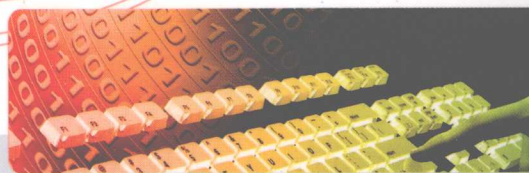
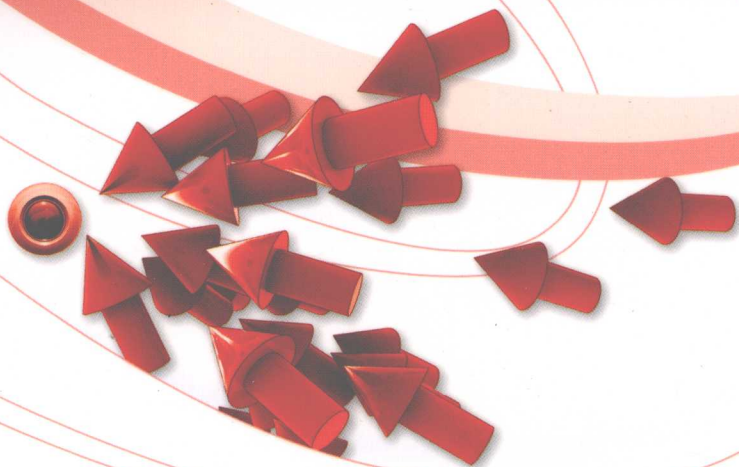


普通高等教育“十一五”国家级规划教材《实用软件工程(第3版)》
配套教材



高等学校计算机规划教材

软件工程实践教程



■ 杨林 赵池龙 主编
■ 姜义平 李冬 肖红玉 编著



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

内容简介

高等学校计算机规划教材

软件工程实践教程 (第2版)

杨林 赵池龙 主编

姜义平 李冬 肖红玉 编著



NLIC 2970686337

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书是普通高等教育“十一五”国家级规划教材《实用软件工程（第3版）》（赵池龙等主编）的姐妹篇，是软件项目综合实践的最新教程。内容包括：软件开发与软件管理全过程；软件工程从业人员的业务素质 and 职业道德规范；建模工具 Power Designer 和 Rational Rose；数据库建模理论及数据库设计模式；网上投票系统、网上论坛系统和学生信息系统三个项目开发案例的立项、需求、设计、实现、测试和源代码分析。本书为任课教师免费提供文档制作模板、电子课件及三个项目的程序源代码。

本书是软件和计算机相关专业本科生“软件工程综合实践”课程教材，也适合于软件工程在职研究生、软件学院师生和 IT 企业软件开发人员作为参考书使用。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有，侵权必究。

图书在版编目（CIP）数据

软件工程实践教程/杨林, 赵池龙主编; 姜义平, 李冬, 肖红玉编著. —2 版. —北京: 电子工业出版社, 2011.4
高等学校计算机规划教材

ISBN 978-7-121-13092-2

I. ①软… II. ①杨…②赵…③姜…④李…⑤肖… III. ①软件工程—高等学校—教材 IV. ①TP311.5

中国版本图书馆 CIP 数据核字 (2011) 第 041978 号

策划编辑: 童占梅

责任编辑: 童占梅

印 刷: 北京市顺义兴华印刷厂

装 订: 三河市双峰印刷装订有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×1 092 1/16 印张: 17.25 字数: 436 千字

印 次: 2011 年 4 月第 1 次印刷

印 数: 4 000 册 定价: 29.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 zlts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。

前 言

《软件工程实践教程》(第2版)进行了较大的改进和优化,内容安排更加紧凑、精炼,实用性更强,篇幅得到有效控制,其主要特点是:

(1) 作为《实用软件工程(第3版)》(赵池龙等主编)的姐妹篇,特别适合于“工程型和应用型大学”本科生作为软件项目实践课程教材,也适合于在职研究生和IT企业软件开发人员使用。

