

·微机软件·

# 军用软件组件化设计方法研究

邵 刚

(南京模拟技术研究所, 南京 210016)

**摘 要:** 传统的军用软件设计方法不具备“规模定制, 按需而变”的能力, 无法根据软件产品需求快速迭代。引入组件化程序设计思想, 提出一种“平台+组件”的软件设计方法及技术体系架构, 实现业务与底层设计逻辑分离, 降低系统耦合性, 并可快捷部署, 针对用户不同需求快速实现软件定制开发。该架构实现了“一体化设计、组件化开发、组合式运用”, 在开发平台提供服务接口的基础上, 设计组件开发向导, 以模板的方式快速构建软件产品, 具有良好的扩展性和可维护性, 提高了软件开发效率、复用水平以及产品可靠性。

**关键词:** 平台; 组件; 软件设计; 体系架构; 快速构建

**DOI:** 10.3969/j.issn.1002-2279.2019.04.009

**中图分类号:** TP311

**文献标识码:** B

**文章编号:** 1002-2279(2019)04-0034-04

## Research on Component Design Method of Military Software

SHAO Gang

(Nanjing Research Institute on Simulation Technique, Nanjing 210016, China)

**Abstract:** Traditional methods for military software designing do not have the power of "mass customization and modified on demand", and cannot rapidly iterate according to the requirements of software products. The concept of component-based programming is introduced, and a software design method and technical architecture of "platform+component" is proposed, which realizes the separation of business and bottom-level design logic, reduces system coupling, can be quickly deployed, and can quickly realize software customization development according to various requirements of users. The architecture realizes "integrated design, component development and combined application". Based on the service interface provided by the development platform, the component development wizard is designed to quickly build software products in the form of templates. It has good expansibility and maintainability, and improves software development efficiency, reuse level and product reliability.

**Key words:** Platform; Component; Software design; System architecture; Fast constructing

## 1 引 言

随着我军信息化水平的不断提高, 军用软件的种类、规模和数量空前增长, 软件作为信息化建设的核心, 在武器装备、组织训练中发挥的作用日益凸显。运用传统设计方法开发的软件产品扩展性和适应性差<sup>[1]</sup>, 无法适应部队军事训练需求的快速变化, 继而会因无法跟随最新要求而被快速淘汰。然而新研或升级相关软件产品的工作量巨大, 无法兼顾部队实战化的训练需求。近年来实装部队和训练基地使用的军用软件种类繁多, 各自独立设计, 互无联系, 缺乏统一标准, 既不利于信息系统的可持续发展, 也无法充分发挥其应有的整体综合功能<sup>[2]</sup>。软件

组件化设计方法以软件平台为中心, 采用组件开发向导的方式开发功能组件<sup>[3]</sup>, 在不改变软件总体架构的基础上, 依靠其所提供的标准化服务与接口, 通过组件即插即用的方式, 实现业务功能<sup>[4]</sup>。采用本组件开发向导可在系统规划时设计各分系统间的标准接口, 减小分系统间的耦合度, 从而减少大量的重复开发工作并提升产品的可靠性, 极大地提高软件的开发效率和复用能力, 进一步减少后期运维工作量, 及时有效地适应业务需求的变化<sup>[5]</sup>。

## 2 “平台+组件”技术概述

### 2.1 “平台+组件”基本概念

平台是指软件运行的公共基础平台, 由基础框

作者简介: 邵刚(1977-), 男, 江苏省灌南县人, 高级工程师, 主研方向: 训练仿真与软件工程化。  
收稿日期: 2019-03-25

架、扩展框架和应用框架组成<sup>[6]</sup>。基础框架将软件系统通用性资源和功能需求进行了标准化集成与封装;扩展框架是对软件系统各种功能资源以及计算机操作系统资源的管理与代理;应用框架是对各种具有共性需求的功能资源的综合集成。这三部分共同作用,为各种面向具体应用的训练信息系统的开发与应用,提供了资源、服务、接口及功能的支撑,以满足训练领域内不同应用目的、对象,以及不同应用方式、内容的信息系统建设需求。

