

ORACLE®

Oracle 技术系列丛书

# Oracle Designer

## 信息系统开发

Oracle Designer  
Handbook

(Second Edition)

(美) Peter Koletzke 著  
Dr. Paul Dorsey

刘晓霞 孙登峰 等译  
曲京 何粼



机械工业出版社  
China Machine Press



McGraw-Hill



Oracle技术系列丛书

# Oracle Designer

## 信息系统开发

(美) Peter Koletzke 著  
Dr. Paul Dorsey

刘晓霞 孙登峰 曲京 何 粼 等译

钟 鸣, 何 燕 华, 廖 校



机械工业出版社  
China Machine Press

本书论述如何利用Oracle Designer来建造大型信息系统。书中将作者丰富的系统开发经验与Oracle Designer工具进行了有机的结合。本书论述透彻、见解独特、深入浅出、举例丰富,不仅是一本Oracle Designer的入门书籍,而且适合大、中型信息系统的组织者、设计者和实施者使用,还可以作为本科生和研究生的教材或参考书。

Peter Koletzke and Dr. Paul Dorsey: Oracle Designer Handbook, Second Edition.

Authorized translation from the English language edition published by McGraw-Hill.

Copyright © 1998 by McGraw-Hill.

All rights reserved.

本书中文简体字版由机械工业出版社出版。未经出版者书面许可,不得以任何方式复制或抄袭本书的任何部分。

版权所有,翻印必究。

**本书版权登记号: 图字: 01-99-0055**

### **图书在版编目(CIP)数据**

Oracle Designer信息系统开发/(美)柯莱兹克(Koletzke, P.), (美)多塞(Dorsey, D. P.)  
著;刘晓霞等译.-北京:机械工业出版社,1999.8  
(Oracle技术系列丛书)  
书名原文:Oracle Designer Handbook, Second Edition  
ISBN 7-111-07298-7

I. O... II. ①柯... ②多... ③刘... III. 数据库系统-软件工具, Designer IV. TP311.56

中国版本图书馆CIP数据核字(1999)第20826号

出 版 人: 马九荣(北京市百万庄大街22号 邮政编码 100037)

责任编辑: 吴 怡

北京忠信诚胶印厂印刷·新华书店北京发行所发行

1999年8月第1版第1次印刷

787mm × 1092mm 1/16 · 34.5印张

印数: 0 001-4 000册

定价: 68.00元

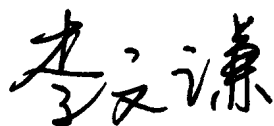
凡购本书,如有缺页、倒页、脱页,由本社发行部调换

# 贺 辞

祝贺中文版 ORACLE 8 系列丛书的出版。希望她能为中国的广大 ORACLE 用户和对数据库技术感兴趣的读者提供最先进的 ORACLE 技术知识。

ORACLE 软件系统有限公司希望通过中文版 ORACLE 8 系列丛书的出版,更好地建立起与广大用户和数据库工作者之间技术交流的桥梁。

Oracle 中国有限公司总经理

A handwritten signature in black ink, appearing to read '李文谦' (Li Wenqian), written in a cursive style.

1998.6

## 序

数据库技术的发展,使它已经成为现代信息技术的重要组成部分,是现代计算机信息系统和计算机应用系统的基础和核心。可以说,如果没有数据库技术的发展,没有优秀的数据库产品的推出和应用,社会信息化的进程将是难以实现的。因此,在衡量一个国家信息化的程度时,其数据库的建设规模、数据库信息量的大小和使用程度也就成为重要的标志之一。

我国引进数据库技术始于70年代末,从微型计算机上运行的数据库到当前的大型数据库系统的引入和应用,已经有20多年的历史。20多年来,虽然在微型计算机数据库知识的普及和应用上取得了很大的进展和成绩,在大型数据库系统的开发和应用上也取得了进步,甚至有了国产化的数据库软件,但如果从对数据库系统的应用效果和对数据库技术的掌握上来比较,则与发达国家之间仍然存在较大的差距,特别是在大型数据库系统的开发、建设和应用水平差距更大。这种差距主要表现在两个方面:第一是数据库的数量及其所收集信息的数量和质量与发达国家相比存在很大的差距;第二是对数据库技术和产品的掌握和应用上更有待于提高和加强,应用人才急需培养,经验有待积累和总结。前者的改善应依靠于对数据库应用基础工作的加强,如重视基础数据的收集和整理,即重视数据工程的建设,并制定相应的数据政策;而后的改善则会更多地依赖于人们对数据库技术和对数据库产品的掌握。为此,我们必须首先占有充足的资料并加以消化。如果数据库厂家和出版机构能提供较完整的、质量较高的技术资料 and 书籍,并为较多的数据库技术人员和应用人员所掌握,无疑将会促进国内数据库技术人才的成长并推动数据库应用水平的提高。

喜闻机械工业出版社华章公司与ORACLE出版公司合作,为配合ORACLE 8在中国的发行,由机械工业出版社买断了ORACLE出版公司出版的ORACLE 8系列丛书的中文简体字版的出版权,并组织国内从事ORACLE应用开发的科技人员和教学人员进行翻译出版,还邀请国内数据库专家对译稿进行了审定,以保证丛书在技术上的权威性。无疑,这对大型数据库系统特别是ORACLE数据库系统的开发和应用将起到很好的推动作用。

众所周知,ORACLE公司推出的ORACLE 8是一种面向网络计算的数据库(the database for Network Computing)并支持对象关系模型的数据库产品。系列丛书全面地介绍了ORACLE 8的功能和技术,现简介如下:

