

<<见微知著>>

图书基本信息

书名：<<见微知著>>

13位ISBN编号：9787121114496

10位ISBN编号：7121114496

出版时间：2011-2

出版时间：电子工业出版社

作者：蔡智明

页数：222

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>

## &lt;&lt;见微知著&gt;&gt;

## 前言

软件工程是一门研究用工程化方法开发、维护、管理大型软件的学问。

软件的开发到底是一门科学还是一门工程，这是一个很难讨论也很难确定的问题。

实际上，软件开发兼有两者的内涵，工程中有科学，科学中有工程，而我们也不一定需要去很严格地鉴定两者之间的界限。

软件工程的目的是开发出有效、实用和高质量的软件，其中需要更多的理论指导，还是本质上是一个实践出真知的行当？

这里有太多理论与实践关系的博弈。

真正的软件工程需求，只有在软件危机的感觉已经“大难临头”的时候，只有在提高软件开发效率和质量的压力已经明确压迫到自身的时候，很多毕业的IT学生才真正开始回味软件工程学习的内容，萌发对软件工程的实质需求。

然而这时他们中的很多人才发现，此前软件工程中学习的理论，在实践中是那么苍白，似乎有些启蒙，但更多地像是泡沫，相当大一部分软件工程学者和书籍是在靠一堆洋精华或洋糟粕的洋“经浜”而扬名万里，理论脱离实践，特别是脱离中国的实践、本土的实践，这是软件工程课程与教材中不争的现实。

软件设计，实实在在是一门实践中的学问，将大型软件开发方法上升到理论后，被称为“软件工程”而作为一门学科存在。

但在软件工程的有关书籍中，我们看到，从理论到理论、从理论到实践描述软件工程的书多，而从实践到理论描述软件工程的书少。

实践是检验真理的唯一标准，普遍真理必须与具体实际相结合。

实践本身就是最好的老师，就是最好的理论！

这个简单的哲学道理应该是学习软件工程和软件开发的基本点。

## &lt;&lt;见微知著&gt;&gt;

## 内容概要

本书以一个非常典型的软件系统——酒店信息管理系统的开发为例，站在欠缺经验的IT学生的角度，介绍了从学生懵头懵脑参与开发，调研规划，建立计划，思考Use Case，构建ER、DFD、面向对象UML模型，到构思软件构架、实现系统界面。

与一般专业书籍完全不同的是，全书采用对比递进写作风格，穿插交互式启发问答。

各章节均以那些糊里糊涂的、片面幼稚的开发思考、文档、模型图、界面为例，首先给出常见的粗劣思考实例，接着加以分析评判，引导读者身临其境地介入思考，再给出改进实例，然后对比、理清、提高，指出实践中带有普遍性、关联性的问题，再提升到软件工程方法的层面加以讨论，并结合了作者多年软件开发的实际经验。

本书以可读性强、易于理解的图、表为主要描述手段。

从这些来自真实实践的糊涂思考及相应图、表、文档的粗拙产生，再到分析、改进、提升的过程中，读者可以举一反三，避免重蹈覆辙，逐步理解、深化、悟透软件工程的真实内涵，切实提高软件工程师的实际修养。

全书将“前后一贯，左右顾盼”的理念贯穿开发活动的始终，十分有助于读者系统化一致化分析能力的提升。

本书可以作为学习、研究软件工程的参考书，或软件工程实践、实习的教材，对于在软件领域摸爬的新老手们，当需要找点软件开发和软件工程的实际参考经验时，本书也是很适合的。

