

基于组件开发

Component-based Development

Principles and Planning for Business Systems

[英] Katharine Whitehead 著
王海鹏 沈华峰 译

IT Trend 系列丛书

基于组件开发

[英] Katharine Whitehead 著

王海鹏 沈华峰 译

人民邮电出版社

IT Trend 系列丛书

基于组件开发

- ◆ 著 [英] Katharine Whitehead
译 王海鹏 沈华峰
责任编辑 俞 彬
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
读者热线 010-67132705
北京汉魂图文设计有限公司制作
北京朝阳展望印刷厂印刷
新华书店总店北京发行所经销
- ◆ 开本: 720×980 1/16
印张: 14.5
字数: 263 千字 2003 年 9 月第 1 版
印数: 1-4 000 册 2003 年 9 月北京第 1 次印刷
著作权合同登记 图字: 01-2003-3179 号

ISBN 7-115-11559-1/TP • 3579

定价: 29.00 元

本书如有印装质量问题, 请与本社联系 电话: (010) 67129223

图书在版编目(CIP)数据

基于组件开发 / (英) 怀特黑德 (Whitehead, K.) 著; 王海鹏, 沈华峰译.
—北京: 人民邮电出版社, 2003.9
(IT Trend 系列丛书)

ISBN 7-115-11559-1

I. 基... II. ①怀...②王...③沈... III. 软件开发 IV. TP311.52
中国版本图书馆 CIP 数据核字 (2003) 第 080307 号

版 权 声 明

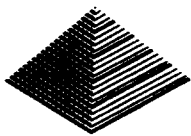
Copyright © 2001 by Unisys Corporation.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise without the prior consent of Addison-Wesley.

Published by arrangement with Addison Wesley Longman, Inc. All Rights Reserved.

本书中文简体字版由 Addison Wesley 出版公司授权人民邮电出版社出版, 未经出版者书面许可, 对书的任何部分不得以任何方式复制或抄袭。

版权所有, 侵权必究。

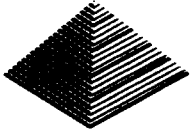


内容提要

本书介绍了基于组件开发的关键概念，要采用基于组件的软件开发模型，就需要理解这些概念。本书也解释了组织采用这种方式所带来的好处。它引导读者走过一个项目规划过程，列举了设计和组装组件时必须知道的一些事项。

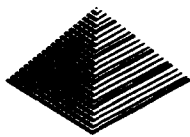
本书共四个部分，第一部分是基于组件开发的概述；第二部分提出一种项目规划的方法，可以极大地提高基于组件开发的有效性；第三部分介绍了组件构建与组装；第四部分是 CBD 案例研究。

本书适合于软件项目经理、设计人员和开发人员阅读参考。



作者简介

Katharine Whitehead 在软件行业已经工作了 20 多年，在此期间她参与了各类 IT 项目，包括面向对象的分析和设计方面的工作。1996 年，她开始成为一个 CBD 顾问，本书就是基于她与业界的客户在形成 CBD 方法过程中所积累的经验。目前她在 **Critical Path** 公司就职，该公司在网络通信方面是全球领先的企业。



出版者的话

在知识经济时代，IT 技术的每一次创新和发展总是带动着生产力巨大的进步和提高，也吸引着大量的优秀人才不断投身技术革命。作为 IT 行业从业者，不管你是经理人还是普通的工程技术人员，都需要敏锐的技术嗅觉、灵活的创新精神和旺盛的学习能力，以适应不断变化发展的技术趋势。

软件产业在 IT 产业中历来占有重要的地位，进入 21 世纪，这种地位继续得到巩固和凸现。纵观软件产业的发展历史，计算技术经历了“以机器为中心—以应用为中心—以企业为中心”的变化，同时，我们不难得出这样的结论：软件开发方法的进步有助于维持软件产品质量、全寿命期、生产成本的平衡。

以企业为中心的计算主要包括以下方面：

- 基于组件开发（component-based development）——将制造业中发展成熟的概念应用到软件开发中去。
- 设计模式（design patterns）——这也和制造业过程有共同之处。
- 中间件（middleware）——它在操作系统之上又迈出了提升计算平台抽象层次的一步。
- 说明性规约（declarative specification）——相对简单的说明取代了相对复杂的程序代码。
- 企业构架（enterprise architecture）——通过分离关注点来组织企业软件。
- 企业应用集成（enterprise application integration）——将孤立的遗产系统集成进企业系统的整体。
- 契约式设计（design by contract）——它有助于推动高质量的软件工程。