- 《ORACLE 8初学者指南》概述了ORACLE 8的功能、结构和各种管理和开发工具。
- 《ORACLE 8数据库管理员手册》提供了高效、高质量发挥和开发数据库的技术。
- 《ORACLE 8 PL/SQL程序设计》详细介绍了ORACLE 8 PL/SQL语言的编程技术。
- 《ORACLE 8数据库构造工具实用指南》介绍了PL/SQL、SQL \* Loader、Developer/2000等ORACLE数据库开发工具的使用方法和技巧。
- 《ORACLE 8参考大全》是内容丰富的数据库应用系统设计和开发的参考手册。
- 《ORACLE 8优化技术》介绍优化ORACLE应用系统的方法、经验和技巧。
- 《ORACLE 8备份与恢复手册》介绍ORACLE的体系结构、备份和恢复的原理和方法。
- 《ORACLE 8故障解决手册》是ORACLE预防性维护和故障检测的指南。

- 《ORACLE Designer信息系统开发》系统介绍ORACLE的CASE工具Designer/2000, 包括系统分析与建模、系统设计以及数据结构和应用程序的自动生成技术。

- 《ORACLE Financial参考手册》介绍ORACLE 8的Client/Server的企业级财务管理系统。

由书目可见, 丛书的内容丰富, 涵盖了大型数据库应用开发中的全部技术内容, 有的资料, 如《Oracle故障检测和维护》以及《Oracle 8网络技术》等都是在国内首次面世, 很值得数据库技术工作者参考和阅读。

无疑, 这套丛书应该有广泛的读者, 它可供大型数据库系统, 特别是ORACLE系统应用的系统管理员、应用程序员、系统分析和设计人员和广大用户学习和参考, 也可做为ORACLE系统培训的教材和高等学校本科和研究生的学习参考书, 同时也是掌握大型数据库系统理论和实践的好材料。

应感谢机械工业出版社的领导和组织翻译丛书的何江华先生以及全体翻译的先生们和女士们, 辛勤的劳动将为我国信息化事业的发展增加动力。ORACLE中国有限公司对丛书的出版给予极大的关注与支持, 李文谦总经理并为丛书的出版题词。

预祝丛书的出版和发行获得成功, 并得到读者的欢迎。

罗晓沛

1998年5月18日

中国科学院研究生院 教授  
ORACLE大学(中国) 校长

## 译者序

本书是一本利用Oracle Designer进行信息系统分析与设计的书籍，根据《ORACLE Designer Hand book》第2版译出。书中将系统开发方法与Oracle Designer进行了有机的组合，将Oracle Designer的介绍融入了方法的讨论之中。

Oracle Designer作为一种功能很强的软件，包含数十种系统开发工具和实用程序。在学习和使用这些工具和实用程序时，我们会感到非常零乱，不易掌握。许多工具或实用程序在功能和外观上极为相似，但为什么不将它们组合在一起呢？还有一些工具或实用程序令人觉得莫名其妙，它们是用来干什么的呢？在信息系统开发过程中Oracle Designer对我们有何帮助？等等。

本书作者以信息系统开发与设计的生命周期方法为主线，结合自己多年的工程实践，详细地介绍了Oracle Designer的每个工具或实用程序的功能、用途和一些使用技巧，以及应该注意的事项。本书的组织方法基本是，每个开发阶段由两章组成，前一章讲述特定的开发阶段的理论，后一章讨论用于该开发阶段的工具或实用程序。这样的组织方法是很有特色的，既介绍了方法，又给出了实施方法所必须的工具，这在其它书中是不常见的。

由于作者具有丰富的信息系统工程经验，书中对生命周期方法的介绍极有特色，其介绍不是采用某种学究式的方式，而是以一种工程的观点进行。譬如，作者就各个开发阶段针对不同规模的信息系统都提出了相应的策略。又譬如，实际的信息系统开发中，很多时候并不是从头开始，有时候可能是对原有系统的改造，有时候可能是从半途接手等等。对于这些情况，作者都给出了很好的方案。在书中，作者对信息系统开发方法提出了很多高见，例如作者在第1章中说，“系统设计项目成功的关键在于几个成功因素的相互配合，它们分别是恰当的人选、合适的技术和工具、正确的方法。”这确实是经验之谈。

由于篇幅所限，这里对书中精辟之处不再一一列举。相信读者阅览本书之后一定会很有收获。本书的内容由浅入深、详尽透彻，不仅可作为了解和掌握Oracle Designer 工具和实用程序的入门书籍，而且特别适合大、中型信息系统的组织、设计、实施人员使用，还可以作为本科生、研究生的信息系统分析与设计的教材或参考书。

参加本书翻译的人员有：刘晓霞、孙登峰、曲京、何艮、郝玉杰、张南、文卫东、田晓涛、徐海琛、张维琪、刘向军、陈登科、汪民红、梅刚、王联华、李晓军、赵燕萍、徐毅、王君、魏允韬、石永平、张文、钟鸣、何江华审校。

