

基于构件组装的信息处理软件开发方法

李晓静, 彭春燕, 田 青

(江南计算技术研究所, 江苏 无锡 214083)

摘 要:基于构件组装的软件开发方法被认为是提高软件开发效率和质量的有效途径,在软件开发中得到了越来越多的应用。文中分析了软件开发的现状,基于信息处理领域提出了一种构件组合式软件开发方法和适合这种开发方法的构件实现模型。介绍了这种构件组合式软件开发方法的基本实现原理,和在该方法指导下的软件开发过程。通过搭建一个实际的智能文本处理系统证明了该开发方法不仅简化了软件复用,提高了软件开发效率和质量,同时基于这种方法开发的软件具有较高运行效率。

关键词:构件;软件开发;信息处理

中图分类号:TP311.52

文献标识码:A

文章编号:1673-629X(2008)11-0128-03

Software Development Approach to Process Information Based on Component Composition

LI Xiao-jing, PENG Chun-yan, TIAN Qing

(Jiangnan Institute of Computing Technology, Wuxi 214083, China)

Abstract: The component-based software development is considered as an effective and efficient approach to improve the productivity and quality of software development, and is applied more and more widely in software development. Analyzes the current software development, and based information-processing it introduces the component composition software development approach and the component model that adapts to this software development approach. The principle of component composition software development approach and the procedure how to develop software are discussed. Through building a text processing system, the approach predigests software reuse, improves the productivity and quality, and the software based this approach holds higher running efficiency.

Key words: component; software development; information process

0 引 言

近年来,基于构件的软件开发(component based software development, CBSD)技术受到越来越多的软件工程人员重视。基于构件的软件开发提高了软件的复用粒度,有利于软件质量和软件开发效率的提高,有助于软件工业化的实现^[1]。

目前,基于构件的软件开发其主流设计和实现语言基于面向对象(Object Orientation, OO)技术,针对现有的异构网络,使不同平台下实现的网络应用程序能够相互作用。比如 CORBA、EJB、DCOM 等都基于上述目的提供了在实现层次上支持构件交互的基础机制^[2]。但是,在基于构件的分布式对象架构中,构件作为服务发布,该方式虽然提高了构件开发语言和运行

环境的兼容性,但不可避免带来了运行效率上的损失。

实际应用中,有些软件程序仅是在单机上运行,对开发语言和运行环境的兼容性要求不高,对程序的运行效率要求较高。基于构件的分布式对象框架显然无法满足这些应用的开发要求,相关应用程序只能是程序开发者从一行行代码写起。所以,一旦应用需求发生变化,应用程序的调整难度相当于重新开发一个新的应用,不仅领域既有成果不能有效共享,而且开发效率低下,软件质量难以保证。所以,在一些应用背景经常发生变化,应用程序对效率要求较高,对兼容性要求较低的领域,有必要寻找一种有效的软件复用机制,以提高软件开发效率和软件质量。

在海量信息处理领域,信息呈现出高带宽、高速率、大流量的特点,要求信息处理程序具有足够高的运行效率,同时信息呈现应用面广、信息量大、多格式的特点,要求信息处理程序能根据数据的变化快速准确地做出调整。基于上述目的,提出了一种适合海量信

收稿日期:2008-02-14

基金项目:总装备部预先研究项目基金(Y-30602)

作者简介:李晓静(1982-),女,陕西岐山人,硕士研究生,研究方向为信息处理;田 青,高级工程师,研究方向为信息处理。

息处理领域的组合式软件开发方法——based component composition software development approach to process information(CSPI),该方法不仅加速了信息处理软件的开发,实现了对不同数据处理手段的敏捷定制,同时避免了分布式对象架构带来的运行效率损失,能够满足信息处理软件对效率的要求。文中所做的工作有助于信息处理领域的软件生产工业化,同时对在相似领域进行软件工业化生产具有一定的借鉴价值。

文中从基本思想、基本概念、转换关系等方面介绍 CSPI 方法,同时简单介绍了 CSPI 方法指导的软件开发流程。

1 基本思想

在基于构件的软件开发中,复用的基本单元是构件,与传统的 API、类的复具有很大的差别。首先从粒度上而言,构件增大了软件复用的粒度;其次,从功能上来看,构件是一个具有完整功能和标准化接口的可独立部署的单元,具有一定的结构,符合一定的标准,隐藏了具体实现,通过提供接口对外部提供服务。构件复用不再是代码复用,而是服务复用;第三,在 CBSD 中,软件开发的重点从程序设计变成构件组装^[3]。