(2) 本书明确提出了“数据库设计模式”的概念,并通过典型实例阐述其用法,从而将数据库设计的理论与实践,推向了一个崭新的阶段。

(3) 在软件项目开发案例中,增加了“网上投票系统”案例,保留了第1版的“网上论坛系统”和“学生管理信息系统”两个案例,三个案例分别运行在.NET、J2EE和Visual Basic平台上,这三个开发平台在国内外IT界,具有广泛的代表性。

本书共分8章,各章的内容安排如下:

第1章,论述软件开发与软件管理全过程。首先要选择合适的软件开发模型,然后按照模型规定的步骤,进行需求、设计、编程、测试、验收,若符合客户要求,则开发过程结束,转入运行与维护阶段。若不符合客户要求,则进入新一轮迭代循环,开始新一轮需求、设计、编程、测试、验收。如此往复循环,直到客户满意为止。要减少往复循环次数,就要加强每一步的评审、审计、配置管理与质量保证工作,尤其是第一步(需求)的质量保证工作。

第2章,论述软件工程从业人员的八条基本业务素质要求,八条职业道德规范,十一条软件工程职业实践准则,以及软件工程职业实践的国际标准。

第3章,详细讲解数据库建模工具Power Designer。

第4章,通过一个“网上求职招聘系统”案例,生动讲解对象建模工具Rational Rose的用法。

第5章,通过案例将数据库设计的理论与实践紧密结合,讲解利用“数据库设计模式”解决各种实际问题。

第6章,介绍软件开发项目实践——“网上投票系统”的立项、需求、设计、编程实现和源代码,重点是让读者在B/S三层体系结构下,学会用.NET平台开发软件。

第7章,介绍软件开发项目实践——“网上论坛系统”的立项、需求、设计、实现、测试和源代码,使读者学会用J2EE平台开发软件。

第8章,介绍软件开发项目实践——“学生管理信息系统”的立项、需求、设计、实现、测试和源代码,使读者掌握用VB平台开发软件。

教学建议:

如果本教材单独作为“软件项目综合实践”课程开设,可从第1章到第8章按自然顺序进行实践教学,建议教学计划为3学分,54学时。第6、7、8章中三个实践项目,对应三种不同的软件开发平台,这对一些老师和学生有一定难度,建议三个项目任选两个项目完成。如果将本教材与《实用软件工程(第3版)》(赵池龙等主编)合在一起教学,建议总教学计划为6学分,108学时。本书为任课老师免费提供课件与三个项目程序源代码,可通过华信教育资源网 <http://www.hxedu.com.cn> 免费注册下载。

本书由一个积极进取的教学团队完成,参加编写的主要有杨林、赵池龙、姜义平、李冬和肖红玉。另外还有王学伟、吴喻、王冬龙、谭兆欣、李耀全、龙茂春、郑智鹏、何侃辉和周远明等人,他们都为本书的出版做出了贡献。由于各种主客观原因,书中难免存在错误或疏漏,诚恳欢迎广大高校师生和IT企业软件工作者批评指正。作者邮箱: zhaochilong@sohu.com, todoubaba@gmail.com 或 jyp2006@163.com。