由于译者水平有限，难免有错误或不当之处，敬请读者批评指正。

1999年3月

# Oracle Designer内容概览

下面列出了主要的Oracle Designer工具和实用程序:

| 工具名称                                                  | 描述章节          |              |        |      |
|-------------------------------------------------------|---------------|--------------|--------|------|
| Application Design Transformer(应用设计转换器)               | 2.2.6         | 12.4         |        |      |
| Application Programmatic Interface(应用程序接口)            | 2.2.7         | 28.1         |        |      |
| C++ Generator(C++生成器)                                 | 2.2.6         | 14.4.2       |        |      |
| Capture Application Logic(捕获应用逻辑)                     | 18.3.7        |              |        |      |
| Capture Design of Form(捕获窗体设计)                        | 2.2.7         | 12.7, 18.3.7 |        |      |
| Capture Design of Report(捕获报表设计)                      | 2.2.7         | 12.7, 18.6.4 |        |      |
| Capture Design of Library(捕获库设计)                      | 2.2.7         | 18.4.3       |        |      |
| Capture Design of Server Model(捕获服务器模型设计)             | 2.2.6         | 8.3.1        | 4.3.9  |      |
| Capture Design of Visual Basic(捕获Visual Basic设计)      | 2.2.7         | 18.7.4       |        |      |
| Create Default Links(创建缺省连接)实用程序                      | 16.2.8        |              |        |      |
| Create Default Module Data Usages(创建缺省模块数据用法)         | 16.2.7        |              |        |      |
| 实用程序                                                  |               |              |        |      |
| Create Function/Attribute Matrix(创建功能/属性阵列)实用程序       | 8.2.3, 10.1.6 |              |        |      |
| Database Administration Guide(数据库管理向导)                | 14.2.11       |              |        |      |
| Database Design Transformer(数据库设计转换器)                 | 2.2.5         | 12.3         |        |      |
| Database Navigator(数据库浏览器)                            | 14.5          |              |        |      |
| Dataflow Diagrammer(数据流图示化工具)                         | 2.2.4         | 8.2          | 10.3   |      |
| Design Editor(设计编辑器), 一般技术                            | 2.2.6         | 12.1         |        |      |
| Entity Relationship Diagrammer(实体关系图示化工具)             | 2.2.4         | 4.2          | 4.3.3  | 10.1 |
| Form Generator(窗体生成器)                                 | 2.2.6         | 18.3         |        |      |
| Function/Attribute Matrix(功能/属性阵列)                    |               |              |        |      |
| 参见Create Function/Attribute Matrix                    |               |              |        |      |
| Function Hierarchy Diagrammer(功能层次图示化工具)              | 2.2.4         | 10.4         |        |      |
| Generate Database Administration Objects(生成数据库管理对象)   | 18.1.2        |              |        |      |
| Generate Database from Server Model<br>(从服务器模型中生成数据库) | 2.2.6         | 18.1         | 20.1.3 |      |
| Generate Database from Type Model(从类型模型中生成数据库)        | 14.4.2        |              |        |      |
| Generate Module Component API(生成模块组件API)              |               |              |        |      |
| Generate Reference Code Tables(生成引用模块表)               | 16.5.1        |              |        |      |
| Generate Table API(生成表API)                            |               |              |        |      |
| Generator Preferences(生成优选项)                          | 2.2.6         | 12.2         |        |      |
| Help Generator(帮助生成器)参见MS Help Generator              |               |              |        |      |
| Help System(帮助系统)                                     | 2.2.3         | 26.4         |        |      |
| Library Generator(库生成器)                               | 2.2.6         | 18.4         |        |      |



(续)

| 工具名称                                              | 描述章节    |             |        |        |       |
|---------------------------------------------------|---------|-------------|--------|--------|-------|
| Logic Editor(逻辑编辑器)                               | 2.2.6   | 14.3        |        |        |       |
| Matrix Diagrammer(阵列图示化工具)                        | 2.2.7   | 10.5        | 16.7   |        |       |
| Module Application Guide(模块应用向导)                  | 16.4    |             |        |        |       |
| Module Component API(模块组件API)                     | 16.3    |             |        |        |       |
| Module Diagram(模块图)                               | 2.2.6   | 16.2        |        |        |       |
| Module Network Viewer(模块网络浏览器)                    | 2.2.6   | 16.1        |        |        |       |
| Module Structure Diagrammer(模块结构图示化工具)            | 16.1.1  |             |        |        |       |
| MS Help Generator(MS Help生成器)                     | 2.2.6   | 18.8        |        |        |       |
| Object Database Designer(对象数据库设计工具)、一般技术          | 2.2.6   | 14.4.2      |        |        |       |
| Oracle Designer window(Oracle Designer窗口)         | 2.3     | 2.2.7, 26.4 |        |        |       |
| Oracle Designer Help System(Oracle Designer 帮助系统) |         |             |        |        |       |
| 参见Help System                                     |         |             |        |        |       |
| Process Modeller(过程建模器)                           | 2.2.4   | 4.1         | 8.2.1  | 10.2   |       |
| Reconcile Report(调和报表)                            | 20.1.3  |             |        |        |       |
| Report Generator(报表生成器)                           | 2.2.6   | 18.6        |        |        |       |
| Repository Administration(信息仓储管理)实用程序             | 2.2.7   | 27.1        | 26.2   |        |       |
| Repository Object Navigator(信息仓储对象浏览器)            | 2.2.7   | 6.1         | 12.8   | 12.8.1 | 26.1  |
|                                                   |         |             |        |        | 6.1.4 |
| Repository Reports(信息仓储报表)                        | 2.2.7   | 6.2         |        |        |       |
| Reusable Component Graphical Editor(可重用组件图形编辑器)   | 16.2.5  |             |        |        |       |
| Server Model Diagram(服务器模型图)                      | 2.2.6   | 14.1        | 14.4.2 |        |       |
| Server Model Diagrammer(服务器模型图示化工具)               | 14.1.1  |             |        |        |       |
| Server Model Guide(服务器模型向导)                       | 14.2.11 |             |        |        |       |
| Table API(表API)                                   | 14.2.9  |             |        |        |       |
| Table to Entity Retrofit(表对实体改进)实用程序              | 8.3.1   |             |        |        |       |
| Type Diagram(类型图)                                 | 2.2.6   | 14.4.2      |        |        |       |
| Update Attributes in a Domain(域属性更新)实用程序          | 10.1.5  |             |        |        |       |
| Update Columns in a Domain(域表列更新)实用程序             | 14.2.3  |             |        |        |       |
| User Extensibility(用户扩展)                          | 2.2.7   | 27.1        | 6.3.1  | 8.4    |       |
| Visual Basic Generator(Visual Basic生成器)           | 2.2.6   | 18.7        |        |        |       |
| WebServer Generator(WebServer生成器)                 | 2.2.6   | 18.5        |        |        |       |