在 CSPI 中,各信息处理构件之间传递的主要是数据,CSPI 方法采用 XML 数据总线技术来实现对数据流的控制管理,如图 1 所示。即所有信息处理构件的输入数据在 XML 数据总线中获取的,输出数据回写到 XML 数据总线中,作为其它信息处理构件的输入数据。构件从 XML 数据总线中获取数据的先后关系及获取的数据位置,由控制部分 Control 负责实现。

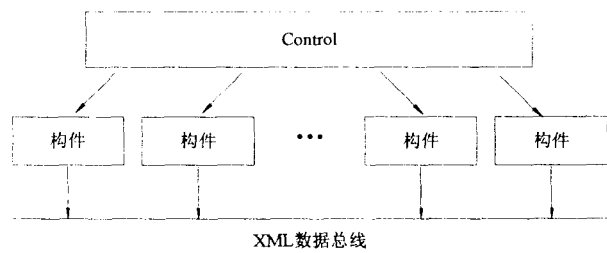


图 1 基于 XML 数据总线的构件运行平台

2 基本概念

作为一种软件开发方法,CSPI 的基本概念主要有构件(component)、连接器(connector)、构件描述语言(component description language, CDL)、控制流(control flow, CF)和流程描述语言(flow description language, FDL)。

构件是一个具有规范接口和确定的上下文依赖的

组装单元,它能够被独立部署和被第三方组装^[4]。构件模型是对构件本质特征及构件间关系的抽象描述,构件模型关注构件的接口,封装构件的实现细节,是构件组装的依据和保证,并为构件的不同生产者和使用者提供了一个为大家所接收的一致性描述^[5]。CSPI 从信息处理软件开发的需要出发,定义构件的实现模型,如图 2 所示。

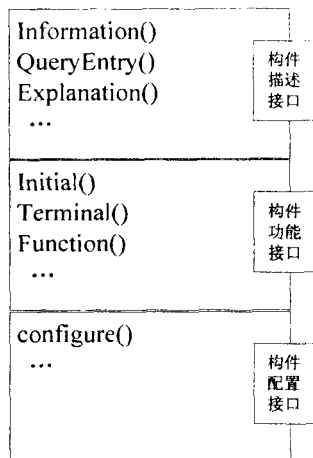


图 2 CSPI 构件模型

这个构件模型对外提供的接口有三类:构件描述接口、构件功能接口和构件配置接口。构件描述接口提供构件的一些基本信息,如版本号、错误信息、构件对外要求的功能和对外提供的功能等;构件功能接口实现构件的主要功能,同时构件功能接口可作为 API 接口用于构件扩展;构件配置接口负责完成构件调用时相关配置信息的更新。

连接器作为和构件同等级别的一阶实体(first-class entity),显式地描述了构件之间的交互关系或交互协议^[6],是 CSPI 控制流实现的基础。如图 1 中所示,CSPI 的控制流 Control 作为应用程序运行中构件的调度管理者,负责实现构件之间的执行顺序。以连接器为基础,CSPI 实现了三种类型的控制流:顺序执行、分支和循环。

构件描述是基于构件的软件开发和构件复用过程中的一项关键技术,它规定了一个软件构件所需的结构和内容,用于指导软件构件的设计、开发、管理、组装、测试等活动。流程描述是用来保存流程信息,它规定了一个完整的应用软件所需的结构和内容,是应用软件生成的基础。鉴于 XML 语言良好的开放性和可扩展性,CSPI 基于 XML Scheme,定义了自己的构件描述语言——CSPI-CDL 和流程描述语言——CSPI-FDL。其中 CSPI-CDL 的基本实现形式如下所示:

```
<component>
//构件基本信息
```

```

<property id="构件唯一标识" name="构件名称" version="版本号" />
<shortDesc shortDesc="构件功能描述" />
//参数列表
<parametersList>
  <parameter name="" type="">
    <defaultValue value="" />
  </parameter>
  .....
</parametersList>
</component>

```

3 转换关系

在 CSPI 中,构件是应用开发的基本单元,可执行应用程序是开发的最终结果,由构件组装而成的流程用流程描述文件保存,通过解析流程描述文件生成可执行应用程序。因此在 CSPI 中,涉及到两个转换过程:由构件描述文件生成流程描述文件以及由流程描述文件生成可执行代码。

(1) 从构件描述文件到流程描述文件。

构件描述文件分两大部分:基本信息部分和功能接口部分。基本信息主要描述构件的基础信息,包括:构件功能说明、构件名称、唯一标识、配置文件、依赖环境等;功能接口部分主要描述构件功能接口相关配置项。流程描述文件分两大部分:构件部分和控制部分。构件部分描述应用软件中涉及到的所有构件及其接口配置信息;控制部分描述构件之间的连接关系。