本书提供电子课件和三个项目程序源代码,任课老师可通过华信教育资源网 <http://www.hxedu.com.cn> 免费注册下载。

作者
于北京师范大学珠海分校

目 录

第 1 章 软件开发与软件管理全过程	1
1.1 选择软件开发模型	2
1.2 选择软件开发方法	4
1.3 三个模型与三层结构	7
1.4 软件开发全过程	10
1.5 软件开发所需基本知识	13
1.6 软件管理全过程	14
第 2 章 软件工程从业人员的业务素质和职业道德规范	17
2.1 软件工程师的业务素质	18
2.2 软件工程师的职业道德规范	18
2.3 软件工程师职业实践的十一条准则	19
2.4 软件工程职业实践的国际标准	20
第 3 章 PowerDesigner 建模实践指南	26
3.1 PowerDesigner 概述	27
3.2 PowerDesigner 的安装与启动	27
3.2.1 PowerDesigner 的安装	27
3.2.2 PowerDesigner 的启动	29
3.3 用 PowerDesigner 进行数据库业务模型设计	29
3.3.1 建立 BPM	30
3.3.2 创建起点	31
3.3.3 定义处理过程	31
3.3.4 定义资源	32
3.3.5 定义终点	32
3.3.6 定义流程	32
3.4 用 PowerDesigner 进行数据库概念模型设计	33
3.4.1 创建概念数据模型	33
3.4.2 创建实体	34
3.4.3 建立实体之间的关系	37
3.4.4 定义域	38
3.4.5 定义业务规则	39

3.4.6	定义 CDM 属性	40
3.5	用 PowerDesigner 进行数据库物理模型设计	41
3.5.1	创建物理数据模型	41
3.5.2	创建表	42
3.5.3	创建列	42
3.5.4	创建索引	44
3.5.5	创建视图	45
3.5.6	创建触发器	46
3.5.7	创建存储过程和函数	47
3.5.8	通过 CDM 生成 PDM	48
3.6	用 PowerDesigner 进行面向对象模型设计	50
3.6.1	创建 OOM	50
3.6.2	设计用例图	50
3.6.3	由 PDM 转换生成 OOM	52
第 4 章	Rational Rose 建模实践指南	54
4.1	Rational Rose 概述	55
4.1.1	初识 Rational Rose	55
4.1.2	安装 Rational Rose	56
4.1.3	Rational Rose 的使用	59
4.2	用 Rational Rose 设计用例模型	61
4.2.1	理解用例模型中的元素	61
4.2.2	用例建模	64
4.2.3	案例分析——“网上求职招聘系统”用例建模	65
4.3	用 Rational Rose 设计领域模型	69
4.3.1	理解领域模型	70
4.3.2	使用 Rational Rose 建立领域模型	70
4.3.3	案例分析——“网上求职招聘系统”的领域模型	71
4.4	用 Rational Rose 设计类模型和包图	72
4.4.1	类建模	72
4.4.2	设计包图	73
4.4.3	案例分析——“网上求职招聘系统”类图和包图	75
4.5	用 Rational Rose 设计系统动态模型	80
4.5.1	时序图建模	81
4.5.2	协作图建模	83
4.5.3	状态图建模	85
4.5.4	活动图建模	86
4.6	用 Rational Rose 进行数据建模	88

4.6.1 创建数据模型	88
4.6.2 案例分析——“网上求职招聘系统”的数据模型	91
4.6.3 数据模型和对象模型的转换	92
4.6.4 由数据模型生成数据库	93
4.7 Rational Rose 的其他功能	94
4.7.1 Rational Rose 逆向工程介绍	94
4.7.2 Java 代码逆向工程	94
4.7.3 设置全局选项	95
第 5 章 数据库建模实践指南	97
5.1 数据库设计规范化理论及其反思	98
5.1.1 为什么要建立规范化理论	98
5.1.2 第一范式	100
5.1.3 第二范式	101
5.1.4 第三范式	102
5.1.5 数据库规范化设计的优点	103
5.1.6 数据库规范化设计的缺点	104
5.1.7 对规范化设计理论的反思	104
5.2 数据库设计评价	105
5.3 数据库设计的基本技巧	106
5.3.1 数据库中的“第三者插足”设计模式	106
5.3.2 数据库中的“列变行”设计模式	108
5.3.3 客户是上帝的数据库设计理念	110
5.3.4 主键设计	111
5.3.5 “西瓜”设计模式	113
5.3.6 树形结构设计模式	114
5.3.7 权限管理设计模式	115
5.3.8 单表设计模式	117
5.3.9 属性拓展设计模式	118
第 6 章 软件开发项目实践——网上投票系统	121
6.1 立项背景	122
6.2 需求分析	122
6.3 数据库的设计与实现	123
6.4 界面的设计与实现	126
6.5 网上投票系统功能的扩展	143