## 书籍目录

引言——走入软件开发的“魔鬼”地带：从哪开始

第1章 从这里开始——基于实践，编好系统开发计划

1.1 调查研究一下这是一个什么项目

1.1.1 了解一下这家酒店的背景

1.1.2 讨论清楚软件的目标

1.1.3 画一张酒店组织构架图

1.1.4 重点是一定要研究清楚在信息系统支持下酒店运行的业务流程

1.1.5 还要画一张系统的网络环境图

1.2 如何组成系统

1.2.1 把系统划分成几个子系统

1.2.2 明确每个子系统的功能

1.2.3 确立系统的软硬件配置

1.3 如何安排好项目计划

1.3.1 安排好时间：软件过程模型的实际应用与管理

1.3.2 安排好资金：为测试与维护留有余地——40-20-40原则

1.3.3 安排好人员：人件与风险管理

1.4 如何考虑知识产权问题

1.4.1 严谨制定开发合同：不要以为合同是虚的

1.4.2 具体确定利益分派：不要以为利益只是钱

1.5 如何编好系统开发计划书——计划书实例

第2章 从这里入手——观察实践，分析软件需求

2.1 从宏观上把握系统——系统Use Case图

2.1.1 在线预订Use Case图

2.1.2 前/后台系统Use Case图

2.2 从微观上明确概念——系统概念模型

2.2.1 系统ER图

2.2.2 概念定义——数据表

2.2.3 概念定义——数据字典

2.3 从流程上明确需求——系统DFD图与相关IPO描述

2.3.1 全系统0层DFD

2.3.2 系统一层DFD

2.3.3 系统二层DFD

2.3.4 IPO描述

第3章 从这里深化——深入实践，建立UML模型

3.1 哪些是我系统中的对象——静态Class图

3.1.1 系统Class图

3.1.2 系统GUI类图

3.2 如何用这些对象实现我的系统——动态Sequence图

3.2.1 预订 (Reservation) 序列图

3.2.2 登记入住 (Check In) 序列图

3.2.3 退房结账 (Check Out) 序列图

3.3 如何编好软件需求规范说明书SRS——SRS实例

第4章 从这里构筑——围绕实践，设计系统构架与部件

4.1 如何考虑系统的构架？

<<见微知著>>

- 4.1.1 以数据为中心的体系架构
- 4.1.2 分层体系架构
- 4.1.3 三层体系 (C/S, B/S) 架构
- 4.1.4 调用/返回体系架构
- 4.1.5 面向对象体系架构
- 4.1.6 面向对象与面向过程
- 4.2 如何组织对象部件——Package图
- 4.3 如何设计对象部件
  - 4.3.1 预订Component图
  - 4.3.2 客房Component设计
  - 4.3.3 预订Activity图
  - 4.3.4 客房State图
  - 4.3.5 入住/退房 Swimlane图
- 第5章 哇噻, 软件出来了——反复实践, 实现系统
  - 5.1 酒店网站首页的设计: 注意全局风格的安排
  - 5.2 客房预订的实现——关联预订的分析与设计
  - 5.3 入住登记的实现——关联入住的分析与设计
  - 5.4 退房收银的实现——关联退房的分析与设计
  - 5.5 如何编好设计说明书——设计说明书实例
- 后记

## &lt;&lt;见微知著&gt;&gt;

## 章节摘录

插图：表1-7在时间安排分配上，呈现的是前期10周一中期30周一后期8周的分配，看上去基本合理。但软件工程管理的计划安排上，有一个40—20:40原则，即前、后期各为总工作量的40%，中期为20%。

对照起来，似乎不合这一原理。

经过仔细分析，可以看到，软件工程的前期40%包括需求分析、建模（也包括原型开发）阶段，这样计算，前期为20周左右，基本合理；但关键在于后期的测试、改进只安排了8周时间，是否足够？这值得商榷。

其实，很多软件项目的开发，后期工作量和难度都是相当大的。

有很多人认为，软件已经实现，系统已经集成成功，开始试用，差不多该“大功告成”了。

这往往导致对后期测试维护工作的轻视，一旦后期工作并不是想象得那么简单和顺利时，项目计划时间已到，资金等也已耗尽，就给进一步工作带来极大的被动。

这也是软件工程管理在原理上之所以强调要给测试维护留有40%的时间、资源的原因。

所以在安排计划上要给后期工作“留有余地”，尽量“赶早不赶后”，把时间往前压，这样，如果提前完成，大家皆大欢喜，如果后期出现困难，也有较充分的时间资金应付，这是计划安排中的一个实用“技巧”。

## <<见微知著>>

### 后记

基于实践，观察实践，深入实践，围绕实践，反复实践，总结起来，其实就是，软件开发、软件工程实实在在是实践出来的，干出来的！

软件工程的方法理论也是前人在实践中总结出来的，当运用于具体的软件实践中时，仍然需要从实践中来，到实践中去。

当深入理解、掌握了方法和理论，又确实使实践有了方向和指南。

但又不能机械地去照搬理论和方法。

理论和实践的辩证法，需要哲人永恒的探索。

软件开发、软件工程是一本永远写不完的书！

本书的部分图表由王孟函、刘苗、李俭荣等同学制作，特此致谢！

最后，再强调一下“反复实践”的问题与“前后一贯，左顾右盼”的原则。

软件开发、软件工程实践是一个复杂的过程，不可能通过计划、分析、建模、设计、实现的一次轮回，就获得理想的软件，一定还要反复测试、调试、改造、逐步完善，有时，到了软件使用中，还会发现，前期的一些模型没有考虑清楚（如押金、会员到底如何处理？

），一些概念也没有定义清楚（如客房到底有几个类型？

软件的使用者到底有哪些角色？

），需要重新考虑模型和设计实现。

<<见微知著>>

编辑推荐

《见微知著:从软件实践到软件工程》：从完整案例实践出发，采用对比递进写作风格，先给出各个开发环节的常见粗劣思考实例，再分析、给出更合适的解决方案；再提升到方法层面讨论。

《见微知著:从软件实践到软件工程》综合作者多年软件开发与教学经验，从来自真实实践的逐步分析构图及相应模型、图、表、文档的实际评判改进中，教会读者举一反三。

结合系统开发实例，综合面向对象与面向过程两种软件开发方法，将两者优势互补，再比较两种方法的特点，比空谈两种理论更具启发性和实践价值。

将“前后一贯，左右顾盼”的理念贯穿开发活动的始终，帮助读者提升周密的系统化分析能力和敏锐的一致化洞察能力，既善于把握全局又精于掌控细节。



<<见微知著>>

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>