

Rational. software

国防单位基于GJB5000 的软件工程规范化解决方案



目录

1	解决方案达成目标	5
2	IBM RATIONAL SDP助力国防单位建立基于GJB5000第二和部分第三级成熟度等级的软件工程平台解决方案	7
2.1	管理类过程的解决方案	8
2.1.1	需求工程	8
2.1.1.1	实施需求工程能够解决国防单位目前存在的那些问题	8
2.1.1.2	IBM Rational需求定义平台设计方案	8
2.1.1.2.1	获取定义业务词汇	9
2.1.1.2.2	通过业务建模, 深入理解用户的需求	10
2.1.1.2.3	从用户角度出发, 准确描述软件的功能需求	10
2.1.1.2.4	利用原型准确把握用户界面技术, 使系统达到最佳的可用性	11
2.1.1.3	IBM Rational需求管理平台设计方案	12
2.1.1.3.1	总体解决方案	12
2.1.1.3.2	需求引入	13
2.1.1.3.3	实施有效地审批流程	13
2.1.1.3.4	有效地组织管理需求	14
2.1.1.3.5	定义文档模板	15
2.1.1.3.6	建立测试方案	17
2.1.1.3.7	审核需求和测试文档	18
2.1.1.3.8	修改需求文档及测试文档	18
2.1.1.3.9	需求和测试文档入库集中管理	18
2.1.1.3.10	在需求变化中维护统一的需求版本	20
2.1.1.3.11	紧急需求	20
2.1.1.4	方案如何提升国防单位需求管理能力	21
2.1.1.4.1	建立一个标准化需求管理流程体系	21
2.1.1.4.2	支持KPI体系	21

2.1.1.4.3	提高团队整体的开发效率和产品的开发质量	21
2.1.1.4.4	建立规范的产品发布管理制度	21
2.1.2	项目管理	21
2.1.2.1	实施项目管理能够解决国防单位目前存在的那些问题	21
2.1.2.2	IBM Rational过程与项目管理平台助力国防单位GJB5000 SPP和SPTO KPA的实现	22
2.1.2.2.1	完整集成化的技改和型号项目综合管理平台	25
2.1.2.2.2	基于业务决策的型号和技改项目组合管理	30
2.2	支持类过程解决方案	31
2.2.1	配置管理	31
2.2.1.1	实施软件配置及变更管理能够解决国防单位目前存在的那些问题	31
2.2.1.2	IBM Rational配置及统一变更管理平台助力国防单位GJB5000 SCM KPA的实现	32
2.2.1.2.1	全面版本控制, 确保软件资产的安全性和完整性, 避免企业资产的流失和意外破坏	34
2.2.1.2.2	完整地记录文件以及系统的变动历史记录, 直观进行文件及系统两个层面的历史回溯和版本比较	34
2.2.1.2.3	一致高效的工作环境管理, 简化开发人员工作	35
2.2.1.2.4	加强开发团队内外的沟通, 及时掌控项目状态, 全面提高开发团队的工作效率	35
2.2.1.2.5	发布版本的构建管理, 建立生产运行代码和源代码的有机联系	36
2.2.1.2.6	引入构件提高软件的灵活性和可重用性, 简化大型软件或多方外包方式下的配置管理	36
2.2.1.2.7	按活动进行工作进度跟踪, 通过灵活流程定制管理活动(如需求、缺陷、变更等)的生命周期	36
2.2.1.2.8	基于相同代码轻松进行多个发布版本的并行开发和维护, 同一修改在多个发布版本上只改一次	36
2.2.2	软件质量保证	36
2.2.2.1	实施软件质量保证能够解决国防单位目前存在的哪些问题	36
2.2.2.2	IBM Rational 软件质量保证平台助力国防单位GJB5000 SQA KPA的实现	37
2.2.2.2.1	软件质量保证活动是有计划的	39
2.2.2.2.2	确保制定了项目的质量目标	39
2.2.2.2.3	与项目的其他计划内容相协调	39
2.2.2.2.4	确定质量保证任务与时间表	39
2.2.2.2.5	软件产品和活动遵守国防单位适用的过程的情况得到客观的验证	41
2.2.2.2.6	处理在软件项目中出现的不符合问题	42
2.2.2.2.7	SQA的目的是通过审计进行循序渐进的过程改进	43
2.2.3	同行评审	44
2.2.3.1	实施同行评审能够解决国防单位目前存在的哪些问题	44
2.2.3.2	IBM Rational 同行评审平台助力国防单位GJB5000 ML3同行评审(Peerreview) KPA的实现	45

2.2.3.2.1 同行评审活动活动是有计划的	46
2.2.3.2.2 借助Rational产品识别和消除在软件工作产品中的缺陷	46
2.3 技术类解决方案	48
2.3.1 软件测试	48
2.3.1.1 实施Rational软件测试解决方案能够解决国防单位目前存在的哪些问题	48
2.3.1.2 IBM Rational 软件测试解决方案助力国防单位测试能力的提升	48
2.3.1.2.1 方案目标	48
2.3.1.2.2 针对测试的最佳经验	48
2.3.1.2.3 测试流程规范的定制	50
2.3.1.2.4 测试管理平台	51
2.3.1.2.5 测试工具平台	52
2.3.2 可视化建模	58
2.3.2.1 有效管理系统复杂度	58
2.3.2.2 增强团队的沟通	58
2.3.2.3 提高系统设计的可重用性	59
2.3.2.4 增强系统架构的灵活性	59
2.3.2.5 建立以构件为基础的软件架构	59
2.3.2.6 保证模型和代码之间的一致性	59
2.3.2.7 使用UML统一软件开发生命周期	60

1 解决方案达成目标

目前国内很多企业都在应用ISO9000、CMMI、6 Sigma等标准来改进自身的过程,较之这些标准,我国的军工企业的武器装备软件开发大多遵循GJB5000-2003《军用软件能力成熟度模型》(以下简称GJB5000)改进自身的软件开发能力。

中国人民解放军总装备部颁布了已于2003年7月21日发布了GJB5000,从2003年10月1日起开始实施。GJB5000的制定,旨在引进国外先进的管理经验,提高我国军用软件的质量,它目前已成为我国武器装备软件建设和发展的事实上的标准。

GJB5000的目标就是不断改进软件生产工艺,提高产品质量。本质上都是以质量管理大师戴明的全面质量控制TQC (Total Quality Control)理论为依据的实践。GJB5000是国家在参考了CMU-SEI的SW/CMM V1.1的基础上制定的,但它除了给出了关于过程成熟度的描述内容外,还给出了改进模式的指导和评估/评价模式的指导,可以说是一个完全本地化的针对武器装备软件开发团队的过程改进体系。

这个体系提供了一个系统的框架,但是它所提供的只是一个过程改进框架,这个框架与软件开发生命周期无关,更与项目管理的过程无关,因此它并不是企业可以直接采纳的软件开发方法和项目管理方法。在实践过程中还需要具体管理和技术平台的支持。例如对于GJB5000中的每一个目标(Goal), GJB5000建议了一些关键实践(Key Practices)来达到该目标,但这些实践只是提出了在具体实践过程应该注意的事项,并没有列出具体可采用的工程技术。

那么,我们在过程改进中应该如何基于GJB5000顺利并结合IBM Rational进行过程改进实施并落地生根,从而提高国防单位的产品开发中软件工程规范化和整体质量呢?

IBM Rational 软件交付平台(以下简称:SDP)是IBM针对软件开发工程标准化而推出的一整套解决方案,它的全称是IBM Rational软件开发平台(Software Delivery Platform)。

IBM SDP的核心是软件开发治理,如下图所示,SDP关注于软件开发过程的治理,目的有三,同时这三方面也恰好符合国防单位基于GJB5000进行改进所要达成的目标:

1. 确保开发活动能够有效支持IT和业务发展目标,同时这也符合国防单位构建自己的软件开发整体平台所要达到的统一规划,放眼未来;
2. 减少风险和变更,从而使软件开发投资受保护;
3. 提升软件开发的规范性和成熟能力,从而达到国防单位持续发展,循序改进的目的;

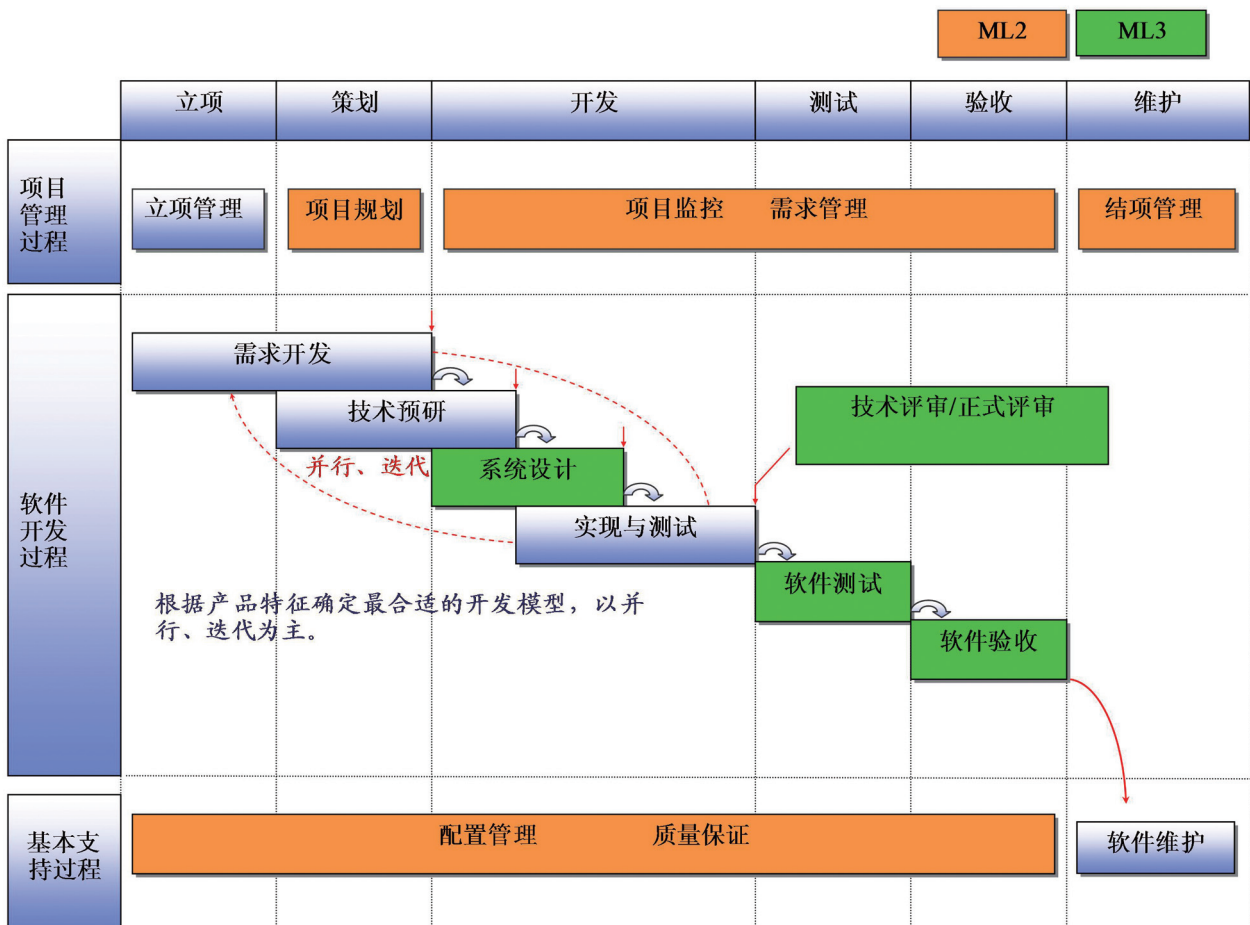


接下来,我们需要怎样的一个方案来达到以上的目标呢?

2 IBM Rational SDP助力国防单位建立基于GJB5000第二和部分第三级成熟度等级的软件工程平台解决方案

该方案主要结合国防单位的现实和希望达成的最终目标,主要面向改进的初期需要涵盖的GJB5000第二级成熟度级别的关键过程域,并适当的引入同行评审这个第三级成熟度级别的关键过程域。

整个GJB5000第二级成熟度级别主要关注软件工程中的管理问题,同时结合国防单位的需求,我们还会在技术领域引入测试和建模解决方案,所以,我们的方案将分为管理类解决方案和技术类解决方案,而在管理类解决方案中有细分为需求管理,项目管理和支撑类解决方案,如下图所示的覆盖范围。



在我们的方案中不仅涉及Rational的相关产品,更重要的是包含了一整套指导软件工程实践的方法论,具体体现在Rational Unified Process(RUP)和Rational Method Composer(RMC)。我们方案中使用的所有工具时都离不开相关方法论的指导,在开发的过程中掌握一个好的开发方法是成功的关键,工具只有在好的开发方法的指导下才能发挥作用;反过来好的方法也需要高效的工具支持来提高工作效率和质量,两者是相辅相成的。

2.1 管理类过程的解决方案

2.1.1 需求工程

2.1.1.1 实施需求工程能够解决国防单位目前存在的那些问题

国防单位通过使用RRC/DOORS和ClearQuest搭建的需求工程平台,实施有效的软件配置及变更管理后可以循序渐进地解决国防单位目前软件开发中的存在问题:

- 如何遵循GJB5000成熟度第二级和第三级的RM、RD关键过程域进行有实效的改进的最佳实践;
- 目前的需求管理松散,没有完整的需求定义和管理体系;
- 没有需求变更管理,造成无法有效记录并跟踪变更请求;
- 无论是技改项目还是产品开发都存在较严重的不受控的需求变更,造成最终形成的产品无法和原始需求对应;
- 对于用户的需求难以正确有效的获取;
- 不规范的需求描述和需求无法条目化,造成需求理解的偏差;
- 缺乏需求属性的定义,难以以不同视角展现需求,如需求风险大小,需求优先实现级别等;
- 缺乏需求到设计建模、需求到测试的追踪能力,使需求的实现验证无法得到保证;
- 无法对需求基线化管理,不了解不同的需求基线差异;

针对以上存在的问题,国防单位如何结合GJB5000中的RM和RD关键过程域切实有效的进行改进?我们认为IBM Rational需求工程平台可以结合国防单位自身具体实际助力我们符合GJB5000成熟度等级2中RM关键过程域的要求。而且在更深层次上解决软件需求工程中的问题。

2.1.1.2 IBM Rational需求定义平台设计方案

对于需求进行有效的工程活动,将奠定项目成功的基石。实现有效的需求工程,第一步就要解决需求的定义问题。项目的干系人众多:国防单位、业务部门、开发人员、维护人员等等,造成了需求的来源众多,而且渠道多样:开会收集的,市场调研的、电话沟通的、电子邮件交流的等等。但由于缺乏必要的系统平台,于是各种各样的需求就四散在各地,难免造成需求的遗漏,更无从进行有效的管理。为了解决需求的定义和确认,通过RRC提供的术语及词汇表维护、业务流程建模、用户界面草图、场景描述、故事板和富文本格式的需求规格实现了:

- 定义业务需求
 - 识别涉众
 - 捕获业务目标和对象
 - 流程优化
- 定义用户需求
 - 识别角色和用例
 - 用例优先级排序
 - 用例细化

- 定义软件需求
 - 从用户需求转化为功能需求
 - 定义非功能需求

同时通过遵循必要的需求评估流程,对定义的需求进行确认和审批,形成需求规格说明并纳入需求管理。

系统越大越复杂,出现的需求类型就越多。一个需求类型不过是需求的一个类。通过确定需求类型,团队可以把大量需求组织成意义明确且更容易管理的组。

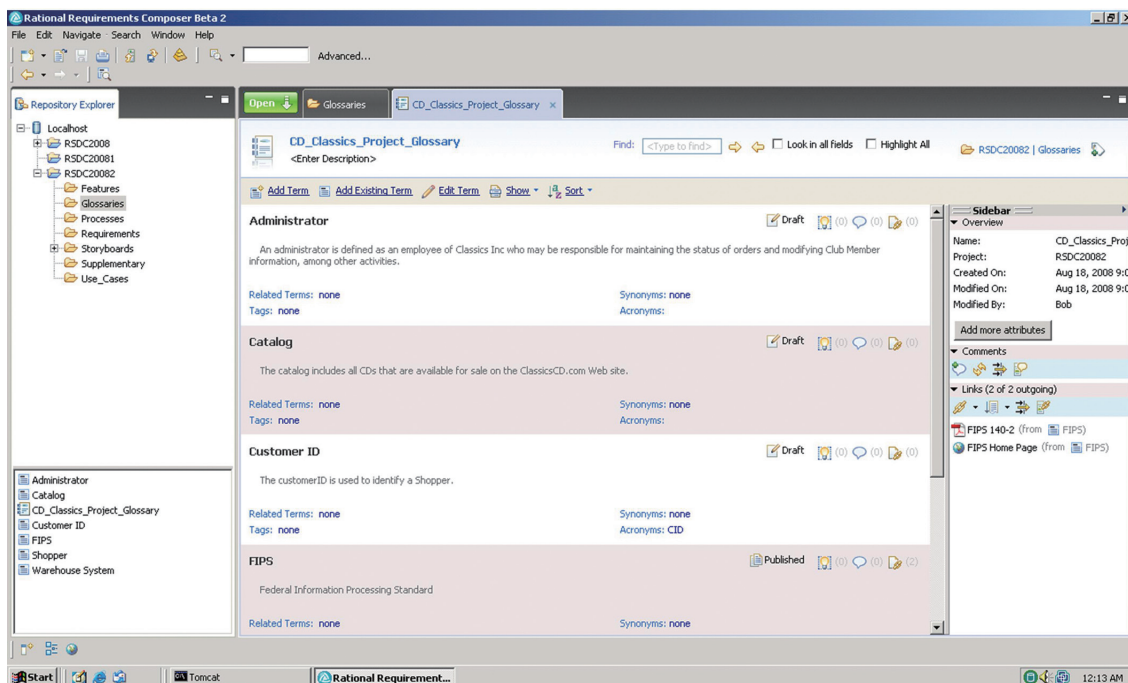
需求管理的重要内容之一就是需求跟踪。可以根据不同需求类型的导出情况,在不同的需求之间建立追踪关系。譬如,用户需要决定了要构建产品的特性,产品的特性又决定了产品的软件需求,等。在这些不同类型的需求之间建立关联,一旦其中的某些需求发生变化,就可以确定它可能带来的影响,从而制定相应的策略。

