

<<大规模系统构架建模及其开发技>>

图书基本信息

书名：<<大规模系统构架建模及其开发技术>>

13位ISBN编号：9787811240979

10位ISBN编号：7811240971

出版时间：2008-3

出版时间：北京航空航天大学出版社

作者：郭秋萍，焦允，王全兰 著

页数：200

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<大规模系统构架建模及其开发技>>

内容概要

从大规模系统构架建模的角度，研究基于构架 / 构件的系统开发方法及其相关技术。

通过研究统一建模语言（UML）与形式化体系结构描述语言（ADL）的融合，建立一种基于构架 / 构件的大规模系统开发模型。

主要包括：国内外研究现状；软件复用思想；系统构架理论与构架建模方法研究；软件构件的抽取、设计与实现方法；基于构架 / 构件的大规模系统开发模型；系统构架的评估等。

可作为高校与软件工程相关专业或从事软件体系结构研究的高年级本科生、研究生、教师等的参考用书，也可为业内相关技术人员提供参考。

书籍目录

第1章 引论 1.1 传统软件开发方法概况 1.1.1 瀑布模型 1.1.2 快速原型模型 1.1.3 螺旋模型 1.1.4 自动程序设计模型 1.1.5 增量模型 1.1.6 演化模型 1.2 传统的软件认识观 1.3 支持复用的软件开发概况 1.3.1 Parnas方法 1.3.2 面向对象的软件开发方法 1.3.3 可视化开发方法 1.3.4 基于构件的软件开发方法 1.4 基于构架 / 构件的软件新认识观第2章 软件复用的思想 2.1 软件复用的定义 2.2 软件复用的分类 2.3 软件复用的优点 2.4 软件复用的技术形式 2.5 复用要求改变过程 2.6 面向复用的软件开发方法 2.6.1 基于合成的方法 2.6.2 基于生成的方法 2.7 面向复用方法的比较与分析 2.8 实现软件复用的关键因素 2.8.1 技术因素 2.8.2 非技术因素 2.9 软件复用与面向对象思想 2.9.1 面向对象对软件复用的支持 2.9.2 软件复用对面向对象的支持第3章 系统构架理论 3.1 系统构架的起源与发展 3.1.1 系统构架研究的必要性和重要意义 3.1.2 系统构架的发展史 3.1.3 系统构架的主要研究方向 3.2 系统构架的基本概念 3.3 构架的重要意义 3.4 构架的构成要素 3.5 构架设计的元模型 3.6 系统构架的设计模型分析 3.6.1 工件驱动的设计模型 3.6.2 用例驱动的设计模型 3.6.3 领域驱动的设计模型 3.6.4 模式驱动的设计模型 3.7 系统构架的形式化描述 3.7.1 系统构架形式化描述的解决方案 3.7.2 系统构架描述和分析的要求 3.7.3 几种主要的系统构架描述语言 3.8 系统构架的形式化描述与UML的集成 3.9 ACME形式化描述语言 3.10 UML与ACME在系统构架建模中的应用 3.10.1 UML与ACME融合的可行性 3.10.2 UML与ACME融合的策略 3.11 系统体系结构风格 3.11.1 管道和过滤器风格 3.11.2 C2体系结构风格 3.11.3 基于事件的隐式调用风格 3.11.4 层次系统风格 3.11.5 C / S体系结构风格 3.11.6 平台 / 插件式构架风格 3.11.7 面向服务的构架风格 3.12 系统构架的发展方向 3.12.1 现存的不足 3.12.2 研究热点 3.12.3 发展方向第4章 构件技术研究 4.1 构件的定义 4.2 构件的特点与分类 4.3 构件模型 4.4 构件技术与面向对象技术的关系 4.5 构件与构架的关系 4.6 构件在软件复用中的适应性问题 4.7 构件的设计与实现原则 4.8 构件的获取和评选 4.8.1 自开发构件 4.8.2 商品化构件 4.9 构件的组装 4.9.1 构件组装的分类 4.9.2 几种构件组装技术第5章 基于构件的系统建模与设计 5.1 基于面向对象的构件分析与设计 5.2 面向构件的建模工具 5.2.1 UML的语义 5.2.2 UML的图形表示 5.3 UML的构件建模机制 5.3.1 用例图 5.3.2 包图 5.3.3 类图 5.3.4 交互图 5.3.5 实现图 5.4 基于用例的需求描述 5.5 基于UML的构件抽取 5.5.1 用例对象类的识别 5.5.2 对象类属性的识别 5.5.3 对象类方法的识别 5.5.4 对象类间关系的识别 5.6 基于对象的构件设计 5.6.1 确定子系统和类 5.6.2 确定用例类图 5.6.3 确定用例顺序图 5.6.4 类的设计 5.6.5 类属性的设计 5.6.6 类操作的设计 5.6.7 关系设计 5.6.8 类的优化 5.7 构件的实现 第6章 主流的构件实现规范 6.1 CORBA 6.2 J2EE / Ja, vaBeans / EJB 6.3 Microsoft COM / DCOM / COM+技术 6.4 三种构件实现规范比较 6.5 构件技术与中间件技术 6.6 基于J2EE规范的应用程序构件 6.6.1 客户层构件 6.6.2 Web层构件 6.6.3 业务层构件 6.7 支持构件技术的运行平台 6.7.1 BEA公司的WebLogic平台 6.7.2 IBM公司的WebSphere平台 6.7.3 Microsoft DNA 2000-7F台第7章 基于构架 / 构件的应用系统开发 7.1 软件生命周期中的系统构架 7.1.1 需求阶段的系统构架 7.1.2 设计阶段的系统构架 7.1.3 实现阶段的系统构架 7.1.4 部署阶段的系统构架 7.2 基于构架 / 构件进行系统开发的两种策略 7.2.1 基于领域构架的系统开发 7.2.2 基于构架 / 构件的系统开发策略 7.3 基于构架 / 构件开发大规模系统的方法模型 7.3.1 系统需求分析 7.3.2 基于ACME与UML融合的系统建模 7.3.3 构件抽取 7.3.4 构件设计 7.3.5 构件实现 7.3.6 订单管理子系统设计的主要构件列表 7.3.7 系统的组装与部署 7.4 系统建模的一致性保证 7.4.1 不一致性的分类 7.4.2 不一致性产生的原因 7.4.3 不一致性问题解决的方法第8章 系统构架评估 8.1 相关术语 8.1.1 质量属性 8.1.2 风险承担者 8.1.3 场景 8.1.4 评估技术 8.2 常用的构架评估方法 8.2.1 软件体系结构分析法SAAM 8.2.2 构架权衡分析法ATAM 8.2.3 基于场景的构架再工程法SBAR 8.2.4 构架层软件维护预测法ALPSM 8.2.5 构架可修改性分析法ALMA 8.2.6 体系结构比较分析法SACAM 8.2.7 基于Rapid描述语言的构架分析法 8.3 常用的构架评估支持工具 8.4 系统构架分析与评估方法的一些问题及讨论 8.4.1 存在的问题 8.4.2 构架分析评价方法的讨论参考文献