# 引 言

## 传统的系统开发方法

传统系统开发方法有12个步骤:

- 1) 命名一个贴切的简称并设计出适合于该简称的基于关键任务(Mission-critical)的系统。
- 2) 获得建造该系统的一笔费用,然后将这些费用转去支付其它超出预算的项目。
- 3) 启用一些新手,因为高级程序设计人员都比较繁忙。
- 4) 雇用一些经验不多的咨询员,因为他们所需费用最低。
- 5) 不必费心去做系统需求,因为用户需要什么你已胸有成竹。
- 6) 煽起咨询人员的热情,因为他们效率太低了。从昂贵的咨询公司雇用一支新的队伍。
- 7) 让每个开发者设计出自己负责的那部分数据库,因为只有他们自己才确切地知道需要什么样的数据表。
- 8) 以MS Access方式做出自己的原型,即使你打算用Developer Web Forms 进行系统开发。
- 9) 赶快开始编码,因为项目进展已经比预期的慢了。
- 10) 不必费心去编写系统文档或用户手册,因为没人会看这些资料。
- 11) 让开发者进行用户认可测试,因为只有开发者才真正了解系统应该如何运行。
- 12) 写几份含混不清、带有误导性的项目现状报告,使用户对项目有点幻想的余地。然后再写一份系统工作需求,说明你实际完成的是什么。回顾一下项目的开发过程,宣布大功告成。

这个设计和建造自动化信息系统的过程极大地受到过程中所用的软件工具的影响。因此,系统分析员、设计者和开发者转向了Computer-Aided Software Engineering(CASE, 计算机辅助软件工程)软件,以获取相关的业务需求,设计出能满足这些需求的数据结构并生成前端和服务端程序代码。因为CASE工具自动化了系统开发所需要的大量手工的、重复的、易错的工作,所以只要使用得当的话,它们不仅能够大大提高系统研制人员的效率,而且还能够增加设计的准确性和系统的可靠性。现在,这些工具已成为建造系统的重要的有机成份,因此要把它们与系统建造过程分离开来几乎是不可能的。

Oracle Designer是Oracle公司的CASE产品,现在的版本号为2.1,它代表了Oracle公司的最新成就,能支持传统的(换句话说,就是所谓的系统开发生命周期)方法的所有阶段。Richard Barker在其有关Oracle CASE的几本论著中提出了这样一些方法,这些论著还给出了系统开发生命周期方法的很精彩的综述。不幸的是,这些书常常被开发者所误用,他们主观地、错误地盲从于这些论著,再结合Oracle Designer产品,形成了一整套开发方法学。其实这些论著压根就不曾打算成为完整的方法学指南,因为开发生命周期中的每一个步骤本身就值得写一本专著。

## 本书的系统开发方法

本书充分结合Oracle公司的Oracle Designer产品,提出了一种重要的修正了的开发方法。其目的是给出软件研制的每一个实际步骤以及方法学意义上的各个阶段,此外还说明了其中Oracle

Designer工具如何提供支持的。这里的工作有两个目的，首先是修正和扩充CASE方法。相对于最先提出CASE的那些书籍来说，这里的工作更为细致，增加了很多提交(Deliverables)，并且除了探讨Oracle Designer v.2.1外，还重新构造了整个开发过程。其次是描述了怎样才能最佳利用Oracle Designer来支持我们提出的修正CASE的方法。

我们称这个修正了的方法为CASE Application Development Method(CADM, CASE应用开发方法)。本书对传统的System Development Life Cycle(系统开发生命周期)方法的书籍以及传统的软件产品的书籍进行了扬弃并综合了这两类书籍的内容。说本书是“论述如何建造系统”或“论述怎样使用Oracle Designer”的书籍都是不全面的。合适的说法是，“这是一本关于如何利用Oracle Designer来建造系统的书籍”。书中将丰富的系统开发实践经验与目前市场上能得到的最好的集成化CASE工具——Oracle Designer进行了有机的融合。

本书的主要目的之一是给出一种开发过程，它能根据业务的原始资料，特别是在分析中所了解到的内容，得出业务的需求以及在系统中最终实现的路径。我们希望有一个规范化的过程，在工程结束的前六个月就能通过与用户交谈所搜集到的资料预见到用户在工程结束时才能提出的问题，例如，用户在工程结束时指出需要某个逻辑处理，并说，“我们在系统中需要这个功能，但没有。可是你已完成了此项功能。它在哪儿？”。为了避免出现这种吓人的问题，我们的整个开发过程都应该以业务需求驱动。本书论述了方法学，讲述了在System Development Life Cycle(系统开发生命周期)的过程中应该如何思考、交谈，应该如何做。

