<<UML 2.2面向对象分析与设计>>

图书基本信息

书名：<<UML 2.2面向对象分析与设计>>

13位ISBN编号：9787302304241

10位ISBN编号：7302304246

出版时间：2013-1

出版时间：清华大学出版社

作者：(英)班尼特(Bennett, S.),(英)麦克罗布(McRobb, S.),(英)法默(Farmer, R.)

译者：李杨

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>

<<UML 2.2面向对象分析与设计>>

内容概要

本书对前一版进行了修订，反映了信息系统开发中最前沿的方法。

Bennett、McRobb和Farmer撰写的这本书是面向对象分析与设计领域的经典图书，是本科生和研究生“系统分析和设计”课程重要的教学用书。

本书为使用UML 2.2中的主要技术进行面向对象分析与设计给出了清晰实用的结构，遵循迭代和增量型方法(它们基于业内标准的统一过程)，将系统分析和设计置于整个系统生命周期的背景中。

本书分为4部分：第1部分为信息系统的分析和设计以及面向对象提供了背景；第2部分关注需求获取和系统分析活动以及UML的基本标记法；第3部分涉及系统架构和设计活动，以及对象设计的UML标记法；第4部分介绍系统的实现，如何组织系统生命周期，以及如何开发可重用

<<UML 2.2面向对象分析与设计>>

作者简介

作者：（英国）班尼特（Simon Bennett）（英国）麦克罗布（Steve McRobb）（英国）法默（Ray Farmer）译者：李杨 Simon Bennett是De Montfort大学技术学院的培训顾问，他在这里提供UML、分析和设计，以及系统架构方面的培训。

他是计算机智能中心的准会员。

之前作为企业架构师供职于Celesio AG，作为ICT主管供职于Leicester议会的复兴和文化事务部，并且在1999年之前作为首席讲师任教于De Montfort大学。

