

基于 SPM-CMMI 的港城经济仿真系统开发

张萍¹, 严以新¹, 许长新², 陈金峰³

(1. 河海大学交通学院, 南京 210098; 2. 河海大学商学院, 南京 210098; 3. 摩托罗拉南京研发中心, 南京 210098)

摘要: 基于 CMM/CMMI 软件开发模型, 构建 SPM-CMMI 软件开发流程, 在此基础上论述 SPM-CMMI 模型在长江三角洲港城经济系统演化仿真开发中的应用。YRDM 项目的开发过程证明 SPM-CMMI1.0 成功地使能了 CMM/CMMI 的软件流程。

关键词: CMM/CMMI; 系统仿真; 软件度量

Research of SPM-CMMI in Simulation of Port and City Economy System

ZHANG Ping¹, YAN Yixin¹, XU Chagxin², CHEN Jinfeng³

(1. Traffic College of Hohai University, Nanjing 210098; 2. Business School of Hohai University, Nanjing 210098; 3. Motorola Nanjing R&D Center, Nanjing 210098)

【Abstract】 This article makes software development process of SPM-CMMI based on CMM/CMMI software development model. Furthermore, it applies SPM-CMMI model to the simulation of port and city economy system evolution in Yangtze River delta. The development process of YRDM project proves that SMP-CMMI1.0 enables software process of CMM/CMMI.

【Key words】 CMM/CMMI; System simulation; Software metrics

自 1990 年美国卡耐基梅隆大学软件工程研究所(SEI)发布 SW-CMM v1.0 以来, SEI 针对不同领域的要求对 SW-CMM 先后进行改进, 并衍生出了一系列成熟度模型^[1,2]。但是, 由于软件企业的业务一般不是单一的, 其业务需采取多个模型, 这样必定会使用很多关键过程域, 关键实践产生重叠, 也必然会增加开发费用。于是 2001 年 11 月起 SEI 开始推出集成多个模型的 CMMI, 以解决多模型之间重叠问题。然而, CMM/CMMI 本身并不是一套软件开发流程, 还不能直接用于软件产品开发的过程, 各软件组织必须根据 CMM 描述的原则, 结合实际业务具体定义自己的流程。

1 CMM/CMMI 模型

1.1 CMM

软件能力成熟度模型(Capability Maturity Model for Software, CMM)最早由美国军方委托美国卡耐基梅隆大学软件工程研究所研制, 最初目的是帮助美国国防部认证软件承包商的过程成熟度, 一度发展成为软件企业成熟度国际认证标准, 也是国际上最流行、最实用的软件生产过程标准。

CMM 为软件企业的过程能力提供了一个阶梯式的进化框架, 阶梯共有五级, 如图 1 所示。

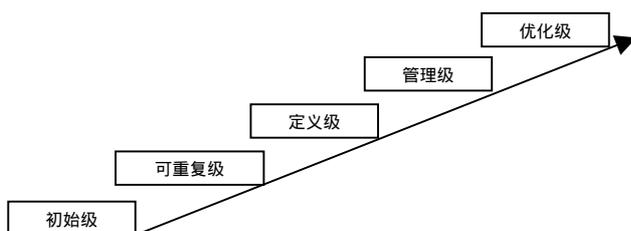


图 1 软件能力过程

CMM 的开发过程划分为关键过程域(Key Process Area, KPA)。对于每个 KPA, CMM 模型先定义其总体目标, 然后

用通用的 5 个特征(Common Feature)将其细分成关键实践(Key Practice)。5 个共同特征是: 承诺执行, 有能力执行, 实际执行, 度量与分析, 执行验证。

1.2 CMMI

随着 CMM 的发展及应用实践, 人们逐渐发现 CMM 并不能涵盖软件产品开发的整个过程, 而同时采取多个模型会由于关键实践的重叠而造成成本增加。因此 SEI 开始推出集成软件能力成熟度模型(Capability Maturity Model for Software Integrated, CMMI), CMMI 是在 CMM 基础上发展起来的模型, 在 CMM 基础上增加了集成的产品和过程开发(IPPD)等专业领域, 不再局限于软件。软件组织对 CMM/CMMI 进行使能, 比较成功的公司如摩托罗拉公司, 其 IPD-CMM 已经通过 CMM5 认证。