组件是指软件业务的功能封装,基于平台才能运行,是软件系统中的功能插件,可根据需要进行定制开发,该组件基于平台提供的组件开发向导创建的工程进行开发,通过向导工程可自动集成公共平台提供的标准服务和接口,在此基础上设计开发组

件特定的功能<sup>[7]</sup>。

## 2.2 “平台+组件”技术

设计“平台+组件”的软件总体逻辑架构如图 1 所示。建立适合军事行业需求的领域系统架构、业务模型和组件库,提供组件向导开发模板、软件运行平台和公共服务接口。具体的技术途径是通过组件开发向导设计组件功能,将设计好的组件文件复制到系统平台指定的文件夹下,当软件运行时会自动识别各个组件,并通过平台提供的组件加载接口将各组件加载到软件系统中,以实现各个组件特定的软件功能,组件具有较强的独立性、功能性、复用性和扩展性,可以方便地实现定制开发和集成其它业务功能。该技术不仅可降低软件开发和维护费用,还能提高开发效率、异构系统的集成能力和标准化程度。

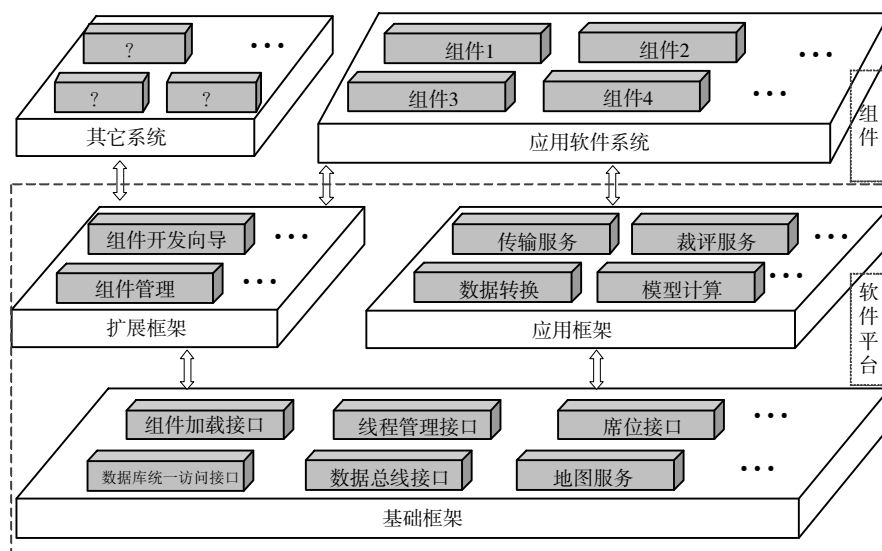


图 1 “平台+组件”总体逻辑架构

## 3 软件组件化设计方法

### 3.1 设计思路

软件组件化设计思路,主要是通过软件服务模块的重用和各组件的独立并行开发,在组件即插即用和组合式应用的基础上,快速构建出功能定制、性能可靠、运行稳定的软件产品<sup>[8]</sup>。组件化的软件设计,其基本思想是将软件系统分解成公共服务、运行平台和功能组件进行设计,再通过运行平台的公共服务调用接口、组件加载接口以及数据通信总线实现系统集成。

### 3.2 设计方法

组件化软件设计与传统方法的不同,突出表现在以下三个方面:以标准化为前提;以服务为基础;

以组合式应用为目的。

以标准化为前提,是指在统一公共服务设计标准、功能组件设计标准和运行平台调用标准的前提下,以接口为中心,实现各模块的统一设计、并行开发,在提高软件质量的同时减少系统测试和集成的工作量。

以服务为基础,是从系统中抽象出功能单一、性能稳定的模块封装成公共服务,建公共服务库,以备检索和调用。在此基础上,进行设计与开发时尽量调用公共服务构建各功能组件,实现由高成本低稳定性的功能重复开发,向低成本高稳定性的服务调用转变。

