大量其他公司的商用软件(及组件)产品。在这种情况下,自行开发复杂的组件模型,效益不明显而风险较大。

1.2 采用标准组件模型

由于采用自定义组件模型有很多限制,所以在选择组件模型时,尽管目前的标准组件模型还不完善,但是绝大多数系统开发仍然优先考虑标准组件模型。

在标准组件模型中,EJB 要求所有组件都用 Java 编写,这限制了组件的实现语言,而且在与遗留系统的集成方面存在严重问题,Java 程序必须运行在 JVM 上,这使得 Java 很难与非 Java 代码实现紧密集成(无法运行在相同的进程空间)。CORBA 不是一个二进制的标准,因此还要选择 ORB 提供商,而 ORB 提供商支持在各种平台上各种语言要比 COM 更费力,所以 ORB 提供商所支持的平台、语言和编译器常常受到限制。相比之下,COM 支持的语言更多,而且是一个二进制的标准,更重要的是市场上存在大量可供利用的基于 COM 的第三方组件。更何况当前主流 CAD 系统都开始转向 Windows 平台的 PC 机上运行了。因此,新兴的 CAD 软件如 solidworks, SolidEdge^[2] 和国内 CAD 软件如开目 CAD2003、CAXA^[3] 等,基本上都是采用 COM 组件模型(包括建立在 COM 基础上的 ActiveX、OLE 技术)。COM 支持两个组合技术:包容和聚合。通过包容操作,可以使一个 COM 对象包含另一个 COM 对象。聚合更复杂一些,外部对象揭示内部对象的接口,就好像外部对象实现了它们,但是避免了间接调用^[1,4]。

2 一种 CAD 系统组件模型的实现方案

考虑到这些,我们认为,如果新开发一个在基于 Windows 平台的 PC 机上运行的 CAD 软件,采用基于组件的开发方法,以尽量利用已有组件为出发点,其组件模型的实现方案如下:

1) 在总体上采用 COM 模型

采用 COM 模型可以解决大多数应用需要,并且可以得到为数众多的第三方组件的支持,而且 Solidworks 等 CAD 软件的成功已说明了 COM 模型在 CAD 软件成功采用的可行性。因此采用 COM 模型无疑可以充分发挥这些优势。例如作为软件总线的组件,采用 COM 模型无疑可以更方便地集成功能组件。对于可从市场上获取的成熟组件和外包开发的组件必须满足 COM 模型的标准。

2) 可以考虑对 COM 模型进行扩展,增加组件的复用和扩展机制

COM 模型的组件组合技术还不完善。对一些客户有扩展需求的组件,客户希望对系统提供的组件进行扩展可以在不修改它的源代码的情况下进行,并且尽管扩展的部分和系统提供的组件可能位于不同的实现模块中(如在不同的 DLL 中),但在使用者看来,可以整体上被看成一个组件。对于自研和二次开发的组件,可能在这方面有更强的需求。

3 扩充技术

正如前面所述,创建自己的组件模型限制了与第三方组件实现互操作的可能性,而且风险很大。当标准组件模型不能完全满足需求时,采用标准组件模型,并在其基础上增加一些规则和推荐的做法是一种很有价值的做法^[5]。因此,我们采用基于 COM 模型,吸取 CATIA 的 OM 模型的优点,在一些方面进行扩展的解决方案。我们称这种新技术为“扩充”

(Add in),对这种新技术的要求是:尽量保持 COM 模型的架构,并且尽可能应用原有的 COM 库函数;尽可能简单,便于应用人员掌握。

3.1 扩充技术的基本原理

由于 COM 模型依靠 QueryInterface 接口方法对组件支持的接口进行查询,而 COM 标准所定义的聚合方法无法在不改动被扩展的组件的 QueryInterface 方法的情况下满足要求。我们采用了一个中介——组件扩展字典来支持这种扩展机制。也就是说,当接口查不到所请求的方法时,再到组件扩展字典中查询,如图 1 所示。当在扩展字典中查到组件后从组件链表中调用相应组件的接口。

