

<<软件工程与开发技术>>

图书基本信息

书名：<<软件工程与开发技术>>

13位ISBN编号：9787560621586

10位ISBN编号：7560621589

出版时间：2009-2

出版时间：江开耀 西安电子科技大学出版社 (2009-02出版)

作者：江开耀 著

页数：380

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>

## <<软件工程与开发技术>>

### 前言

经过40余年的发展，软件工程学已经取得了长足的进步。

对软件工程过程、软件开发技术和软件工程工具的研究与改进正引导着我们逐渐克服软件危机的影响，从传统的手工业生产模式向大工业化的软件生产模式转化。

由传统的软件工程到计算机辅助软件工程、从结构化技术到面向对象技术、从只注重产品到同时关注产品与过程，软件工程学本身也在实践中不断发展并走向成熟，已经成为计算机科学技术的一个重要分支学科。

近年来，在传统软件工程的基础上，新的软件工程方法、新的软件体系架构不断出现，软件工程过程模型不断趋于成熟，计算机辅助软件工程技术突飞猛进，使得软件工程效率和软件工程质量都有了质的飞跃。

但我们也客观地认识到，即使是在信息技术得到了飞速发展的今天，软件危机的阴影依然未能完全消除；进一步提升软件研发的工程化程度，从根本上提升软件工程的效率可控、质量可控程度的任务仍然有待完成；学习并掌握现代软件工程理论，理解并学习使用计算机辅助软件工程方法与工具也仍然是计算机应用相关专业学生的重要任务之一。

本书注重从实用角度介绍软件工程的基础知识和实用的软件工程技术方法，希望能够使本书的读者对现代软件工程有一个较为全面的理解，并对实际的软件工程活动有所帮助。

全书共四篇，23章，对软件工程进行了较全面的介绍。

第一、二篇主要介绍了软件工程基础知识与软件工程方法，主要内容包括软件工程的基本概念、结构化的软件工程方法和面向对象的软件工程方法。

第三篇综合介绍了软件工程项目管理方法，主要内容包括工程估算、软件度量、风险防范、软件质量保证和软件配置管理方面的知识。

## <<软件工程与开发技术>>

### 内容概要

《软件工程与开发技术（第2版）》从软件工程方法、软件工程过程层面对现代软件工程学进行了较为系统和全面的介绍。

全书共分为四篇，23章。

第一篇介绍了传统的软件工程知识，包括软件工程的由来与发展、与软件工程学科相关的基础知识以及传统的结构化软件工程方法，具体内容有基于结构化方法的可行性分析、需求分析、设计与编码和软件测试知识等。

第二篇以RuP为蓝本，介绍了现代面向对象的软件工程方法，重点就业务模型、用例模型、对象模型、包模型、动态模型、构件模型、部署模型的建模方法进行了详细讲述，最后介绍了面向对象测试的概念。