第7章 软件开发项目实践——网上论坛系统	144
7.1 系统立项背景.....	145
7.2 系统需求分析.....	146
7.2.1 需求分析的任务.....	146
7.2.2 需求分析文档.....	147
7.3 系统设计.....	155
7.3.1 开发环境和运行环境.....	155
7.3.2 整体架构设计.....	156
7.3.3 数据库设计.....	159
7.3.4 设计文档.....	163
7.4 代码实现.....	170
7.4.1 系统目录结构.....	170
7.4.2 实现数据库连接.....	171
7.4.3 实现用户注册登录功能.....	172
7.4.4 实现版块管理子系统.....	191
7.5 系统部署.....	209
7.6 本章小结.....	212
第8章 软件开发项目实践——学生管理信息系统	213
8.1 系统立项背景.....	214
8.2 需求分析.....	215
8.2.1 理解需求.....	215
8.2.2 分析需求.....	215
8.2.3 用例分析.....	216
8.2.4 需求分析文档.....	217
8.3 概要设计.....	225
8.3.1 概念数据模型 CDM 设计.....	225
8.3.2 物理数据模型 PDM 设计.....	226
8.3.3 创建数据库.....	227
8.3.4 创建表.....	228
8.3.5 概要设计文档.....	229
8.4 详细设计.....	241
8.5 编程实现.....	243
8.5.1 控件级定义.....	243
8.5.2 窗体级定义.....	244
8.5.3 编程实现.....	245
8.5.4 用 VB 语言实现部分代码.....	245

8.5.5 用户手册	249
8.6 系统测试	259
8.6.1 概述	259
8.6.2 测试计划	260
8.6.3 测试报告	261
8.6.4 测试小结	263
8.7 本章小结	264
参考文献	265

第1章

SE

软件开发与软件管理全过程

本章导读

软件工程是研究软件开发与软件管理的工程科学。软件工程认为：只要将软件开发过程与软件管理过程改善了，软件企业的项目、产品和服务就能令客户满意，软件企业就能发展壮大。那么，软件开发过程与软件管理过程到底包含哪些内容呢？这就是本章要解决的问题。

表 1-1 列出了读者在本章学习中要了解、理解和关注的主要内容。

表 1-1 本章对读者的要求

要求	具体内容
了解	1) 软件开发过程包括哪几个阶段？ 2) 软件开发过程中有哪几份重要文档？ 3) 软件管理过程中有哪几个重要的里程碑？
理解	1) IT 企业常用的软件开发模型有哪几种？它们各自的优点、缺点、适用场合怎样？ 2) IT 企业常用的软件开发方法有哪几种？它们各自的优点、缺点、适用场合怎样？
关注	软件工程中两种主要的开发平台 .NET 和 J2EE

软件开发的一般过程是：首先选择合适的软件生命周期模型，即软件开发模型；然后按照该模型规定的步骤，进行需求、设计、编程、测试、验收。若符合客户要求，则开发过程结束，转入运行与维护阶段；反之，进入新一轮迭代循环，开始新一轮需求、设计、编程、测试、验收；如此往复循环，直到客户满意为止。若要减少往复循环次数，则要加强每一步的评审、审计、配置管理与质量保证工作，尤其是第一步（需求）的质量保证工作。

1.1 选择软件开发模型

软件生命周期（Software Life Cycle）与软件开发模型（Software Development Model）有关，不同的开发模型，对应的生命周期略有差异。

生命周期指软件开发全部过程、活动和任务的结构框架。软件开发包括需求、设计、编程和测试阶段，有时也包括软件实施与维护阶段。目前IT企业在软件开发实践中使用的各种生命周期模型，基本上是瀑布模型、增量模型、原型模型与迭代模型。