定义需求时无论怎样谨慎小心,也总会有可变因素。变更的需求之所以变得难以管理,不仅是因为一个变更了的需求意味着要花费或多或少的时间来实现某一个新特性,而且也因为对某个需求的变更很可能影响到其他需求。应确保赋予需求一个有弹性的结构,使它能适应变更,并且确保使用可追踪性链接可以表达需求与开发生命周期的其他工件之间的依赖关系。管理变更包括建立基线、确定需要追踪的重要依赖关系、建立相关项之间的可追踪性,以及变更控制等活动。

2.1.1.2.1 获取定义业务词汇

在需求定义中,必须使用业务中最常用的术语和表达方法来定义常用词汇。此后您还应该在业务的所有文本说明中始终如一地使用常用词汇。这样可以需求工程的一开始就保持需求领域说明的一致性,从而避免项目成员对术语的使用及其含义产生误解。

RRC提供了术语和词汇表的定义、同义词关联、缩略语对等以及状态等级等功能,同时通过方便的超链接功能使词汇表的履历与项目需求的定义保持动态的一致。并且可以通过标签(Tag)功能对术语和词汇表进行基线化。

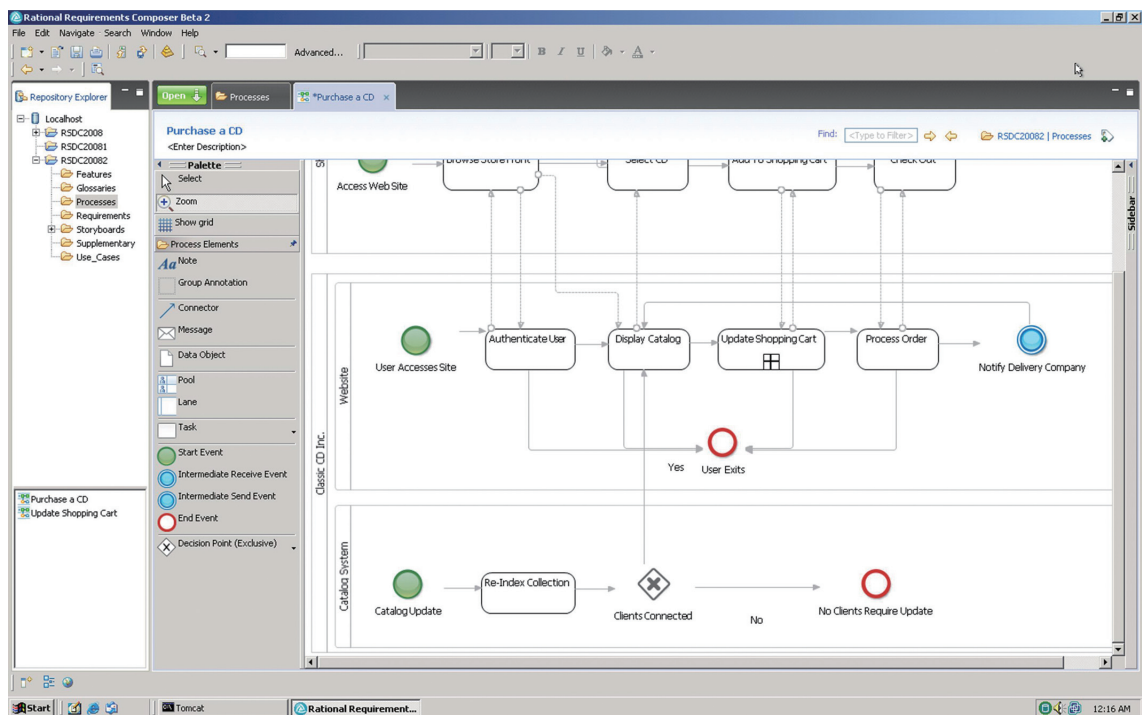


2.1.1.2.2 通过业务建模, 深入理解用户的需求

由于企业业务活动的多变性与复杂性, 要准确理解目标系统的业务需求, 首先必须了解目标系统的应用环境, 也就是企业的业务过程与协作流程。因此, 对于收集后经过评估和分派的原始国防单位需求, 就需要由领域专家对国防单位的业务流程进行业务建模, 与国防单位深入交流, 以深入理解国防单位现有的工作流程, 并对其进行相应的优化, 发现可以通过IT系统实现的环节, 进而将业务需求转化为软件需求, 从而为今后的设计、实现指明方向。

Rational Requirement Composer(RRC)提供基于Business Process Modeling Notation (BPMN) 2.0的子集符号协助国防单位业务人员和开发人员创建和优化业务流程, 同时将业务流程与业务目标和业务对象相关联, 并通过泳道图、对象流、时序流、信息流等建模方式构建业务模型, 从而实现:

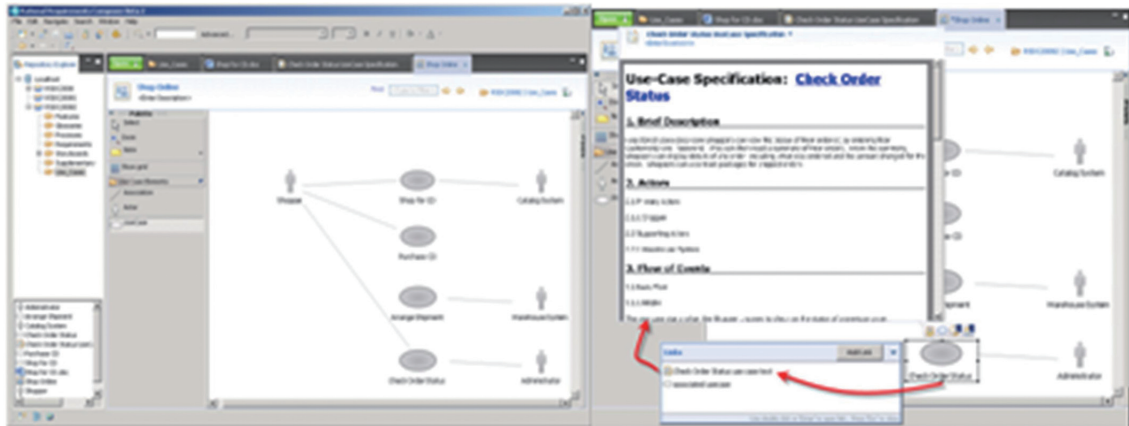
- 调研国防单位各部门的业务活动情况, 了解国防单位的组织中当前存在的问题并确定改进的可能性。
- 确保国防单位、用户和开发人员就项目的业务需求达成共识。
- 导出国防单位所需的业务需求。
- 有助于明确项目或产品的用户需求和功能性需求。



2.1.1.2.3 从用户角度出发, 准确描述软件的功能需求

用例建模是被业界广泛接受的需求开发与描述方法, 用例建模站在用户的角度, 从目标系统的外部以一种可观测和可验证的方式描述目标系统的预期行为, 描述系统如何与最终用户以及/或其他系统进行交互。用例易于被业务部门理解, 可以清晰定义目标系统的边界。用例模型在系统功能与最终用户之间建立了明确的关联, 便于企业业务部门安排合适的需求评审人员对其相关需求进行评审, 既可以提高需求评审的效率, 又可以保证需求的正确性。用例主要描述系统的功能性需求, 系统的非功能性需求则通过需求补充规约进行描述。

RRC以基于用例(Use Cases)的需求组织方式和及其灵活的超链功能, 将最初的基于富文本(RichText)的需求描述按照国防单位的业务流程逐步细化, 并结合以原型、词汇术语的有机组合, 最终以立体的方式展现国防单位的软件需求规格。



2.1.1.2.4 利用原型准确把握用户界面技术,使系统达到最佳的可用性

原型作为一种软件需求定义方法,代表了开发者对于如何实现一个特定的用例的一种观念。另一方面用户对原型的评价可以指出软件需求中的许多问题,在你开发真正产品之前,可以最低的费用来解决这些问题。

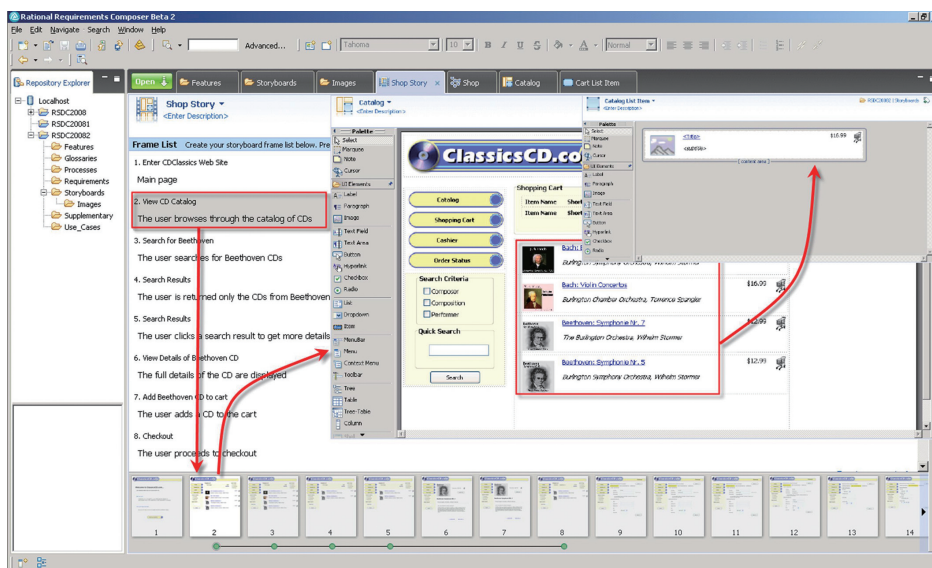
同时原型作为一种软件需求定义工具,用它它可以探索不同的用户界面技术,使系统达到最佳的可用性,并且可以评价可能的技术方案。

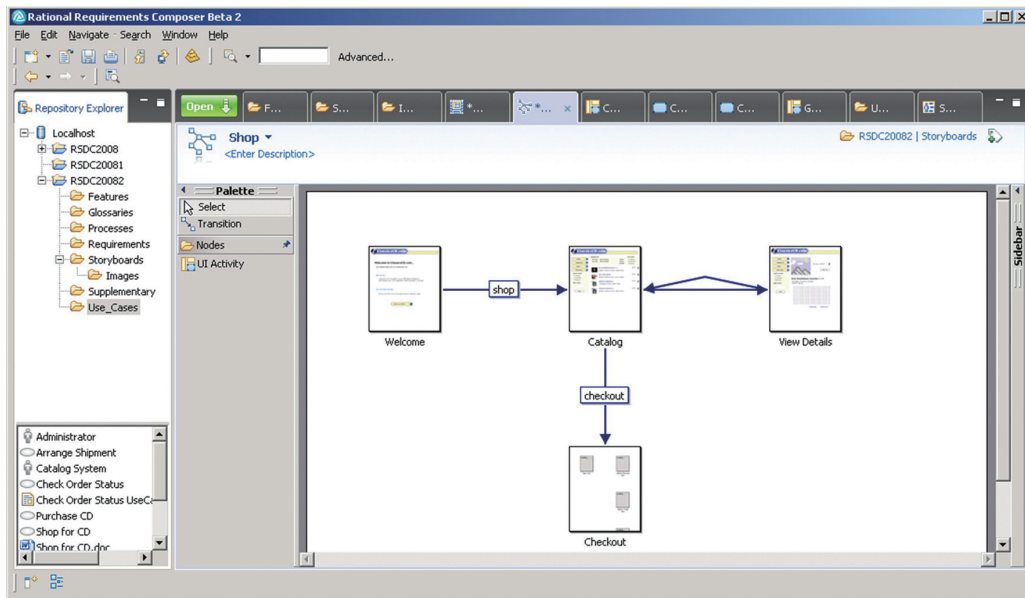
RRC提供了故事板(Storyboards)、草图(Sketch)、屏幕流(Screen Flow)和组件(Part)等多样手段,为开发国防单位原型提供了最佳平台。

当用户在考虑原型中所提出的功能可否使他们完成各自的业务任务时,原型使用户所探讨的问题更加具体化。您可以利用故事板并结合“行为原型”(behavioral prototype)在屏幕上显示出用户界面的正面像,并允许这些界面之间的一些导航,在包含必要的功能基础上对对话元素细化为特定的屏幕显示、菜单和对话框。一旦确定了软件需求并勾画出屏幕的大体布局,你就可以从最佳使用的角度设计每一个用户界面元素的细节。比起直接从用例的描述跳跃到完整的用户界面的实现,然后在软件需求中发现重大错误,利用逐步求精的方法所花费的努力将会更小。

利用草图(Sketch)可以摒弃书面原型(paper prototype)以及简单的画图工具的缺陷,它可以把一个系统某部分是如何实现的呈现在用户面前。成为探索特定的用户界面设计的合适工具。同时结合组件(Part)功能可以使你迅速地实现并更改用户界面,而不管在界面后面的代码效率的高低。

最终您可以将以上的各种原型元素以屏幕流(Screen Flow)的方式来展现用例中的活动顺序,提供一个基于文字需求的满足业务场景的功能场景。



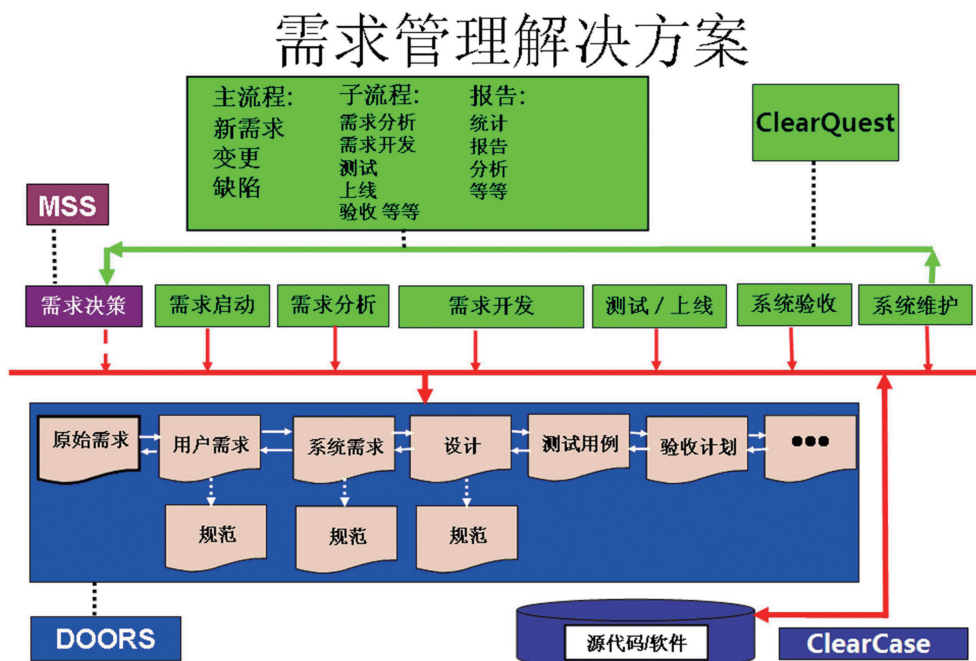


2.1.1.3 IBM Rational需求管理平台设计方案

2.1.1.3.1 总体解决方案

解决方案是设计在IBM Rational需求管理工具DOORS、变更配置管理工具ClearQuest的基础上。DOORS和CLEARQUEST都是在软件工程管理业界技术上主导和领先地位的产品。在OVUM及YPHISE的报告中都连续多次名列榜首的解决方案，并在国内外也得到了成功的应用，涉及的领域有航空航天、电信、汽车、政府、电力等各个行业。

在理解国防单位目前管理体系的基础上，我们确立了需求管理的整体结构和流程，如下图。



IBM Rational通过DOORS建立统一的需求追踪模型,记录用户需求、系统需求、设计、测试用例等资产,并进行追踪和覆盖分析;通过ClearCase收集系统代码和程序,进行系统发布和版本管理;通过ClearQuest管理新需求的总体流程、需求变更流程、缺陷流程、紧急需求处理等流程。