以组合式应用为目的,是以组件组合应用作为构建软件系统的核心,将所需的公共服务和功能组

件通过运行平台进行调用整合,满足用户特定需求和系统的规模定制。

### 3.3 生命周期

基于组件化设计过程是选择公共服务、组件设计开发、组件集成的过程,与传统软件设计过程相比,具有更强的科学性和灵活性。其生命周期通常包括:需求分析,运行平台设计与开发,公共服务的设计开发,构件的开发、选择和组装,系统集成,系统测试,软件维护等。

## 4 工程技术实现

在此,以某系统(略称为“系统 A”)为例进行组件化软件设计方法的说明。系统 A 体系架构如图 2 所示。系统主要提供通信和地图导航,其总体设计采用三层结构体系:第一层为公共服务层,主要负责提供基础服务、数据库服务、通信服务、地图服务;第二层为运行平台层,实现功能组件的加载,同时为功能

组件层提供服务接口;第三层为功能组件层,即应用层,直接为客户提供所需功能应用。

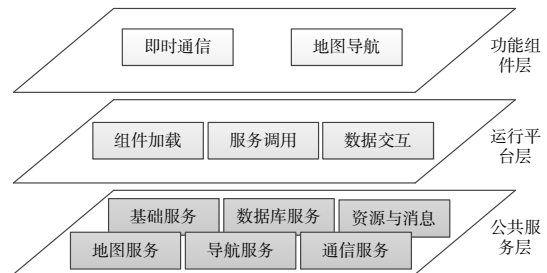


图 2 “系统 A”体系架构

### 4.1 公共服务

基于“平台+组件”的开发,就是按照平台提供的统一技术体制和方式,开发用户按需定制的功能,并按要求调用上述各类基础服务,开发出属于用户领域应用功能构件,然后再通过平台提供的重构与定制功能构建用户领域的应用系统。系统的 UML 图如图 3 所示。

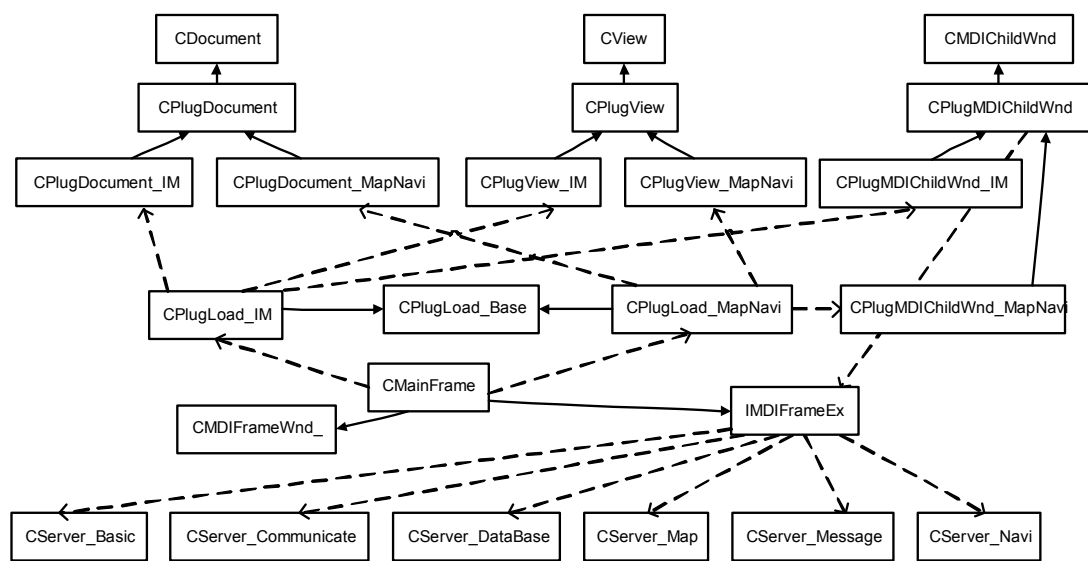


图 3 系统 UML 图

为了功能组件能够共享和调用公共服务,系统 A 按照统一的模式设计了各类基础功能和系统管理服务接口,共六大类,分别是:基础服务、数据库服务、通信服务、地图服务、导航服务和资源与消息服务,与之相对应的是六个接口类:CServer\_Basic、CServer\_Communicate、CServer\_DataBase、CServer\_Map、CServer\_Message 和 CServer\_Navi,并由框架接口 IMDIFrame 包含并提供调用接口。