1. 瀑布模型

瀑布模型（Waterfall Model）又称流水式过程模型，它形象地用阶梯瀑布描述，水由上向下一个阶梯接着一个阶梯地倾泻下来。在瀑布模型中，软件开发的各项活动严格按照线性方式进行，当前阶段的活动接受上一阶段活动的工作结果，实施完成所需的工作内容。对当前阶段活动的工作结果需要进行验证，如果验证通过，则该结果作为下一阶段活动的输入，继续进行下一阶段的活动，否则返回上一阶段进行修改。

瀑布模型选取的条件是：在开发周期内，客户的需求是稳定的。

2. 增量模型

增量模型（Incremental Model）是遵循递增方式来进行软件开发的。软件产品被作为一组增量构件（模块），每次需求、设计、编程、集成、测试和交付一块构件，直到所有构件全部实现为止。

增量模型选取的条件是：项目本身可拆卸成多个模块，并且客户同意一个模块接着一个模块地交付。

一般而言，增量模型与迭代模型一起使用。

3. 原型模型

原型模型（Prototype Model）的本意是：在初步需求分析之后，马上向客户展示一个软件产品原型（样品），对客户进行培训，让客户试用，在试用中收集客户意见，根据客户意见立刻修改原型，之后再让客户试用，反复循环几次，直到客户确认为止。

原型模型选取的条件是：开发者手中有相近或相似的产品原型。

一般而言，原型模型与迭代模型一起使用。

4. 迭代模型

迭代模型（Iterative Model）的所谓迭代，是指活动的多次重复。从这个意义上讲，原型不断完善和增量不断产生，都是迭代的过程。因此，快速原型法和增量模型都可以看成是部

分的迭代模型。但是，这里所讲的迭代模型是 RUP (Rational Unified Process)，它是一种最有代表性的迭代模型。RUP 推出的一种“逐步求精”的面向对象的软件开发过程模型，被认为是软件界迄今为止最完善的、商品化的开发过程模型。

迭代模型的特点是：迭代或迭代循环驱动，每一次迭代或迭代循环，均要走完初始（规范业务与需求）、精化（架构设计与详细设计）、构建（编程与测试）、移交（产品化）这 4 个阶段。

迭代模型选取的条件是：在开发周期内，客户的需求是有变化的。

5. 瀑布模型与迭代模型之间的关系

在宏观上，迭代模型是动态模型，瀑布模型是静态模型。一方面，迭代模型需要经过多次反复迭代，才能形成最终产品。另一方面，迭代模型的每一次迭代，实质上都是执行一次瀑布模型，都要经历初始、精化、构造、移交 4 个阶段（或需求、设计、实现、交付 4 个阶段），走完瀑布模型的全过程。

在微观上，迭代模型与瀑布模型都是动态模型。迭代模型与瀑布模型在每一个开发阶段（初始、精化、构造、移交）的内部，都有一个小小的迭代过程，只有经历这个小小的迭代过程，该阶段的开发工作才能做细做好。

瀑布模型与迭代模型之间的这种微妙关系，如图 1-1 所示。

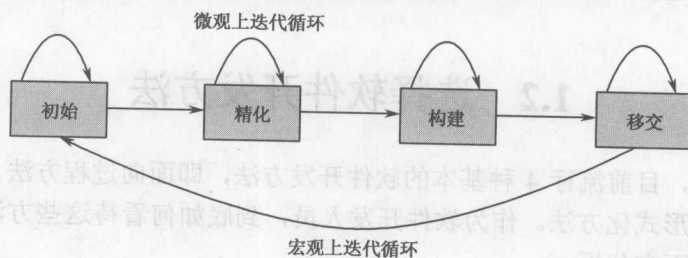


图 1-1 瀑布模型与迭代模型之间的关系

从图 1-1 可见，迭代中有瀑布，瀑布中有迭代，即你中有我、我中有你。

瀑布模型与迭代模型之间的关系，反映了人们对客观事物的认识论：要认识与掌握某一客观事物，必须经历由宏观到微观的多次反复过程。只有从宏观上反复迭代几次，才能看清全貌，掌握事物的宏观发展规律。只有从微观上反复迭代几次，才能吃透每个细节，掌握事物的微观发展规律。