2.1.1.3.2 需求引入

对于已经开发出来的需求,需要进行有效的管理,借助DOORS我们可以把RRC中的各类需求逐条记录到用数据库实现的需求库中,每条需求对应数据库中的一条记录,而对于一般用户、业务人员以及企业领导,DOORS还提供了方便易用的Web界面,可以查看项目需求的进展情况。

为了对需求进行进一步的管理,我们还可以通过对需求的深入分析,给每条需求添加必要的属性,例如:优先级、来源、稳定性、成本、难易程度、风险等等。然后,我们就可以利用DOORS提供的数据库的强大检索能力,通过选择不同的属性作为过滤的条件,建立不同的视图,让我们能够从不同的视角对现有的需求加以审视和权衡,以确定项目的范围和开发的重点。

DOORS系统也提供二次开发脚本接口,可以定制数据导入接口,可以使用该接口将原有系统中的数据导入到需求管理系统中。

DOORS可以设置需求标准模板,对进入系统的需求进行规范化,并可通过追踪视图,可便捷地对需求的满足性进行分析,从而确保需求的规范管理。

2.1.1.3.3 实施有效地审批流程

当导入了RRC中采集的各类需求后,接下来的工作是如何有效的实施审批过程,有效的审批方法来自:

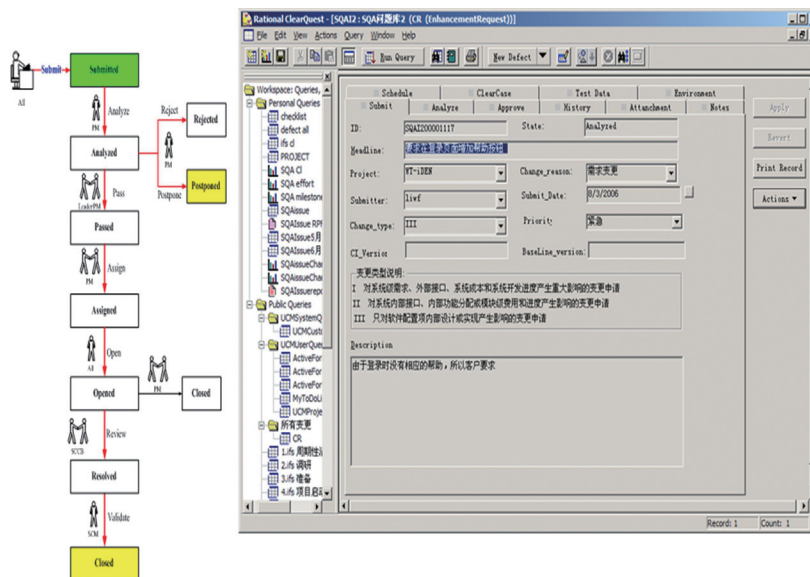
- 建立请求评审小组(变更管理组和需求评审管理组)
- 快速查询急需评审项

ClearQuest具备优秀的各种查询功能,评审小组通过ClearQuest界面,进行各种条件查询,立即定位所寻的评审项。当发现不足信息时,可以从ClearQuest发送E-mail通知项目人员,以附件的方式或在表单中追加内容的方式补足信息。

- 需求评审组寻找重要级高的需求请求和开发问题
- 变更管理组寻找优先级和重要级高的变更请求
- 不足信息迅速通知相关人员补齐

- 确认请求会涉及到哪些需求,分配给相关的人员进行处理

评审小组在ClearQuest中,通过执行“分派”动作,可以将变请求的状态改变,使请求自动流到下一个环节。



2.1.1.3.4 有效地组织管理需求

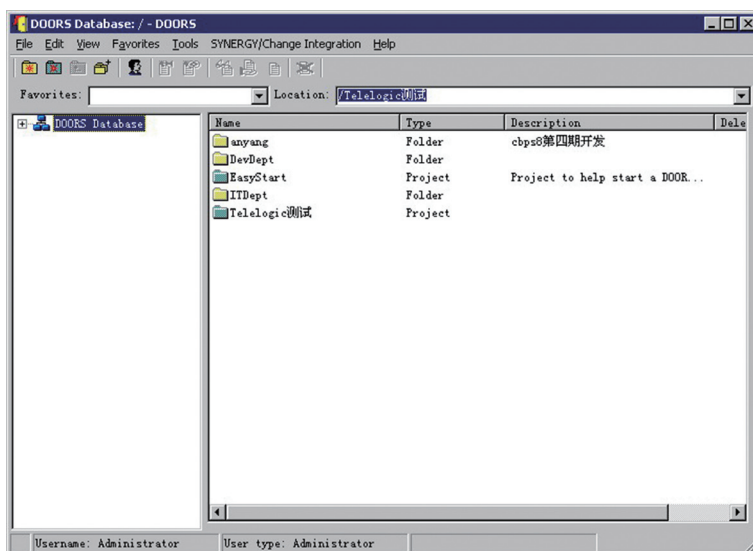
在审核部门审批完成后,当评审组决定实施该请求并将其分派至相关需求负责人后,该负责人应实施有效的需求管理。

DOORS可以从以下方面强化需求的组织和管理:

- 需求定义

作为业界领先的需求管理工具,DOORS具有灵活的需求数据库管理能力,并具有独一无二的可同时打开多个需求项目的能力,以帮助用户最终实现项目级或产品级的需求重用和需求追踪。

在DOORS中,项目是从属于需求数据中的,如下图所示,绿色文件夹代表的是项目,黄色文件夹代表的是目录,需求数据可存放在上述两种文件夹中。

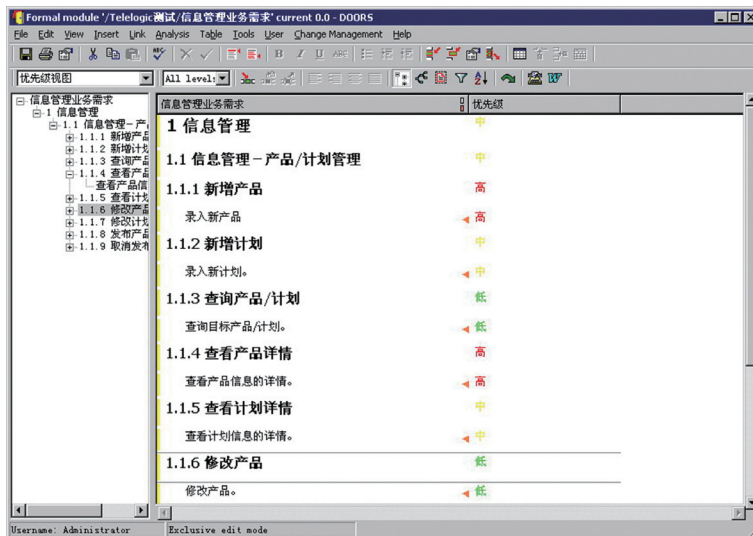


在访问控制权限允许的前提下,用户可以同时打开不同项目的需求数据,即可同时管理多个项目。

在DOORS中,用户可灵活地定制各种文档类型,包括用户需求、系统需求、测试用例、设计等。

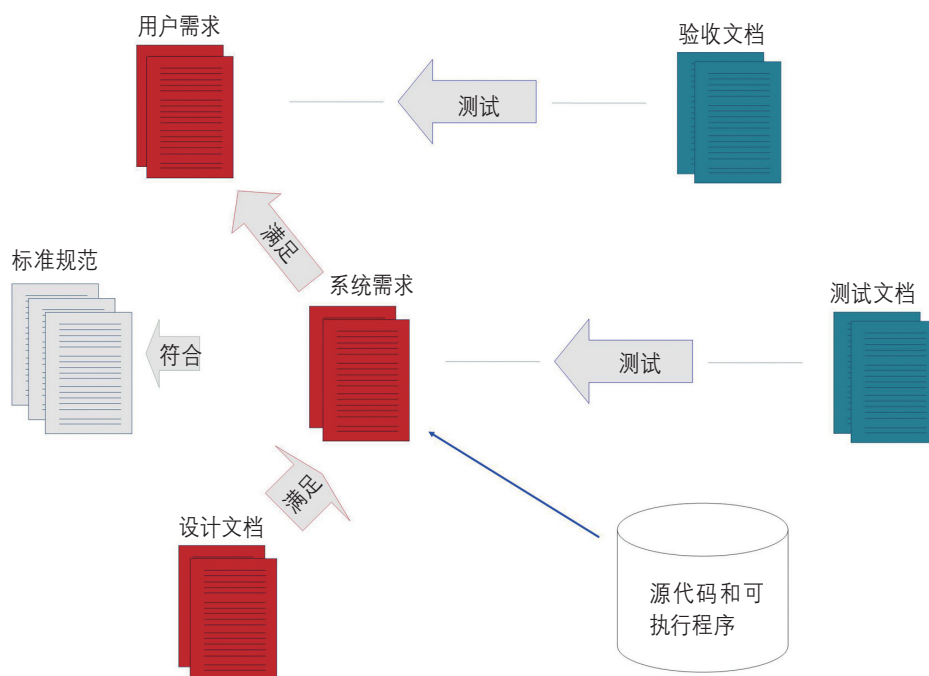
DOORS提供了强大的定义需求属性的能力,用户可以自定义属性的名称、类型、缺省值等等。基于需求属性,用户可以对需求进行查询、排序和过滤等等。可以这样说,需求属性可以体现用户需求管理的深度。

例如,对于用户需求来说,必要的属性即是表示出各需求的优先级。DOORS可以灵活地定义出优先级,如高、中、低,并可用不同颜色来表示优先级,如下图:



直接点击优先级栏,可选择升序/降序排列。

在DOORS中,可通过定义项目中的目录、文档模块和链接关系模块来定义项目的模板。例如,在下图的项目中定义了包含用户需求、系统需求、系统测试、验收测试及标准规范等文档模块,以及满足关系、合规关系和测试关系的链接模块。



另外, DOORS中提供了项目创建的向导功能,可迅速定制项目模版。

2.1.1.3.5 定义文档模板

DOORS中缺省支持许多业界标准规范的模板,用户还可以定义自己所习惯的文档模板。

DOORS缺省支持IEEE, ISO, 以及RUP等规范的文档模板,也允许用户自定义文档模板。

• 需求分析

需求负责人得到的需求是用户需求,或者说用户需求。用户需求的描述是基于业务语言的,非技术性的,功能描述性的。描述了对一项具体的业务功能。

在开发工作开始前,用户需求必须转化为系统需求,为技术开发人员所理解。系统需求是基于技术性语言,术语;描述了对系统的功能要求。系统需求必须严谨,清晰,无二义性。

整个开发活动是以用户需求来驱动的,任何歧义,非单一性都可能导致需求误解,需求丢失,导致后续开发活动的无效性。

一项用户需求的实现,可能要求多个应用系统工作,例如,计费,网管都需要做相应的变更。基于此,用户需求的分析主要是厘清用户需求的范畴,消除歧义,在某些情况下,大的用户需求要界定为多个子系统业务的描述,实现需求描述的精确,单一,唯一,可对应性。

DOORS提供需求分析的方法培训以及工具的实现:

- DOORS对需求分析提供一种可视化的手段和方法,通过各种属性及链接过滤分析、统计图表对需求进行分析和说明,并使用DOORS中的链接将用户需求与系统需求建立关联关系,进行追踪分析;
- 在需求分析的结果整理成文档时,可以从文档模板库中选择使用业界标准的需求模板或者自定义的模板,规范任何需求(或测试)文档的框架或结构。

- 需求分解

需求分解包含2个层面的含义:

- 一个用户需求可能是基于多个系统的同时更新实现的。因此要将用户需求分解成对不同系统的需求
- 对同一个系统, 一个用户需求也可能是基于多个系统需求来实现的。因此也存在需求分解工作。

语素转换或翻译。需求分解, 不仅要将一个用户需求对应到一个或多个系统需求。而且, 要实现用户需求到系统需求的转换。这是两种不同的语言。用户需求是面向业务人员, 面向国防单位的; 而系统需求是面向技术人员, 基于应用系统的语言。在需求分解过程中, 要实现此2种语言的转换。最终将来自于市场的业务描述, 转换为设计开发人员能理解的要求, 分解成可执行的任务。

整个需求分解过程存在一定的方法论, IBM Rational将提供该方法论, 并提供相应的工具支持。借助于 DOORS, 基于一定的方法论, 将:

- 需求条目化。单一需求描述单一功能, 消除歧义。
- 需求分级分解。单一的用户需求分解成清晰, 可执行的系统需求。

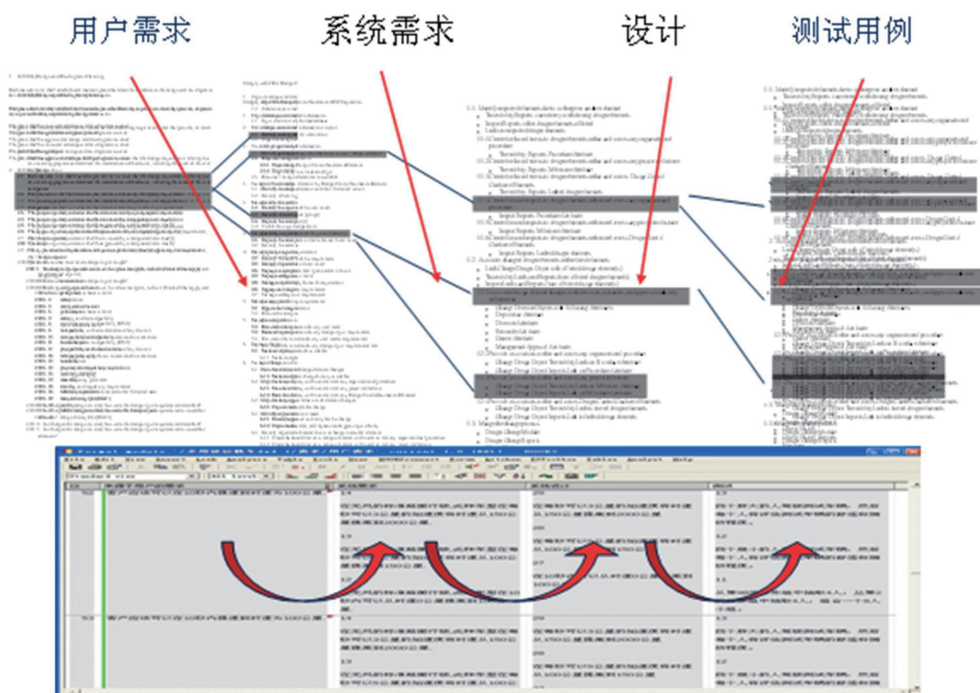
需求语素转换。通过需求分解的环节, 一项对业务的具体描述生成了对系统的开发或维护请求。业务语言转换成为技术语言, 从而能够实现需求从业务部门到设计开发人员的无歧义传递。

- 建立需求项的跟踪关系

DOORS中将映射关系直接显示在文档中使得对映射关系的追踪不必脱离文档而能够直接即时操作完成。

- 通过DOORS可以建立各个分解需求项之间的关系, 快速而简洁的操作建立它们之间的映射(关联)关系。从而使国防单位在修改和变化某一个需求时, 可以迅速定位该需求会关联的到哪些需求项, 不会有需求变更的漏查和漏执行。
- DOORS能够有若干种不同的方式分析和观看映射关系, 其中最有效的方式是在一个视图上显示所有从用户需求到系统需求(或测试用例)的映射关系(通过一个表单的显示方式), 从而找出系统需求满足用户需求(或测试用例)的覆盖关系(即每一用户需求项都映射到至少一个系统需求项)和覆盖率, 或从系统需求项(或测试用例)到用户需求的覆盖关系(即每一个系统需求都是用来满足至少一个用户需求的)。

如下图所示。



- 建立需求项与需求请求之间的关系

为了对整个需求执行状况实施完整的跟踪管理, 需要建立需求与原始的需求之间的关系, 从而在改变需求或需求请求的状态时, 二者才能形成互动。ClearQuest与DOORS的无缝集成, 充分满足了这一链接关系的建立和管理。

- 需求文档的权限管理

此外, 在管理需求文档资料时, DOORS中设定了严格的文档访问权限, 该权限能够控制只有被授权的用户才能对其进行修改和更新。DOORS的访问权限能够应用在数据库的各层结构。

2.1.1.3.6 建立测试方案

需求组织管理环节完成后, 国防单位应将转向如何建立有效的测试方案上。该环节的活动内容包括:

- 建立测试计划、测试用例和测试报告格式

- DOORS中可以实现对测试计划和方案的管理。对测试计划的管理是通过属性的自定义和使用来管理的, 作为DOORS中的一个文档, 测试计划能够创建属性“测试执行状态”, 分别对应“通过”, “未通过”等状态。使用DOORS中对属性的过滤或搜索功能, 可以将该属性所有对应“通过”或“未通过”的测试用例过滤或搜索出来供相应人员参阅。
- DOORS的测试跟踪套件能够在DOORS中管理测试运行结果。用户使用配置后的表单来填写测试运行结果并能够比较同一测试用例不同次运行的结果
- 创建测试用例
- 测试用例也可以作为DOORS一种文档存储在DOORS中。
- 创建测试报告文档格式
- 报告文档格式作为测试方案的一部分, 也应存储在DOORS中

