

软件过程技术及环境研究

潘秋菱¹, 刘宗田², 贾 亮¹, 许 庆¹, 袁兆山¹

(1. 合肥工业大学微机所, 安徽合肥 230009; 2. 上海大学计算机学院, 上海 200072)

摘 要: 软件过程已成为软件质量控制的有效方法,但在以往的工作中,由于理论和技术上的困难,未能充分发挥度量在软件过程质量控制中的作用. 本文介绍了自主开发的基于 UML 的软件过程工程环境 Softev. 为软件开发者提供了定义、运行和监控过程的平台. 同时,提出了在软件开发中进行度量的一组指标,并在 Softev 中予以实现,以客观的度量数据,更科学地评价软件产品(包括中间产品)和实施过程,指导软件开发的全过程.

关键词: 软件过程; 软件质量; UML; 度量

中图分类号: TP311.5 **文献标识码:** A **文章编号:** 0372-2112 (2001) 11-1575-03

Research on Software Process Technology and Environment

PAN Qiu-ling¹, LIU Zong-tian², JIA Liang¹, XU Qing¹, YUAN Zhao-shan¹

(1. Research Institute of Microcomputer, Hefei University of Technology, Hefei, Anhui 230009, China; 2. Shanghai University, Shanghai 200072, China)

Abstract: Software development processes, as a means to ensure software quality and productivity, have been widely accepted within the software development community. This research introduces how to assist developers to establish quality management plan and monitor the implementation of software processes quantitatively in a CASE environment. A PSEE (process-centered software engineering environment) named Softev has been brought forward to provide developers the environment to define, execute and monitor their processes. As UML is becoming a standard diagrammatic notation for object-oriented modeling and is spreading rapidly, it has been chosen to express the artifacts developed during the requirement analysis and system design stages. In the early products of software process engineering, there was not enough attention to be given to the metrics of software processes and artifacts. This research proposes a suite of metrics and gives the implementation in Softev. Managers are able to predict and supervise the software project more precisely.

Key words: software process; quality control; UML; metrics

1 引言

随着计算机技术的发展,软件产品的质量控制越来越引起人们的重视. 为保证软件质量,软件过程技术着眼于对软件开发过程的组织、监视和评估,软件过程研究者将软件看作是能够通过系统有序的运行过程而建立的错综复杂、相互联系的信息集合体. 从某种意义上讲,高质量的软件产品是由高质量的各个部分及各部分之间的协调一致所组成的. 为支持软件过程技术提供的软件工程环境就称为以过程为中心的软件工程环境 PSEE (Process-centered Software Engineering Environment). 近年来软件过程技术发展很快,每年都有许多相关论文提交到 ISPW (International Software Process Workshop), ICSP (International Conference on Software Process), EWSPT (European Workshop on Software Process Technology), ICSE (International Conference on Software Engineering) 等会议. Figgetta 总结了软件过程研究取得的成就和未来的发展方向^[1].

PSEE 还正处在发展阶段,现有的环境主要支持复杂过程

的描述,自动触发、过程仿真及监控等功能,或者面向过程的体系结构的开发支持^[2,3]. 还有一些注重配置管理等特定功能. 然而,在软件过程中的软件度量是本文强调的重点. 只有度量和记录现行的项目,以后的类似项目才可以借鉴以往的经验教训,而且,在每个关键过程结束的时候,对其进行度量,评估已完成的过程并对以后的过程进行及时的预测,能够保证软件高效高质地完成. 近年来软件质量保证发展迅速,尤其是卡内基梅隆大学的软件工程研究所 (SEI) 建立的 CMM (能力成熟度模型) 质量体系,明确提出对软件度量的要求. 软件成熟度评估和不断的软件过程改进都离不开定量的度量技术和工具. 在软件度量的发展中,大多讨论的是源代码的度量. 本文从实践应用出发,将度量分为过程和产品两方面,并实现相应的工具. 然而在对中间产品的度量时,由于中间产品以往大多用自然语言表示,使度量难以进行. UML 这种建模语言的兴起和广泛应用,使客观度量需求设计成为可能.

UML (Unified Modeling Language) 是一种面向对象的统一建

模语言. 在 1996 年提出后, 到年底, 已有 700 多个公司表示支持采用 UML 作为建模语言. 1997 年被 OMG(对象管理组织) 采用为基于面向对象技术的标准建模语言. 由于其在国际上被广泛地接受和在国内正在普及, 为具有可比性地客观度量中间产品提供了基础, 所以在进行需求和设计的度量时选用其最常用的产品 Rational Rose 作为中间产品的表述形式^[4].

本文实现了一个开放分布式的支持软件开发者工作过程的软件过程工程环境 Softev. 还提出了一组针对过程和产品进行软件度量的指标. 并在 Softev 里实现了相应的工具. 其详细内容将在第三节进行讨论, 下一节介绍 Softev 的概要和应用流程.