Simon是Schaum's Outline of UML（2nd edition）书的合著者，该书也由McGraw.Hill出版。

Steve McRobb是De Montfort大学信息学院的首席讲师。

他教授面向对象系统分析与设计课程十几年，目前是“计算、信息技术和信息系统管理”研究生课程的课程负责人。

他最近的研究范围主要集中在在线隐私以及ICT对权利关系的影响方面。

Steve是Yorkshire Dales国家公园的前任首席行政主任。

Ray Farmer是Coventry大学工程和计算学院的副院长。

他的研究兴趣包括信息系统分析与设计、面向服务架构以及工程和计算教育方面的教学研究。

他定期作为英国和国际上面向对象分析与设计方面的咨询师。

Ray之前在De Montfort大学的信息系统学院担任不同的职位。

<<UML 2.2面向对象分析与设计>>

书籍目录

案例A1 Agate公司案例研究——简介 A1.1 Agate公司介绍 A1.2现有计算机系统 A1.3使用当前系统的业务活动 A1.4需求总结 案例B1 FoodCo公司案例研究——简介 B1.1 FoodCo公司介绍 B1.2现如今的FoodCo公司 B1.3建议 第1章信息系统 1.1 介绍 1.2信息系统的历史 1.3信息系统现状 1.4何为系统 1.4.1系统思考 1.4.2系统思考的益处 1.4.3信息系统发展过程中的系统 1.5信息与信息系统 1.5.1信息 1.5.2组织中信息系统的工作 1.5.3信息技术 1.6成功战略 1.6.1确认商业战略 1.6.2信息系统的贡献 1.6.3信息系统和信息技术战略 1.7本章小结 问题回顾 案例研究、练习和项目 拓展阅读 第2章信息系统开发面临的挑战 2.1 介绍 2.2挑战 2.2.1终端用户的视角 2.2.2客户的视角 2.2.3开发者的视角 2.3 出错的原因 2.3.1质量问题 2.3.2生产率问题 2.4道德层面 2.4.1项目内的道德问题 2.4.2更为广泛的道德问题 2.5失败的代价 2.6本章小结 问题回顾 案例研究、练习和项目 拓展阅读 第3章面对挑战 3.1 介绍 3.2 问题响应 3.2.1质量问题 3.2.2安装和运行问题 3.2.3生产率问题 3.3项目的生命周期 3.3.1瀑布型生命周期模型 3.3.2原型 3.3.3迭代和增量型开发 3.4方法论 3.4.1统一软件开发过程 3.4.2敏捷方法 3.5信息系统开发管理 3.6用户参与 3.7软件开发工具 3.7.1模型和技术支持 3.7.2软件构建 3.7.3使用软件开发工具的益处和困难 3.8本章小结 问题回顾 案例研究、练习和项目 拓展阅读 第4章面向对象 4.1 介绍 4.2基本概念 4.2.1对象 4.2.2类和对象 4.2.3类成员 4.2.4一般化和特殊化 4.2.5封装、信息隐藏和消息传递 4.2.6多态 4.2.7对象状态 4.3面向对象的起源 4.4当前的面向对象语言 4.5本章小结 问题回顾 案例研究、练习和项目 拓展阅读 第5章建模的概念 5.1 介绍 5.2模型和图 5.2.1模型的含义 5.2.2图 5.2.3模型和图的区别 5.2.4 UML中的模型 5.2.5开发模型 5.3绘制活动图 5.3.1活动图的目的 5.3.2活动图的标记法 5.4开发过程 5.4.1基本原则 5.4.2主要活动 5.5本章小结 问题回顾 案例研究、练习和项目 拓展阅读 第6章需求获取 6.1 介绍 6.2用户需求 6.2.1 当前系统 6.2.2新的需求 6.3事实发现技术 6.3.1背景阅读 6.3.2访谈 6.3.3观察 6.3.4文档抽样 6.3.5调查问卷 6.3.6记住上述技术 6.3.7其他技术 6.4用户参与 6.5 文档需求 6.6用例 6.6.1 目的 6.6.2标记法 6.6.3使用原型支持用例 6.6.4 CASE工具支持 6.6.5使用用例图的业务模型 6.6.6测试和用例 6.7需求获取和建模 6.8本章小结 问题回顾 案例研究、练习和项目 拓展阅读 案例A2 Agate公司案例研究——需求模型 A2.1 介绍 A2.2需求列表 A2.3执行者和用例 A2.4术语表 A2.5最初架构 A2.6需求建模活动 第7章需求分析 7.1 介绍 7.2分析模型 7.2.1分析模型和其他模型的区别 7.2.2好的分析 7.3分析类图：概念和标记法 ” 7.3.1类和对象 7.3.2特性 7.3.3特性和状态 7.3.4实例之间的链接 7.3.5类之间的关联 7.3.6关联和状态 7.3.7多重性 7.3.8操作 7.3.9操作和状态 7.3.10分析类图的稳定性 7.4用例实现 7.5绘制类图 7.5.1健壮性分析 7.5.2分析类stereotype 7.5.3确认类 7.5.4从通信图到类图 7.5.5查找对象和类的其他方法 7.5.6添加和定位特性 7.5.7添加关联 7.5.8确定多重性 7.5.9寻找操作 7.5.10初始操作分配 7.6 CRC卡 7.7整合分析类图 7.8本章小结 问题回顾 案例研究、练习和项目 拓展阅读 案例A3 Agate公司案例研究——需求分析 A3.1 介绍 A3.2用例实现 A3.3整合分析类图 A3.4需求分析活动 第8章完善需求模型 8.1介绍 8.2软件和规范重用 8.2.1为何重用 8.2.2难以重用的原因 8.2.3面向对象对重用的贡献 8.3进一步向结构中添加内容 8.3.1寻找和建模一般化 8.3.2寻找和建模组合 8.3.3组合或聚集与一般化的合并 8.3.4组织分析模型——包和依赖关系 8.4重用软件组件 8.4.1组件的UML标记法 8.4.2基于组件的开发 8.4.3组件建模实例 8.5软件开发模式 8.5.1模式的起源 8.5.2什么是软件模式 8.5.3分析模式 8.6本章小结 问题回顾 案例研究、练习和项目 拓展阅读 第9章对象交互 9.1介绍 9.2对象交互和协作 9.3交互顺序图 9.3.1基本概念和标记法 9.3.2管理顺序图 9.3.3分支 9.3.4延续 9.3.5异步消息 9.3.6时间约束 9.3.7实时系统和并发的建模 9.3.8准备顺序图的指导原则 9.4通信图 9.4.1基本概念和标记法 9.4.2通信图中的信息标签 9.5交互概览图 9.6时序图 9.7模型一致性 9.8本章小结 问题回顾 案例研究、练习和项目 拓展阅读 第10章规范对象操作 10.1介绍 10.2操作规范的角色 10.3合同 10.4描述操作逻辑 10.4.1非算法型方法 10.4.2算法型方法 10.5对象约束语言 10.6创建操作规范 10.7本章小结 问题回顾 案例研究、练习和项目 拓展阅读 第11章规范控制 11.1介绍 11.2状态和事件 11.3基本标记法 11.4深度标记法 11.4.1组合状态 11.4.2并发状态 11.4.3进入和退出伪状态 11.4.4交叉和选择伪状态 11.4.5历史伪状态 11.4.6状态机的特殊化 11.5准备状态机 11.5.1行为型方法 11.5.2生命周期方法 11.6协议型和行为型状态机 11.7一致性检查 11.8质量准则 11.9本章小结 问题回顾 案例研究、练习和项目 拓展阅读 案例A4 Agate公司案例研究——深入分析 A4.1介绍 A4.2顺序图 A4.3状态机 A4.4操作规范 A4.5对类图的进一步修订 A4.6需求分析的深入活动 第12章设计 12.1介绍 12.2设计和分析有何不同 12.2.1设计 12.2.2迭代型生命周期中的设计 12.3逻辑设计