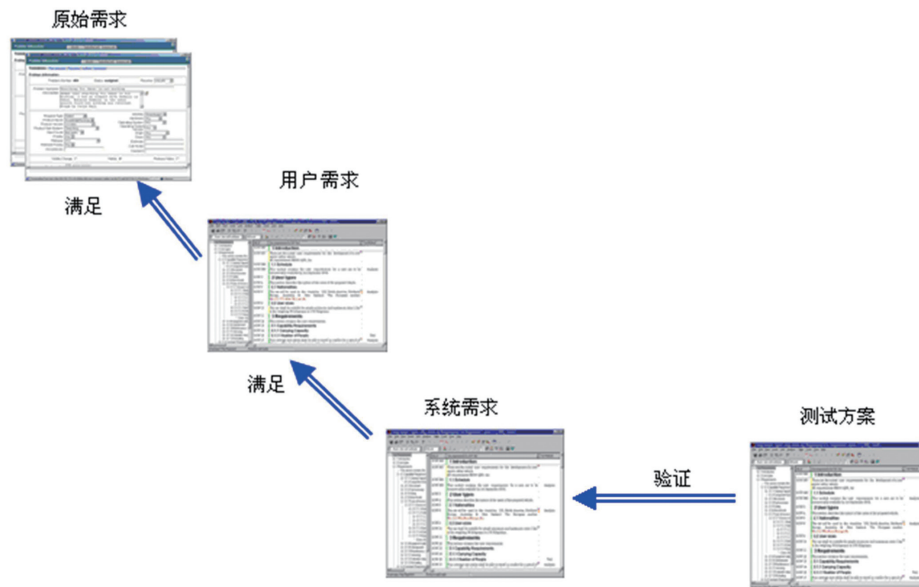
- 建立需求与测试方案的跟踪关系

- 将测试项文档保存在DOORS中将测试项条目化管理, 并建立从用户需求项到相应(一个或多个)测试项的映射关系;
- DOORS中建立从需求到测试用例的双向映射关系, 并且通过映射关系从一个需求项追踪到相应的一个或多个测试用例, 或者从一个测试用例追踪到相应的一个或多个需求项。这种双向映射关系也可在测试报告和测试计划之间建立并进行追踪
- 这些映射关系的建立提供了从用户需求到系统需求, 和从用户需求到测试用例的追踪机制, 某个用户需求项产生变更后, 用户能够沿着映射关系自动触发并打开对应的系统需求项(或测试用例)进行跟踪

- 提供联合测试管理方法

在DOORS中, 通过创建和使用需求文档的“完成状态”属性, 以及测试用例的“测试状态”属性, 再通过DOORS简单的二次开发能力将这两种属性连系起来, 实现只有当若干个测试用例的“测试状态”都变为“通过”后, 对应的需求项的“完成状态”才能够自动变为“已完成”。实现, 一项业务, 多级测试, 统一通过的联合测试方法。保证测试的完整性和产品的高质量。

下图描述了从原始需求到用户需求、系统需求以及系统方案的全部追踪关系。



2.1.1.3.7 审核需求和测试文档

在需求文档和测试文档完成之后,国防单位还需对需求及测试文档进行进一步审核,审核内容应包括:

- 需求项与原始的需求请求是否相吻合

使用DOORS和ClearQuest的集成功能,可以从DOORS的画面中,启动ClearQuest,查找与该需求相关联的需求请求信息;

使用DOORS的需求项目之间的连接和跟踪功能,可以审核:

- 需求分解后的跟踪关系是否正确
- 需求与测试方案的跟踪关系是否正确
- 测试计划与测试用例、测试报告之间的关系是否正确

该审核过程是分派给集成商的需求准确性和可跟踪性的有效保证,也是国防单位控制集成商开发质量的基础。

2.1.1.3.8 修改需求文档及测试文档

在审核需求和测试文档中,如果发现问题,需要对这些文档进行修改,在修改过程中,DOORS可以帮助你迅速查找一个需求修改会辐射到何处,并帮助你准确定位到辐射点。

同时,需求修改的过程,在DOORS中都留下完整的历史。修改,删除,移动和复制等变更历史记录可以对需求项的任何变更记录在案以供查询,包括:

- 变更的时间
- 谁做的变更
- 变更前后的内容比较

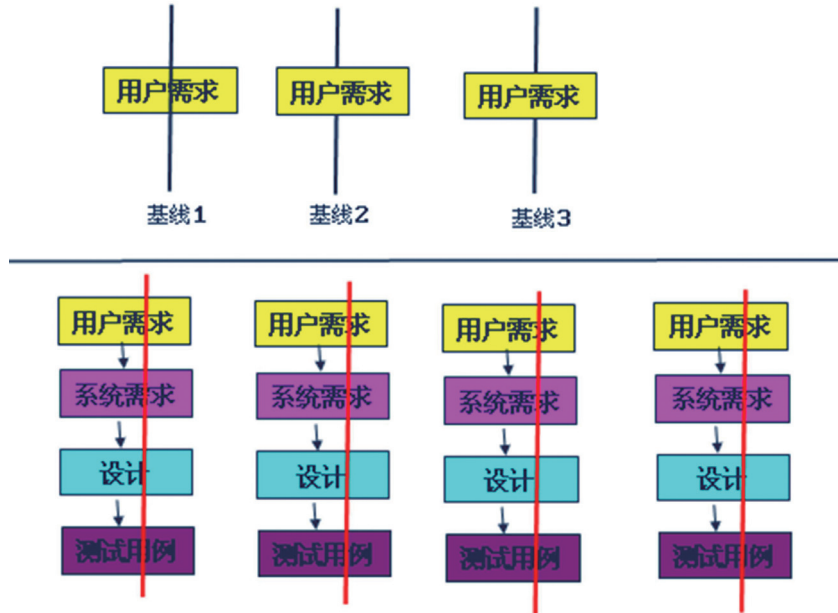
2.1.1.3.9 需求和测试文档入库集中管理

需求和测试文档评审的结束,标志着国防单位项目管理的一个里程碑,该控制的主要活动包括:

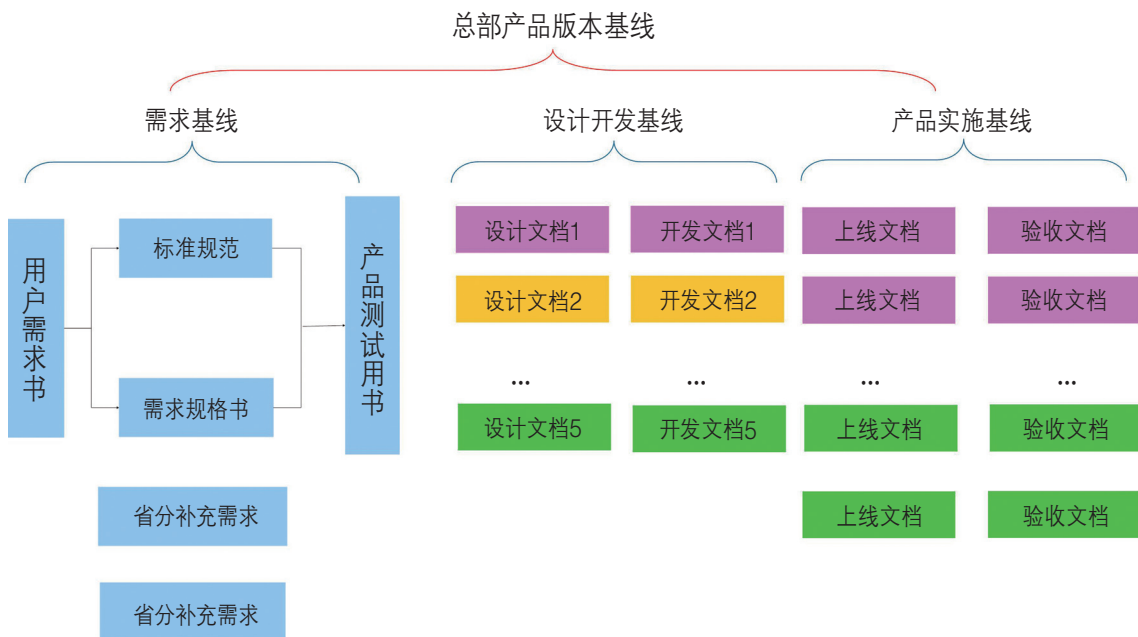
- 需求和测试文档入库管理
- 建立里程碑基线

DOORS的版本管理功能可以帮助将文档入库管理,并可建立基线。DOORS的基线提供审核功能,通过电子签名来确保基线的有效审核。

DOORS的基线功能提供两个纬度的管理,提供单独文档基线和项目部分或全部文档的基线集,基线集是作为项目的统一基线,可确保项目成果的一致性。



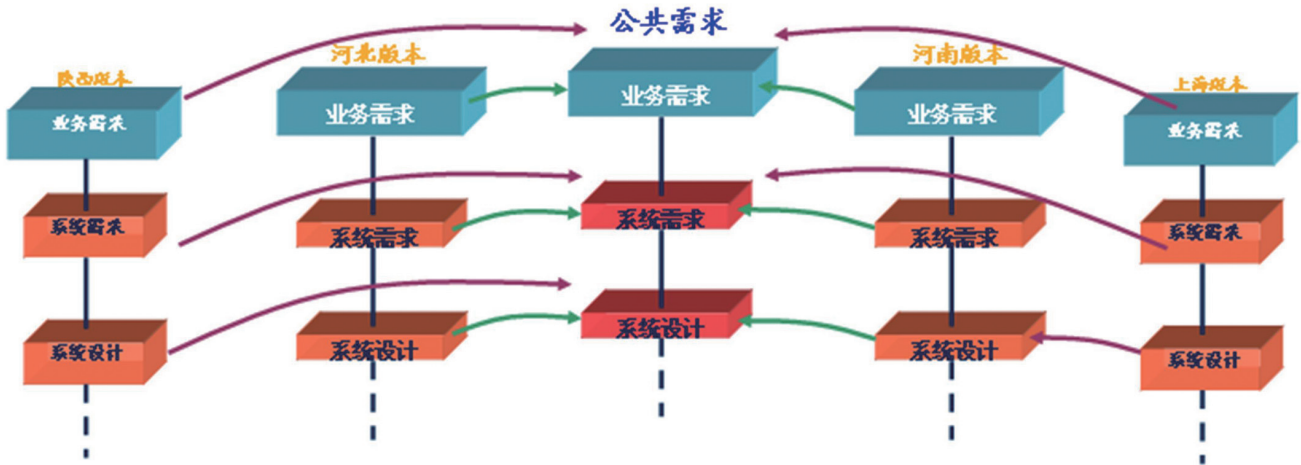
DOORS中的基线集能够建立起整个项目的多层次的基线结构,包括需求基线、设计开发基线、产品实施基线,以及总部产品版本整体基线。



在ClearQuest 中同时记录里程碑基线和时间,以便将来必要时随时查阅。

2.1.1.3.10 在需求变化中维护统一的需求版本

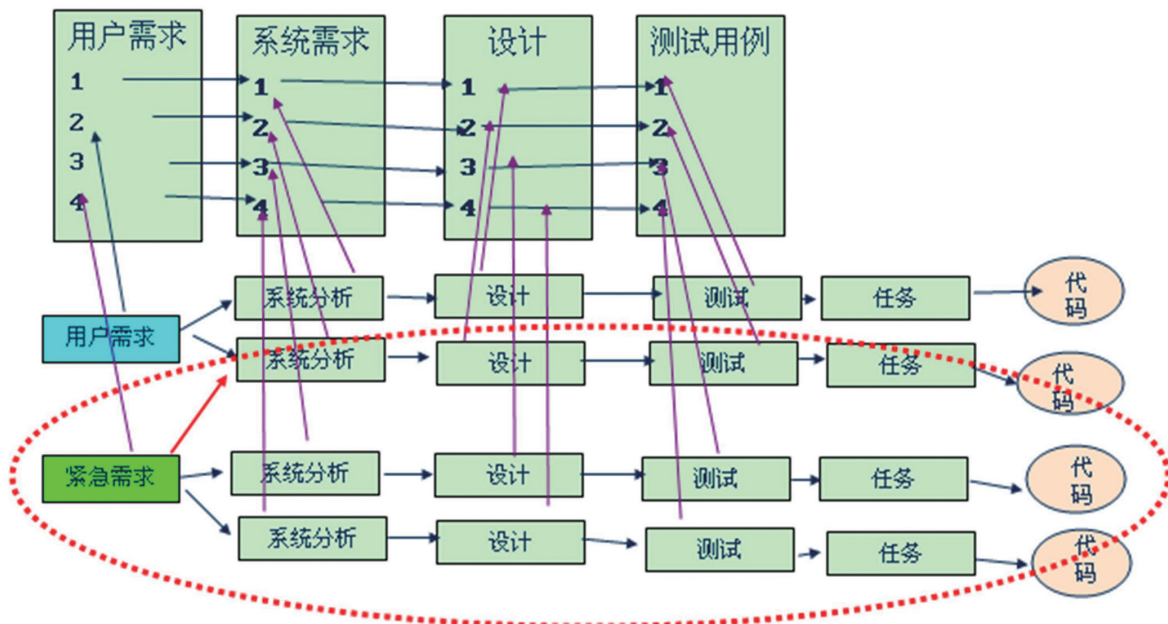
在需求的运行维护中, 无论来自总部和省分需求的变化, 通过总部需求和省分需求的继承关系都可以实时地捕获到, 从而保持需求版本的完整性和一致性。



2.1.1.3.11 紧急需求

由于IBM Rational方案技术灵活的特点, 可以自由地处理紧急需求。当紧急需求到来时, 由于需求的整个进展过程已经被分解为多个开发请求, 因此如果紧急需求的一部分需要应用原有的开发请求, 则可以方便地重用原有开发请求, 并提高相应开发请求的优先级, 同时再为紧急需求的其余内容创建新的开发请求。

由于IBM Rational方案中的流程和需求都可以很方便的分解、建立关联重新组合, 同时支持一对一、一对多、多对一和多对多的关联, 因此对紧急需求的支持并不会破坏原有需求的开发和进展。



2.1.1.4 方案如何提升国防单位需求管理能力

本方案的目的是帮助国防单位在短期内建立一个需求管理流程, 以提升国防单位在项目管理上的以下几方面的能力:

2.1.1.4.1 建立一个标准化需求管理流程体系

在国防单位引入IBM Rational的需求工程解决方案后, 来自各方(国防单位或集成商)的各类请求和信息都可集中搜集管理到统一的数据库中。当各类信息集中管理在数据库后, 国防单位的各类信息执行状况可以根据建立好的流程管理体系进行跟踪进行管理。

国防单位的团队人员, 工作在标准化的流程系统上, 这将极大的促进国防单位的团队人员的工作行为的规范化和职业化, 这将带动国防单位团队人员整体素质的提高

IBM Rational的技术专家还可以通过工具的引入帮助国防单位建立起一整套的流程管理体系, 实现对集成商开发项目状况的完全控制。该流程是一个灵活的、可拓展的管理流程, 可以支持团队的不断发展, 适应于国防单位发展过程的需要。

2.1.1.4.2 支持KPI体系

当ClearQuest汇总了项目和团队人员的各种信息后, 团队人员的工作效率和工作成果可以用ClearQuest的条件查询和图表功能量化地展示出来; 如:

- 到那天为止谁完成了多少工作
- 目前在谁手上有多少正在进行的工作
- 谁在忙、谁在闲

等等。ClearQuest还可以提供多种统计信息, 为KPI的管理提供有力的原始数据。 这些数据可以有效地帮助国防单位合理的评估团队的工作效率以及分析如何分配合理的资源。

2.1.1.4.3 提高团队整体的开发效率和产品的开发质量

需求管理是建立在标准流程基础上的需求、测试、配置、变更管理的综合系统。以需求管理为主线, 对变更、配置和测试管理实施有效的控制后, 国防单位整体会在:

- 团队的整体效率提高
- 需求定义明确
- 测试与需求跟踪力度较大
- 开发资产共有化和合理重用化得到相当程度上的改善, 使国防单位对项目的开发管理效率和开发质量的到极大的控制和改善。

2.1.1.4.4 建立规范的产品发布管理制度

伴随着ClearCase的实施, 也是IBM RATIONAL将其软件的配置管理和发布管理的最佳实践经验向国防单位的知识传递过程, 在实施的过程中, IBM RATIONAL公司会引导国防单位建立规范的产品发布管理规章制度, 实现国防单位内外管理的统一;

2.1.2 项目管理

2.1.2.1 实施项目管理能够解决国防单位目前存在的那些问题

国防单位通过使用Rational Team Concert和Rational Method Composer以及Rational Focal Point搭建的项目及项目组合管理平台, 实施有效的技改项目和项目组合管理后可以循序渐进地解决国防单位目前软件开发中存在的问题:

- 如何遵循GJB5000成熟度第二级的SPP和SPTO关键过程域进行有实效的改进;

- 项目信息只有项目执行科室掌握, 对于高级管理层信息不可见;
- 项目管理者本身不能够有效掌握项目进展的真实状况;
- 项目的执行资源只能在项目执行科室的范围内考虑, 难于就大项目开展更大组织范围内的协同;
- 资源不能够得到高效使用, 管理者无法掌握资源的实际工作状态;
- 项目计划不清, 项目团队对于任务分工和进度安排缺乏清晰的了解, 项目团队难于对于项目建立统一的目标, 影响了项目中的协同工作效率;
- 项目的进展状况难于及时掌握, 在开发过程中容易出现超期现象;
- 项目的风险、变更、问题等相关项目范围元素的管理能力不强, 在大型项目上问题较多;
- 项目管理能力不强, 限制了在项目上新技术的引入和采纳, 项目开发工作长期在低水平上重复;
- 缺乏必要的项目最佳实践管理能力, 没有实现最佳实践的有效积累, 更缺乏对于最佳实践的复用;
- 缺乏院所级的项目管理能力, 对于一些项目共同面临的院所级问题, 不能够有效地利用高级管理的资源进行统一的协调解决, 问题处理的效率低下;

针对以上存在的问题,国防单位如何结合 GJB5000中的SPP和SPTO关键过程域进行切实有效的进行改进?