本书另一个目标是为开发过程的每一个实际阶段的方方面面提供一种高质量的保证。当然，如果你有更好的应用开发方法的话，不理睬本书中所介绍的内容也没什么关系。CADM成功的关键在于确保每一个阶段在进行下一个阶段前都已正确地完成。

虽然本书提出了传统“瀑布(Waterfall)”式生命周期法的一个扩充了的变种，但我们也认识到，还有很多其它成功使用过的有效的方法。不过，我们发现，这些可选用的方法与瀑布方法具有许多相同的特征、相同的案例和相同的阶段，尽管这些阶段可能进行了扩展或压缩。因此，这里提出的技术在某种意义上是传统的，但是，完善的方法应该与其它方法互相可以移植。这种观点我们认为是经得起论证的。

令人激动的是，Oracle Designer 支持这些可供选用的方法以及本书所介绍的CADM。但是，关于它如何支持这些不同的方法却很复杂。我们所做的一切系统开发工作都是基于如何明智地使用正在建造的或维护的构造合理的系统中心信息仓储。而这正是Oracle Designer的优秀之处，也正是使它成为一个适合于每种设计和开发环境的理想工具的原因。

本书还包含怎样处理Oracle Designer使其能够支持各CADM阶段，以及如何支持的内容。书中对有效地利用Oracle Designer的各种工具进行了详细的介绍，并说明在每个阶段要重视收集哪些信息，这些内容在别的资料中是看不到的。本书还指出了在开发过程的每一个步骤中如何找到这些信息。在上述这些内容的介绍和对方法本身的讨论中，常常穿插一些如何高效地利用工具的提示和技术。

## 关于第2版的说明

自本书出了第1版后，在我们的开发环境中已经有了两个重要的变化。首先是Oracle Designer已经成长成为一个软件产品。其用户界面的质量已经得到了极大的改善，能够生成代码。其次，在最近几年中，我们已经改变了某些关于系统开发的观念。

现在我们认识到,开发阶段并非像我们原来所认为的那样是离散的或不连续的。事实上,许多活动和提交并不完全属于某个特定的阶段,而是横跨整个系统开发生命周期的。在特定的情况下,我们甚至会抽出项目的某个部分,将其一口气做出来,以使开发的效果更为显著。正如“自顶向下”的开发方法(传统的SDLC模型)的反方法“自底向上”方法(一种更为面向RAD的方法)一样,这是一种“中间开花(Middle-out)”的开发方法,也是一种很值得重视的形式方法。

与系统开发类似,本书也处于不断完善之中。在这一版中,由于我们的观念已经发生了变化,所以增加了一些章节,论述我们认为在第1版中遗漏了的问题。例如数据移植和使CADM适合于某种快速的开发环境,以及对开发阶段的某些方面进行重新认识等等。还有许多东西需要我们去学习。正如我们在第一版中所论述的那样,工具极大地影响方法。这已成为CADM背后的核心原理之一。在这个后继的版本中,它也将无可置疑地会对我们的论述产生重大的影响。

尽管已经大大地增加了本书的长度,但是就如何利用Oracle Designer 来进行有效的系统开发这个论题来说,本书的论述还是显得有点粗糙。许多章节的内容完全值得写专门的论著。事实上,已经有人这么干了,可参看Oracle Press的Oracle8 Designer Using UML Object Modeling中的Database Design(Dr. Paul Dorsey&Joseph Hudicka,1999)和其它的Oracle Press的书籍,特别是关于Oracle Designer生成器的书籍。此外,还可以很容易地就Analysis(分析)、Testing(测试)、Application Design Standards(应用设计标准)(包括GUI和编码标准)、Data Migration(数据移植)以及WebServer Generation(Web服务器生成器)这些论题写出不少的书来。

## 本书的结构

本书分为四个部分。第一部分有两章,分别综述了CADM方法和Oracle Designer。第二部分(第3章~第21章)提供了本方法每个阶段以及支持这些阶段的Oracle Designer工具和实用程序的详细介绍。每个阶段都在一章或者两章中进行讨论,虽然Test(测试)、Implementation(实现)、Maintenance(维护)安排在一起介绍,但是用两章讨论了这些内容。关于Oracle Designer的资料都融入了方法的讨论之中,并且都是从安装过程结束时选取出来的。介绍每个阶段的章节后所跟的章节介绍支持该阶段活动的相应Oracle Designer工具。

我们在本书中提出的CADM生命周期法从一个阶段到另一个阶段以顺序方式给出。然而,我们认为,系统开发并不是如此简单的、线性的过程。项目的范围在任何时候都可能会产生变化。在进程的任何时刻都可能会发现新的用户需求,甚至有的会在测试阶段发现。此外,质量控制检测也可能失败。通常,现实的东西总会超出我们的理论预期。因此,第21章讨论了如何对预料之外的问题进行处理。

第三部分是这一版中的新内容,讨论超出传统SDLC内容的论题。第22章给出针对中小规模项目的一种Rapid Application Development(RAD,快速应用开发)方法的改进方法。第23章讨论对已经开始的项目需要“从中间开始”时,如何更改CADM步骤。第24章讨论Business Process Reengineering(业务过程重构),第25章介绍常被忽略、但对项目的成功又具有关键性作用的一个论题——数据移植。

