

<<软件工程>>

图书基本信息

书名：<<软件工程>>

13位ISBN编号：9787121130557

10位ISBN编号：7121130556

出版时间：2011-4

出版时间：弗利特(Hans Van Vliet)、郭文明 电子工业出版社 (2011-04出版)

作者：弗利特

页数：447

译者：郭文明

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>>

内容概要

《软件工程：原理与实践（第3版）》是荷兰著名软件工程学家Hans van Vliet多年从事软件工程教学和经验的提炼和总结，书中引用了大量的文献和生动的事例，详尽介绍了关于如何设计软件的知识，以及在编程领域之外的、一名全面发展的软件工程师应该具备的素养。

《软件工程：原理与实践（第3版）》全书分三大部分，第一部分详细介绍了软件工程中的软件管理问题；第二部分重点说明软件生命周期各个阶段的相关技术；第三部分给出了软件工程中的现代论题，包括用户界面设计、基于中间件的软件设计、面向服务的软件设计及全球化软件开发。

《软件工程：原理与实践（第3版）》知识点丰富，内容新颖，讲解严谨，被欧洲和北美许多大学采用作为教材，是软件工程领域的研究、教学、管理和技术人员十分难得的最新教材、实践指南和参考文献。

作者简介

作者：(荷兰)弗利特(Hans Van Vliet) 译者：郭文明

<<软件工程>>

书籍目录

目录第1章 导论 11.1 什么是软件工程 31.2 软件开发生的过程 71.3 维护还是革新 101.4 失败案例 111.5 软件工程的道德 161.6 君往何处 171.7 小结 191.8 扩展阅读 19第2章 软件工程管理概述 242.1 软件开发项目的计划 252.2 软件开发项目的控制 272.3 小结 29第3章 软件生命周期回顾 313.1 瀑布模型 323.2 敏捷方法 343.3 统一软件开发过程 433.4 模型驱动架构 453.5 插曲:维护或演进 453.6 软件产品线 483.7 过程模型 493.8 小结 513.9 扩展阅读 51第4章 配置管理 544.1 职责与任务 544.2 配置管理计划 584.3 小结 594.4 扩展阅读 59第5章 人员管理与团队组织 615.1 人员管理 625.2 团队组织 655.3 小结 705.4 扩展阅读 71第6章 软件质量管理 726.1 度量与数字 736.2 质量属性的分类 776.3 软件质量评价 816.4 软件质量体系 846.5 软件质量保证 846.6 能力成熟度模型 856.7 使用入门 906.8 小结 926.9 扩展阅读 93第7章 成本估算 957.1 算法模型 977.2 成本估算指南 1097.3 人力资源分配 1127.4 敏捷开发中的成本估计 1147.5 小结 1157.6 扩展阅读 116第8章 项目计划与控制 1188.1 项目控制的系统观念 1188.2 软件开发项目的分类法 1208.3 风险管理 1238.4 项目规划和控制技术 1258.5 小结 1298.6 扩展阅读 130第9章 需求工程 1329.1 需求获取 1369.2 需求文档和管理 1509.3 需求说明的技术 1569.4 验证和确认 1589.5 小结 1599.6 扩展阅读 160第10章 建模 16310.1 经典建模技术 16410.2 对象及相关材料 16810.3 统一建模语言 17210.4 小结 17910.5 扩展阅读 180第11章 软件体系结构 18111.1 软件体系结构和软件生命周期 18311.2 体系结构设计 18411.3 体系结构视图 18811.4 体系结构风格 19011.5 软件体系结构评估 19811.6 小结 20011.7 扩展阅读 201第12章 软件设计 20312.1 设计依据 20512.2 经典设计方法 21912.3 面向对象分析和设计方法 23012.4 如何选择设计方法 23712.5 设计模式 24012.6 设计文档 24412.7 验证与确认 24612.8 小结 24612.9 扩展阅读 249第13章 软件测试 25313.1 测试目标 25613.2 测试和软件生命周期 26113.3 验证和确认规划以及文档编制 26513.4 人工测试技术 26613.5 基于覆盖率的测试技术 27013.6 基于故障的测试技术 27413.7 基于错误的测试技术 27613.8 测试技术的比较 27713.9 测试阶段 28213.10 软件可靠性评估 28313.11 小结 28813.12 扩展阅读 289第14章 软件维护 29314.1 维护种类回顾 29514.2 维护问题的主要原因 29614.3 逆向工程与重构 29914.4 重温软件演进 30614.5 组织和管理问题 30714.6 小结 31614.7 扩展阅读 317第15章 软件工具 31915.1 工具包 32115.2 以语言为中心的环境 32215.3 集成环境和工作平台 32315.4 以过程为中心的环境 32715.5 小结 32815.6 扩展阅读 329第16章 用户界面设计 33216.1 用户界面在哪里 33416.2 什么是用户界面 33516.3 人机交互中人的因素 33616.4 人机交互中模型的作用 33916.5 交互系统的设计 34516.6 任务分析 34716.7 用户界面的细节规范 35116.8 评价 35216.9 小结 35516.10 扩展阅读 356第17章 软件重用性 35817.1 重用的维度 35917.2 中级产品重用 36117.3 重用和软件的生命周期 36517.4 重用工具和技术 36717.5 软件重用前景 37017.6 软件重用的非技术因素 37317.7 小结 37617.8 扩展阅读 377第18章 基于组件的软件工程 37918.1 为什么要有基于组件的软件工程 38018.2 组件模型和组件 38118.3 基于组件开发过程和组件生命周期 38718.4 基于组件的开发中的架构方法 39118.5 小结 39218.6 扩展阅读 392第19章 面向服务 39419.1 服务、服务描述和服务通信 39519.2 面向服务的架构 (SOA) 39819.3 Web 服务 39919.4 面向服务软件工程 40519.5 小结 40619.6 扩展阅读 407第20章 全球化软件开发 40920.1 全面系统开发的挑战 41020.2 如何克服差距 41420.3 小结 41720.4 扩展阅读 418中英文术语对照表 419参考文献 426