我们认为IBM Rational Team Concert、IBM Rational Method Composer可以助力我们符合GJB5000成熟度等级2中SPP和SPTO关键过程域的要求。而且在更深层次上解决当前项目管理中所遇到的问题, 并进而可以解决在ML3中所涉及的组织一级的相关的过程域的问题。

2.1.2.2 IBM Rational过程与项目管理平台助力国防单位GJB5000 SPP和SPTO KPA的实现

关键过程域	目的	目标	共同特征	关键实践	IBM SDP Solution		
					RUP	RMC	RPM
SPP	为完成软件工程和管理工作制定合理的计划。并将计划付诸实施。	目标1: 用于策划和跟踪软件项目用的软件估计已建立文档。	Co. 1 Co. 2 Ab. 1 Ab. 2 Ab. 3 Ab. 4 Me. 1 Ve. 1 Ve. 2 Ve. 3	活动9 按照已文档化的规程导出对软件工作产品规模（或对软件工作产品规模的更改）的估计。 活动10 按照已文档化的规程导出对软件项目的工作量及成本的估计。 活动11 按照已文档化的规程导出对项目的关键计算机资源的估计。 活动12 按照已文档化的规程导出项目的软件进度表。 活动15 记录软件策划数据。	SPP流程支持和方法支持	定义和发布项目标准化流程	参见: 2.2 2.4 2.6

关键过程域	目的	目标	共同特征	关键实践	IBM SDP Solution		
					RUP	RMC	RPM
SPP	为完成软件工程和管理软件项目制定合理的计划。并将计划付诸实施。	目标1：用于策划和跟踪软件项目用的软件估计已建立文档。	Co. 1 Co. 2 Ab. 1 Ab. 2 Ab. 3 Ab. 4 Me. 1 Ve. 1 Ve. 2 Ve. 3	<p>活动9 按照已文档化的规程导出对软件工作产品规模（或对软件工作产品规模的更改）的估计。</p> <p>活动10 按照已文档化的规程导出对软件项目的工作量及成本的估计。</p> <p>活动11 按照已文档化的规程导出对项目的关键计算机资源的估计。</p> <p>活动12 按照已文档化的规程导出项目的软件进度表。</p> <p>活动15 记录软件策划数据。</p>	SPP流程支持和方法支持	定义和发布项目标准化流程	参见： 2.2 2.4 2.6

关键过程域	目的	目标	共同特征	关键实践	IBM SDP Solution		
					RUP	RMC	RPM
SPP	为完成软件工程和管理软件项目制定合理的计划。并将计划付诸实施。	目标3：受影响的组和个人同意他们的关于软件项目的约定。	Co. 1 Co. 2 Ab. 1 Ab. 2 Ab. 3 Ab. 4 Me. 1 Ve. 1 Ve. 2 Ve. 3	<p>活动1 软件工程组参加项目建议群组。</p> <p>活动3 在项目的整个生存期内，软件工程组和其受影响影响的组一起参加整个项目的策划。</p> <p>活动4 高级管理者参加按照已文档化的规程评审对组织外部的个人和组所作的软件项目约定。</p>	SPP流程支持和方法支持	定义和发布项目标准化流程	参见： 2.2 2.3 2.4

关键过程域	目的	目标	共同特征	关键实践	IBM SDP Solution		
					RUP	RMC	RPM
SPP	为完成软件工程和管理软件项目制定合理的计划。并将计划付诸实施。	目标3: 受影响的组和个人同意他们的关于软件项目的约定。	Co. 1 Co. 2 Ab. 1 Ab. 2 Ab. 3 Ab. 4 Me. 1 Ve. 1 Ve. 2 Ve. 3	活动1 软件工程组参加项目建议群组。 活动3 在项目的整个生存期内, 软件工程组和其综合受影响的组一起参加整个项目的策划。 活动4 高级管理者参加按照已文档化的规程评审对组织外部的个人和组所作的软件项目约定。	SPP流程支持和方法支持	定义和发布项目标准化流程	参见: 2.2 2.3 2.4

关键过程域	目的	目标	共同特征	关键实践	IBM SDP Solution		
					RUP	RMC	RPM
SPP	为完成软件工程和管理软件项目制定合理的计划。并将计划付诸实施。	目标3: 受影响的组和个人同意他们的关于软件项目的约定。	Co. 1 Co. 2 Ab. 1 Ab. 2 Ab. 3 Ab. 4 Me. 1 Ve. 1 Ve. 2 Ve. 3	活动1 软件工程组参加项目建议群组。 活动3 在项目的整个生存期内, 软件工程组和其综合受影响的组一起参加整个项目的策划。 活动4 高级管理者参加按照已文档化的规程评审对组织外部的个人和组所作的软件项目约定。	SPP流程支持和方法支持	定义和发布项目标准化流程	参见: 2.2 2.3 2.4

关键过程域	目的	目标	共同特征	关键实践	IBM SDP Solution		
					RUP	RMC	RPM
SPP	为完成软件工程和管理工作制定合理的计划。并将计划付诸实施。	目标3: 受影响的组和个人同意他们的关于软件项目的约定。	Co. 1 Co. 2 Ab. 1 Ab. 2 Ab. 3 Ab. 4 Me. 1 Ve. 1 Ve. 2 Ve. 3	活动1 软件工程组参加项目建议群组。 活动3 在项目的整个生存期内, 软件工程组和其综合受影响的组一起参加整个项目的策划。 活动4 高级管理者参加按照已文档化的规程评审对组织外部的个人和组所作的软件项目约定。	SPP流程支持和方法支持	定义和发布项目标准化流程	参见: 2.2 2.3 2.4

2.1.2.2.1 完整集成化的技改和型号项目综合管理平台

RTC是一个实时协作及集中式的技改或型号项目管理综合性的平台, 它提供一个综合的信息管理平台, 内建了自动数据收集和汇报功能, 提供进行有效项目管控时所需要的实时洞察能力。

实时协作功能有效地帮助团队减少出错机会, 不同角色能够更加密切地在一个实时工作环境里遵循敏捷流程进行高效协作, 使得开发团队的生产率和创新能力提高到一个新的水平:

- RTC继承了IBM开放技术平台Jazz Team Server提供的强大协作能力和相关基础设施, 并新增加了集成的项目管理(Agile Planning)、工作项管理(Work Item Management)、构建管理(Build Management)和配置管理(Configuration Management)等功能, 使支撑开发团队的平台能力得到大大的增强;
- RTC还提供了连接器(Connectors)和桥接(Bridges)等工具, 用于扩展和集成其它外部系统;

项目管理不仅仅是管理单个的技改或型号项目, 而是管理整个国防单位内的所有技改、型号、自主研发和外包项目, 并且使国防单位在IT方面的投入更好地适应业务发展方向和目标。

IBM Rational Team Concert作为一个企业级的项目管理平台为国防单位项目管理提供了一个基础平台。

- 提供高效的甲乙双方团队协作能力(Collaboration)
- 支持敏捷开发, 并做到适度管控(Right-Size Governance)
- 透明化项目状态, 提高外包项目可预见能力(Transparency & Visibility)
- 基于开放的Jazz技术平台(Open Standard-based)
- 实现软件生命周期的工件双向追踪能力(Traceability)
- 集成的工作项管理, 并提供灵活的定制能力(Work Item)



利用这个综合性的项目管理平台, 不单能够让国防单位在项目管理工作上游刃有余, 而且能够为所里基于CMMI/GJB5000的过程改进工作, 提供极大的辅助作用, 对于提升企业在大型项目上的执行能力和跨部门的协同工作能力, 也会带来巨大的帮助。主要体现在以下维度:

2.1.2.2.1.1 转变开发团队工作方式, 实现高效团队协作(Team Collaboration), 获得更高的软件生产率

Jazz平台能够理解开发团队(Team Awareness)和他们的组织结构。你可以在Jazz平台清晰地定义和划分每一个项目和它所包含众多开发团队的层次结构, 包括: 哪些开发团队, 子团队(树状结构), 每个开发团队包含哪些成员和其角色, 每一个开发团队采用什么样的开发过程, 以及每个成员的个人信息设置等等。

Jazz平台能够与即时通讯工具(例如: Jabber, Lotus Same, Goggle Talk等)进行一体化的集成, 在开发团队的实际IDE环境中(如: Eclipse环境)提供在线提示(presence)和实时通信(messaging)的能力, 实现高效的团队协作能力。开发团队成员可以在IDE任何工作状态下, 快速地实现一对一、多对多的在线实时谈话, 可以通过谈话窗口发送各种Jazz对象链接(如: 变更集、工作项等), 在实际工作环境中实现有上下文关系(In context)的快速团队沟通与协作。

Jazz平台提供了Feeds服务支持(Atom 1.0和RSS 2.0-compliant), 提高团队协作沟通和响应的速度。当Jazz库中任何对象发生数据变化后(如: 源代码发生变更、工作项状态发生变化等等), Jazz都会通过Feeds服务主动在团队范围内或根据订阅记录进行广播, 让给所有相关开发人员能够在最短的时间内掌握最新动态, 开发人员也可以查阅每天团队、Jazz对象具体发生了什么事情。

Jazz平台还支持邮件通知功能(Email Notification)。例如: 当你关注某项工作的任何进展和变更时, 你可以订阅(subscribe)该项工作, Jazz平台就会自动地给你发送相关提醒邮件, 让你及时掌握工作最新状态。

能力	功能	收益
理解团队	团队成员可以看到其它哪些项目成员已经登录平台, 他们正在做哪些工作; 当有影响自己工作的变更发生时, 平台自动通知他们。	使得团队可以在具体的开发环境中(特别是在全球的、组织分布的跨地域开发环境)直接地进行协作开发。
协作特性	提供了与项目集成的在线(presence)、即时通信(messaging)、Feeds服务和邮件通知等协作功能。	提升开发团队的内聚力和快速响应, 特别是对跨地域的开发团队, 使他们能够更加容易地去共享项目文档, 实时地协作, 以及获得一些热点问题的快速回答。

2.1.2.2.1.2 天生能够理解和支持各类开发过程, 使国防单位对开发可以做到适度的管控(Right-Size Governance)

Jazz被设计用来理解并支持各种类型的开发过程, 包括从小型的敏捷(Agile)风格的项目, 到大型的并带有复杂遵从需要的企业级项目。这些开发过程都是可定制、可配置的, 能够适应不同企业环境, 而Jazz平台对开发过程支持是中性的(process neutral), 并没有绑定某种开发过程。企业每个项目都有自己的开发过程, 为项目工作的众多开发团队也可以有各自不同的开发过程, Jazz平台允许企业实施这些管控理念, 满足不同项目、不同团队的不同需要。

Jazz有强大、灵活的过程定制(Process Customization)能力。Jazz的开发过程是基于一系列过程规则(rules-based), 开发团队可以定制自己的全新开发流程, 也可以对已有开发流程进行必要的修改。

Jazz的开发过程不是要去控制开发团队, 相反, 它是响应式的, 当需要的时候它就会发生作用, 帮助减少开发人员发生错误。Jazz的开发流程向导(Team Advisor)就是在开发人员某个操作违反开发过程要求时触发, 中止正在执行的操作, 并解释错误的原因和提出相应解决办法。

能力	功能	收益
理解和自动化开发过程	通过自定义的过程来引导开发团队的工作流程; 自动化实现流程步骤和不同严格程度的检查点。	自动化个人和开发团队的工作流程; 提升开发团队的开发透明度。
弹性的、基于规则的过程定义	可以为不同的开发团队、不同项目、不同项目阶段指定不同的过程规则; 随时调整你的管控规则, 避免需要预先定义所有东西。	帮助企业适度地管控开发项目、开发团队和不同开发周期的需要; 可以在前期指定较少限制和在后期设置较紧的规则, 来鼓励前期的创新和降低开发后期面临的风险。

2.1.2.2.1.3 基于开放标准的技术平台让国防单位能够更加灵活地组装自己的软件交付平台, 降低投资成本

Jazz技术构建在国际开放的Web和OSGi标准之上, 并充分利用了开放的Eclipse技术, 因此Jazz本身就是一个开放的技术平台。它的开放性、可扩展的架构将导致许许多多基于Jazz的新产品诞生, 让你可以选择自己喜爱的解决方案来组装自己独特的软件交付平台, 并拥有着足够的灵活性。

Jazz平台的客户端、服务器和Web UI都是基于开放标准Eclipse的插件式扩展架构技术, 因此, 我们的客户、合作伙伴或其它企业都可以通过增加插件(Plugin)的方式来对Jazz平台进行功能扩展。

Jazz平台还支持现有广泛的开源和商业软件用户群。Jazz有一个开放的中间件接口, 这意味着你能够使用开源中间件, 例如Tomcat™, Derby, and Jabber®, 商业中间件, 例如IBM WebSphere®, DB2®, 或者是第三方商业产品如Oracle?等来安装和配置各种基于Jazz的产品。

能力	功能	收益
可扩展的基础平台设施(Extensible infrastructure)	Jazz平台的客户端和服务端都是基于开放的Web标准, 都可以通过额外的插件进行扩展。	把开发团队协作的好处扩展到更加广泛的业务和技术用户, 使用他们可以使用不同的客户端共享项目文档; 使他们可以在自己的开发交付平台上增加商业和开源的产品来提升交付平台的弹性。
连接器框架(Connector Framework)	通过连接器框架可以集成其它配置管理(Configuration Management)和变更管理(Change Management)系统, 例如: ClearCase, ClearQuest, Subversion等;	使你的软件交付平台能够快速转移到Jazz平台, 来提升在组装你的理想软件交付平台上的灵活性。
开放的中间件接口(Open middleware interface)	基于Jazz的产品能够运行在各种平台, 包括开源的中间件产品: Tomcat™, Derby, Jabber®等, 以及商业中间件产品, 如: WebSphere®, DB2®, Lotus and Oracle®等。	使你能够在现在商业平台或开源平台的基础上, 快速地启动和运行Jazz平台, 帮助降低管理成本和总的投资成本。

2.1.2.2.1.4 Jazz影响着IBM Rational软件交付平台的现在与未来

Jazz是一种技术创新, 它通过从桌面功能部件的集成(Desktop Integration)到服务端完整生命周期集成(Lifecycle Integration)的设计思想转变作为起点, 将把Rational软件交付平台的能力提高到一个前所未有的水平。

我们都知道, Eclipse平台是Rational桌面版工具(如: Rational Software Architect等)的基础平台; Jazz平台则是未来Rational团队版产品(如: RequisitePro, ClearQuest, ClearQuest等)甚至是整个SDP的基础平台。

我们设计了两条路径让不同类型客户能够充分利用Jazz平台带来的好处: 一条针对Rational客户的进化路径; 另一条是针对敏捷团队的革命性路径。最终, 这两条路径会走向融合, 产生下一代的IBM Rational SDP。

对于我们的Rational老客户来说已经是走在进化路径上了。Rational很多产品例如ClearCase, ClearQuest等都会逐步转到Jazz平台, 通过Jazz平台与广泛的第三方产品进行互连互通, 从而带来不可估量的好处。同时, 我们也会把更多Jazz技术的新功能、新特性加入到Rational现有产品的新版本中。

喜欢敏捷开发方法的团队可以直接使用Jazz平台和它上面的RTC产品, 这是一个为敏捷开发团队量身定做的、易学易用的团队协作开发产品。

总体来说, Jazz技术是Rational现有产品如ClearCase, ClearQuest等的重要补充, 对于需要更加先进SCM能力的企业和团队还说, 他们可能会选择使用ClearCase和ClearQuest。通过RTC提供的互通能力, 在一个企业中可以同时使用RTC和ClearCase和ClearQuest, 甚至是Subversion。