作为将这一系列新的趋势性技术整合到一起的开发方法，Model Driven Architecture (MDA) 的出现，为提高软件开发效率，增强软件的可移植性、协同工作能力和可维护性，以及文档编制的便利性指明了解决之道。也正因为如此，MDA 被面向对象技术专家预言为未来两年里最重要的方法学。

面对这一系列新的技术名词，你是否感到有些无所适从？或者你只是听说过，也许你对它们还略知一二，但相信大多数读者还没有系统地学习过相关的理论知

识。没关系，只要具备前面提到的 IT 从业人员的优秀素质，你肯定会对这些技术话题产生浓厚的兴趣，那么，你最需要的就是比较系统的技术书籍。

“IT Trend”系列图书是人民邮电出版社从 Addison Wesley 和 John Wiley 两大知名的国外专业出版公司引进、精心策划和出版的一套高技术定位、高制作品质的系列图书。图书内容定位覆盖了我们前边提到的那些 IT 趋势性技术，而且原书几乎都是全世界范围内第一本论述相关技术的专著。

这套图书的首批出版计划包括：

《IT 体系结构与中间件——建设大型集成系统的策略》(*IT Architectures and Middleware: Strategies for Building Large, Integrated Systems*)

《应用 MDA》(*Model Driven Architecture: Applying MDA to Enterprise Computing*)

《解析 MDA》(*MDA Explained: The Practice and Promise of The Model Driven Architecture*)

《Design by Contract 原则与实践》(*Design by Contract by Example*)

《基于组件开发》(*Component-based Development: Principles and Planning for Business Systems*)

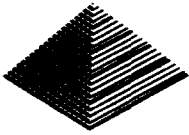
我们希望这套新技术图书能够为广大读者打开一扇窗口、提供一个机会，以了解那些正在和将要对 IT 产业发展产生重大影响的技术趋势，帮助你准确展望和把握行业的技术前景。我们期待“IT Trend”系列图书“一步领先”的精彩，能够带给你“步步领先”的收获。



致 谢

本书源自于我在 Seer Technologies 公司工作时，我的同事和我在组件方面所做的工作。我从 Jon Cogdale 那里受益颇多，在我们最初研究组件时，他是我的合作者。我在第 7 章的例子就是源自于他。Graham Berrisford 提出了许多有创意的意见，对概念进行了发人深省的讨论，第 12 章中关于订单处理的例子主要是得益于与他的讨论。在我的同事中，Doug Kime、Rob Allen、Tony Phillips、Bernard Londeix、Mark Gilbert 和 Martin Carter 也提供了许多有价值的建议。

从我们的客户与我们分享的经验中，从 David Sprott 和 Lawrence Wilkes 负责的 CBDi 论坛中，从这个领域的其他文献和工作中，我也学到了很多。



前 言

本书试图回答两个问题：我们使用组件期望达到什么目的？我们怎样定义和实现组件才能达到这个目的？本书希望能深入讨论基于组件开发（CBD）的特殊好处，以及成功的 CBD 所必须理解的一些概念问题。

本书的目标是：

- 解释 CBD 的基本原理以及隐藏在其后的关键概念；
- 提出一种项目规划的方法，可以极大地提高 CBD 的有效性；
- 提供在组织中引入基于组件方法的指导；
- 提供对设计组装组件所涉及的一些问题的个人观点，包括确定组件范围和管理组装的策略。

对 J2EE、.Net 等技术方面的深入探讨，或者提供一种完整的方法来强调建模技术的使用，并不是本书的目标。本书将假定您已经使用了基于模型的开发方法（最好是面向对象的，至少也是混合面向对象和面向过程的）。这种假定让本书能够专注于采用基于组件方法的一些特定问题，而不是讨论已经被广泛理解的软件开发技术。

CBD 常常被认为主要与新软件的开发有关。但是，软件的开发不会凭空产生。它必须考虑现有应用的组合，以及组织中现有的软件基础设施。出于这个原因，本书对项目规划给予了特别关注。在本书，CBD 和集成现有系统被看作是一对自然的搭档。软件开发被视为逐渐从现有的应用组合进化到改进的应用组合（更加组件化）的一个过程。CBD 提供了一种方法来实现这种渐近式的改进，而且基于组件的视图为我们描绘了应用组合最终看起来的样子，指导我们向这个目标前进。

