

# 浅谈架构高效应用系统平台的方法

## Expounding on the High-efficient Application System Structure

周志刚  
ZHOU Zhi-gang

(南宁铁路局信息技术所, 广西南宁 530003)  
(Department of Information Technology, The Railway Bureau of Nanning, Nanning, Guangxi, 530003, China)

**摘要:** 构筑高效应用系统平台的方法是依据软件管理规范化原则和开发方法可重用性原则, 先设计合理的总体框架, 全面分析系统内外所涉及的使用对象、处理数据和解决的问题, 再逐一建立针对性的功能模块。这样可以提高软件应用开发的效率, 较好地构筑高效应用系统平台。

**关键词:** 应用系统 软件开发 功能模块

**中图分类号:** TP393 **文献标识码:** A **文章编号:** 1002-7378(2008)04-0317-03

**Abstract:** According to the principles whose base on the standardization of software management and the reused developing methods, there is one of methods to build a high efficient application system platform. At first, a reasonable total structure is designed. Then users, data and problems surrounded the application system are totally analyzed. Third, modules of function is directly built one by one. In this way, the efficiency of software development is improved and a high efficient application system platform is build.

**Key words:** application system, software development, modules of function

近年来, 各行各业的信息化建设取得了长足的进步, 逐步投产了许多服务于相关专业的管理信息系统项目, 原来的传统工作模式大为改观、工作效率显著提高。构筑应用系统平台如同建筑大楼, 基础必须扎实、结构必须合理、施工必须规范, 否则整个应用系统的大厦必将隐患无穷。应用软件在研发初期, 如果系统架构不合理, 就会导致系统功能不可控、开发周期长、返工率高; 系统投产后, 稳定性低、维护成本高, 造成投产系统越多, “辫子”越多的局面, 因此软件开发成本居高不下、利润率偏低。许多单位及部门对进一步拓展内部信息化管理范围的热情高涨, 但是众多用户不断扩展的需求和软件开发人员的不足已成为制约信息化管理水平提高的瓶颈。如何提高软件开发机构的研发效率、软件管理水平和软件产品质量, 大幅降低开发风险、缩短开发周期, 成为

亟待解决的问题。

本文从系统基础架构的角度出发, 采用数据流程图<sup>[1]</sup>的分析方法, 对如何架构高效率的应用系统平台进行探讨。

### 1 应用系统架构的指导原则

一般来说, 应用系统的解决方案千差万别, 由于设计不一定合理, 管理不一定规范, 开发人员的技术水准不一定相同, 就会出现实施周期长短不一, 使得整个系统投产后在安全、稳定、效率方面的表现参差不齐。如何高效率地开发好系统, 形成高效稳定的应用系统平台并兼顾今后系统的功能扩充, 同时为其他应用系统的开发, 提高系统架构及代码的可重用性打下良好的基础, 可以从软件管理规范化和软件方法可重用性这两方面来考虑。

#### 1.1 软件管理规范化

软件开发过程是复杂、细致而又反复优化的过程。一般来说, 对于大多数规范化管理程度不高的软件开发机构, 管理人员和应用系统的用户难以有效

收稿日期: 2008-10-06

作者简介: 周志刚(1962-), 男, 高级工程师, 主要从事计算机应用方面的研究。

地深入了解和控制此过程,往往到整个应用系统即将交付的时候,被疏忽的问题才暴露出来,此时必须通过修补甚至返工才能解决问题。这样会导致应用系统的开发周期长、成本高、效率低。

DFD(数据流图)是软件开发常用的分析建模工具,作为软件工程学的一种分析方法得到广泛的应用,可以用来规范软件开发管理,从而达到持续地控制软件开发过程、保证软件质量、降低成本、提高软件开发效率的目的。采用DFD分析方法可以使得包括软件开发机构的管理和开发人员、软件用户在内的任何一个该项目的参与者,根据开发过程的阶段和分工充分把握整个过程的时间节点、技术手段、文档规范、人员、工时、预算、成本等各个方面的要求,从而实现软件开发过程的规范和可控,极大地降低开发过程中的差错率、提高开发效率。

1.2 开发方法可重用性

对于软件开发机构来说,同一工程中不同的开发者之间和不同时期的应用系统之间要实现规范化的高效率开发,就必须实现模块构件和设计思想的共享,解决软件开发方法的可重用性问题。为此,可以通过建立软件知识库的方法,使开发团队对于类似的应用需求共享知识库、模型构件以至程序代码,从而大幅度地提高整个应用系统的开发效率、缩短开发时间。新的应用系统开发的重点则可以转移到应对未曾遇到的新需求,突出应用系统的创新点及提高开发团队的创新能力上来。

2 应用系统架构方法

依据指导原则,构筑高效率的开放式应用系统,可以采用数据流图<sup>[1]</sup>的分析方法。首先设计合理的总体框架,全面分析系统内外所涉及的使用对象、处理数据和解决的问题,其次,逐一建立针对性的功能模块,进行分析处理。

2.1 构建总体框架

通过需求分析,采用自上而下的方法对该系统进行分解后得到4大功能模块(图1),它可以通过数据接口与外部应用系统进行信息交换,即用户认证、例外处理、功能平台和公共接口和外部应用系统。4个模块基本涵盖了相关业务的全部功能,并且还可以适用于其他类似的开放式应用系统。这种架构的核心就在于强调开发思想甚至代码的可重用性,这种可重用性构成了软件开发机构的知识库。因此,整个应用系统结构简洁高效,而且稳定性、通用性好,符合模块化开发思想,能够充分发挥团队优