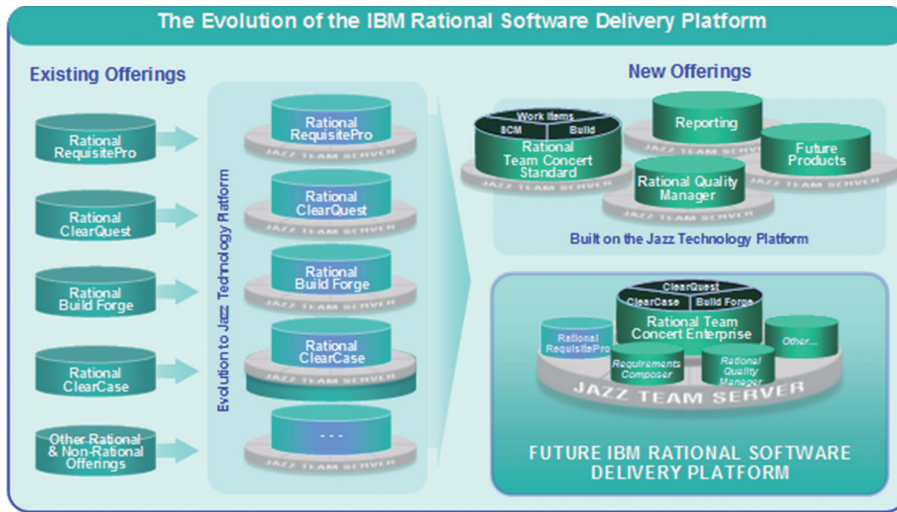


图1 IBM Rational软件交付平台的未来

2.1.2.2.1.5 提供不同版本, 满足国防单位从科室到所乃至院所一级不同团队规模需要

目前RTC 1.0提供了三个版本: Express-C(Community), Express和Standard三个版本(Editions), 分别满足不同规模(支持最大用户数分别是: 10人, 50人和250人)开发团队需要。这三个版本的主体功能都是相同的, 功能差异在于Standard版本多了一些定制、报表和集成功能等, 如: 工作项定制、ClearCase/ClearQuest连接器、报表、仪表盘、对LDAP的支持等。将来IBM Rational还会发布企业版(Enterprise Edition), 支持更大规模的开发团队。

这些版本可以运行在Windows和Linux平台, 支持开源的Tomcat, Derby或其它商业软件, 如: IBM WAS, DB2, Oracle等。具有描述请参考后面相关内容。

Features	Express-C	Express	Standard
Wrong operating system? Select your OS: <input type="button" value="Windows"/>	<p>DOWNLOAD Express-C</p> <p>EARLY ACCESS TRIAL Get set up quickly with the all-in-one Express-C client/server download. Just unzip the package and run!</p>	<p>DOWNLOAD Express</p> <p>EARLY ACCESS TRIAL Ease into Express using IBM Installation Manager to install client, server, and optional components. Makes future updates easy to manage! Other Download Options...</p>	<p>DOWNLOAD Standard</p> <p>EARLY ACCESS TRIAL Get started with Standard using IBM Installation Manager to install client, server, and optional components. Makes future updates easy to manage! Other Download Options...</p>
License	Early Access Trial	Early Access Trial	Early Access Trial
Max User Limit	10	50	250
Database included (optional)	Derby only	Derby and DB2 Express (DB2, Oracle)	Derby and DB2 Express (DB2, Oracle)
App Server included (optional)	Tomcat only	Tomcat (WebSphere)	Tomcat (WebSphere)
Source Code Management	✓	✓	✓
Work Item Tracking	✓	✓	✓
Build Management	✓	✓	✓
Agile Planning	✓	✓	✓
Subversion Integration	✓	✓	✓
Server Level Permissions	✓	✓	✓
LDAP Authentication	✓	✓	✓
Customizable Process	✓	✓	✓
Customizable Work Item Attributes and Workflow			✓
Reports			✓
Dashboard			✓
Role-based Process Permissions			✓
ClearCase Connector			✓
ClearQuest Connector			✓
LDAP Import			✓

图2 Rational Team Concert的3个版本功能对照

2.1.2.2.1.6 采用工作项集中管理各种开发活动, 并提供灵活而强大的定制能力

开发过程中会产生各种类型的工作, 例如: 任务、缺陷、需求、变更等, RTC通过集成的工作项管理(Work Item Management)对这些工作进行统一的跟踪和管理。在RTC中工作项的作用类似Hub, 它可以把RTC中的各种工件(如: 其它工作项、变更集、构建结果等)有机地关联在一起, 建立强大的追踪关系(Traceability)。另外, 工作项还是制定项目迭代计划的基础。

工作项管理提供了灵活而强大的工作项定制能力。工作项有不同的工作项类型(Work Item Type), 如: 任务(Task)、缺陷(Defect)等, 用户可以根据需要自由增加和定制新的工作项类型。每一种工作项类型都有自定义的字段、状态转换矩阵(State Transition Matrix), 我们还可以定制查询、编辑界面, 以及工作项在不同团队、不同迭代中的权限与行为等。

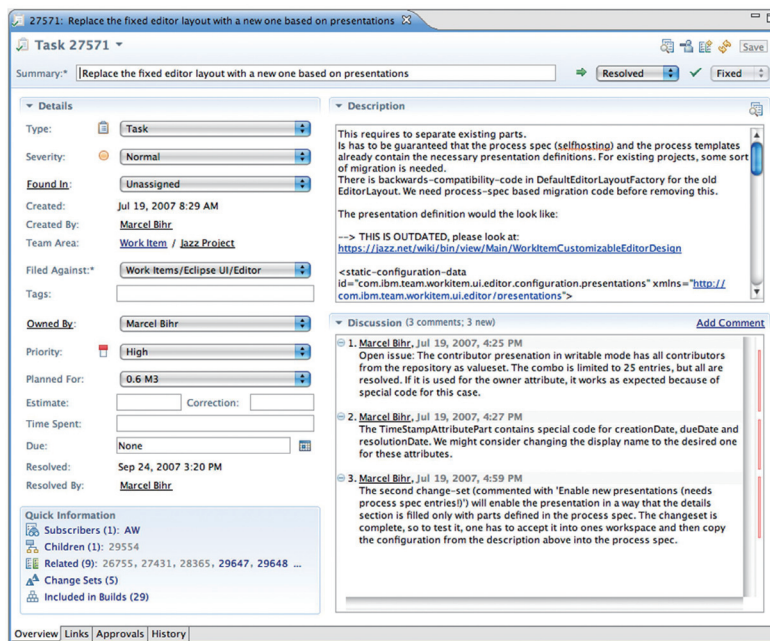


图3 编辑工作项的用户界面

2.1.2.2.2 基于业务决策的型号和技改项目组合管理

IBM Rational Focal Point是目前市场上用于进行项目组合管理和产品管理的最全面的决策制定解决方案。作为唯一完全基于Web的决策支持系统, Focal Point可帮助组织根据商业目标、客户需求、成本和可用资源来准确判断、管理和监视最佳产品或项目组合。这种决策制定软件将独特的“假设”方案分析功能与可视化技术结合在一起, 加强了利益相关者协作、优先级排序、决策制定、竞争对手情报、方案规划、信息可视化、产品组合优化、客户简档创建和其它商务智能活动, 从而改进了型号项目和技改项目管理以及两者的项目组合管理。

作为总师或决策者者, 您可能会将产品要求记录在电子数据表或文本文件中, 将更多信息记录在损失报告和财政报告中, 针对院所及军方决策和消息信息由于缺乏一个中央位置来集中保存这些独立的信息, 因此, 您很难收集并分析它们。如果使用单一数据存储库, 您将能够与客户和相关利益群体共享决策信息, 并将硬件产品或软件版本的提供情况及时通报给他们。

Focal Point解决方案支持有关产品人员使用Web界面或电子邮件来提交产品信息。用户可将电子邮件自动路由到Focal Point, 由系统生成要求信息以及介绍产品、军代表的链接。军代表可通过Web圈定军方所需的产品范围并描述产品价值 — 以便您进一步了解产品特性的市场价值。

通过使用Focal Point, 您可将更多时间用在市场和竞争对手分析等战略活动上, 从而为产品取得长期成功奠定基础。

2.1.2.2.2.1 基于价值做决策

在产品规划流程中, 决定目标市场对产品特性是否买账是最为棘手的一个环节。如果价值评估工作不充分, 那么, 您所提供的特性和产品主观性很强, 不是基于真实情况作出的明智决策。Focal Point提供竞争分析和盈/亏报告, 您可以更好地了解竞争对手在相同市场领域的战略及其优缺点。此外, 准确定义军品面向的市场领域和用户群, 将帮助您进一步了解该领域的客户和用户需求。

Focal Point不仅允许您集中管理这些信息, 而且还能帮助您将这些不同来源的信息关联在一起, 以便您基于军方及民用市场需求做出特性决策, 不受产品指导会议上不同阵营意见的左右。

2.1.2.2.2.2 构建能够实现的产品路线图

您在构建产品路线图时, 应该想到未来形势可能会发生变化, 需要您更新路线图。但是, 您却很难修改路线图、推出新特性或新版本、牢记常见的局限性, 如资源和预算的可用性及全新版本或特性的范围及其可能创造的目标市场价值等。Focal Point提供交互式的线路图直观显示工具, 允许您通过放大路线图来观看产品细节、查看资源和财务需求曲线图, 并且通过拖放产品版本来观看特定区域内的产品信息。

交互式产品路线图提供影响评估支持, 允许边使用边修改, 可帮助您提高市场灵活性, 获得竞争优势。由于路线图的成功率极高, 因此, 您有理由相信自己能够在适当时间推出适当产品, 从而放开手脚去推广产品, 并且进军全新市场领域。

2.1.2.2.2.3 工作原理

Focal Point提供强大的集成功能, 能够帮助总师及决策人捕获信息, 并且与国防单位内部的其他信息源共享信息, 如CRM、企业资源规划(ERP)及财务和缺陷跟踪系统。

Focal Point提供Web服务应用编程接口(API), 能够轻松集成其他的数据源, 以便馈送数据或提取数据。

您可通过Web接入Focal Point基于角色的功能, 以便提供多用户支持。

Focal Point基于最佳业务实践在Web应用服务器上安装了预配置的视图、报告和表单, 随时可供使用。

基于Web的管理和配置工具可帮助客户基于现有流程定制Focal Point。配置工具易于使用, 配置变化可由一名主要用户进行集中管理, 从而最大限度地降低配置变化对IT人员的影响, 并且简化新型信息的捕获工作。

2.2 支持类过程解决方案

2.2.1 配置管理

2.2.1.1 实施软件配置及变更管理能够解决国防单位目前存在的那些问题

国防单位通过使用ClearCase和ClearQuest搭建的配置及统一变更管理平台, 实施有效的软件配置及变更管理后可以解决循序渐进地解决国防单位目前软件开发中的存在问题:

如何遵循GJB5000成熟度第二级的SCM关键过程域进行有实效的改进同时又能遵循《军用软件配置管理GJB1268A-2004》的最佳实践;

目前国防单位配置管理制度难于实施的问题;

- 仅有软件配置管理没有变更管理, 造成无法有效记录并跟踪变更请求;
- 建立国防单位配置和变更审计流程;
- 开发人员未经授权修改代码或文档;

- 军工企业的软件核心技术杜绝泄密的问题;
- 找不到某个文件的历史版本;
- 无法重现历史版本;
- 无法重新编译某个历史版本,使维护工作十分困难;
- “合版本”时,开发冻结,造成进度延误;
- 软件系统复杂,编译速度慢,造成进度延误;
- 因一些特性无法按期完成而影响整个项目的进度或导致整个项目失败;
- 已修复的Bug在新版本中出现;

针对以上存在的问题,国防单位如何结合GJB5000中的SCM关键过程域如何切实有效的进行改进?我们认为IBM Rational配置及统一变更管理平台可以结合国防单位自身具体实际助力我们符合GJB5000成熟度等级2中SCM关键过程域的要求。而且在更深层次上解决软件配置及变更管理中的问题。

2.2.1.2 IBM Rational配置及统一变更管理平台助力国防单位GJB5000 SCM KPA的实现

关键过程域	目的	目标	共同特征	关键实践	IBM SDP Solution		
					RUP	ClearQuest	ClearCase
SCM	建立和维护在项目的整个软件生存周期中软件项目产品的完整性。	目标1 软件配置管理活动是有计划的。	Co. 1 Ab. 2 Ab. 3 Ab. 4 Me. 1 Ve. 2 Ve. 4	<p>活动1 按照文件化的规程对每个软件项目准备一份SCM计划。</p> <p>活动2 用文件化的经批准的SCM计划作为进行SCM活动的基础。</p>	SCM流程支持和方法支撑	参见章节 2.7 3	参见章节 2.1 2.7 3

关键过程域	目的	目标	共同特征	关键实践	IBM SDP Solution			
					RUP	RPM	ClearQuest	ClearCase
SCM	建立和维护在项目的整个软件生存周期中软件项目产品的完整性。	目标2 所选定的软件工作产品是已标识的、受控的和适用的。	Co. 1 Ab. 1 Ab. 2 Ab. 3 Ab. 4 Ab. 5 Me. 1 Ve. 4	活动2 用文件化的经批准的SCM计划作为进行SCM活动的基础。 活动3 建立配置管理系统作为软件基线用的仓库。 活动4 标识将置于配置管理之下的软件工作产品。 活动7 按照文件化的规程根据软件基线库生成产品并控制其的发放。	RUP	项目管理中配置管理及相关活动的管理	参见章节3	参见章节2.1 2.2 2.5 2.6 2.8 3
					SCM流程支持和方法支撑			

关键过程域	目的	目标	共同特征	关键实践	IBM SDP Solution		
					RUP	ClearQuest	ClearCase
SCM	建立和维护在项目的整个软件生存周期中软件项目产品的完整性。	目标3 对已标识的软件工作产品的更改是受控的。	Co. 1 Ab. 1 Ab. 2 Ab. 3 Ab. 4 Ab. 5 Me. 1 Ve. 4	活动5 按照文件化的规程，起动、记录、审查、批准和跟踪关于所有配置项或单元的更改请求和问题报告。 活动6 按照文件化的规程控制对基线的更改。	RUP	参见章节2.3 2.4 2.7 3	参见章节2.2 2.8 3
					SCM流程支持和方法支撑		

关键过程域	目的	目标	共同特征	关键实践	IBM SDP Solution		
					RUP	ClearQuest	ClearCase
SCM	建立和维护在项目的整个软件生存周期中软件项目产品的完整性。	目标4 受影响的组和个人得到软件基线的状态和内容的通知。	Co. 1 Ab. 2 Ab. 3 Ab. 4 Ab. 5 Me. 1 Ve. 1 Ve. 2 Ve. 3 Ve. 4	活动8 按照文件化的规程记录配置项或单元的状态。 活动9 编写SCM活动和软件基线内容的标准报告，并使其可供受影响的组和个人使用。 活动10 按文件化的规程进行软件基线审核。	SCM流程支持和方法支撑	参见章节 2.2 2.4 3	参见章节 2.1 2.2 2.8 3

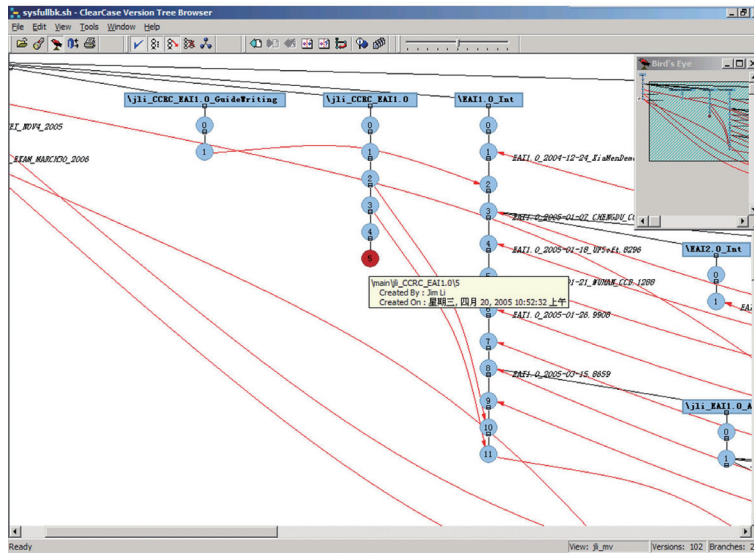
2.2.1.2.1 全面版本控制, 确保软件资产的安全性和完整性, 避免企业资产的流失和意外破坏

包括源代码、模型、文档以及运行文件的软件资产是软件开发企业最重要的资产, 如同银行电信企业的数据库同等重要, IBM Rational配置管理解决方案可以

- 提供对所有文件系统对象的版本控制, 包括源代码文件、分析设计模型、二进制代码和测试脚本等。
- 自动跟踪目录名更改、移动和删除。
- 独特的触发器、锁机制以及版本库/目录/文件分级安全管理保证资产的安全性。

2.2.1.2.2 完整地记录文件以及系统的变动历史记录, 直观进行文件及系统两个层面的历史回溯和版本比较

ClearCase可以对源代码、文档、二进制文件等进行文件级版本控制, 通过文件/目录版本树浏览界面, 可以直接以树形结构描述一个文件/目录的版本成长历史、每个版本的跟踪审计信息等等。另外, 为了更为直观地从系统或构件层面进行历史发布的跟踪, ClearCase还可以将文件/目录等组织为构件(component)进行管理。通过构件基线浏览器, 可以全面看到某个系统/子系统或者构件所有基线的情况, 包括基线标识、建立时间、创建人、基线所包含的开发活动、基线成熟级别。同时在文件和构件浏览器中可以方便地进行文件版本或系统发布的比较。



