

《软件开发过程与配置管理》

图书基本信息

书名：《软件开发过程与配置管理》

13位ISBN编号：9787121123078

10位ISBN编号：712112307X

出版时间：2011-2

出版社：刘江华、王立、马玲、等 电子工业出版社 (2011-02出版)

页数：403

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：www.tushu000.com

《软件开发过程与配置管理》

前言

软件开发过程和配置管理是软件开发产业中的一个永恒的话题。随着软件产品日益复杂化，软件开发团队不断庞大化，软件配置管理在软件开发过程及整个软件生命周期中越发处于举足轻重的地位。优秀的软件配置管理方案有助于优化软件开发过程，提高软件产品质量，改进团队开发效率；低效的软件配置管理方案将使软件开发过程紊乱，项目进度失控，团队协作不畅，使所开发出的产品漏洞百出，最终甚至导致开发机构人心涣散，业务发展举步维艰。近些年来，随着我国科技和经济的繁荣发展，我国软件开发企业急剧增多，规模不断扩大，涉及行业和领域也日益广泛。如何实现软件开发团队在数量和质量上同步提高是多数企业所有者或管理者十分关心的问题。笔者十几年前刚踏入软件开发行业时，软件开发项目中开发工具的应用还极其有限，开发过程和配置管理工具的应用就更少了。笔者当时在一个规模较小的系统集成公司从事软件开发工作，参与的开发项目都是通过手工的方式进行代码管理的。通常的做法是每个开发人员每天或定期将自己负责的代码复制一份到软盘上，交给项目负责人或集成人员去拼装整个项目的代码，而公共模块的修改完全依靠相关开发人员手工进行合并。由于缺乏工具的支持，根本不可能在开发项目中形成成熟的开发流程和稳定的开发团队，最终的结果是不断地招募新的开发人员重复地进行失败的项目。本书中虚构的“陷入泥潭的项目”就是这种早期开发项目的缩影。

《软件开发过程与配置管理》

内容概要

《软件开发过程与配置管理:基于Rational的敏捷方案设计与应用》以软件配置管理对软件开发过程和团队开发的支持为主旨，借助IBM Rational软件开发和配置管理工具ClearCase和ClearQuest，详尽阐释了一种能有效支持软件开发过程的集成式软件配置管理方案的设计、实现和应用。全书分为4篇，分别为理论与背景、设计与实现、实施与应用、扩展与升华。全书力图以由浅入深、由理论到实践、由传统到现代的方式为读者展示软件开发过程和配置管理的方法和工具的使用。

《软件开发过程与配置管理:基于Rational的敏捷方案设计与应用》可供软件开发过程和配置管理系统设计和实施专家、软件开发人员、软件开发管理者，以及转型期软件企业的决策者参考，也供软件工程和软件配置管理教学研究人员参考。

《软件开发过程与配置管理》

作者简介

刘江华，2002年加入IBM中国开发中心（CDL）并服务至今。先在WPLC部门负责软件开发过程和配置及构造管理的流程设计和系统实现及运营维护等相关工作，之后转入Rational实验室服务部从事软件开发方法、工具和实践等方面的咨询工作。加入IBM之前先后在北京吉通、朗讯科技中国贝尔实验室、清华同方等IT研发机构从事软件开发实践和相关管理工作。王立，1998年毕业于西安交通大学电子工程系，2002年于西安交通大学计算机科学与技术系获硕士学位，2005年获博士学位。同年7月加入IBM中国开发中心，先后在IBM Lotus和Rational产品线从事企业级协作与软件开发过程的研发工作。目前服务于IBM中国系统与科技中心，从事平台管理系统软件的研发工作。马玲，2006年毕业于北京航空航天大学计算机系，并获硕士学位。之后加入IBM中国开发中心并服务至今，在Lotus部门，负责开发过程，配置管理以及构造管理的研发工作，并从事系统管理和运营维护等相关工作。