第三篇就项目估算、项目策划、品质管理、配置管理、风险管理等项目经理必备的项目管理知识作了较全面的介绍。

现代软件工程十分重视不断提升个人与组织的工程过程能力，因此，第四篇用较大篇幅论述了软件能力成熟度模型、个人软件过程PsP和小组软件过程TsP的相关内容。

《软件工程与开发技术（第2版）》适合作为计算机相关专业本科教学中软件工程课程的教科书，也可作为软件从业人员的参考书。

## 书籍目录

第一篇 传统的软件工程第1章 软件工程引论1.1 软件产品的概念与特征1.1.1 软件产品的概念与分类1.1.2 软件产品的特征1.1.3 软件发展的阶段划分1.2 软件危机1.2.1 软件危机及其表现1.2.2 生软件危机的原因1.2.3 解决软件危机的途径1.3 软件工程的产生及其发展1.4 软件工程技术基础1.5 软件工程过程的概念1.6 几种软件过程模型1.6.1 线性顺序模型1.6.2 原型模型1.6.3 快速应用开发模型1.6.4 演化软件过程模型1.7 过程技术1.8 软件重用技术1.9 计算机辅助软件工程工具1.10 小结习题第2章 系统工程基础与软件可行性研究2.1 基于计算机的系统2.1.1 基于计算机的系统概述2.1.2 计算机系统工程2.2 系统需求识别2.2.1 系统分析的目标2.2.2 系统分析过程2.3 可行性研究与分析2.3.1 效益度量方法2.3.2 成本—效益分析2.3.3 技术分析2.3.4 方案制定与评估2.4 系统体系结构建模2.4.1 建立系统结构流程图2.4.2 系统结构的规格说明定义2.5 系统定义与评审2.5.1 系统定义文档模板2.5.2 系统定义的评审2.6 小结习题第3章 结构化需求分析与建模3.1 需求分析3.1.1 需求分析的任务3.1.2 需求分析的步骤3.1.3 需求分析的原则3.2 数据建模3.2.1 实体—关系模型3.2.2 数据建模的其他工具3.3 功能建模3.3.1 数据流图的基本符号3.3.2 数据流与加工之间的关系3.3.3 数据流模型的建立方法3.3.4 建立数据流模型的原则3.4 行为建模3.4.1 状态迁移图3.4.2 Petri网3.5 数据字典3.5.1 数据字典的基本符号3.5.2 数据字典中的条目及说明格式3.5.3 加工逻辑的描述3.5.4 数据字典的建立3.6 结构化需求分析的若干技术3.7 验证软件需求3.7.1 软件需求规格说明的主要内容3.7.2 软件需求的验证3.8 小结习题第4章 结构化软件设计4.1 软件设计中的基本概念和原理4.2 体系结构设计概述4.2.1 体系结构设计的任务4.2.2 体系结构设计中可采用的工具4.2.3 体系结构设计的原则4.2.4 体系结构设计说明书4.3 面向数据流的体系结构设计方法4.3.1 数据流图的类型4.3.2 面向数据流的体系结构设计过程4.4 详细设计概述4.4.1 详细设计的任务4.4.2 详细设计可采用的工具4.4.3 详细设计的原则4.4.4 详细设计说明书4.5 面向数据流的详细设计方法4.6 面向数据流的设计方法4.7 小结习题第5章 软件编码5.1 程序设计语言5.1.1 程序设计语言的分类5.1.2 程序设计语言的特性5.1.3 程序设计语言的选择5.2 编码风格及软件效率5.2.1 编码风格5.2.2 软件效率5.3 程序复杂度的概念及度量方法5.3.1 程序图5.3.2 程序复杂度的度量方法5.4 小结习题第6章 软件测试技术6.1 软件测试基础6.1.1 软件测试的概念、目的和原则6.1.2 软件测试的过程6.1.3 软件测试的方法6.2 白盒测试技术6.2.1 白盒测试概念6.2.2 白盒测试的用例设计6.3 黑盒测试技术6.3.1 黑盒测试概念6.3.2 黑盒测试的用例设计6.4 软件测试计划和测试分析报告6.5 软件测试策略6.5.1 单元测试6.5.2 集成测试6.5.3 确认测试6.5.4 系统测试6.6 小结习题第二篇 面向对象的软件工程第7章 面向对象技术总论7.1 概述——面向对象方法论7.2 面向对象技术的基本概念7.2.1 类7.2.2 对象及对象实例7.2.3 消息机制7.3 面向对象技术的基本特点7.3.1 封装性7.3.2 继承性7.3.3 多态性7.3.4 抽象性7.4 面向对象分析方法7.5 面向对象技术与程序结构7.5.1 概述7.5.2 重构7.5.3 个程序结构改进(重构)的例子7.6 面向对象软件工程7.6.1 传统的面向对象软件工程7.6.2 现代的面向对象软件工程7.6.3 RuP过程7.6.4 LIML简介7.7 设计模式与框架7.7.1 设计模式的基本概念7.7.2 设计模式举例——AbstractFactory(抽象工厂)模式7.7.3 框架的基本概念7.7.4 框架的应用7.7.5 框架开发与软件重用7.7.6 框架的分类及开发原则7.8 基于构件的软件体系结构(Com / Dcom、Corba、Internet)7.9 面向对象分析解决(描述)问题的模式7.10 小结习题第8章 业务模型8.1 业务模型概述8.2 业务建模的目的及内容8.2.1 业务建模的目的8.2.2 业务建模的内容8.3 业务建模流程和任务8.4 业务建模中使用到的UML元素和版型8.4.1 业务系统8.4.2 业务目标8.4.3 业务规则8.4.4 业务参与者8.4.5 业务工人8.4.6 业务实体8.4.7 业务事件8.4.8 业务用例8.4.9 业务用例实现8.4.10 业务用例与业务用例实现的区别8.5 业务建模举例8.5.1 业务目标(部分)8.5.2 组织结构8.5.3 岗位设置和职责8.5.4 业务参与者(部分)8.5.5 业务用例模型8.5.6 业务对象模型8.6 小结习题第9章 需求分析与用例模型9.1 需求分析9.1.1 系统需求和需求描述9.1.2 需求类型9.1.3 需求与用例模型9.2 Actor及其关系9.2.1 Actor9.2.2 如何发现Actor9.2.3 Actor之间的关系9.3 用例及其关系9.3.1 用例(UseCase)9.3.2 用例的版型(Stereotype)及用例观点9.3.3 用例之间的关系9.4 用例图和用例模型9.4.1 参与者与用例之间的关联9.4.2 用例图9.4.3 用例模型9.5 用例规格说明9.5.1 概述9.5.2 用例的描述模板9.5.3 通过用例描述来获取系统的功能9.5.4 用例的描述方法及举例9.6 用例描述中常见的错误举例9.7 使用用例方法发现和确定系统功能需求9.7.1 通过Actor来发现用例9.7.2 通过业务用例和业务流程来发现用例9.7.3 多视角的建模9.8 小结习题第10章 分析设计与对象模型10.1 类和对象的定义IO.1.1 类的定义10.1.2 类的定义讨论10.1.3 类的程序语言定义及和现实世界类的映射10.1.4 类的高级概念10.2 对象、类之间的