2 港城经济仿真系统(YRDM)的开发模式

2.1 YRDM 的软件开发模式 SPM-CMMI

YRDM(Yangtze River Delta Model)是项目组目前开发的一个港城经济仿真软件。港口建设和

城市协调发展涉及港口、城市人口、城市经济、综合运输网以及区域土地能源等多个方面, 是一个多因素、多目标、高阶非线性的动态大系统问题^[3]。其仿真系统结构较为复杂, 且软件开发人员不稳定, 若采用传统的瀑布式软件开发模式, 会导致开发成本上升, 甚至项目失败。

YRDM 项目组根据 CMM 五级及 CMMI 模型使能定义了自己的开发流程 SPM-CMMI1.0(Software Project Management

基金项目: 高等学校重点学科基金资助项目“长江三角洲港城经济系统演化仿真”(G030811)

作者简介: 张萍(1979—), 女, 博士生, 主研方向: 港口、海岸工程及系统工程仿真; 严以新、许长新, 教授、博导; 陈金峰, 高工

收稿日期: 2006-02-09 **E-mail:** zhangpinghu@126.com

CMMI 1.0)。如图 2。

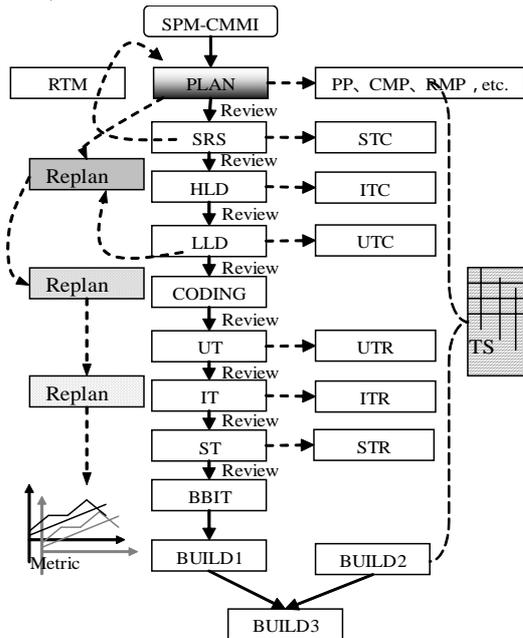


图 2 SPM-CMMI 开发流程

SPM-CMMI 分为以下几个阶段：

- 计划阶段(PPLAN)；
- 系统需求分析阶段(SRS)；
- 概要设计阶段(HLD)；
- 详细设计阶段(LLD)；
- 编码阶段(CODING)；
- 单元测试阶段(UT)；
- 集成测试阶段(IT)；
- 系统测试阶段(ST)；
- 模块集成构建测试(BBIT)。

以上每个阶段都需要进行输出物的评审(Review)，评审之后进行返工修改，PM 进行缺陷率的分析与工作度量。同时，开发过程始终都贯穿着需求跟踪管理(RTM)，以保证需求不被遗漏。此开发模式中比较重要的一个 KPA 就是软件度量，KPA 贯穿整个软件开发的过程，并为整个软件开发过程提供分析数据。

如图 2，结合系统动力学理论的反馈，我们增加了计划与质量目标的动态反馈与调整。由于开发初期很难确定一个较为复杂系统的实际工作量，容易出现较大计划偏差；而且初期我们只能制定系统级的质量目标，不能控制整个质量目标的达成。随着软件设计的逐步深入与推进，对工作量的估计越来越接近实际值，我们可以根据新的情况调整计划，使实际过程与计划的偏差始终保持在偏差范围内。而通过重计划调整系统下一阶段的阶段质量目标可以更好地分解总体质量目标，保证质量可控。

SPM-CMMI1.0 区别于其他软件模式的优点主要体现在如下方面：

- (1) 开发过程量化管理，动态监控项目进程与风险，质量目标阶段化分解与监控。
- (2) 系统反馈计划与质量目标动态调整机制，可保障项目的最终如期完成与质量可控。

2.2 YRDM 对 SPM-CMMI1.0 的应用

YRDM 采用 SPM-CMMI1.0 软件流程开发，该流程融合了 CMM 与 CMMI 的软件开发模型，充分显示了 CMM 开发模型的优越性，使该项目在高效与较低成本控制之下顺利运