6. 软件开发模型总结

以上介绍的 4 种开发模型，是 IT 企业常用的模型，它们的性能比较，如表 1-2 所示。

此外，自从敏捷文化出现之后，在一些中小型软件企业，XP 模型 (Extreme Programming Model) 也迅速得到了应用。

除了上面介绍的几种常用模型之外，还有螺旋模型 (Spiral Model)、喷泉模型 (Fountain Model) 等，目前这些模型在 IT 企业很少使用。

表 1-2 4 种开发模型性能对比

模型名称	优点	缺点	适合的场合
瀑布模型	开发阶段清晰，便于评审、审计、跟踪、管理和控制	逆转性差，错误会发散式传播	瀑布模型适合于面向过程开发方法，要求在开发时间内需求没有或很少变化
增量模型	将一个大风险分解为多个小风险，从而降低了开发难度，人员分配灵活，刚开始不用投入大量人力资源	系统的组装和拆卸性必须强，或开发人员全局把握水平必须高，客户必须同意分阶段提交产品	系统的组装和拆卸性强，客户接受分阶段交付
原型模型	开发速度快，用户意见反馈实时，有利于开发商在短时间内推广并实施多个客户	因为事先有一个展示性的产品原型，所以在一定程度上，不利于开发人员的创新	凡是有软件产品积累的软件公司，他们在投标、开发、实施项目的过程中，通常采用原型模型法
迭代模型	在开发早期或中期，用户需求可以变化；在迭代之初，它不要求有一个相近的产品原型；模型的适用范围很广，几乎适用于所有的项目开发	要求采取循环工作方式，每次循环均使工作产品更靠近目标产品一次，这就要求项目组具有很高的水平并掌握先进的开发工具。反之就会存在较大的技术风险和技能风险	多次执行各个开发工作流程，从而更好地理解需求，设计出更为强壮的软件构架，逐步提高开发组织能力，最终交付一系列逐步完善的实施成果

1.2 选择软件开发方法

在软件工程界，目前流行 4 种基本的软件开发方法，即面向过程方法、面向对象方法、面向元数据方法和形式化方法。作为软件开发人员，到底如何看待这些方法，进而如何选择这些方法呢？请看下文分析。

1. 面向过程方法

面向过程方法（Procedure-oriented Method），来自于面向过程的程序设计语言，如汇编语言、C 语言。面向过程方法包括面向过程的需求分析、设计、编程、测试、维护和管理。

面向过程方法，习惯上也称为传统软件工程开发方法，或结构化方法。它包括结构化分析、设计、编程、测试、维护。它有时又称为面向功能的方法，即面向功能的分析、设计、编程、测试、维护。由此可见，面向过程方法、面向功能方法和结构化方法，三者含义相同。

面向过程方法，强调模块化的思想，采用“自顶向下，逐步求精”的技术对系统进行划分，分解和抽象是它的两个基本手段。编程时采用“单入口、单出口”的控制方式，并且只包含顺序、分支和循环三种结构，目标之一是使程序的控制流程线性化，即程序的动态执行顺序符合静态书写结构。

面向过程方法的优点是：以处理流程为基础，简单实用。

面向过程方法的缺点是：只注重过程化信息，因而忽略信息的层次关系以及相互联系。它企图使用简单的时序过程方法（顺序、分支、循环三种结构），来描述关系复杂（随机）的信息世界，因而对于关系复杂的信息系统来说，其描述能力不强，最后可能导致软件设计、

开发和维护陷入困境。

自从面向对象方法出现之后，面向过程方法的应用范围开始萎缩，在许多领域，它逐渐被表述能力更强的面向对象方法所取代。当前，面向过程方法主要用在过程式程序设计中，如对象方法（函数）、科学计算、实时跟踪和实时控制的实现。