章节摘录

第1章 引论 信息化是整合大规模企业各种主要业务的重要手段之一，能有效降低企业运行成本，提高管理效率。

如今企业级应用系统正逐渐发展成在Intranet和Internet环境下的各种客户端可远程访问的分布式、多层次异构系统。

尤其在电子商务、Web信息系统等领域，网络技术已成为现代企业管理的基础，并越来越深入地影响其未来走向。

面向对象方法的成熟，虽然为软件开发方式带来了一场技术上的变革，但基于网络的计算环境又为系统开发提出许多新的课题。

它要求软件实现跨空间、跨平台、跨用户的共享，导致软件在规模、复杂度、功能要求的极大增长，需要软件生产走异构协同工作、多层次集成、可反复重用的工业化道路。

与此对应的是，传统软件开发过程不注重系统体系结构的总体规划，不注重利用以往软件开发所积累的知识和经验成果。

一方面导致系统体系结构不优化，开发成功率低，适应性不强，难以扩展和维护；另一方面导致在分析、设计、编码、测试等软件开发各阶段重复劳动，极大降低系统开发效率，而且由于未充分重用现有的高质量软件，容易引入各种错误，从而影响软件质量。

为解决传统软件开发中存在的一系列问题，一种基于构架/构件的系统开发思想被提出，是目前企业级应用系统建设的主流方向。

在过去几年中，已产生了许多支持构架、构件的软件开发方法或相关工具，如RUP开发过程、UML建模语言、Rational ROSE建模工具等。

本书从覆盖整个软件开发生命周期的角度，围绕软件开发的三个关键元素：过程、表示法和技术，探讨应该如何将其系统地加以应用，以一种系统化的方式有效地指导，并实现大规模软件的构件化开发。

。

<<大规模系统构架建模及其开发技>>

编辑推荐

《大规模系统构架建模及其开发技术》从大规模系统构架建模的角度，研究了基于构架 / 构件的系统开发技术。

全书共分8章，分别介绍了传统系统开发概况与基于构架的软件新认识观；软件复用思想以及面向复用的系统开发方法；系统构架理论；构件技术；基于构件的系统建模与设计；主流构件实现规范；基于构架 / 构件的大规模系统开发方法模型；系统构架的评估。

《大规模系统构架建模及其开发技术》的读者对象包括：与软件工程有关专业的本科生、硕士研究生、博士研究生，从事软件体系结构研究的学者或从事该领域相关研究的技术人员，以及其他对系统构架、软件构件感兴趣的读者。

<<大规模系统构架建模及其开发技>>

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>