行。下面以开发中的项目计划及其各阶段的调整为例，说明 SPM-CMMI 的执行过程。

2.2.1 初试计划阶段

项目计划的输入为对 YRDM 系统客户需求分解后形成的任务书“YRDM-v1.0-工作任务”。工作任务书中包含了软件的客户需求与性能要求、质量要求以及系统规格。项目计划阶段主要进行了以下几项工作：

(1) 制定项目开发进程阶段时间点里程碑计划。主要是根据工作量及开发规模在整个开发时间限定的情况下为各阶段分配时间，制定 SPM-CMMI 对应的各阶段完成计划。

(2) 资源投入与分配计划。根据项目的开发规模及可获得的人力设备资源制定不同阶段投入的人力及其他资源，根据 SPM-CMMI 要求指定开发过程中的各个角色(PM、CMO、DBA、MC、TC、QA)。

(3) 质量目标。任何需要交付的软件产品都需要对其质量有一个评测，制定质量目标就是为了能够实现需求的前提下控制软件的最终缺陷率。质量目标也需要参照软件规模来制定，且本文提出的 SPM-CMMI 要求：质量目标必须分解为阶段性目标并动态监控调整。

(4) 风险管理计划(RMP)与缺陷预防计划(DPP)。分析项目开发过程中存在的风险，将风险按照发生的可能性与危害性进行排序，同时根据风险的级别制定前 10 个风险的规避措施与应急计划，制定风险跟踪计划。缺陷预防计划主要是根据以前的经验对可能出现的缺陷制定预防与跟踪计划。

以上计划均需根据项目规模制定，然而项目初期我们不知道项目的实际规模，SPM-CMMI 在项目初期制定计划时采用项目分解估计工作量的方法。对工作任务的分配需求以及业务了解之后，采用业界常用的 Wideband Delphi 估计方法，对各分配需求进行代码量的初始估计，得出 YRDM 的总工作量为 6.5k 行代码。根据工作量我们制定初始的项目开发进程计划及质量目标如表 1、表 2。

表 1 YRDM 初始计划

阶段	初始计划		
	开始时间	结束时间	投入人力
项目计划	2005-2-16	2005-2-20	2
需求分析	2005-2-21	2005-3-8	5
详细设计	2005-3-9	2005-3-23	5
编码阶段	2005-3-24	2005-4-1	4
单元测试	2005-4-2	2005-4-16	4
系统测试	2005-4-17	2005-4-28	2
产品发布	2005-4-29	2005-5-8	2

表 2 YRDM 质量目标

质量活动	目标	最高	最低	实际
SRS Review(个/页)	0.5	0.7	0.4	
STP Review(个/页)	0.3	0.4	0.2	
LLD Review(个/页)	0.6	0.8	0.5	
UTP Review(个/页)	0.55	0.7	0.45	
CODE Review(个/KL)	5.5	7	5	
UT 单元测试(个/KL)	5	6	3.5	
ST 系统测试(个/KL)	2	3.5	1.5	
验收测试(个/KL)	1	2	0	
维护(个/KL)	1	1.5	0	

2.2.2 详细设计阶段结束计划调整

根据 SPM-CMMI 制定的要求，在每个阶段结束均进行一

次工作量度量，图 3 为详细设计结束后的工作量度量表^[4,5]。

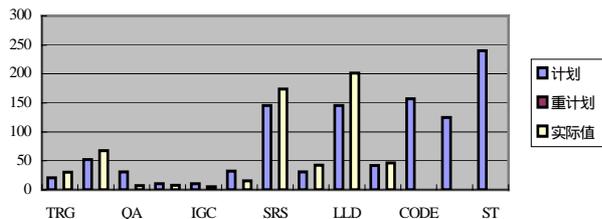


图 3 详细设计阶段结束工作量度量表

由图 3 可计算出实际的 SRS 与 LLD 工作量超出了计划，并且偏差最大达到了 40%，且前期质量目标没有达到。分析发现其原因主要是因为前期用户对自己的需求理解与开发人员产生偏差，导致设计时需求的变更，从而使进度受到了较大影响。根据实际情况我们重新调整了后期计划与质量目标，见表 3、表 4。

表 3 YRDM 重计划

阶段	重计划		
	开始时间	结束时间	投入人力
单元测试	2005-4-12	2005-4-25	5
系统测试	2005-4-25	2005-5-10	2
产品发布	2005-5-10	2005-5-12	2