2. 面向对象方法

面向对象方法（Object-oriented Method），在不少软件工程教材中，也称为现代软件开发方法。该方法包括面向对象的需求分析、设计、编程、测试、维护和管理。面向对象方法是一种运用对象、类、消息传递、继承、封装、聚合、多态性等概念来构造软件系统的软件开发方法。

面向对象方法的特点是：将现实世界的事物（问题域）直接映射到对象。分析设计时由对象（Object）抽象出类（Class），程序运行时由类还原到对象。

面向对象方法的优点是：由于每个对象可以接收信息、处理数据和发送信息给其他对象，可作为一个独立的单元使用与运行，可起到效果独特的作用，可以处理大量既离散又关联的实体，可重复使用编程的逻辑单位，以及该方法侧重的不是过程的连续数据，而是客观的离散数据，所以它能描述无穷的信息世界。同时，又由于对象间具有继承性，所以易于代码重用和扩展。

面向对象方法的缺点是：设计和实现的复杂性较高，对于习惯于面向过程方法的人，较难掌握。

面向对象方法是当前计算机界关心的重点，是软件工程方法论的主流。面向对象的概念和应用已超越了程序设计和软件开发，扩展到更宽的范围。如交互式界面、应用结构、应用平台、分布式系统、网络管理结构、CAD技术、人工智能等领域。

3. 面向元数据方法

这里讲的面向元数据方法（Meta-data Oriented Method），既不是传统软件工程中的“面向数据流”方法，也不是传统意义上的面向数据结构的 Jackson 方法，它们两者都是面向过程的方法，而且这两种方法都出现在关系数据库管理系统 RDBMS (Relational Database Management System) 成熟之前。

元数据（Meta-data）是关于数据的数据，组织数据的数据，管理数据的数据。

这里的元数据，泛指一切组织数据的数据，如类的名称、属性和方法，实体的名称、属性和关联；数据库中的表名、字段名、主键、外键、索引、视图；数据结构中存储数据的框架等。但是，我们研究的重点，是指数据库中的元数据。

面向元数据方法来源于面向元数据的程序设计思想，即来源于关系数据库语言的程序设计思想。当关系数据库管理系统和数据库服务器出现之后，面向元数据方法才被人们所发现与重视。当数据库设计的 CASE 工具 Power Designer、Oracle Designer 和 ER win 出现之后，面向元数据设计方法才开始流行。面向元数据方法包括面向元数据的需求分析、设计、编程、测试、维护。

面向元数据的需求分析，就是在需求分析时，找出信息系统所有的元数据，使其完全满足信息系统对数据存储、处理、查询、传输、输出的要求。有了这些元数据，信息系统中的

一切原始数据不但都被组织起来了,而且能完全派生出系统中的一切输出数据。

面向元数据的设计,就是利用需求分析获得的元数据,采用面向元数据的CASE工具,设计出信息系统的概念数据模型CDM(Conceptual Data Model)和物理数据模型PDM(Physics Data Model),以及从原始数据到输出数据的所有算法与视图。

面向元数据的编程,就是在物理数据模型PDM的基础上,根据信息系统的功能、性能、接口和业务规则,建立数据库表和视图,再利用数据库编程语言,编写出存储过程和触发器。

面向元数据的测试,就是对数据库表初始化并加载之后,运行相关的存储过程和触发器,测试信息系统的各种功能需求与性能指标。

面向元数据的维护,就是对数据库表中的记录进行统计、分析、审计、复制、备份、恢复,甚至对表结构及视图结构,也可以进行必要的调整。

事实上,20多年来,面向元数据方法已经是建设信息系统、数据库、数据仓库和业务基础平台的基本方法。

面向元数据方法的缺点是:只能实现二维表格,不能实现窗口界面。

面向元数据方法,与关系数据库管理系统紧密地捆绑在一起,只要面向对象数据库不能完全替代关系数据库,这种方法就不会终结。目前数据库管理系统的发展趋势是:在关系型数据库的基础上,将面向对象的某些特性(如继承)添加上去,称为“对象-关系型数据库”,但本质上仍然是一个关系型数据库。