2 软件过程工程环境 :Softev

2.1 过程定义

在 SEI 的 CMM 模型中, 软件过程被定义为一组人们用于开发和维护软件及相关产品的活动、方法、实践或转换. 将软件过程形式化定义为一个十元组, 如下:

定义 1 $PS = \{ PN, PD, PC, AN, IA, OA, RN, VM, SC, EC \}$

这里: *PN*: 过程名、*PD*: 过程描述、*PC*: 过程特性、*AN*: 组成步骤、*IA*: 输入产品、*OA*: 输出产品、*RN*: 所需资源、*VM*: 校验方式、*SC*: 开始条件、*EC*: 结束条件.

定义过程时应考虑以上要素, 过程由 *PN* 和 *PC* 唯一决定. Softev 支持图形化过程定义和显示, 可以采用不同粒度层次化定义过程.

2.2 模块分析及应用流程

Softev 包含九个模块: ServerConfiguration、ResourceAllocator、Engine、ProcessMaker、ResourceManager、Monitor、Client、MeasurementManager and WorkloadStatistic. 前三个模块运行在服务端, 其他是客户端模块. 其中负责过程定义和运行的基本模块是 Engine、ResourceAllocator 和 ProcessMaker. Engine 是 PSEE 的心脏, 根据所定义的过程模型, Engine 和 ResourceAllocator 互相协作, 进行调度和过程驱动. ResourceAllocator 根据资源模型和状态, 选择一个执行任务的合适人选. ProcessMaker 通过可视化的方法, 提供平台定制化过程模型. 客户端 Client 模块为用户提取和提交任务的接口, Monitor 监视过程运行, MeasurementManager 管理度量的设定和工具运行. ServerConfiguration 和 ResourceManager 用来初始化 Softev, 前者用于运行环境的设置, 后者进行资源建模. 最后, WorkloadStatistic 进行总结工作量等数据分析.

对于成熟企业来说, 开发业务集中在有限的领域, 项目开始的时候, 管理者应首先查看以往类似项目的数据, 参考以往经验定义该项目的过程. 根据过程定义, 系统自动或允许用户手工分配工作人员, 定义完成后, 由项目管理者启动过程, 开发者通过系统客户端提取分配给自己的任务, 按照描述执行, 完成后提交. 系统记录所需数据, 在关键步骤中根据定义对输出产品进行度量. 严重偏离时给负责人或管理者发警告信息, 否则, 进入下一个步骤. 有权限的用户也可随时查询度量数据, 进行评估或对后续的过程进行指导. 从而能够控制项目开

发, 降低风险. 图 1 给出了一个过程定义的示例, 图中定义了项目 CIMS(Container Terminal Management System) 的顶层过程定义, 以及需求分析过程的子过程复审的定义.

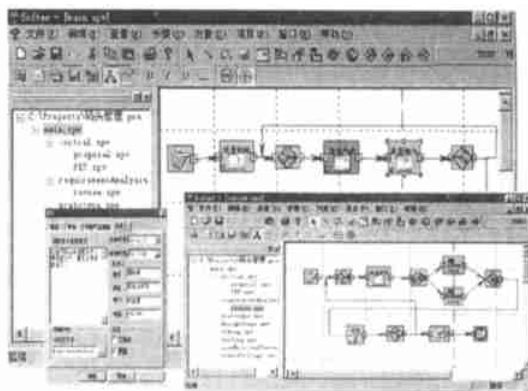


图 1 Softev 过程定义和显示界面示例

3 软件度量指标

软件度量在软件开发中占有重要的地位, DeMarco 曾说过: “任何不可度量的东西都是不可控制的^[5]”. 在 C&K 理论基础上, 结合大量的软件开发经验^[6], Softev 度量的指标分为两组: 一组是过程指标, 其度量实体为开发的不同阶段, 属性包括阶段成本、时间等. 另一组为中间产品, 实体是主要的过程中间产品, 属性即为这些产品的属性. 建议进行度量的指标为:

⑧ 过程度量: 针对给定过程, 度量计划时间、实际时间、计划成本、实际成本、修改数、用户反馈数、后续过程发现的错误数、最终用户报告的错误数、生产率、项目复用率、项目返工率. Softev 提供界面收集和存储这些数据以备以后分析和参考.

⑨ 产品度量: 实际应用时, 可根据情况进行裁剪.

用户需求: 用例数、执行者数、用例(执行者)的相关执行者数(相关用例数)、消息维大小、相对消息维大小、用例的复杂度、无二义性、完整性、可校验性、可实现性、可重用性.