<<UML 2.2面向对象分析与设计>>

和物理设计 12.4系统设计和详细设计 12.4.1 系统设计 12.4.2详细设计 12.5设计的质量和目標 12.5.1 目标和约束 第13章系统架构 第14章详细设计 第15章设计模式 第16章人机交互 第17章设计边界类 第18章数据管理设计 第19章实现 第20章软件重用 第21章软件开发过程 附录A标记法汇总 附录B部分解决方案和答案指导 术语表

章节摘录

版权页：插图：本书的作者在购买一双鞋子的时候，店员为正确录入商品价格而费尽力气。会有新的现金金额录入系统，但是因为这双鞋子是促销产品，所以录入起来很困难。

顾客购买这种鞋子，并且免费附赠一双袜子。

因为袜子也是常规商品，所以需要袜子也进行记录。

这意味着袜子通过促销被“卖出”，尽管是免费的。

处理这种问题的一个简单方法就是在出售时由助理人员将价格置零。

助理人员会尝试这样做，但是系统可能会特别阻止商品以零售价卖出。

助理求助于经理。

在经过一些类似情况后，会发现解决这种交易的唯一方法就是把鞋子的价格降低1便士，而袜子以1便士的价格售出，因此总的售价是正确的。

既然店员明白如何做，那么在以后类似的情况下，他们处理起来也就简单多了。

但是处理一项经常出现的任务是很繁琐的。

这种糟糕的设计还有很多例子，它们给用户带来了不少不快，并且浪费了时间。

“这个系统看起来很美——但是能用来完成一点有用的事情吗？”

”系统可能设计得美观，易于使用，但是仍然不能处理“正确的”事情，这对于应该由系统执行的任务来说是一个问题。

例如图书馆分类查询系统，如果只有在书的名称和作者姓名都拼写无误的情况下才能检索出书号，那么这一系统的功能就受限。

读者在查询的时候很可能不知道书的名称，即便知道作者的姓名，也可能拼写错误。

系统失败的另一个例子是，虽然满足了用户的需求，但是系统性能很差（这一点与刚才系统可用性的问题重复）。

系统对用户来说，作用可能值得怀疑，因为系统要求他们按照看起来毫无意义的方式工作。

用于佐证例子虽然很古老，但是依旧很有价值，因为作者可以清晰地描述出来。

设计仓库管理系统的部分原因是，提高管理员对仓库中紧张空间的使用控制。

工人发现新的系统为了以最佳的方式最大化存储空间的利用率，但这却剥夺了他们的自由决定权：因为他们会看到，节省这些空间能为公司带来改善，他们找到了使用系统的方法。

他们因为态度不好而饱受管理人员的责难，而且作为部门的主要职员，工人被自认为，不必要的废文和程序而烦恼，但是使用它们却又是他们对公司的责任。

如果软件错误和失败危及生命，那么将更加令人忧心。

一个极端的例子是伦敦救护车服务计算机辅助调度（London Ambulance Service Computer Aided Dispatch, LASCAD）系统，该系统在1992年发布之后不久就被弃置。

估计总的开发成本为4300万英镑。

该系统本来是为了加快为紧急情况派出救护车的流程，但实际上却降低了响应时间。

在出现几次因为病人长时间等待而导致死亡的情况之后，该系统被弃置了。

虽然这些声明从未被证实过，但是继续运行该系统的风险是不可接受的。

关于是否是因为软件错误而导致英国皇家空军Chinook直升机在Kintyre半岛的Mull坠机的争论，至今尚未停息。

机上所有的29人全部丧生，包括高级警官和军方情报长官。

英国政府能接受的定论是飞行员的疏忽大意。

然而，事故调查组的结论是机组人员可能因主要的技术（例如软件）故障而分心。

Computer Weekly的一系列报道以及British TV的4频道新闻，宣称英国国防部内部报告已经对此类型直升机在特定情况下引擎控制软件的稳定性产生了顾虑。

英国上议院选举委员会建议驳回国防部的观点。

在最近，2007年6月份，又有新的证据可能推翻官方结论（Collins，2007），但是至今政府拒绝重启调查，而将责任继续归咎于飞行员。

<<UML 2.2面向对象分析与设计>>

编辑推荐

《国外计算机科学经典教材:UML 2.2面向对象分析与设计(第4版)》对前一版进行了修订,反映了信息系统开发中最前沿的方法。

Bennett、McRobb和Farmer撰写的这本《国外计算机科学经典教材:UML 2.2面向对象分析与设计(第4版)》是面向对象分析与设计领域的经典图书,是本科生和研究生“系统分析和设计”课程重要的教学用书。

<<UML 2.2面向对象分析与设计>>

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>