4. 形式化方法

软件工程的形式化方法(Formalized Method),是建立在严格数学基础上、以逻辑推理为出发点、并且具有精确数学语义的开发方法。

形式化方法在IT企业很少使用,在此也不进一步介绍。

5. 各种方法之间的关系

到目前为止,软件工程的基本方法,包含面向过程方法、面向对象方法、面向元数据方法和形式化方法。至于面向业务基础平台方法,它只是面向元数据方法与面向对象方法的具体应用案例,不能单独作为一种基本方法。

在大型多层(B/A/S)结构的信息系统建设中,这4种方法的关系是:面向元数据方法用在数据库服务器S层面上系统的分析、设计与实现,面向对象方法用在除数据库服务器层之外的其他层面上(B/A)系统的分析、设计与实现,面向过程方法用在其他两种方法本身内部函数的分析、设计与实现,形式化方法用在某些核心程序的正确性证明上。

由上可知,在实现一个较大的软件项目时,常常根据项目的特点,采取以某一种方法为主,以其他方法为辅。例如,开发信息管理类项目时,宜采取面向对象方法为主、面向过程和面向元数据方法为辅。对于实时跟踪与自动控制的项目,宜采取面向过程方法为主、面向元数据和面向对象方法为辅。对于纯数据库项目,宜采取面向元数据方法为主、面向过程和面向对象方法为辅。

因此,面向过程方法、面向对象方法、面向元数据方法和形式化方法,它们各有优缺点,适合于不同的场合。4种方法的比较,如表1-3所示。

表 1-3 4 种开发方法的对比

方法名称	优点	缺点	适合的场合
面向过程的方法	简单好学	不适应窗口界面， 维护困难	大型工程计算，实时数据跟踪处理，各种自动化控制系统，系统软件实现，以及过程、方法与函数的计算
面向对象的方法	功能强大，易于维护	不易掌握	互联网时代，完全由用户交互控制程序执行过程的应用软件和系统软件的开发。例如，在 B/S 结构中数据层之外的其他各层的需求、设计与实现
面向元数据的方法	通俗易懂	不适应窗口界面	以关系数据库管理系统为支撑平台的信息系统建设。例如，B/S 结构中数据层的需求、设计与实现
形式化方法	准确、严谨	难于上手和应用	对安全性要求极高，不容许出错的软件系统，如军事、医药、交通等领域

4 种基本的开发方法各有优缺点，各有生存时间和空间，它们不是互相孤立、毫无联系、彼此对立的，而是互相帮助、取长补短、彼此有关的，所以它们在软件开发领域能和平共处，互相促进，共同构成了一个多极化的、丰富多彩的软件工程方法论世界。

1.3 三个模型与三层结构

在信息系统需求与设计，一般需要建立三个模型，即功能模型、业务模型和数据模型，分别对应 B/A/S 三层体系结构中的三个不同层次，即浏览层、中间层和数据层，如图 1-2 所示。



图 1-2 三层结构示意图

其中：功能模型主要表现在浏览层；业务模型对应中间层，即业务逻辑层，功能模型中的功能，基本上都在业务模型中加以具体实现；数据模型对应数据层，实现持久型数据的存储与运算。

1. 三个模型

(1) 功能模型

功能模型 FM (Function Model) 实质上是用户需求模型，是描述系统能做什么，即对系统的功能、性能、接口和界面进行定义。因此，功能模型反映了系统的功能需求，它是划分系统模块和用户界面设计的主要依据。

功能模型在需求分析时的表示方法为系统功能需求列表、性能需求列表、接口需求列表、界面需求列表。统一建模语言 UML 主要采用“用例图”来描述功能模型。

功能模型既是动态的，又是静态的。因为有的功能与系统运行的时间序列有关。