第四部分介绍几个Oracle Designer的功能和实用程序,它们在开发方法的各个阶段都要用到。第26章略述了应用系统和信息仓储管理,第27章介绍Oracle Designer的User Extensibility(用户扩展)选项。第28章介绍Application Programmatic Interface(应用程序接口)。第29章论述关于信息仓储内信息如何流动,某个阶段的元素定义如何拷贝到另外阶段的所有重要的论题。

因为许多Oracle Designer工具和实用程序不止在一个阶段使用,所以特定工具的介绍可能会分布在不同的章节。如果读者想了解本书中某个特定的工具或实用程序在何处介绍,可以参看紧跟在目录后的“Oracle Designer主要工具和实用程序概览”。

### 样例数据模型

在本书的Oracle Designer章节中,有一些基于一个称为CTA(Computer Training Associates)的学生注册追踪系统例子,CTA也是一个假设的公司名称。这些例子并不采用案例研究方法,因此其业务规则和实现细节并非是介绍的重点。不过对CTA系统的功能进行一个简略的介绍还是很有用的。

CTA系统追踪学生按计划报到的情况、班级学生登记、课程研修情况等等。它管理教师为课程安排的课外作业、实验等等。它还存放学生和教师的简介信息,并能够根据学生完成学业情况的登记信息和评分报告给出单据。

该系统中各主要实体由一个存放其信息的表来描述,信息分别为:STUDENTS、INSTRUCTORS、COURSES、SECTIONS、ENROLLMENTS、WORK\_GRADES。因为学生的数量相当多,将邮政编码数据存放在一个单独的结构ZIPCODES中,可将其连接到STUDENTS和INSTRUCTORS表以查找地址中的城市和州部分。了解这些规定对理解本书中的例子应该会有所帮助。

### 使用Oracle Designer的意义

Oracle Designer仍然是目前市场上最好的集成CASE产品,它的性能会极大地影响CADM过程。Oracle Designer能够追踪单个信息仓储中的数据和处理信息,使其成为一种无法衡量其价值的资源。除了固定的功能和实用程序外,Oracle Designer还提供了用户扩展选项和应用程序接口(API)。这些功能使用户能扩充Oracle Designer的功能以支持Oracle产品的设计者有意不予实现的功能,或支持用户的特殊需求。

总的来说,Oracle Designer使应用系统开发更为精致和灵活,如下表所示。这是因为Oracle Designer提供了一种获取和管理新系统或当前系统的庞大数据的方式。数据有时相当难于管理,以至于系统设计者忽略或遗漏掉关键的业务需求,或者说,花费过多的时间去开发系统,最终丢掉了重要的需求。

| 开发因素  | 使用Oracle Designer | 不用Oracle Designer |
|-------|-------------------|-------------------|
| 系统开发  | 结构化               | 一般是特殊化的           |
| 维护代价  | 低                 | 高                 |
| 数据库生成 | 自动(代价低)           | 手工(代价高)           |
| 代码生成  | 部分自动化             | 全手工               |
| 系统文档  | 主要存放在信息仓储中        | 专门存放              |

**说明** Oracle Designer 2.1,有时可能会在帮助系统和一些其它的Oracle文献及第三方资料中发现这种称呼。

### CASE方案失败的原因

过去,CASE工具一直被视为是一种浪费时间和资源的东西,怀有这种看法的项目领导者不



可能支持其员工进行所需的学习。如果没有专门的时间学习这些工具，那么它们在工作中当然不会起作用，结果是工期明显拖延或者是完全失败。由于这些早期的负面影响，“CASE”被打上了不公正的烙印，许多公司碰都不去碰它。Oracle公司(以及其它的CASE软件商)因此重新改写了这个概念，并从它们的文档或产品介绍中擦去了“CASE”这个词。尽管如此，Oracle Designer还是一个CASE工具，只不过较之以前的产品，它具有更多的内涵和更易于使用罢了。但是该产品也需要花时间学习，忽略这一点，仍然会导至项目的失败。

CASE工程失败的另一主要原因是有的CASE用户不了解软件开发过程，完全依赖CASE产品引导他们一步一步地通过开发生命周期。这在当前的Oracle Designer 产品情况下，有时是可能的。但是不能，也不可能完全期望将CASE工具作为完全的驱动。首先，CASE是“计算机辅助”而不是“计算机驱动”的软件工程。CASE工具并不能完全替代方法。是方法而不是工具定义了我们的工作过程。

### 方法的决定

在开发一个系统时，主要精力必须放在开发的方法上。方法的重要任务之一是为工作选择工具。特别是CASE工具对开发方法有重要的影响，因为它们直接涉及工程的各个阶段。正如我们刚刚提及的，危险在于将CASE工具当成了方法。如果工具执行许多开发任务，就很容易会得出它们支持和驱动所有开发任务的结论。这种工具驱动的方法使得用户在开发过程中会跳过一些重要的步骤，因为这些步骤并不明确地通过工具处理。

我们都喜欢简易的、傻瓜式的设计过程。喜欢表格、详细的文档提交和精确的标准。喜欢放弃必须思考将要做什么事的职责。可是，事实上这是不可能的，因为每个系统都是不同的，系统开发是一项充满智力的活动。开发者必须回答下列问题：