关系10.2.1 泛化10.2.2 关联10.2.3 聚集10.2.4 依赖10.3 抽象类和接口10.4 分析模型10.4.1 边界类10.4.2 控制类10.4.3 实体类10.4.4 分析模型举例10.5 设计模型10.6 抽象类和接口的设计原则10.6.1 缺省抽象原则DAP10.6.2 接口设计原则IDP10.6.3 黑盒原则BBP10.6.4 不要具体化超类原则DCSP10.7 类图10.8 领域设计10.8.1 概述10.8.2 领域建模10.8.3 领域分析10.8.4 领域工程10.9 面向对象设计的原则10.9.1 单一职责原则(sRP)10.9.2 开闭原则(OCF)10.9.3 Liskov替换原则10.9.4 依赖倒置原则(DIP)10.9.5 接口分离原则(ISP)10.10 对象模型与关系模型10.10.1 概念模型的表示方法10.10.2 对象模型和关系模型的相互转换10.11 小结习题第11章 系统结构与包模型11.1 包的概念11.2 包之间的依赖关系11.3 包的版型11.4 用包表示的系统高层结构11.5 设计包的原则11.5.1 重用等价原则(REP)11.5.2 共同闭包原则(CCP)11.5.3 共同重用原则(CRP)11.5.4 非循环依赖原则(ADP)11.5.5 稳定依赖原则(SDP)11.5.6 稳定抽象原则(SAP)11.6 小结习题第12章 系统动态特性与对象交互模型12.1 动态模型概述12.2 交互图12.2.1 概述12.2.2 顺序图12.2.3 顺序图中的对象12.2.4 顺序图中的消息12.2.5 建立顺序图的方法和步骤.....第三篇 软件工程项目管理第四篇 软件工程过程模型参考文献

## &lt;&lt;软件工程与开发技术&gt;&gt;

## 章节摘录

插图：第1章 软件工程引论1.1 软件产品的概念与特征1.1.1 软件产品的概念与分类就本质而言，软件就是一个信息转换器，它的功能不外乎是产生、管理、获取、修改、显示或转换信息。

它担任着双重角色，首先，它是一种产品，表达了由计算机硬件体现的计算潜能；其次，它又是开发和运行产品的载体，是计算机控制（操作系统）、信息通信（网络）的基础，也是创建和控制其他软件（软件工具和开发环境）的基础。

对于软件的一种公认的解释是：软件是计算机系统中与硬件相互依存的另一部分，它是包括程序、数据及其相关文档的完整集合。

其中，程序是为实现设计的功能和性能要求而编写的指令序列；数据是使指令能够正常操纵信息的数据结构；文档是与程序开发、维护和使用有关的图文资料。

根据用途划分，软件可以大致划分为如下类别：（1）系统软件：就一般情况来说，系统软件是为其他软件服务的软件。

系统软件与计算机硬件交互频繁，处理大量的确定或不确定的复杂数据，往往需要具有多用户支持、资源精细调度、并发操作管理、多种外部设备接口支持等项功能。

（2）实时软件：管理、分析、控制现实世界中所发生的事件的软件称为实时软件。

它一般有数据采集、数据分析、输出控制等三方面的功能。

实时软件需要保持一个现实任务可以接受的响应时间，即必须保证能够在严格限定的时间范围内对输入做出响应。

（3）商业管理软件：商业信息处理是最大的软件应用领域，包括常规的数据处理软件和一些交互式的计算处理（如POS软件）软件。

它的基本功能是将已有的数据重新构造，变换成一种可以辅助商业操作和管理决策的形式。

在这个过程中，几乎都要涉及到对于大型数据库的访问。

各类管理信息系统（MIS）、企业资源计划（ERP）、客户关系管理（CRM）等都是典型的商业管理软件。

## <<软件工程与开发技术>>

### 编辑推荐

《软件工程与开发技术(第2版)》：普通高等教育“十一五”国家级规划教材。

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>