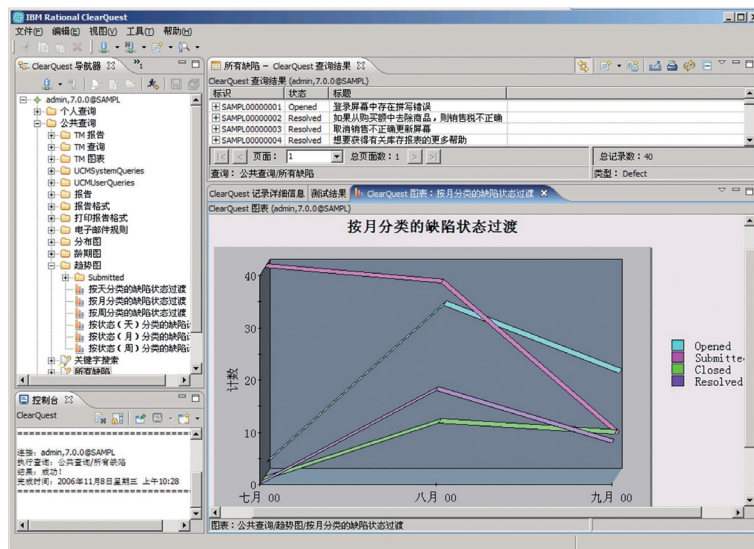
2.2.1.2.3 一致高效的工作环境管理, 简化开发人员工作

ClearCase和ClearQuest提供了IDE集成环境、命令行、Windows客户端、Eclipse客户端、Web客户端等多种界面, 方便了使用人员的日常使用。另外

- 利用ClearCase动态视图可以对文件版本进行透明访问, 即在不下载文件的情况下直接从客户端透明访问服务器资源。
- 支持灵活的ClearCase快照视图以进行迅速简便的远程和脱机访问。
- 生成人员任务列表以安排每个人的工作量。

2.2.1.2.4 加强开发团队内外的沟通, 及时掌控项目状态, 全面提高开发团队的工作效率

通过统一的变更管理平台增强项目开发过程中各部门之间, 各部门与集成商之间有效的沟通, 实现流程的自动化操作, 而且在多个流程间的信息可以进行交互。全面记录完整的管理信息, 使各级领导、业务人员和项目管理者及时、自动地了解项目管理状态, 量化内部项目人员及供应商项目组成员工作量, 工作进度。



2.2.1.2.5 发布版本的构建管理, 建立生产运行代码和源代码的有机联系

- 基于现有构建工具进行版本化审计。建立可执行代码与源代码版本之间的跟踪关系。
- 支持主机上的远程构建能力, 从而跨平台地进行主机LOAD模块的构建审计。
- 为每个可执行构建对象创建详细的材料清单, 以保证构建的可重现性并支持可审计性。
- 构建对象可以在开发人员之间进行共享, 避免重复构建, 提高构建速度。

2.2.1.2.6 引入构件提高软件的灵活性和可重用性, 简化大型软件或多方外包方式下的配置管理

- 通过构件(component)提升配置项的管理层次, 更易于与分析设计中的构件或子系统进行对应。
- 构件可以供多个项目进行共享, 安全性控制可以在构件层面实现, 便于多方参与下清晰的分工合作。
- 构件基线以及复合基线(包含基线的高层基线)的引入为大型项目的开发组织提供了灵活性和便利性。
- 构件基线浏览器可以完整浏览构件的变更历史, 方便进行基线的比较和基线所包含的开发活动。轻松比较两次发布版本之间在功能上的差异以及在文件和版本上的差异。

2.2.1.2.7 按活动进行工作进度跟踪, 通过灵活流程定制管理活动(如需求、缺陷、变更等)的生命周期

- 利用ClearQuest根据要求定制活动(包括需求、缺陷、变更等)的处理流程, 包括流程处理状态、流程涉及的数据以及流程涉及的表单布局及设计等。
- 通过项目管理、历史记录、附件、审计跟踪、电子签名、Email通知等几十个预置模型包快速定义用户自己的管理流程。
- 利用ClearCase和ClearQuest的集成活动会自动传入开发人员工作环境。开发人员以分配给自己的活动为依据进行代码修改, 所做修改会自动关联到相应活动。
- 开发活动通过提交自动嵌入所发布的基线中, 从而随时掌控基线或发布版本中所包含的功能。

2.2.1.2.8 基于相同代码轻松进行多个发布版本的并行开发和维护, 同一修改在多个发布版本上只改一次

- 利用ClearCase丰富的构件及项目机制, 可以在同一系统或构件上可以同时展开多个开发项目, 如基于1.0发布同时开始针对1.0缺陷的维护版本1.1和基于1.0的新需求和变更的2.0开发等。
- 基于前一发布版本快速展开后续版本的开发, 自然保持关联发布版本之间的继承关系。
- 同一种修改只需要做一次。例如在1.1版本中修改的Bug可以被自动地施加到2.0版本中, 而不需要手工来重复这种修复操作。
- 利用构件和强大的并行开发功能可以支持多种开发组织形式。如小规模团队的开发、基于软件特性的大型开发、基于发布版本的大型开发等等。
- 轻松比较多个发行版本之间在功能和源代码上的差异

2.2.2 软件质量保证

2.2.2.1 实施软件质量保证能够解决国防单位目前存在的哪些问题

在整个GJB5000-2003中, 软件质量保证是和同行评审一样贯穿所有级别的关键过程领域, 其目的是对过程进行相应的产品审计和活动评审, 使管理者对软件项目或组织级活动正使用的过程和正构造的产品有适当的了解。国防单位也正需要这样的平台对所里所有项目输出的一系列工作进行产品的进行检查, 评估, 以审计其与已经制定的规格、标准、或其他规范的符合性, 从而解决国防单位目前存在以下普遍问题:

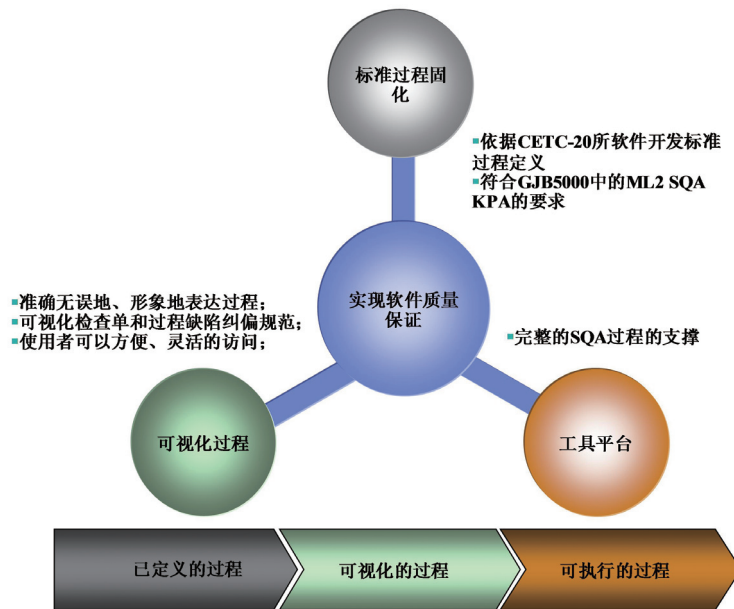
- 开发团队对已发布的管理和开发流程的遵从和符合程度较低, 造成开发过程随意性大, 过程难于控制;
- 流程执行力较低;
- 流程和实际能力不对等, 即我们常说的“两层皮”。

以上问题集中起来反映了国防单位在流程改进中需要加强的两方面:

- 过程的执行力如何得到保障?
- 如何通过过程纠偏保证软件质量?

IBM Rational的解决方案是通过如下图所示的三个环节有机衔接的,并且达到:

- 固化过程,使过程成为强制标准
- 可视化过程,使过程的执行周期透明化,
- 可执行过程,通过工具平台使过程实际可操作,杜绝有流程不执行的尴尬境地.
- 灵活定义、发布和管理过程,使国防单位通过迅速制定合适的过程满足其业务发展需求,并且能够持续改进并维护过程;



2.2.2.2 IBM Rational 软件质量保证平台助力国防单位GJB5000 SQA KPA的实现

软件质量保证的目的是向管理者提供适当的对软件项目正使用的过程和正构造产品的可视性。

关键过程域	目的	目标	共同特征	关键实践	IBM SDP Solution			
					RUP/RMC	RPM	ClearQuest	ClearCase
SQA	使管理者对软件项目正使用的过程和正构造的产品有适当的了解	目标 1 软件质量保证活动是有计划的。	Co. 1 Ab. 1 Ab. 2 Ab. 3 Me. 1 Ve. 2 Ve. 3	活动1 按照文件化的规程为软件项目制订SQA计划。 活动2 按照SQA计划进行SQA组的活动。	SQA流程支持和方法支撑。参见章节 2.1 2.4 和 4	结合软件开发计划对SQA计划和管理进行参见章节 2.1和4	管理过程缺陷参见章节4	SQA相关配置项的管理参见全文

关键过程域	目的	目标	共同特征	关键实践	IBM SDP Solution			
					RUP和RMC	RPM	ClearQuest	ClearCase
SQA	使管理者对软件项目正使用的过程和正构造的产品有适当的了解	目标2 软件产品和活动遵守适用的标准、规程和需求的情况得到客观证实。	Co. 1 Ab. 1 Ab. 2 Ab. 3 Ab. 4 Me. 1 Ve. 2 Ve. 3	活动2 按照SQA计划进行SQA组的活动。	SQA流程支持和方法支撑. 参见章节2.1和2.4和4	结合软件开发计划对SQA计划和活动进行管理 参见章节2.1和4	管理过程缺陷 参见章节2.3, 2.4和4	SQA相关配置项的管理 章节2
				活动3 SQA组参与准备和审查项目的软件开发计划、标准和规程。				
				活动4 SQA组审查软件工程活动以证实符合性。				
				活动5 SQA组审核指定的软件工作产品以证实符合性。				

关键过程域	目的	目标	共同特征	关键实践	IBM SDP Solution			
					RUP和RMC	RPM	ClearQuest	ClearCase
SQA	使管理者对软件项目正使用的过程和正构造的产品有适当的了解	目标3 受影响的组和个人接到软件质量保证活动和结果的通知。 目标4 高级管理者处理在软件项目内部不能解决的不符问题。	Co. 1 Ab. 1 Ab. 2 Ab. 3 Ab. 4 Me. 1 Ve. 1 Ve. 2 Ve. 3	活动6 SQA组定期向软件工程组报告其活动的结果。	SQA流程支持和方法支撑. 参见章节4	结合软件开发计划对SQA计划和活动进行管理 参见章节2.1和4	管理过程缺陷 参见章节2.2, 2.3, 2.4和4	SQA相关配置项的管理 参见全文
				活动7 按照文件化的规程把在软件活动和软件工作产品中鉴别出的偏差形成文件并加以处理。 活动8 合适时, SQA组与顾客的SQA人员一起对SQA组的活动和发现进行定期审查。				

2.2.2.2.1 软件质量保证活动是有计划的

质量保证计划用于筹划复审和审计活动, 复审和审计的目的在于检查国防单位开发团队是否完全遵照该计划引用的各种辅助计划中的预定方式执行了项目流程。

SQA人员可以借助RUP中SQA活动的相关流程并结合以下三方面的最佳实践, 有机而且充分地对SQA活动进行策划。

2.2.2.2.2 确保制定了项目的质量目标

SQA人员和项目负责人通过为项目制定质量目标, 确保已制定获得客户认可的质量目标, 且最终记录在软件需求规约说明书中。

2.2.2.2.3 与项目的其他计划内容相协调

软件质量保证和国防单位项目开发的其他计划是密切相关的, 一般还引用了其他一些计划, 主要说明了项目标准与如何处理各种支持流程(如配置管理)。只有这样才有助于确定应采用的质量保证审计的类型及其频率。比如下图所示的SQA报告方法和频率。

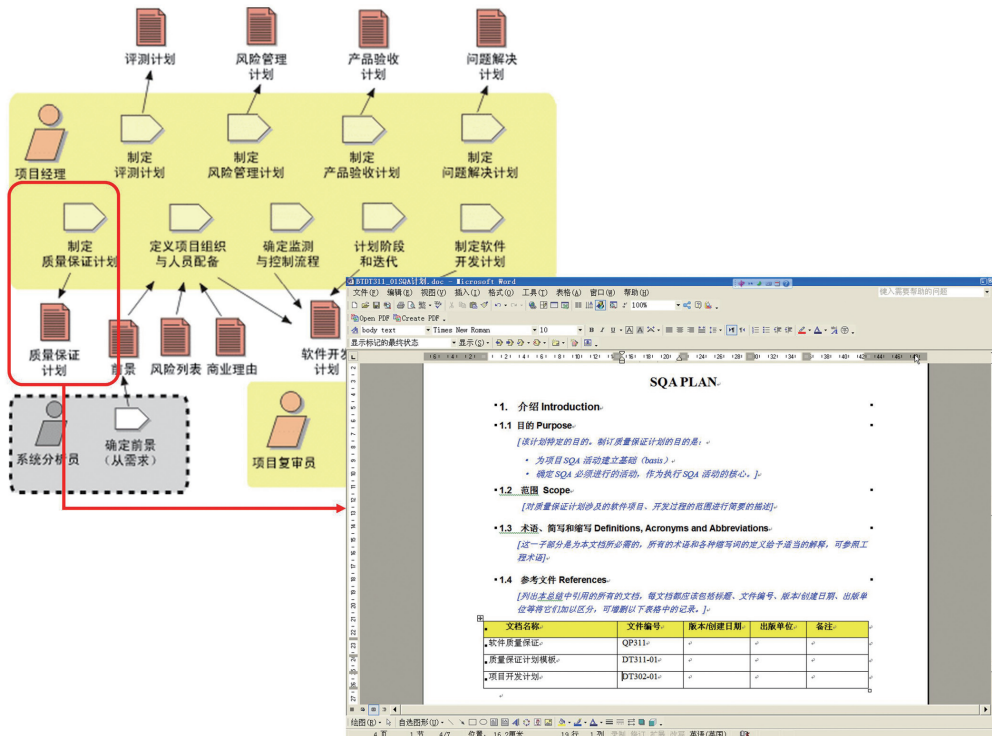
报告类型	报告对象	频率
Checklist	项目负责人、项目组所有成员	每周, 或事件驱动(发现不符合项时)
软件项目状况一览表	高级经理、项目负责人、软件项目负责人	每周
周例会上报告SQA发现(依据checklist)	参加项目周例会的项目组所有成员	每周
SQA跟踪报告	国防单位11室相关人员、项目负责人、项目组成员、SQA主管	每两周
质量问题报告	根据不符合报告规范	事件驱动, 质量问题发生时
项目里程碑报告及总结报告——SQA部分	国防单位11室相关人员、项目负责人、SQA主管、所领导、SEPG	项目阶段末, 项目顾客验收后
客户回访记录	项目负责人、所领导(军代表有投诉时)	每月

2.2.2.2.4 确定质量保证任务与时间表

国防单位SQA人员定义质量保证的任务与活动和时间表。这些活动通常包括:

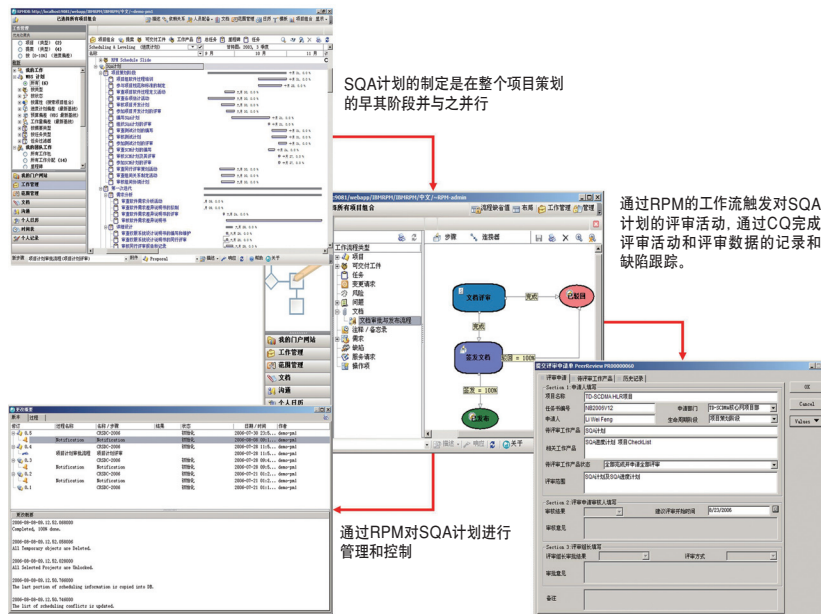
- 审计项目计划, 确保计划符合项目的预定流程。
- 审计项目, 确保执行的工作符合项目计划的规定。
- 审批与标准组织项目流程之间的偏差。
- 流程改进评估。