读者对象

本书的目标是满足实践者的需要，特别是项目规划人员和架构设计人员的需要，他们可能负责将 CBD 引入组织。本书也适用于经理、设计人员和开发人员。

主要内容

本书包含以下部分和章节。

第一部分 基于组件开发概述

第 1 章 基于组件开发：尝试管理混沌

本章解释了 CBD 出现的业界背景，以及为什么 CBD 提供了一种方法来对待软件开发者所面临的冲突压力。

第 2 章 什么是组件

本章提供了组件的定义，并讨论了组件的特征。本章还讨论了组件所需的支持环境，并简要地列出了现存的一些组件模型，如 COM、CORBA 和 EJB 等的优缺点。

第 3 章 从整体观点来看组件

CBD 并不是有效的软件开发惟一要考虑的东西。本章讨论了 CBD 与其他开发技术和面向对象的关系，对 CBD 进行了定位。本章还讨论了 CBD 的局限性，以及不适用 CBD 的一些情况。

第二部分 规划过程

第 4 章 在组织中定位组件

本章讨论了 CBD 初始阶段的范围和由谁负责，以及它在组织中的位置。

第 5 章 软件架构和基础设施

本章讨论了对架构的需求，以及一个 CBD 架构需要涉及的方方面面。

第 6 章 定义组件

本章介绍了影响组件定义的原则：

- 粗粒度和细粒度的组件；
- 组件的类型以及如何让它们一起工作；
- 定义组件的范围。

同时本章还讨论了目前出现的一些标准。

第 7 章 实践基于组件开发

本章建立在前面三章的基础之上，列出了一组任务，用以创建一个初始 CBD 策略，随着时间推移不断地优化这个策略。

第 8 章 项目规划的实用研究

本章讨论了 CBD 策略作为项目规划的一部分，如何能够做到长期有效。这是一个在长期的远景、近期的需求和混乱的现实之间进行折衷的过程。

第 9 章 基于组件开发的组织

本章讨论了实现 CBD 规划的一些可选的组织结构。本章还讨论了策略规划的角色划分和团队结构，组件获取和组件组装。

第三部分 组件构建与组装

第 10 章 获取组件

本章介绍了关于获取组件与获取应用软件包不同的一些特定问题。

第 11 章 设计组件

本章讨论如何在设计组件时考虑灵活性——如何确定设计与复用之间的关系。

第 12 章 组装组件

本章讨论了让组件一起工作所引发的一些问题。

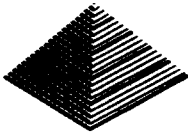
第 13 章 测试、认证并维护一个组件目录

本章列举了管理和维护一个组件组合时遇到的一些相关问题，包括让组件可以复用或销售所需要考虑的一些问题。

第四部分 案例研究

第 14 章 使用组件管理混沌

本章包含了在组织中使用 CBD 的一个简单（虚构的）案例研究，随着该组织引入了呼叫中心和基于因特网的系统，组织中现存的基于大型主机的应用正朝着基于组件的方式演进。



目 录

第一部分 基于组件开发概述	1
第 1 章 基于组件开发：尝试管理混沌	3
1.1 软件开发的问题	3
1.1.1 软件难以维护	3
1.1.2 应用集成的需要	4
1.1.3 表示方式的改变	7
1.1.4 技术的变化	7
1.1.5 重复不必要的工作	7
1.2 改进软件开发方面的进展	8
1.2.1 提高产能的工具	8
1.2.2 复用	11
1.2.3 灵活性	13
1.2.4 企业范围的信息模型	13
1.3 基于组件开发的承诺	14
1.4 基于组件开发的可行性	15
第 2 章 什么是组件	17
2.1 定义组件	17
2.1.1 组件是预先构建的	17
2.1.2 组件是黑盒，只能通过其接口访问	19
2.1.3 组件是可分离的	19
2.1.4 组件组装和部署	19
2.1.5 组件上下文环境	20
2.1.6 支持技术	20
2.1.7 组件的定义	20