<<软件工程>>

章节摘录

版权页：插图：软件开发项目的进展（时间方面）是难以估量的。

在目标系统完成以前，仅有一堆白纸。

像“90%的编码已经编译完成”的言论只能视为一小撮盐这样的调味品，现实事务中有太多的这种情况。

第1章中介绍的软件开发阶段划分方式和各个阶段的要素变量，均旨在为管理者提供衡量和控制进展的工具。

软件开发系统的时间消耗和对人力资源的需求，与系统的大小有关系，即使我们可能采取分配任务给更多的人力来缩短开发时间，大系统需要的开发时间也显然要比小系统更长。

软件开发项目的控制在人力与时间方面似乎没有替代性，因为增加更多的人力以缩短开发时间并不能轻松实现。

越多的人涉入其中，也就需要更多的时间去协调和交流。

超过某一临界点时，增加更多的人力实际上会延长开发时间。

一些时间控制问题在布鲁克斯法案中有着经典表述：“在新项目中增加人手，只会拖延它的时间”。

第7章将回到这个问题的讨论。

资料需要管理，尤其是文档。

除了技术和用户文档以外，同样包括项目自身的文档。

涉及到的项目文档包括以下内容：当前事态、达成一致的变动、已作出的决议，这类文档最好整理在配置管理内容中。

在灵活的小项目中，很少会注意到开发中的文档，许多必要的知识是不言而喻的，它存在于项目开发人员的头脑中。

但从道理上讲，无论站在提高系统可读性角度看，还是站在方便系统发布给用户角度看，文档资料理所当然地应当提供。

开发团队的所有成员都必须明确知道他们各自的角色以及他们被寄予的期望。

所有工作人员都明确知道自己的任务，这一点非常重要，没有说出口以及含糊不清的期望将会导致独立团队有意无意地制定他们自己的目标。

这些状态都值得项目经理持续注意。

团队组织与人员的合作程度，部分来说依赖于项目的属性和它的环境。

当组建项目团队时，这种依赖关系必须加以认识并且考虑。

质量是最为重要的一个方面。

客户不满足于由计算机专家提供的纯技术解决方法，他们需要系统能够满足他们需求。

软件本身的质量需求和软件开发过程中的质量控制往往会发生冲突。

在架构设计时期，质量需求一般在与相关利益者的对话中实现平衡。

在项目中，我们必须理清面对的质量需求。

对于质量的要求应建立在规则的基础上，以便能够控制开发过程中的行为。

质量不是附加的特性，是应该控制的。

控制开支（资金方面）很大程度上意味着控制员工支出。

尽管硬件和工具的成本不应被忽视，但这些成本在项目中一般能够精确地估算好。

而且，它们的确要比员工开支少很多。

估算软件成本意味着需要估算建设软件所需的劳动力。

所需人力十分依赖于软件的规模，例如所需要的编码的总数。

更多情况下，其他因素也影响着软件的成本或生产软件的生产力。

一个均衡的有着丰富经验的团队，将比一个新设立的没有经验的团队更有生产力。

极端严密的质量约束限制，例如非常高可靠的或一个非常快速的反应时间，同样也可能影响生产力。

面对人力资源需求，大量的关于不同角色的量化成本模型提了出来（见第7章）。

相对于比较精确地估算软件规模大小以及软件的成本，人们更倾向于首先给出一个成本极限，再根据

<<软件工程>>

用户需求、时间限制进行平衡。
如果工作量加大，就压缩用户需求。
在这三者之间达成一致后再决定是否投资以及目标完成时间。
这种增量成本估算方法比较适合小型的灵活项目。

<<软件工程>>

编辑推荐

《软件工程:原理与实践(第3版)》对于成功的软件开发而言,“软件工程”是一个富有挑战性的话题。

《软件工程:原理与实践(第3版)》是完整介绍软件生命周期各个方面的一部力作,着重阐述了软件项目能够在不超预算的情况下按时开发完成所需要的各种技术和因素。

《软件工程:原理与实践(第3版)》第三版的亮点主要包括:完整展现了需求工程和软件体系结构;增加了基于组件的软件工程、面向服务的软件开发和全球化软件开发的相关内容;扩展了影响软件开发的人文因素和社会因素的论述;比较了传统的重量级开发方法(瀑布模型)与新型的轻量级开发方法(敏捷开发方法和极限编程)。

《软件工程:原理与实践(第3版)》可以作为初级和高级软件工程课程的教材,适合于从事大规模软件开发和维护的各类人员。

对于将该书用于教学和自修的读者,《软件工程:原理与实践(第3版)》还配有专门的网站提供该书的相关附加资源。

教师可通过填写《软件工程:原理与实践(第3版)》末尾的“教学支持说明”信息获取相关资源。

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>