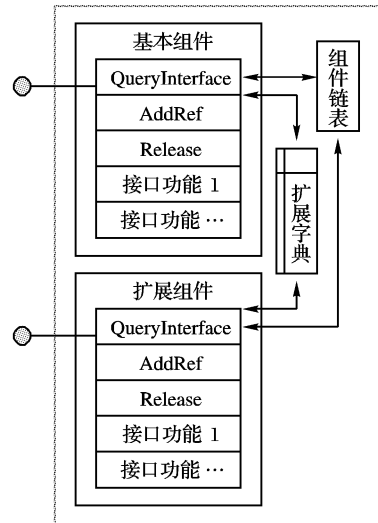


图 1 组件模型原理示意图

这种方法通过扩展字典规定了基本组件和扩展组件的扩展关系,而在组件客户的角度来看,他们是一个整体,组件的扩展组件是看不见的,似乎所有的功能都是由基本组件来实现的。另一方面,扩展组件的发布不需要更改基本组件的代码,而只需要在部署时在扩展字典中(可以是文本文件)添加该扩展组件的 ID 即可。这可以由手工完成,也可以由程序完成。换句话说,这种扩展机制具有增加扩展功能后不改变基本组件的 ID 和不改变基本组件的代码。

3.2 扩充技术的实现

扩充技术实现的关键是,在组件的基类中添加成员变量 ChainCmp(组件链表)和成员函数 Init()。ChainComponent 存放着 base 组件和所有扩展组件的指针,它是类厂创建组件并查询扩展字典之后调用成员函数 Init() 初始化的。该成员变量的数据结构如下:

```
struct ChainCmp
{
    const CLSID * cmpID;           //组件 GUID
    CAXUnknown * cmp;             //指向组件的指针
    ChainCmp * next;              //指向链表下一节点
}
```

客户利用 CoCreateInstance 创建组件和使用接口的过程如图 2 所示(为了简化,图中省略了一些幕后的调用)。

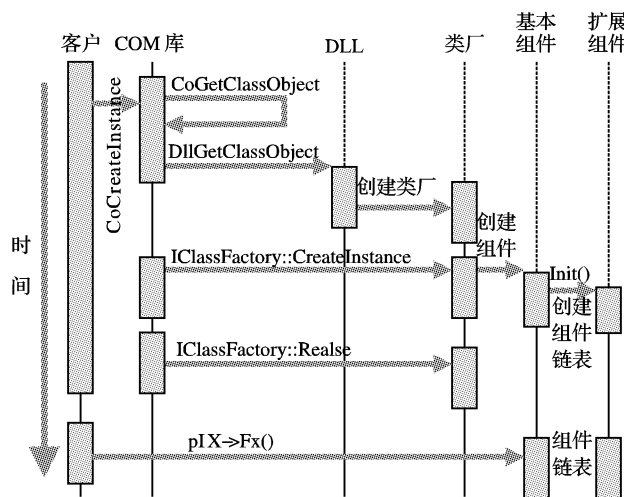
第一步,客户调用 COM 库实现的 CoCreateInstance,返回接口指针。

CoCreateInstance 首先调用 COM 库函数 CoGetClassObject, CoGetClassObject 在注册表中查找指定的组件,找到后,它装载实现此组件的 DLL。装载成功之后,它调用在 DLL 服务程序中实现的 DLLGetClassObject,此函数的作用是创建相应的

类厂,并查询类厂接口。

然后,CoCreateInstance 使用返回的类厂接口,来调用类厂的 CreateInstance 函数。该函数先调用组件的 CreateInstance 函数(在具体组件实现的文件中定义)创建基本组件,接着调用组件的 Init 函数(在组件的基类中实现)创建扩展组件并建立组件链表。最后查询组件的 QueryInterface(在组件的基类中实现,并在具体组件实现的文件中重载),返回接口指针。

第二步,客户调用返回的接口指针,执行接口定义的功能。



5个结构化元素:客户、COM库、DLL、类厂、组件

实线——元素已经建好并活动的

虚线——元素不再活动

灰框——每个操作生命周期

图2 CoCreateInstance 创建组件的过程

从上面过程可以看出,这与原来 COM 调用组件的过程几乎一样,只不过增加了基本组件初始化创建扩展组件的一步,而且这一工作主要由基类提供,对具体组件实现的工作量增加很少。

3.3 在 CAD 软件中如何应用扩充技术

当不同的软件功能由不同的团队开发,并且独立发布时,采用这种技术可以方便的添加组件功能,而无需更改被扩展组件的代码,甚至可以实现扩展组件的动态发布。例如,一个机翼可以用一个组件表示。一个开发团队可以开发机翼的空气动力学基本性能(如基本流场计算),其他团队开发独立出售的高级功能(如振动性能)。这些独立开发的组件只要遵循上述机制,发布后就可以无缝的连接起来,从外部观点来看,就像一个组件。

