

<<软件工程基础与实用教程>>

图书基本信息

书名：<<软件工程基础与实用教程>>

13位ISBN编号：9787302287339

10位ISBN编号：7302287333

出版时间：2012-9

出版时间：清华大学出版社

作者：沈文轩 等编著

页数：440

字数：722000

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>

<<软件工程基础与实用教程>>

内容概要

《软件工程基础与实用教程——基于架构与mvc模式的一体化开发》是一本将软件工程基础理论与现代企业级开发技术相结合而诞生的软件工程实用教程。

全书分为上下两篇。

上篇是软件工程基础篇，系统地介绍了软件开发必须遵循的软件工程基本原理和开发方法，包括软件的需求、设计、实现、维护等开发过程和项目管理；下篇是体现本书特色的应用篇，介绍了应用企业级软件架构和mvc设计模式等现代开发技术进行软件开发过程，即基于架构与mvc模式的一体化开发方法。

一体化开发方法是本书的核心内容，它——方面是指从分析、设计到实现全程遵循mvc模式，而不仅仅指某个开发阶段；另一方面是指架构与mvc模式的相互结合，架构可使得mvc的应用走向简单化、实用化。

在一体化的基础上进而介绍了一种正交设计方法，它可以轻松处理复杂领域问题的解耦，将软件结构按mve三大空间进行隔离设计，然后采用组装方式，以两大装配将mve三大空间联系为一个软件整体。正交设计具有很强的规律性、可操作性。

本书在最后介绍了一体化开发的应用案例，该案例不仅体现了一体化的具体应用，也展示了软件开发从抽象到细化的真实过程，让读者深入理解软件工程的一系列重要思想。

《软件工程基础与实用教程——基于架构与mvc模式的一体化开发》可以作为计算机或软件工程等相关专业的本科生、研究生的教材，同时也可作为软件开发领域的各类专业人员的参考书。

<<软件工程基础与实用教程>>

书籍目录

上篇软件工程基础篇

第1章软件工程概述

1.1软件的概念

1.2软件的发展与软件危机的产生

1.3解决软件危机的途径

1.4软件工程的定义

1.5软件工程的内容

1.6软件发展的新阶段与新问题

1.7软件工程的演变发展

本章小结

习题

第2章软件过程

2.1软件过程概述

2.2软件过程与软件生存周期

2.3软件过程与软件工程

2.4软件开发模型

2.5软件过程与开发方法

2.6软件过程改进与cmm

本章小结

习题

第3章软件计划

3.1软件计划概述

3.2问题定义

3.3可行性分析

本章小结

习题

第4章软件需求基础

4.1需求分析概述

4.2需求获取

4.3需求建模

4.4需求说明

4.5需求评审

4.6需求变更管理

本章小结

习题

第5章软件设计基础

5.1软件开发的逐步求精

5.2软件设计过程和任务

5.3模块化与信息隐藏

5.4详细设计

本章小结

习题

第6章面向结构化从分析到设计

6.1结构化开发概述

6.2面向数据流的结构化分析

<<软件工程基础与实用教程>>

6.3面向数据流的结构化设计

本章小结

习题

第7章面向对象从分析到设计

7.1面向对象开发简介

7.2统一建模语言uml

7.3面向对象分析(ooa)

7.4面向对象设计

本章小结

习题

第8章软件实现

8.1编码

8.2测试

本章小结

习题

第9章软件维护

9.1软件维护概述

9.2软件维护的分类

9.3软件的可维护性

9.4软件维护的实施过程

9.5配置管理

本章小结

习题

第10章软件项目管理

10.1项目和项目管理的基本概念

10.2软件度量

10.3软件估算

10.4开发进度计划

10.5团队组织与人力资源计划

本章小结

习题

下篇软件工程实用篇

第11章架构与mvc模式

11.1软件架构

11.2架构的分层

11.3java ee架构简介

11.4面向企业集成的架构简介

11.5mvc模式

11.6架构分层与mvc分层

11.7开源架构ssh简介

本章小结

习题

第12章面向架构与mvc的一体化开发方法

12.1一体化开发介绍

12.2一体化与正交设计

本章小结

习题

第13章 一体化开发方法应用实例

13.1 用户需求

13.2 ooa分析过程

13.3 ood设计过程

本章小结

习题

参考文献

本书特色

教学建议

跋

章节摘录

版权页：插图：2) 更合理的生命周期模型 RUP模型图中初始、细化、构造、交付4个阶段表达了RUP的生命周期。

重新审视2.2.1节中介绍的基本生存周期模型，该模型将整个生命周期划为三个时期。

问题定义、需求分析、设计、实现、维护等各项活动分别属于某个时期，而且是非此即彼的界限。

而RUP模型模糊了这种界限，单独看某种活动的阴影线，不是随阶段的进展戛然启动或截止，而是在不同阶段所占的比重不同，表达出逐步介入或结束的过渡时间，展现了与其他活动的合作关系。

如“实现”活动在“细化阶段”开始介入，在“构造阶段”达到最大，在“交付阶段”逐渐减少。

这样的表达比基本生存周期模型要合理得多。

比如“需求分析”活动，在做项目计划时首先必须开展需求分析，而在开发时期还要继续完善需求分析。

用2.2.1节中介绍的基本模型，“需求分析”活动要么划分在“计划时期”，要么划分在“开发时期”，而划分在哪个时期都有它的理由又都不甚合理，但也没法表达为两个时期共有。

用RUP模型就不必为某个活动属于哪个Phase而做无谓的争论了。

3) 结束关于RUP没有维护活动的争论 有的研究者认为RUP模型忽略了维护，其实不然。

RUP一个迭代周期(cycle)经过4个阶段(Phase)，相比2.2.1节介绍的基本生存周期模型，表面上这4个阶段相当于基本生存周期模型的计划时期+开发时期，似乎缺少了重要的运行维护。

但实际上RUP模型中的“配置和变更管理”就是针对软件维护的。

软件维护就是一次开发过程的迭代，衍生一个新的版本。

只要把“配置和变更管理”做好，维护就是重新走一遍初始、细化、构造、交付4个阶段而已。

4) 完美的增量开发——双重迭代 RUP一个重要的特点是迭代式开发，并且是双重迭代，迭代效果更好。

按时间组织经过初始、细化、构造、交付4个阶段的迭代是一个大的迭代，迭代的增量是提交一个新的软件版本；在各阶段内各活动还可能经过若干次迭代，这是辅助迭代，辅助迭代产生一个增量，通过迭代逐步逼近里程碑，如图2-9所示。

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>