第1篇 理论与背景第1章 典型软件开发过程及主要模型1.1 软件生命周期及主要过程1.1.1 软件计划阶段1.1.2 需求分析阶段1.1.3 软件设计阶段1.1.4 软件编码阶段1.1.5 软件测试阶段1.1.6 软件维护阶段1.2 软件开发过程模型1.2.1 瀑布模型1.2.2 螺旋模型1.2.3 RUP模型1.2.4 敏捷开发第2章 软件配置管理的基本概念原理2.1 软件配置管理概览2.1.1 什么是软件配置管理2.1.2 软件配置管理的起源与发展2.1.3 软件配置管理能解决哪些问题2.2 软件配置管理的任务和活动2.2.1 软件和配置项2.2.2 标识2.2.3 变更控制2.2.4 状态报告2.2.5 配置审计2.3 软件配置管理的核心要素2.3.1 版本和版本树2.3.2 里程碑与基线2.3.3 软件配置库2.3.4 工作空间2.3.5 变更请求与变更集2.3.6 软件配置管理工具2.4 软件配置管理的主要过程2.4.1 配置项标识与存储过程2.4.2 版本管理过程2.4.3 变更控制过程2.4.4 基线管理过程2.5 软件配置管理中的角色2.5.1 配置管理专职人员2.5.2 机构运营管理人员2.5.3 项目开发人员第3章 常用软件配置管理工具简介3.1 软件配置管理工具的发展历程3.2 面向文件的配置管理工具及功能3.2.1 早期的软件配置管理工具3.2.2 CVS3.2.3 VSS3.3 面向变更集的配置管理工具及功能3.3.1 ClearCase UCM3.3.2 其他面向变更集的工具3.4 面向开发流程的配置管理工具及功能3.4.1 集成的ClearCase和ClearQuest3.4.2 其他面向开发流程的工具第2篇 设计与实现第4章 面向软件开发过程的软件配置管理系统4.1 陷入泥潭中的项目——没有完善配置管理的软件开发4.2 软件开发过程的基本要素与本质特性4.2.1 什么是软件开发过程4.2.2 软件开发过程的四要素4.2.3 软件开发过程的本质特性——复杂性4.3 软件配置管理与软件开发过程要素的关系4.3.1 对人员的支持4.3.2 对任务的支持4.3.3 对方法的支持4.3.4 对产品的支持4.4 软件配置管理驱动软件开发过程的量化和改进4.4.1 度量在软件开发过程中的作用4.4.2 如何建立有效的度量体系4.4.3 软件配置管理为度量提供客观数据4.4.4 基于度量的过程改进4.5 ISO和CMMI中的软件配置管理4.5.1 ISO 9000中对配置管理的要求4.5.2 CMMI中对配置管理的要求4.6 定制配置管理系统以支持软件开发过程4.6.1 角色与职责4.6.2 任务和变更管理4.6.3 过程支持4.6.4 团队开发4.6.5 构造和发布4.6.6 项目管理第5章 基于ClearCase/ClearQuest的方案总体设计5.1 ClearCase/ClearQuest固有的过程支持及可扩展性5.1.1 ClearCase UCM模型5.1.2 基于UCM模型的开发流程5.1.3 ClearQuest简介5.1.4 定制和扩展ClearQuest5.2 完善任务管理5.2.1 定义任务范围和类型5.2.2 设定任务属性5.2.3 确定任务完成状态5.2.4 建立任务间的关系5.2.5 任务实施过程中的数据记录5.2.6 流程方案中的任务设计5.3 明晰开发团队角色5.3.1 