表 4 YRDM 质量目标调整

质量活动	目标	最高	最低	实际
CODE Review(个/KL)	6.5	7	5.5	
UT 单元测试(个/KL)	5.5	6	4.5	
ST 系统测试(个/KL)	2.5	3.5	2	
验收测试	1	2	0	
维护	1	1.5	0	

2.2.3 项目结束度量分析图

根据 SPM-CMMI 要求，我们在项目结束时对整个项目进行了度量分析，下面看一下进度分析图，如图 4。

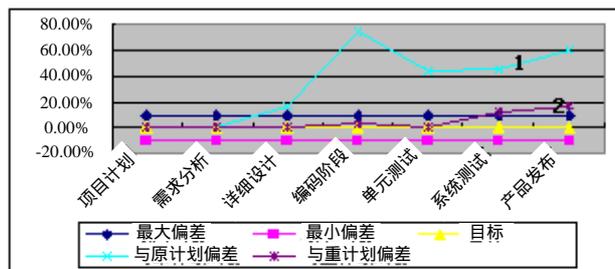


图 4 度量分析图

线 1 是我们的初始计划与目标的偏差，线 2 是我们重计划后进度与目标的偏差。可见，如果不进行度量分析及及时调整计划，我们的项目将会与预期完成日期形成较大偏移，从而无法满足项目合同，而调整后的计划与目标的偏差仅为 20%，最终保证了项目的成功。

3 小结

在 YRDM 项目开发中软件度量是一个重要的部分，本文结合系统动力学原理进行软件度量的设计，从而对软件项目过程进行有效的控制与管理。软件开发过程管理的有效性是 CMM/CMMI 所追求的目标之一，也是开发组织一直追求的一种高效管理方式。在本项目中采用了 SPM-CMMI 的软件开发流程，使整个项目开发过程实现低成本、少资源、高效率的目标。YRDM 项目的开发过程证明 SPM-CMMI1.0 成功地使能了 CMM/CMMI 的软件流程。

参考文献

- 1 郑人杰, 王 纬, 王方德. 基于软件能力成熟度模型(CMM)的软件改进方法与实施[M]. 清华大学出版社, 2003.
- 2 USAF Software Technology Support Center (STSC). Mapping CMM-SE/SE/IPPV 1.1 to SW-SMM V1.1[Z], 2002-01.
- 3 许长新. 港航经济系统论[M]. 北京: 海洋出版社, 2004.
- 4 徐如志, 钱乐秋. 基于度量的软件项目过程优化控制研究[J]. 电子学报, 2003, 31(12).
- 5 Sukhodolsky J, Smari W. Software Project Control Optimization[C]. Proceeding of the 2000 IEEE International Conference on Information Reuse and Integration, Honolulu, Hawaii, 2000: 42-49.

(上接第 270 页)

3 账户管理

服务器端读取客户端提交的用户名和口令，通过查询数据库来验证用户名与口令是否正确，并与防火墙进行通信，完成授予用户访问 Internet 的权限。用户授权功能通过调用防火墙来实现，即动态的修改防火墙的过滤规则，通过认证的用户的的数据包通过。

4 结束语

由于 Linux 的源代码开放，为免费软件，能在配置很低的电脑上运行，提高了系统的使用效率。而 Linux 拥有的 Unix 安全机制以及它自带的防火墙保证了系统内部数据的安全。本文基于 Linux 的开放实验室计费系统为例，探讨 Linux 环境的开发方法。利用嵌入的 Linux 编程实现开放实验室管理信息自动采集，让管理人员随时了解上机者的上机频率、上

机时间、机房忙时间段等决策信息。既为机房监督提供了数据，又为任课教师提供了必要的统计信息，同时为实训中心提供了宏观调控的现实依据。这种开放实验室管理模式，希望能够为高校开放实验室管理工作提供有益的思路。

参考文献

- 1 李全忠. 直接读取 PCI 网卡的 MAC 地址的原理与方法[J]. 计算机工程, 2005, 31(18).
- 2 李善平. 边学边-Linux 内核源指导[M]. 浙江: 浙江大学出版社, 2005-03: 95-120.
- 3 井口信和. TCP/IP 网络工具篇[M]. 北京: 科学出版社, 2003-04.
- 4 咎世刚. 基于 Linux 防火墙的内部邮件监控与阻断系统[J]. 计算机工程, 2005, 31 (24).
- 5 宋 军. 显式流量控制协议 XCP 研究[J]. 计算机科学, 2005, 16 (7).