势,便于开发管理,可以有效控制风险。

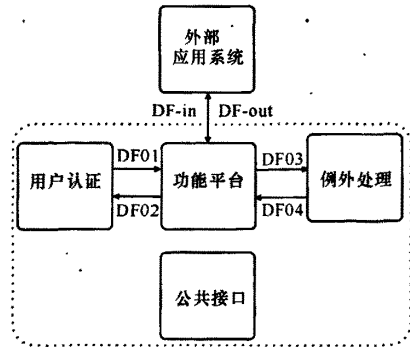


图1 铁路业务管理信息系统

2.2 建立功能模块

2.2.1 用户认证模块

用户认证模块用于用户权限认证的管理,包括用户组织管理、对象管理和权限管理等三个部份,如图2所示。该模块对系统的安全性起着至关重要的作用,不同的应用系统对此模块功能的要求各不相同,但基本功能大同小异,通过用户权限认证模块,用户可以对增加或修改用户、组织、对象、角色、身份等进行有效地管理,并能通过多种灵活的授权方式对用户赋予相应的权限。

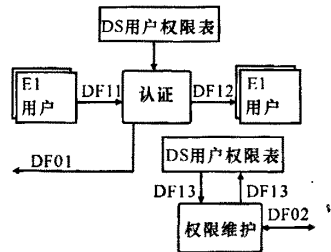


图2 用户认证模块数据流

用户权限的认证极为重要,一个应用系统能够安全可靠地运行有赖于此功能的有效发挥,因此有必要独立作为一个单独的模块从用户管理、对象管理和权限管理三方面认真细致地设计开发,并形成公共接口供其他应用系统共享,其处理过程主要是将用户输入的信息传递给认证处理程序,它根据用户权限角色表提供的权限判断用户的授权合法性并通知给有关调用程序,将一切非法用户拒之门外。该模块还负责根据使用者的要求更新维护用户权限角色表。

2.2.2 例外处理模块

例外处理模块用于应用系统的例外处理,也就是错误处理模块,如图3所示。该模块可处理应用系

统运行中所发生的例外事件,从而发现隐患,保障应用系统的稳定性。

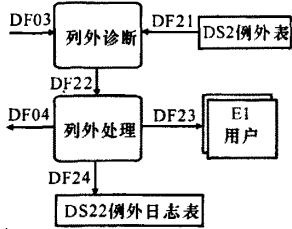


图3 例外处理模块数据流

该模块处理过程主要是例外诊断程序从调用程序读取例外代码,通过查询例外表后产生诊断结果,例外处理程序将诊断结果分解程序例外信息,用户通知信息和例外日志信息并分别输出给调用程序、用户及例外日志表。应用系统投产后发生异常情况通常是不可避免的,这样的处理流程可以使开发者迅速定位应用系统的异常位置,及时解决开发时编程人员考虑不周之处。该模块的规范化和标准化将使应用系统的稳定性显著提高。同时,该模块还可作为标准件运用于其他应用系统,提高开发效率。

### 2.2.3 功能平台模块

由于不同的应用系统的功能需求大不相同,因此,整个开发的大量工作就是高效地架构功能平台模块这个用户极为重视的应用系统功能平台。在大多数 MIS(管理信息系统)中,如信息的输入输出、统计分析及打印报表等常用功能均具有相当的通用性,因而即使在这个功能平台中也有许多模块可从知识函数库中共享,而开发的重点应放在如何创新地解决应用系统中遇到的新难题。创新的开发过程将使软件开发人员的水准和知识积累得以提高,这也将为软件开发机构的持续发展提供源源不绝的强劲动力。

功能平台模块该模块是本系统的功能控制中心,用于满足用户的业务需求,其中上层的初始化模块的作用是对整个系统的运行环境进行初始化设置;例外捕获模块用于捕获系统的例外事件和错误处理,提高系统运行的稳定性;功能控制模块的作用在于响应用户的功能调用请求,将用户的业务请求与不同的子功能相联系;这三个模块也具有通用性,略作修改即可重复使用。因此,系统真正的工作在于开发功能平台中的各子功能处理模块,它们分别对

应用户的具体业务需求,由于重复的开发工作相对减少,开发人员可以将精力集中于详细了解用户的业务需求,通过高质量、高效率地研发工作,最大限度地满足用户的需要。模块具体流程如图4所示。

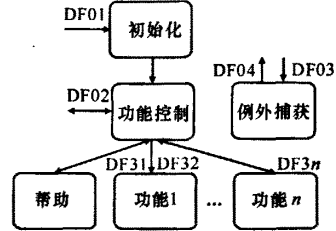


图4 功能控制平台

### 2.2.4 公共接口模块

公共接口是一个容纳了众多编程智慧结晶的成熟知识函数库,随着时间的推移,公共函数不断增多并逐步完善,它在应用系统整个开发过程中可重复利用的部份也越来越多。其比重在全部开发工作量中占越大,就表明开发工作周期越短、效率越高,更意味着软件开发风险的降低,软件开发机构的 CMM(软件能力成熟度模型)级别的提高,经济效益不言而喻。

通过公共接口形成一个与时俱进的公共函数库,积存着多种类型的函数模块,随着时间的推移,它不断积累封装开发人员的经验以便重用,同时又作为知识平台通过对团队成员的学习培训,形成软件开发机构金字塔形的人才结构,迅速提高开发人员的整体研发水平。

## 3 结束语

通过合理地设计模块化、规范化的总体架构,就可以极大地提高软件应用开发的效率,较好地解决构筑高效应用系统平台的架构问题,而且通过知识和经验的积累,又可以不断地丰富软件开发机构的知识宝库,加强核心竞争力。

### 参考文献:

- [1] 张友生. 系统分析师考试辅导[M]. 北京:清华大学出版社,2005.

(责任编辑:尹 闯)