软件项目中的角色和职责5.3.2 角色在任务处理中的分工5.3.3 流程方案中的角色设计5.4 加强产品和基线管理5.4.1 软件产品的演化过程5.4.2 通过基线记录产品的演化过程5.4.3 管理复杂的产品和基线5.4.4 流程方案中的基线设计5.5 收集开发过程管理数据5.5.1 常用过程管理数据5.5.2 如何收集生成数据5.5.3 如何获取与展示数据第6章 工作流程的详细设计6.1 变更记录的流程处理与状态流图6.1.1 缺陷更改记录的状态流程6.1.2 功能更改记录的状态流图6.2 变更流程的状态模型6.2.1 定义状态模型6.2.2 使用状态转移矩阵图6.2.3 添加、更改和删除状态6.3 流程中的动作机制6.3.1 变更记录的动作6.3.2 ClearQuest支持的动作类型6.3.3 添加新的动作6.3.4 创建状态迁移6.3.5 更改动作6.3.6 通过添加Hook定制动作6.3.7 使用默认的动作6.3.8 删除动作第7章 数据模型设计7.1 ClearQuest 数据存储机制7.1.1 模式库7.1.2 用户数据库7.2 ClearQuest数据库设计7.2.1 数据库设计简介7.2.2 记录类型7.2.3 数据类型7.2.4 数据项操作7.3 新方案数据模型7.3.1 基础数据模型7.3.2 数据表设计7.3.3 关联数据表7.4 数据项设计7.4.1 任务相关数据7.4.2 产品相关数据7.4.3 人员相关数据第8章 变更控制信息设计8.1 基本的变更控制信息8.1.1 标志信息8.1.2 范围信息8.1.3 类型信息8.1.4 状态信息8.1.5 描述信息8.1.6 人员信息8.1.7 执行属性8.2 变更附加信息8.2.1 附加开发数据8.2.2 附加测试数据8.2.3 附加发布数据8.2.4 附加二进制信息8.3 任务关联信息8.3.1 任务的层次结构8.3.2 任务与代码的关系8.3.3 任务与结果关系8.3.4 特定的任务关系8.4 过程相关信息8.4.1 任务状态8.4.2 任务决策数据8.4.3 任务实施数据8.4.4 任务验证数据8.4.5 任务历史第9章 基于角色的权限控制9.1 变更控制委员会9.2 开发相关角色9.2.1 开发小组负责人9.2.2 开发人员9.3 测试相关角色9.3.1 测试负责人9.3.2 测试人员9.4 集成人员9.5 项目管理者9.6 角色权限控制的实现9.6.1 确定允许的操作9.6.2 动作的权限控制9.6.3 数据行为设置第10章 实现对项目整体的支持10.1 UCM中的项目和相关对象10.1.1 项目10.1.2 流10.1.3 活动10.2 UCM对项目模块结构的支持10.2.1 确定版本控制工件 (Artifact) 10.2.2 UCM组件10.2.3 如何划分组件10.3 软件开发项目与UCM项目的对应10.3.1 UCM项目的应用方式10.3.2 ClearQuest中的开发项目10.3.3 关联UCM项目与ClearQuest中项目10.4 任务分解与团队分工10.4.1 按产品功能划分任务10.4.2 基于角色和功能模块的人员组织10.4.3 复杂任务支持第11章 关键变更控制点的设计与实现11.1 UCM中的变更状态和相关ClearQuest软件包11.1.1 ClearQuest软件包11.1.2 匹配状态类型11.1.3 设定状态对应的默认动