CSPI 提供的可视化组装工具依据构件描述生成构件基础模型并在组装工具中以图标方式显示,通过在组装工具中以画流程图的方式拖拽相关构件和连接子,完成应用流程设计,设计好的流程由流程描述文件保存,从构件描述到界面图标和从界面流程元素到流程描述由组装工具提供的自动映射规则实现。构件描述中的基本信息对应生成构件图元显示的相关信息,功能接口信息作为构件属性由用户根据流程需要进行配置;流程中所包含的所有构件及连接子,其信息在流程描述文件的构件部分保存,它们之间的连接关系在流程描述文件的控制部分保存。

(2) 从流程描述文件到应用程序。

CSPI 提供的流程部署模块负责将流程描述文件生成可执行应用程序,并将该应用部署到指定的服务器上,流程部署功能分三个步骤实现:

① 构件检测:构件检测服务负责确认流程描述文件中记录的所有构件在所部署的环境中确实存在,如果准备部署的流程中存在无法获得的构件实体,将提示用户并终止流程部署。

② 流程解析:流程解析服务负责将流程描述文件解析生成应用程序代码。通过解析流程描述文件,流程解析服务将 CSPI - FDL 描述层次的流程解析为代码层次的流程,在代码中实现构件的功能接口调用,将流程描述文件中记录的构件信息和控制信息转化为代码实现。CSPI 的构件模型和 XML 数据总线技术是流程解析服务实现基础。流程解析服务的实现原理如图 3 所示。

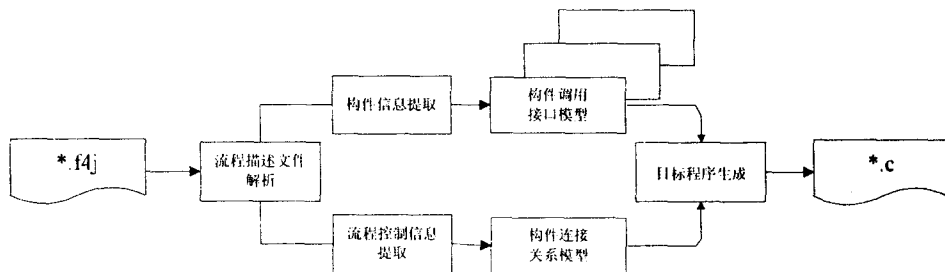


图 3 流程解析服务实现原理图

通过解析流程描述文件,提取构件信息生成构件调用接口模型,提取流程控制信息生成构件连接关系模型,以构件接口调用模型和构件连接关系模型为基础生成目标程序。

③ 编译部署:流程解析服务生成的代码文件,通过编译,在指定的服务器上生成可执行程序。由于采用运行时编译的方式,提高了运行环境的兼容性,在一定程度上克服了环境异构。

部署成功的应用流程以可执行程序方式存放。通过操作系统或者集群的相关调度功能则可以启动该应用程序。

4 CSPI 指导的软件开发过程

CSPI 方法实现了自顶向下的应用软件开发,其支持的软件开发流程如图 4 所示。

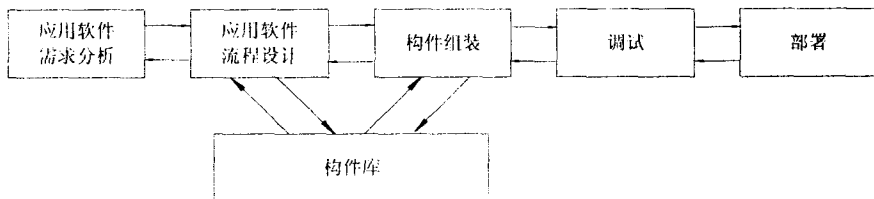


图 4 CSPI 软件开发流程

(下转第 179 页)

它实现了 CAD/CAM 软件的基本功能:对 AutoCAD R14 输出的 STL 文件的读取,STL 文件的 OpenGL 三维显示和视图变换操作,将 STL 文件数据输出给底层的刀补、差补软件并形成数控机床加工代码。在软件中,根据实际加工中的需求,预先存储了 27 种零件。

在仿真前,首先使用 AutoCAD 绘制出加工零件的三维模型,然后以 STL 文件保存,通过菜单栏调用 STL 文件,在 OpenGL 中实现对 STL 文件的三维显示和视图变换(显示在 3D 模型显示区),同时在模型参数显示区显示模型参数。最后实现的软件仿真界面如图 5 所示。