- 在理论上是否存在一种可靠的方案，此方案将把工程引向正确的方向。
- 我们怎么才能知道某个任务或阶段已经完成了呢？
- 我们怎样评判项目是否成功？

每个阶段都有必须回答这几个问题。可惜的是，这些问题都不是通过制定工作计划、交付期限以及要求立即拿出结果的请求所能够解决的。

### 何时使用CADM

在写完本书后，我们认识到，首先CADM是一种传统的系统开发生命周期(SDLC)方法。我们的经验是，开发者和用户更喜欢在小心地做完分析和设计后就结束项目。大多数人都遇到过尽是败笔的系统，这些失败通常是由于走捷径不当和没有采用合适的方法，或者对正确的方法遵循得不够仔细而造成的。Oracle Designer 通过提供存放有大多数分析和设计信息的信息仓储，帮助我们更好地进行分析和设计。

虽然CADM不强调原型开发，但我们确实相信，为了给出某个概念的证明，可以进行原型研究。本书第22章描述了RAD-CADM方法。为了更好地与系统的用户进行沟通，原型开发可以在任何系统的整个生命周期中进行。它可以在花大量的经费建造系统以前，向用户提供一些信息，使他们了解建成的系统将会是什么样的。在Pre-Design(预设计)阶段，我们建议开发者生成一个原型，以便用户能够对系统的概貌有大致地了解。在此系统即将建造之时，用户对系统在其生命周期中的演化一直有很大的作用。

冒着被视为保守分子的风险，我们竭力提倡一种传统的SDLC方法，因为它做得更好、更快和更有效，而且组织更为严密。我们认为，必须进行仔细和正确的系统分析。如果这样，就能节省宝贵的时间和金钱，不用重做那些第一次就应该做对的事。

在我们一开始撰写本书时，认为它会是一本很小的著作。然而，完成之后，我们不得不承认，虽然它比我们所设想的要大得多，但也仅仅是一个应用开发过程概览。就某个题目我们经常想多写一点，但是受现实因素的制约，不能这样做。即使在这个修订和扩充了的版本中，我们也不能说本书对如何利用Oracle Designer建造系统作了完整的论述，因为那样篇幅会更大。然而，我们确实认为，本书是我们关于利用Oracle Designer建造系统的最好方法的观点的完整综述。

本书的英文原名为Oracle Designer Hand book, Second Edition。英文版的出版社为Osborne, a Division of The McGraw Hill Companies. 网址为<http://www.osborne.com>.

# 目 录

贺 辞

序

译者序

Oracle Designer内容概览

引言

## 第一部分 基础知识

|                                  |    |
|----------------------------------|----|
| 第1章 系统设计方法简介 .....               | 1  |
| 1.1 CADM概述 .....                 | 1  |
| 1.2 超越传统方法 .....                 | 2  |
| 1.3 策略阶段的概念 .....                | 4  |
| 1.4 预分析阶段的概念 .....               | 5  |
| 1.5 分析阶段的概念 .....                | 6  |
| 1.6 预设计阶段的概念 .....               | 8  |
| 1.7 设计阶段的概念 .....                | 9  |
| 1.8 建造阶段的概念 .....                | 10 |
| 1.9 测试阶段的概念 .....                | 11 |
| 1.10 实现阶段的概念 .....               | 12 |
| 1.11 维护阶段的概念 .....               | 13 |
| 1.12 本章小结 .....                  | 13 |
| 第2章 Oracle Designer简介 .....      | 14 |
| 2.1 信息仓储 .....                   | 15 |
| 2.1.1 信息仓储的构成 .....              | 15 |
| 2.1.2 信息仓储存放的内容 .....            | 15 |
| 2.1.3 信息仓储的表征 .....              | 16 |
| 2.2 前端工具 .....                   | 17 |
| 2.2.1 安装和硬件系统需求简述 .....          | 17 |
| 2.2.2 前端程序 .....                 | 17 |
| 2.2.3 Oracle Designer功能的种类 ..... | 18 |
| 2.2.4 系统需求建模 .....               | 19 |
| 2.2.5 产生初步设计 .....               | 22 |
| 2.2.6 设计和生成 .....                | 23 |
| 2.2.7 实用程序 .....                 | 32 |
| 2.3 Oracle Designer界面 .....      | 36 |

|                                         |    |
|-----------------------------------------|----|
| 2.3.1 Oracle Designer窗口 .....           | 36 |
| 2.3.2 启动Oracle Designer窗口 .....         | 37 |
| 2.3.3 Help系统 .....                      | 38 |
| 2.3.4 浏览器界面 .....                       | 40 |
| 2.3.5 图示化工具界面 .....                     | 43 |
| 2.3.6 实用程序界面 .....                      | 49 |
| 2.4 Oracle Designer工具如何<br>配合CADM ..... | 50 |
| 2.5 本章小结 .....                          | 51 |