作11.1.4 集成UCM策略11.1.5 集成UCM软件包11.2 任务的接收和分配11.2.1 任务自动分配的项目需求11.2.2 版本和功能组件共同决定是否自动分配11.2.3 自动分配的实现11.3 UCM Project的检查和重用机制11.3.1 对版本实现不同级别的锁11.3.2 检查版本和UCM项目的对应关系的唯一性11.4 代码审查11.4.1 开关机制强制代码复查11.4.2 代码复查相关的状态和动作11.5 父子任务同步11.5.1 开发人员相关的动作11.5.2 测试人员的动作11.6 新功能请求的特殊控制11.6.1 限制普通开发人员创建新功能11.6.2 变更请求类型的变化11.6.3 新功能记录从Submitted状态到Closed状态的直接跳转第12章 代码集成与构造过程12.1 集成和构造的基本原则12.1.1 根据项目管理的需求实现构造管理12.1.2 自动化构造过程减少人工操作12.1.3 构造过程可重复12.1.4 构造的周期性和连续性12.1.5 按照产品组件进行构造12.1.6 自动化部署并进行初步验证12.1.7 生成清晰、完整的构造信息12.2 ClearCase和ClearQuest中与集成构造相关的对象12.2.1 构造支持对象：元素、分支和版本12.2.2 构造处理对象：标签、属性、超链、触发器12.2.3 工作流和活动12.2.4 构件和基线12.3 基于构造流的构造过程12.3.1 UCM中构造策略的选择12.3.2 引入构造流的好处12.3.3 如何设置构造流12.3.4 基线管理策略12.3.5 基于构造流的构造过程12.4 构造清单和文件列表12.4.1 构造清单的意义12.4.2 生成构造清单并添加到ClearQuest系统中12.4.3 变更文件列表12.5 封装构造过程12.5.1 前处理命令12.5.2 产品构造程序调用命令12.5.3 后处理命令12.6 构造结果示例第13章 查询与统计图表设计13.1 创建和使用查询13.1.1 选择数据来源13.1.2 确定结果格式13.1.3 设定查询条件13.1.4 操作查询结果13.2 各类图表的创建方法13.2.1 图表机制和图表分类13.2.2 缺陷分布图的创建13.2.3 缺陷趋势图的创建13.2.4 缺陷期龄图的创建13.3 常用项目管理数据和图表13.3.1 项目质量指标13.3.2 项目进度指标13.3.3 效率指标13.4 典型缺陷图表分析第3篇 实施与应用第14章 模拟项目“电网计费系统”介绍14.1 系统需求14.2 项目涉众14.3 开发环境和方法第15章 开发团队部署15.1 部署开发工具15.1.1 选择开发工具15.1.2 组建部署小组15.1.3 部署方法和过程15.2 选定开发模型15.2.1 获取其他项目的经验15.2.2 领会敏捷开发模型15.2.3 寻求工具对开发模型的支持15.3 确定团队结构和角色15.3.1 多敏捷团队结构15.3.2 不同角色的主要职责15.3.3 独立测试团队15.4 规划配置库和组件模块15.4.1 VOB库和UCM组件15.4.2 UCM和ClearQuest项目15.4.3 ClearQuest模块第16章 安装基本配置管理系统16.1 准备安装环境16.2 安装配置域服务器16.2.1 扩展AD域的Schema16.2.2 用户的UNIX属性16.2.3 配置ClearCase主机使用LDAP认证16.2.4 配置Vob服务器上的Samba使用域认证16.3 ClearCase主要服务器介绍16.3.1 ClearCase VOB服务器16.3.2 ClearCase 注册服务器16.3.3 注册域（Region）16.4 配置VOB服务器支持环境16.4.1 ClearCase 用户和组权限设置16.4.2 Linux VOB 服务器端的Samba共享配置16.4.3 Linux VOB 服务器端NFS共享配置16.5 安装ClearCase客户端和服务端16.5.1 创建ClearCase发布区16.5.2 升级发布区16.5.3 安装ClearCase服务器端16.5.4 ClearCase客户端安装16.6 ClearQuest安装及初始化16.6.1 ClearQuest的安装16.6.2 ClearQuest数据初始化16.7 安装许可证服务器第17章 搭建集成式配置管理环境17.1 创建配置管理库17.1.1 准备VOB存储空间17.1.2 创建VOB和PVOB17.1.3 在VOB中创建组件（Component）17.2 创建并配置UCM 项目17.3 创建ClearQuest项目17.3.1 创建“电网计费”项目对应的Product记录17.3.2 创建“电网计费”项目对应的Version记录17.3.3 创建“电网计费”项目对应的Component记录17.3.4 创建“电网计费”项目对应的Lineitem记录17.4 建立CC/CQ项目关联17.4.1 ClearCase端关联ClearQuest数据库17.4.2 在ClearQuest中关联UCM Project17.5 设置特殊对象和属性17.5.1 准备构造流和相关的属性17.5.2 应用触发器第18章 日常敏捷开发18.1 制订迭代计划18.2 开发实现18.3 缺陷修正18.4 每日站立会议（Scrum）18.5 沟通协作第19章 每日集成构造19.1 建立构造环境19.1.1 安装构造所需的工具19.1.2 建立工作空间19.2 自动化集成构造过程19.2.1 构造过程自动化19.2.2 代码集成自动化19.2.3 构造结果部署和测试自动化19.2.4 部署自动化脚本19.3 构造结果保存和发布19.4 构造异常处理19.4.1 查找构造中断的引入者19.4.2 检入补丁代码并继续构造过程19.5 实现持续集成19.5.1 检索是否有开发人员交付工作19.5.2 缩短构造过程第20章 日常敏捷测试20.1 测试介入阶段20.2 测试驱动的开发20.2.1 概述20.2.2 过程20.2.3 优点20.3 测试自动化20.4 独立团队测试20.5 缺陷跟踪第21章 项目管理和产品发布21.1 项目发布规划21.1.1 发布计划21.1.2 进度管理21.2 功能划分与人员设定21.2.1 功能划分21.2.2 人员设定21.3 进度跟踪报告21.4 风险管理21.5 发布管理第4篇 扩展与升华第22章 ClearCase与其他SDP工具的集成22.1 与需求管理工具的集成22.1.1 Base ClearCase和RequisitePro集成22.1.2 ClearCase UCM和RequisitePro集成22.2 与开发设计工具的集成22.2.1 与Rational RSA和RAD的集成22.2.2 与微软Visual Studio的集成22.3 与测试工具集成22.3.1 ClearCase与CQTM的集成22.3.2 ClearCase与Rational RFT、RPT的集成第23章 构建管理工具的使用与集成23.1 BuildForge的功能介绍23.1.1 功能特性和产品架构23.1.2 对象与概念23.2 自动化构建示例23.2.1 “电网计费”系统原有构建过程23.2.2 创建BuildForge对象和属性设置23.2.3 设置定时构建启动23.3