当产品采用打包配置时,不同的配置组件的数量不同,同

一组件实现的功能可能也有不同。因此,对有些组件采用这种扩展机制,可以根据企业的需要,方便、低成本的升级,而不影响以前的投资。

软件给用户提供二次开发的平台,如果支持这种技术,可以使用户的扩展与原来的系统无缝连接,并且不会影响到原有的应用。

3.4 扩充技术的不足和需要进一步研究的问题

尽管这种新技术有上述的优点,但也存在着一些不足。例如:

当分别在基本组件和扩展组件中实现的接口,被频繁的交替查询时,这种机制的查询效率会降低。幸运的是,这种情况出现的几率比较低。

由于要维护一个组件字典,给使用和维护带来了一些不便。不过由于使用和维护都很简单,而且只对组件“服务器端”有影响,对使用这些组件的“客户”而言,几乎没什么影响。

对于已有的组件和购买的组件,由于基类没添加相应的数据和方法,所以无法在不改变源代码的情况下使用该方法。

组件字典的定义、内容和如何维护是需要仔细研究的一个重点,这关系到性能、功能和易用性。另外,通过组件字典是否也可以实现其他技术(例如,条件接口)也是一个需要研究的问题。

4 结语

组件模型对软件开发来说是一个关键因素,本文针对目前 CAD 开发的趋势,提出采用在 COM 组件模型的基础上增加适当的特征的组件模型选择方案,可以最大限度地利用当前市场上已有组件,便于与其他专业公司合作开发,而且比自行开发组件模型的风险要小的多。在 COM 组件模型的基础上,我们提出一个组件复用和扩展的新技术,并在 VC++ 6.0 环境中,实现了这种技术。结果表明,采用它可以有效的支持组件的组合、复用和动态扩展。

参考文献:

- [1] IVICA C, MAGNUS L. Building reliable component-based software systems[M]. London: Artech House publishers, 2002. 375-386.
- [2] 叶修梓,彭维,唐荣锡.国际 CAD 产业的发展历史回顾与几点经验教训[J].计算机辅助设计与图形学学报,2003,15(10):1185-1193.
- [3] 白跃伟,陈明,陈卓宁,等.基于 Windows 组件可复用的 CAD 系统开发[J].计算机工程与应用,2004,(13):99-101.
- [4] ROGERSON D. Inside COM[M]. Washington: Microsoft Press, 1997.
- [5] WHITEHEAD K. Component-based Development[M]. London: Addison Wesley, 2003. 17-113.

(上接第 242 页)

- [8] 陈广明,陈生庆,张立臣.Z 实时扩展及基于多视点的应用模式[J].计算机应用,2005,25(2),362-365.
- [9] Ftp://ftp.irt.uni-hannover.de/pub/pearl/report.pdf[EB/OL].
- [10] FISCHER C. How to combine Z with a process algebra[A]. BOWEN JP, FETT A, HINCHEY MG, eds. ZUM98 The Z Formal Specification Notation, volume 1493 of LNCS[C], 2000.
- [11] STANKAVIC J, ZHU R, POORNALINGAM R, et al. VEST: an aspect-based composition tool for real-time systems[A]. the 9th Real-Time Applications Symposium 2003[C]. Toronto, Canada: IEEE Computer Society Press, 2003. 110-123.
- [12] STOYENKO AD, MARLOWE TJ. polynomial-Time Transformations and Schedulability Analysis of Parallel Real-Time Programs with re-

- stricted Resource Contention[A]. Real-Time Systems[C], 1992. 307-329.
- [13] SCHNEIDER S. An Operational Semantics for Timed CSP[A]. number 1453 in Lecture Notes in Computer Science[C]. Springer-Verlag, 1997. 45-61.
- [14] 陈广明,张立臣,陈生庆.面向方面的实时软件开发方法[J].计算机科学,2005,32(7).
- [15] STOYENKO A, HALANG W. High-Integrity PEARL and its Schedulability Analyzer: Transferring State-of-the Art Real-Time Software Technology from University Laboratories to industry[R]. Tech. report CIS-91-16, CIS Dept, NEW Jersey Institute of Technology, Newark, 1991.