## 第二部分 生命周期的各个阶段

|                      |    |
|----------------------|----|
| 第3章 策略阶段 .....       | 53 |
| 3.1 策略阶段概述 .....     | 53 |
| 3.2 提交 .....         | 54 |
| 3.2.1 初始项目文档 .....   | 54 |
| 3.2.2 项目合同 .....     | 55 |
| 3.3 策略文档 .....       | 56 |
| 3.3.1 策略文档的目标 .....  | 56 |
| 3.3.2 策略文档的结构 .....  | 56 |
| 3.3.3 执行概要 .....     | 57 |
| 3.3.4 原有系统的描述 .....  | 59 |
| 3.3.5 相关的项目 .....    | 59 |
| 3.3.6 业务和财政支持人 ..... | 60 |
| 3.3.7 动机 .....       | 60 |
| 3.3.8 项目范围 .....     | 61 |
| 3.3.9 解决方案 .....     | 61 |
| 3.3.10 成本效益分析 .....  | 62 |
| 3.3.11 项目组织和成员 ..... | 64 |
| 3.3.12 工作计划 .....    | 65 |
| 3.3.13 业务效果 .....    | 66 |
| 3.3.14 策略文档结束语 ..... | 67 |
| 3.4 应用系统分割 .....     | 67 |
| 3.5 范围 .....         | 67 |
| 3.6 策略文档的例子 .....    | 68 |

|                                             |     |                                                                      |     |
|---------------------------------------------|-----|----------------------------------------------------------------------|-----|
| 3.7 对小系统的改动 .....                           | 71  | 6.2.4 其它菜单和工具栏的功能 .....                                              | 123 |
| 3.8 策略阶段何时结束 .....                          | 72  | 6.3 将系统需求映射为信息仓储元素 .....                                             | 123 |
| 第4章 策略阶段中的Oracle Designer .....             | 73  | 6.3.1 使用Oracle Designer系统需<br>求的元素 .....                             | 123 |
| 4.1 Process Modeller .....                  | 73  | 6.3.2 使用用户需求扩展 .....                                                 | 123 |
| 4.1.1 相对于功能的业务过程 .....                      | 73  | 6.3.3 在用户扩展和基于表的系统<br>之间选择 .....                                     | 124 |
| 4.1.2 基本技术 .....                            | 74  | 第7章 分析阶段——信息收集 .....                                                 | 125 |
| 4.1.3 高级技术 .....                            | 79  | 7.1 提交 .....                                                         | 125 |
| 4.1.4 在Process Modeller中保存图形优<br>选项集合 ..... | 81  | 7.2 信息收集概述 .....                                                     | 126 |
| 4.1.5 建立多媒体表现 .....                         | 81  | 7.3 从用户那里得到需求 .....                                                  | 126 |
| 4.1.6 其它菜单和工具栏的功能 .....                     | 83  | 7.3.1 会谈 .....                                                       | 126 |
| 4.1.7 所提供信息的用途 .....                        | 85  | 7.3.2 调查表 .....                                                      | 131 |
| 4.2 Entity Relationship Diagrammer .....    | 85  | 7.3.3 电子通信 .....                                                     | 132 |
| 4.2.1 基本技术 .....                            | 86  | 7.3.4 联合应用开发会议 .....                                                 | 132 |
| 4.2.2 其它菜单和工具栏功能 .....                      | 89  | 7.4 原有系统检查 .....                                                     | 133 |
| 4.2.3 所提供信息的用途 .....                        | 90  | 7.5 安全 .....                                                         | 137 |
| 4.3 在RON中跟踪文档 .....                         | 91  | 7.6 单元级分析 .....                                                      | 137 |
| 第5章 预分析阶段 .....                             | 92  | 7.7 对小系统的改动 .....                                                    | 138 |
| 5.1 预分析阶段概述 .....                           | 92  | 7.8 信息收集何时结束 .....                                                   | 138 |
| 5.2 提交 .....                                | 93  | 第8章 分析阶段——信息收集中的<br>Oracle Designer .....                            | 139 |
| 5.3 预分析的目标 .....                            | 93  | 8.1 信息收集使用的Process Modeller和<br>Entity Relationship Diagrammer ..... | 139 |
| 5.4 信息搜集 .....                              | 93  | 8.2 Dataflow Diagrammer .....                                        | 140 |
| 5.4.1 用户提供的需求 .....                         | 94  | 8.2.1 Dataflow Diagrammer和<br>Process Modeller之间的区别 .....            | 140 |
| 5.4.2 对原有系统的重新检查 .....                      | 95  | 8.2.2 基本技术 .....                                                     | 141 |
| 5.5 需求分析 .....                              | 95  | 8.2.3 Function/Attribute Matrix实用<br>程序 .....                        | 147 |
| 5.6 分析计划 .....                              | 97  | 8.2.4 其它技术 .....                                                     | 147 |
| 5.7 分析计划样例 .....                            | 97  | 8.2.5 其它菜单和工具栏功能 .....                                               | 148 |
| 5.8 对小系统的改动 .....                           | 98  | 8.2.6 所提供信息的用途 .....                                                 | 149 |
| 5.9 预分析阶段何时结束 .....                         | 99  | 8.3 原有系统的ERD .....                                                   | 150 |
| 第6章 预分析阶段中的Oracle Designer .....            | 100 | 8.3.1 设计捕获实用程序 .....                                                 | 150 |
| 6.1 Repository Object Navigator .....       | 100 | 8.3.2 Table to Entity Retrofit<br>实用程序 .....                         | 153 |
| 6.1.1 基本技术 .....                            | 101 | 8.3.3 Entity Relationship Diagrammer<br>和原有系统的ERD .....              | 154 |
| 6.1.2 其它菜单和工具栏功能 .....                      | 111 |                                                                      |     |
| 6.1.3 使用RON抽查策略工作 .....                     | 114 |                                                                      |     |
| 6.1.4 使用RON创建新版本 .....                      | 114 |                                                                      |     |
| 6.2 Repository Reports .....                | 115 |                                                                      |     |
| 6.2.1 基本技术 .....                            | 116 |                                                                      |     |
| 6.2.2 输出窗口 .....                            | 120 |                                                                      |     |
| 6.2.3 自制报表 .....                            | 120 |                                                                      |     |