### 4.2 功能组件

即时通信组件(CPlugLoad\_IM)和地图导航组件(CPlugLoad\_MapNavi)同为继承自统一调用接口

(CPlugLoad\_Base)的动态库;AddTemplate 作为主要接口函数,负责向运行平台提供添加文档模板功能(CMultiDocTemplate),其中所指的模板包含三个基类(CMDIChildWnd、CDocument、CView)的派生类,用于实现即时通信(CPlugMDIChildWnd\_IM、CPlugView\_IM、CPlugDocument\_IM)和地图导航(CPlugMDIChildWnd\_MapNavi、CPlugDocument\_MapNavi、CPlugView\_MapNavi)的相关功能。

系统 A 提供“功能组件开发向导”的实际工作界面如图 4 所示,其实现了类组件项目、资源、类以及组件界面风格的自定义创建,从而简化开发步骤,

提高开发效率。



图4 功能组件开发向导

### 4.3 运行平台

运行平台为多文档 MFC 应用程序,提供统一的集成框架和系统管理,平台初始化时通过调用功能组件文档模板添加接口,实现文档模板(CMultiDoc Template)的添加。

组件加载:通过 MFC 应用程序的多文档加载机制,调用 CMultiDocTemplate 类的 OpenDocumentFile (NULL)函数实现组件加载显示。

服务调用:运行平台的 CMainFrame 派生于虚接口基类 IMDIFrameEx,实现基础服务、数据库服务、通信服务、地图服务、导航服务动态库的加载与初始化,各功能组件通过类 IMDIFrameEx 指针进行服务模块的创建、调用和析构工作。

## 5 结 束 语

提出的软件组件化设计方法能够很好适应军事需求的快速变化,最低限度地减少软件定制开发的工作量。以平台为基础,通过组件开发快速构建软件产品的方法具备一定的先进性,可以提高软件的开发效率、复用水平以及产品可靠性,对新形势下的军用软件开发具有一定的参考价值。

### 参考文献:

[1] 李劲涛. 基于构件的软件工程技术探究[J]. 信息通信,

2016 (10):170-171.

LI Jintao. Research on component-based software engineering technology[J]. Information & Communications, 2016(10): 170-171.

[2] 徐雪峰, 冯书兴, 沈世禄, 等. 基于构件的软件开发方法和技术研究[J]. 装备指挥技术学院学报, 2007, 18(5):94-98. XU Xuefeng, FENG Shuxing, SHEN Shilu, et al. Research on the ways and technologies of component-based software development[J]. Journal of the Academy of Equipment Command & Technology, 2007, 18(5): 94-98.

[3] 田容雨. 基于软件构件技术的 Web 系统开发平台的研究[D]. 济南:山东大学, 2011.

TIAN Rongyu. Research on Web system develop platform based software component technology[D]. Ji 'nan: Shandong University, 2011.

[4] 徐勇. 军用软件管理中构件化技术应用研究[J]. 计算机与数字工程, 2013, 41(4):587-590.

XU Yong. Component technologies of the military software management[J]. Computer and Digital Engineering, 2013, 41 (4):587-590.

[5] 杨芙清, 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.

YANG Fuqing, MEI Hong, LI Keqin. Software reuse and software component technology[J]. ACTA ELECTRONICA SINICA, 1999, 27(2):68-75.

[6] 张华欣. 软件复用与软件构件技术探讨[J]. 信息通信, 2013(6):175.

ZHANG Huaxin. Discussion on software reuse and software component technology[J]. Information & Communications, 2013(6):175.

[7] 徐晓军. 探究基于软件总线技术的软件工程方法[J]. 信息化建设, 2015(9):288-289.

XU Xiaojun. Research on software engineering methods based on software bus technology[J]. Informatization Construction, 2015(9):288-289.

[8] 高欣. 基于可复用软构件的软件开发技术研究[J]. 信息与电脑, 2016(3):104-105.

GAO Xin. Research on software development technology based on reusable software components[J]. China Computer & Communication, 2016(3):104-105.