系统设计: 类数、类的加权方法数、孩子数、继承深度、封装度、依赖度、被依赖度、鲁棒性、可复用性、界面质量.

源代码: 大小(LOC)、类数、加权方法数、继承深度、孩子数、耦合度、响应集、相对内聚缺乏度、鲁棒性、可读性.

测试报告: 测试用例数、测试密度、测试频度、加权错误数、测试工作量、用户满意程度.

维护报告: 正确性、稳定性、维护工作量、用户满意程度.

Softev 对以上的斜体部分的指标进行客观度量. 其中, 对于用户需求和系统设计, MeasurementManager 模块通过 Rational Rose add-ins 提取数据进行处理, 如图 2 所示. 对于某些源代码, 如 C++, 如图 3 所示, 系统通过词法分析和语法分析, 度量出结果^[7]. 其他的客观数据主要从收集的第一手数据总结而得. 遗憾的是, 并非所有的指标都能够客观测定的, 一些指标必须依赖于主观的判断. Softev 提供录入各阶段数据的界面.

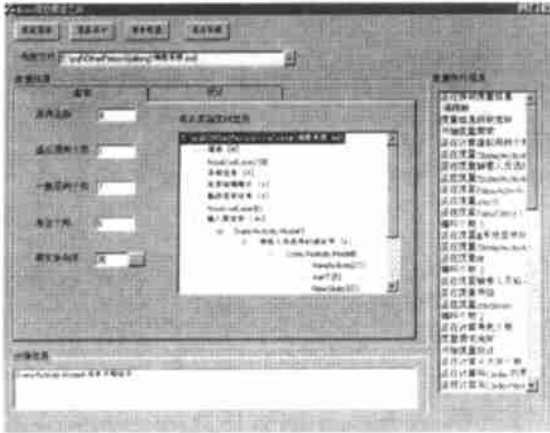


图 2 需求和设计度量模块



图 3 VC++ 源码度量模块

4 总结

本文实现了一个 PSEE 工具: Softev, 介绍了其功能和工作流程. 提出了一套度量指标, 通过不同阶段不同粒度的度量能够对正在进行或进行过的工作进行评价, 指导后来的过程, 有效地控制软件开发质量.

Softev 主要用 VC++ 6.0 完成, 数据库为 SQL SERVER 7.0/2000. 已经实现的主要功能包括图形化方式定义过程模型、过程启动、运行、监控, 工作进度度量、中间产品度量, 数据分析 and 处理.

Softev 为自由传播软件, 一些比较完善的模块可以通过 E-mail 索取试用. Softev 的完成需要很大的工作量, 现在只是初成框架, 目前正与一软件开发公司合作试用, 但是这涉及工作方式的改变, 需要较长的时间.

在进一步工作中, 首先许多功能还需要继续开发. 另外, 在新的版本中拟引入 PETRINET 的扩展形式进行过程定义, 加强过程定义的有效性检验. 同时, 过程的演化一直是 PSEE 的问题, 现阶段采取简易的“eager”方式, 下阶段应更好地处理. 最后, 不断扩大度量范围, 在实践中巩固或修正度量指标, 同时加强在实践中的应用, 不断积累项目数据, 是一个永无止境的过程.

参考文献:

- [1] A Fuggetta. Software process :a roadmap [A]. ICSE 2000 [C], Limerick (Ireland) ,June 2000. Also available at <http://www.cefriel.it/~alfonso/documents/Presentations/ISR/ISR.htm>.
- [2] G Bolcer ,R Taylor. Endeavors :A process system integration infrastructure [A]. In 4th International Software Process Workshop [C], Brighton ,1996 :76 - 85.
- [3] P K Garg ,et al. Process-Centered Software Engineering Environments [M]. IEEE Computer Society ,1996 :131 - 140.
- [4] James Rumbaugh ,et al. The Unified Modeling Language Reference Manual [M]. Addison-Wesley ,1999.
- [5] DeMarco T. Controlling Software Projects [M]. Prentice-Hall , Englewood Cliffs ,NJ . 1982.
- [6] Chidamber S R ,C F Kemerer. A metrics suite for object-oriented design [J]. IEEE Trans. Software Eng. ,1994 ,20(6) :476 - 493.
- [7] LI Xin-ke ,LIU Zong-tian ,et al. Implementation of a measurement tool for object oriented software and research on measure experiments [J]. Chinese Journal of Computer ,2000 ,10 :1220 - 1225.

作者简介:



潘秋菱 女. 1973 年 8 月生于河南. 合肥工业大学博士研究生, 主要研究领域为软件过程、质量保证、Petri 网、软件度量等. E-mail : qipan@mail.shu.edu.cn



刘宗田 男. 1946 年 6 月生于山东. 研究员, 博士生导师, 上海大学计算机学院, 研究方向为知识工程、软件工程.