《软件开发过程与配置管理》

BuildForge与变更和配置管理系统集成23.3.1 什么是适配器23.3.2 适配器环境变量23.3.3 在项目中使用适配器第24章 如何支持远程团队24.1 使用Web方式支持远程团队24.1.1 CC Web Server 安装和配置24.1.2 ClearCase远程客户端 CCRC24.1.3 ClearQuestWeb 服务器安装和配置24.1.4 ClearQuestWeb的使用24.2 CCRC和CQ Web集成应用24.2.1 准备UCM视图24.2.2 远程客户端关联ClearQuest记录24.2.3 CCRC中的交付和同步24.3 使用Multisite支持远程团队24.3.1 ClearCase和ClearQuest Multisite简介24.3.2 VOB副本24.3.3 使用Mastership实现独立开发24.3.4 站点间的同步模式24.3.5 ClearCase Multisite站点的复制和同步24.3.6 UCM的应用实例24.3.7 ClearQuest的Multisite第25章 基于度量的项目管理和过程优化25.1 可度量的能力改进框架25.2 Rational Insight介绍25.2.1 Rational Insight架构25.2.2 Rational Insight4的安装25.2.3 Rational Insight的管理25.3 Rational Insight应用举例25.3.1 定制样本报告25.3.2 定制仪表盘25.3.3 报告追溯第26章 如何与Jazz & RTC集成26.1 Jazz概述26.1.1 Jazz平台的目标26.1.2 Jazz体系结构26.2 第一个基于Jazz平台的产品——RTC26.2.1 RTC简介26.2.2 RTC中的主要概念26.2.3 使用RTC进行团队协作26.2.4 使用RTC进行软件配置和变更管理26.2.5 软件构建26.3 RTC与ClearQuest和ClearCase的集成26.3.1 使用ClearQuest Bridge来连接RTC和ClearQuest26.3.2 将RTC与 ClearCase关联26.4 其他Jazz产品参考文献

章节摘录

插图：传统的软件配置管理系统大多侧重于产品代码的管理，缺乏对软件开发过程的完整支持。纯粹依靠传统的软件配置管理系统会使软件组织的各个部门之间或角色之间没有连接关系，组织所拥有的只是相互独立的信息资源，从而形成了信息“孤岛”。理想的软件配置管理系统将提供过程支持，使组织之间能通过系统的过程驱动力建立紧密的联系，便于数据共享和开发人员协作。为了完善配置管理系统以加强对开发过程的支持，需要在传统配置管理系统的基础上，完善对人员和任务的管理，强化变更过程，并提供对团队开发、过程模式、构造发布和项目管理等方面的支持。下面来分别看看软件开发项目在这些方面的实际需求。

4.6.1 角色与职责常见的配置管理系统中通常只区分普通用户和系统管理员两种角色。为了加强过程的支持，至少必须在系统中定制出下列与过程相关的角色和职责。

项目经理：负责项目的创建、协调和进度管理，包括任务分配、风险分析和生成向上级主管汇报的报表。

变更控制委员会（CCB），负责分析评判需求和变更，使开发任务与项目的商务需求保持一致。

《软件开发过程与配置管理》

媒体关注与评论

配置管理是软件开发项目的枢纽，形象地比喻应该是和军队的供给线一样重要。“兵马未动，粮草先行”，任何一个软件开发组织，都需要先具备良好的开发流程和配置管理工具，而且要在管理层和所有的开发人员中普及软件配置管理的知识。刘江华所带领的小组在IBM内部做了很多的相关培训和知识分享，这也是软件配置管理在IBM CDL非常成功的原因之一。我也很欣喜地看到，他们终于联手IBM Rational开发部门和其他团队的专家，将多年的经验和积累凝于笔端，付梓成书，让更多的人受益。 ——IBM协同办公产品总监寇育新

《软件开发过程与配置管理》

编辑推荐

《软件开发过程与配置管理:基于Rational的敏捷方案设计与应用》由电子工业出版社出版。

《软件开发过程与配置管理》

精彩短评

- 1、读这本书的主要目的是为了弄清楚配置项和基线的确切含义。
- 2、作者对敏捷开发的理解或许还需要提升
- 3、书中全部都是IBM工程师的经验之谈，值得精读。

《软件开发过程与配置管理》

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:www.tushu000.com