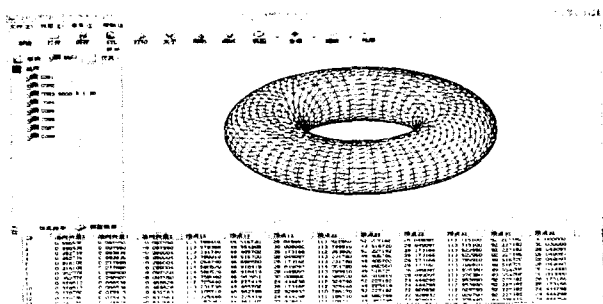


图 5 CAD/CAM 软件实现界面

4 结束语

为了实现数控机床加工零件产品的快速原形制造系统,文中提出了一种基于 AutoCAD、Visual C++ 6.

(上接第 130 页)

开发者首先进行需求分析,明确应用软件需要完成的功能;接着根据构件库的构件提供情况进行流程设计,并根据设计的流程从构件库中获取相应的构件完成构件组装;调试流程,修改流程中构件的相关配置,确保流程运行的正确性;最后将调试通过的流程部署到指定的集群上。该方法使用户只需要对整个应用的构造策略和原则进行精心的描述和制定,将所需要的构件按这些原则、策略组织起来,便可以推出一个新应用软件,从而大大提高软件生产效率。

5 结束语

基于构件的软件开发是近年来软件工程界研究的热点之一。文中针对信息处理领域,提出了一种基于构件的组合式软件开发方法——CSPI 方法。该方法以构件组装为指导,结合信息处理的特殊性,利用工具支持的自动转换机制,提供了一整套从流程设计到最终应用软件生成的解决方案。基于 CSPI 方法的基本思想,开发实现了它的支持工具 CSPI-Tool,在 CSPI-Tool 的支持下,通过开发实现一个文本智能处理系

统和 OpenGL 语言的 RPM 技术实现方案。这种方案可以方便地将 AutoCAD R14 输出的 STL 文件利用 OpenGL 进行三维显示,并可以根据需要实现各种视图变换操作;在确认设计图形后,可以将对应的 STL 文件数据输出给底层的刀补、差补软件并形成数控机床加工代码,提高数控机床加工产品的效率并降低企业的生产成本。

参考文献:

- [1] 张兴国,臧铁生,刘明.基于 OpenGL 仿真的装配机器人离线编程系统[J].系统仿真学报,2005(10):2433-2436.
- [2] Nakano A, Kalia R K, Vashishta P. Scalable Molecular Dynamics, Visualization, and Data-Management Algorithms for Materials Simulations[J]. Computing in Science & Engineering, 1999(9/10):37-39.
- [3] 李新春,赵冬斌,易建强.基于 OpenGL 的移动机械手路径规划仿真[J].系统仿真学报,2006(1):196-199.
- [4] 许万荣,樊自田.基于 OpenGL 的消失模铸造生产线设计的三维实现[J].特种铸造及有色合金,2006(1):17-18.
- [5] 叶文,马登武,陶永华.基于 PC 机的飞机低空突防航路规划仿真系统[J].系统仿真学报,2006(2):343-346.
- [6] 康凤举.现代仿真技术与应用[M].北京:国防工业出版社,2006:291-309.
- [7] 康凤举,高丽娥,杨惠珍,等.一种新型水下航行器半实物仿真系统的实现[J].系统仿真学报,2004(7):1453-1456.

统测试证明,基于 CSPI 开发软件系统,不仅简化了软件复用,提高了软件开发的效率和质量,同时系统具有较高的运行效率,能够满足信息处理的效率要求。

参考文献:

- [1] 杨美清,梅宏,李克勤.软件复用与软件构件技术[J].电子学报,1999,27(2):68-75.
- [2] 杨美清,梅宏,吕建,等.浅论软件技术发展[J].电子学报,2002,30(12A):1901-1906.
- [3] Clements P C. From subroutines to subsystems: component-based software development[M]//In: Brown A W. Component-Based software Engineering: Selected Paper from the Software Engineering Institute. Los Alamitos, CA: IEEE Computer Society Press, 1996:3-6.
- [4] Sametinger J. Software Engineering with Reusable Components[M]. Berlin, Germany: Springer-Verlag, 1997.
- [5] Szyperski C. Component Software: Beyond Object-Oriented Programming[M]. [s.l.]: Addison Wesley, 2002.
- [6] 梅宏,陈锋,冯耀东,等. ABC: 基于体系结构、面向构件的软件开发方法[J].软件学报,2003,14(4):721-731.