2 基于组件开发

2.2	组件粒度	20
2.3	组件环境	22
2.3.1	互操作性	22
2.3.2	不同技术之间的互操作性	22
2.3.3	组件模型	23
2.3.4	包装现有的应用	24
2.3.5	平台无关性	25
2.3.6	服务提供	25
2.4	组件和它们在整体中的位置	26
2.5	部署组件	27
2.5.1	定制和配置	27
2.5.2	组装: 把组件粘在一起	27
2.5.3	处理改动	29
2.6	组件模型	30
2.6.1	COM 和 MTS	31
2.6.2	CORBA	31
2.6.3	Enterprise JavaBeans 和 JavaBeans	32
2.7	组件规范	33
第 3 章	从整体观点来看组件	35
3.1	与基于模型开发的关系	36
3.2	需求规格和设计复用	37
3.3	与面向对象技术的关系	38
3.4	软件基础设施和开发自动化	39
3.5	区分不同类型组织的需求	39
3.6	基于组件开发不适用之处	40
3.7	基于组件开发的局限性	41
第二部分	规划过程	43
第 4 章	在组织中定位组件	45
4.1	基于组件开发初始阶段的范围和负责人	45
4.2	跨越组织部门协调软件	46

4.2.1	使软件结构与组织相符	47
4.2.2	变更管理	47
4.2.3	使用公共的软件解决方案与复用组件	48
4.3	内部市场模型	48
4.3.1	为何与内部市场有关	49
4.3.2	内部市场模型要关注的问题	50
第 5 章	软件架构和基础设施	53
5.1	架构考虑的两个方面	53
5.1.1	技术架构	54
5.1.2	应用架构	55
5.2	技术环境问题	56
5.2.1	可以互操作的组件的图景	57
5.2.2	实际中的组件互操作	61
5.3	软件基础设施问题	63
5.4	应用架构问题	65
5.4.1	可分离性与架构领域的定义	65
5.4.2	不同领域的组件之间的交互	67
5.4.3	分层的可选项	67
5.4.4	使用事件来控制点到点的通信	68
5.4.5	在实践中使用分层	70
第 6 章	定义组件	71
6.1	分层的架构	71
6.1.1	业务逻辑层	72
6.1.2	表示层	73
6.1.3	数据访问层	74
6.1.4	分层与 workflow	74
6.1.5	事务边界	77
6.1.6	分布问题	77
6.1.7	分层架构和 J2EE	78
6.2	组件类型	80
6.2.1	业务组件	80
6.2.2	应用组件	83

6.2.3	数据访问组件	84
6.2.4	工具组件	84
6.3	包装与适配器	84
6.3.1	包装	84
6.3.2	适配器	86
6.4	定义组件范围	87
6.4.1	业务组件	87
6.4.2	应用组件	92
6.4.3	数据访问组件	94
6.5	组件规格说明	94
6.5.1	组件接口基础	95
6.5.2	客户组件与组件组装	95
6.5.3	组件规格说明	96
6.5.4	定义服务	96
第7章	实践基于组件开发	99
7.1	定义基于组件开发初始阶段的范围和预期的好处	100
7.1.1	采用基于组件方式的关键驱动力	100
7.1.2	不要采用基于组件开发的情况	101
7.1.3	采用组件的初始阶段的范围	101
7.2	定义组件蓝图	101
7.2.1	确定业务过程	102
7.2.2	创建一个业务对象模型	102
7.2.3	形成一个组件蓝图	103
7.3	审查当前系统	107
7.3.1	创建当前系统的详细目录	107
7.3.2	评估组件化的恰当性	108
7.4	建立一个架构框架	109
7.4.1	技术环境	109
7.4.2	软件基础设施和开发标准	111
7.4.3	组件模型	112
7.5	组织结构上的考虑	112
7.6	项目规划	113

第 8 章 项目规划的实用研究	115
8.1 应用集成和更新	115
8.1.1 为什么还要保留遗留应用	115
8.1.2 如何管理遗留应用	116
8.2 集成软件包	119
8.3 外包	120
第 9 章 基于组件开发的组织	121
9.1 组织基于组件开发	121
9.1.1 优点示范中心	121
9.1.2 将组件开发与组装分离	122
9.1.3 基于应用项目的组件开发	123
9.1.4 技术架构和基础设施	124
9.2 组织规划	124
9.3 组织组件获取和组装	125
第三部分 组件构建与组装	127
第 10 章 获取组件	129
10.1 筛选候选组件	130
10.1.1 寻找组件	130
10.1.2 初选候选组件	130
10.1.3 正式评估	130
10.1.4 拜访参考顾客	132
10.1.5 选择一个候选对象	132
10.2 评估符合程度并确定改动之处	133
10.2.1 评估符合程度	133
10.2.2 确定改动之处	135
10.3 获取组件	136
第 11 章 设计组件	139
11.1 为复用而设计	140
11.1.1 为复用而通用化	141