最后通过对RUP提供的如下图所示的SQA计划模板的裁剪形成项目的SQA计划



然后通过Rational RPM/ClearQuest和ClearCase的整合操作达到如下图所示的:

1. SQA计划的策划和编写是在整个项目策划的早期阶段并与之并行。
2. 通过RPM的工作流触发对SQA计划的评审活动, 用经批准的SQA计划作为进行SQA活动的基础。
3. 通过CQ完成对SQA计划的评审活动和评审数据的记录以及缺陷跟踪。
4. 通过ClearCase对SQA计划进行管理和控制。

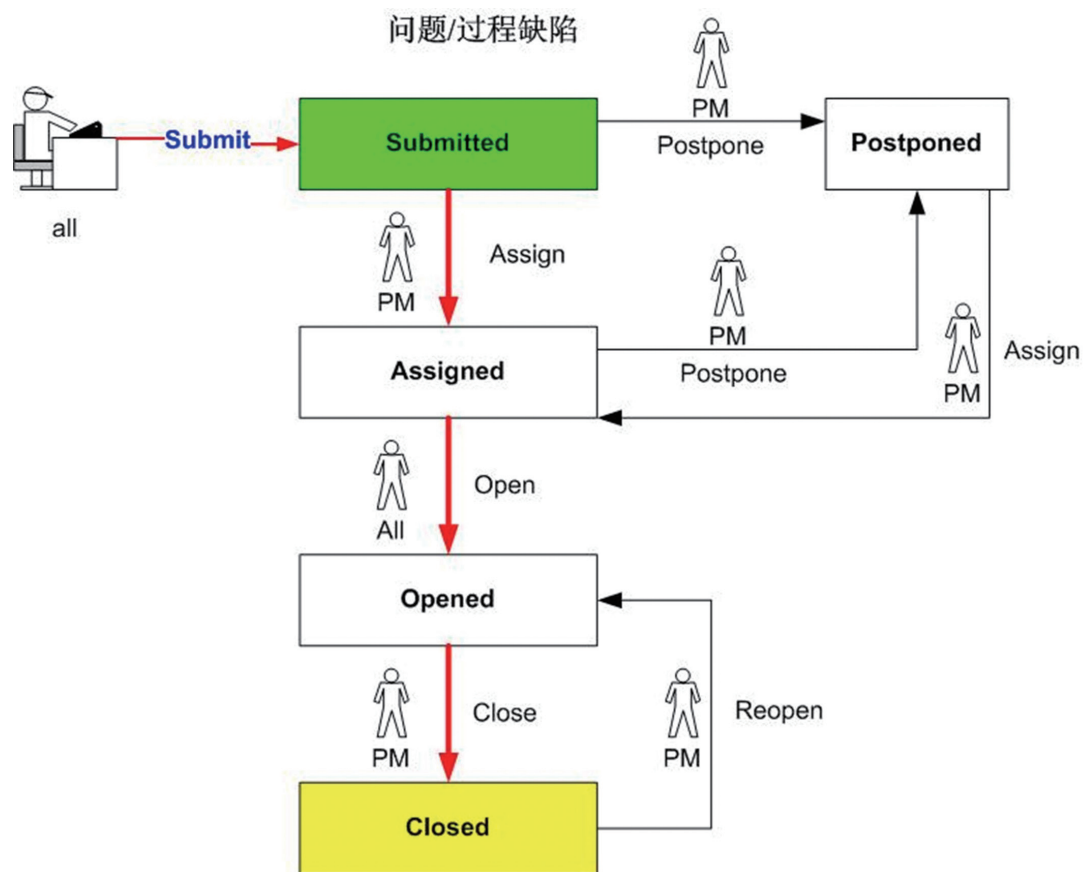


2.2.2.2.5 软件产品和活动遵守国防单位适用的过程的情况得到客观的验证

过程审计活动是一个以SQA为核心,与开发团队的诸多角色有机关联的软件质量保证活动,一个显著特点就是有计划地对开发团队对过程的遵循进行验证并对出现的过程问题进行纠偏,这是一种“鞭策”性的治理,由于SQA活动覆盖了软件开发的整个生命周期,所以,更需要一种平台能够类似于企业质量信息“集线器”,以可视化和自动化的方式贯穿开发周期的始终,使得过程的执行和发现的质量缺陷通过对过程的电子化,以符合戴明环的方式保证国防单位软件质量。

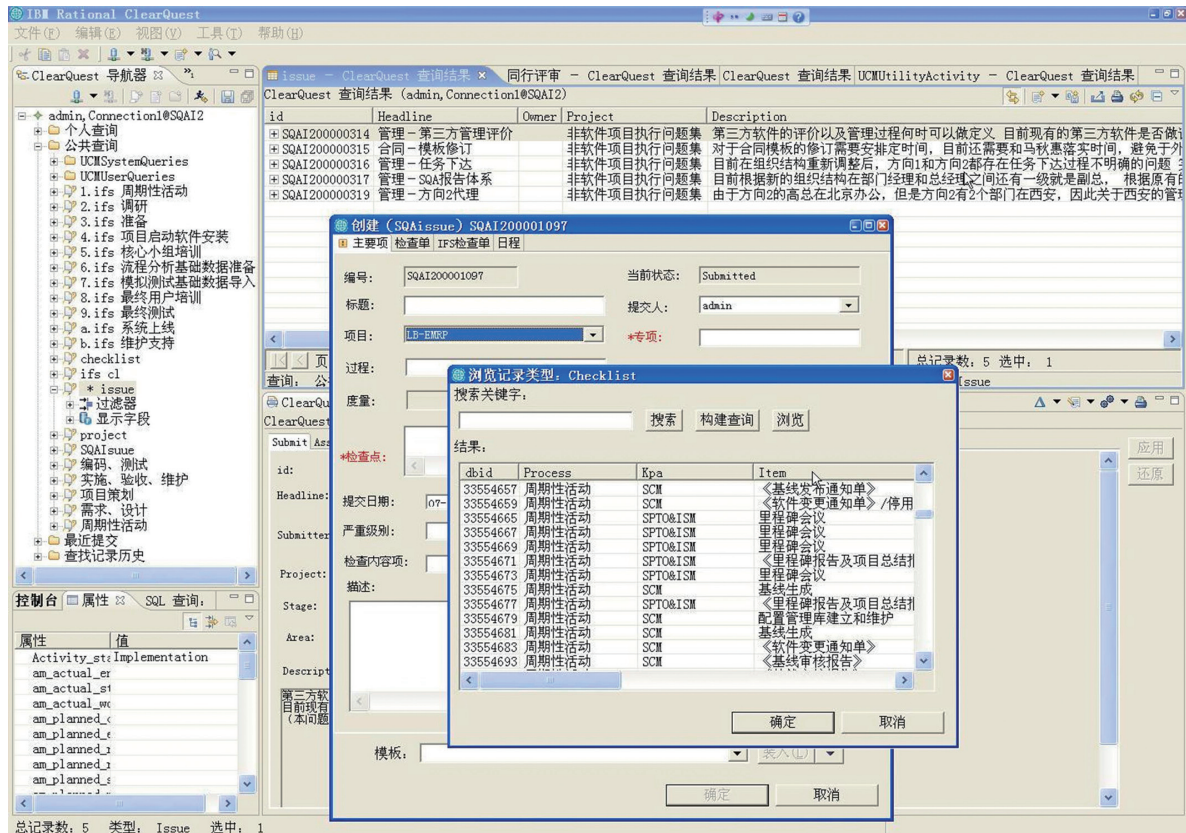
Rational ClearQuest是一个强大而高度灵活的缺陷跟踪系统,能在整个开发周期内捕获、跟踪并管理过程缺陷,帮助国防单位以更高的效率交付出更高质量的软件。

通过ClearQuest可完全自主定制的界面和工作流程引擎我们可以借助类似下图的过程缺陷管理流程帮助我们解决以下问题:



- 项目: 项目例会中提出的问题、项目里程碑会议中提出的问题、项目内部验收会议上提出的问题、顾客投诉反映的问题、项目实施过程中遇到的问题、与其他组(部门)配合问题等。
- SQA: 跟踪审查项目工作产品及活动发现的问题、与项目组的沟通协调问题等。
- SCM: 跟踪项目配置管理活动发现的问题、基线审核发现的问题、与项目组的沟通、协调、配合问题。

借助以上的平台,国防单位的SQA人员依照《SQA计划》进行审计。国防单位的SQA人员确保在审计活动进行前,根据下图所示的已经电子化的项目检查单(CHECKLIST)和SQA计划以及相关的开发标准和规范,实施相应的审计,审计中发现的缺陷记录于检查单(CHECKLIST),不符合项使用ClearQuest工具进行记录和跟踪。对于发现的不符合,按照本章节第三章描述《质量问题报告及处理规范》中相应规定进行处理。在纠正措施完成后进行验证,确保问题得到解决。

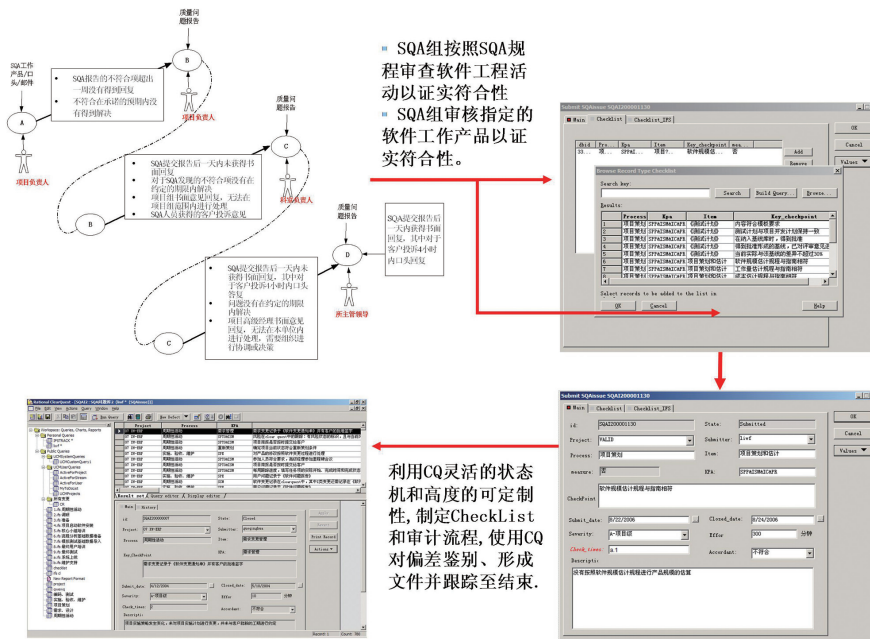


2.2.2.2.6 处理在软件项目中出现的不符合问题

Rational ClearQuest助力国防单位针对审计中发现的不符合项、质量问题和客户投诉按照以下4个级别进行分类后,按照下图中所示的过程,结合ClearQuest的过程定制和数据测量达到为产品满足需要的质量提供足够信任的所有活动。

质量问题的报告及处理可分为A、B、C、D四级进行。其中:

- A级: 对SQA审查和审计活动中发现的不符合项的处理
- B级: 对在A级预期未解决或未获得项目组明确回复的不符合项的处理
- C级: 对在B级未按要求时间获得回复或预期未解决或项目组书面回复无法在项目组范围内解决的不符合项和SQA获得的客户投诉意见的处理
- D级: 对C级未按要求时间获得回复或预期未解决或书面回复无法在软件单位范围内解决的质量问题的处理



2.2.2.2.7 SQA的目的是通过审计进行循序渐进的过程改进

国防单位根据SQA过程中审计的问题,可以不断改善自己的流程.根据我们所的产品开发是兼具硬件开发的系统工程,所以,对于软件开发和管理的改进可以基于IBM Rational系统工程统一开发过程并利用IBM Rational RMC这样的过程改进工具,能够将系统工程过程,以及各类开发和管理角色统一整合在一个无缝连接的过程改进平台上协同工作。

Rational Method Composer(简称RMC)提供一个功能强大的、可视化的、易用的过程定义工具,方便国防单位根据项目的实际需要,定制出自己的结合RUP的软件项目开发和过程管理过程。我们可以借助下表所列的功能和益处搭建一个良性循环的过程改进环境。

功能	RMC	益处
提供软件开发过程平台	<ul style="list-style-type: none"> RMC是一种软件开发方法论框架,一种经过验证的、灵活而实用的成功软件项目的过程平台; 通过可配置的构架, RMC允许您针对每个软件的每个阶段仅仅选择和部署所需要的过程组件; 以业经验证的软件工程最佳经验为核心, RMC平台包括针对项目的特定需要来配置RUP的工具、把自己的内部知识开发到过程组件中的工具、强大而且可定制的基于Web的部署工具, 以及与同行及业界权威交流最佳经验的在线社区; 	<ul style="list-style-type: none"> 协助国防单位快速实现过程改进; 协助国防单位快速引入软件工程最佳经验;
支持业界通用开发及管理过程	RMC包含了以下预定义的通用过程: <ul style="list-style-type: none"> 包含了预定义的小项目过程、大的分布式项目过程、系统工程(System Engineering)、项目群(Program)管理和项目组合(Portfolio)管理过程; 	<ul style="list-style-type: none"> 协助国防单位快速建立所需管理过程; 协助国防单位快速共享业界最佳经验;
过程的执行能力	定义的过程可以直接转变成项目计划模板, 成功经验帮助团队快速定义项目目标、规划项目资源和确定项目里程碑, 提高过程的可执行性;	协助国防单位快速完成过程的推广、执行过程;
业务驱动开发过程的支持	通过提供完成的IT生命周期管理过程, 帮助业务和IT部门的有效合作, 保持IT项目和业务发展目标一致;	协助国防单位快速建立IT生命周期管理过程, 实现业务驱动的软件开发过程, 提高IT的整体投资回报率;

2.2.3 同行评审

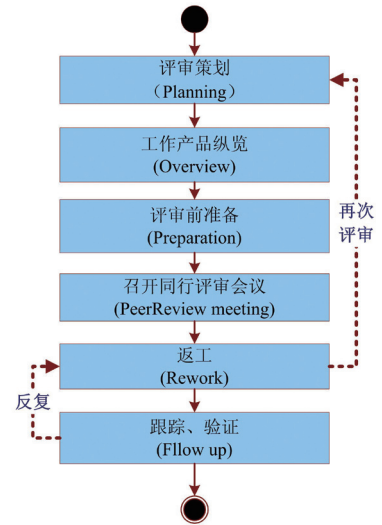
2.2.3.1 实施同行评审能够解决国防单位目前存在的哪些问题

同行评审的目的是为了及早地和高效率地从软件工作产品中消除缺陷。一个重要的伴随结果是对软件工作产品及可防止的缺陷得到更好的了解。

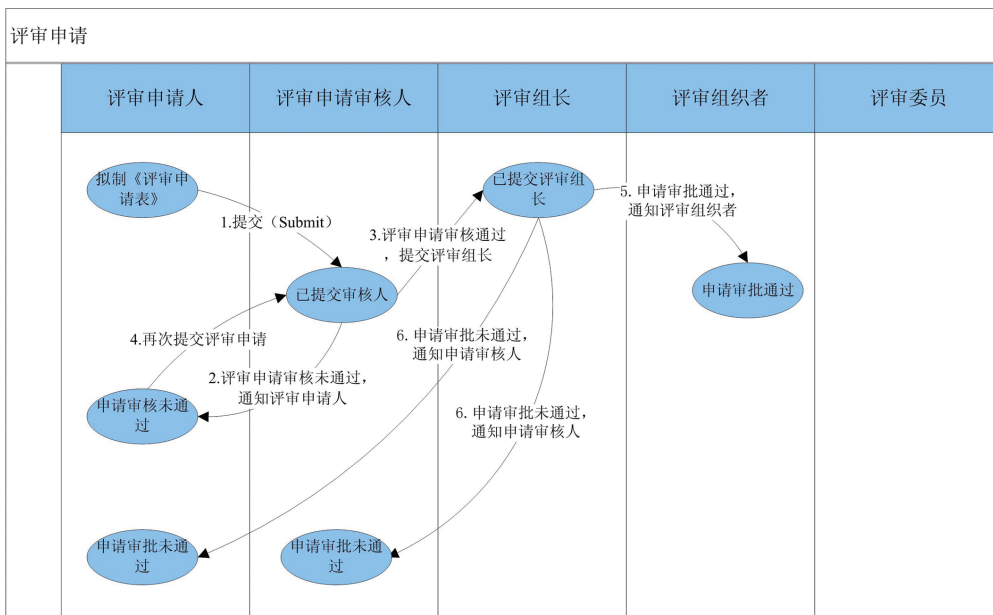
同行评审和SQA是软件工程过程的一种不可忽视的质量保证手段。它是将一个软件工作产品或一组工作产品或一个软件工作产品的一部分提交给作者的同事们以识别出缺陷。它的实质是集中时间发挥集体的智慧。客观及时地揭示隐藏的缺陷。也就是说，软件评审可以用在软件开发过程的各个不同阶段以发现可以排除的错误。但在很多的开发团队中，由于缺少正规和标要的评审方式，导致以下软件开发过程中常见的问题：

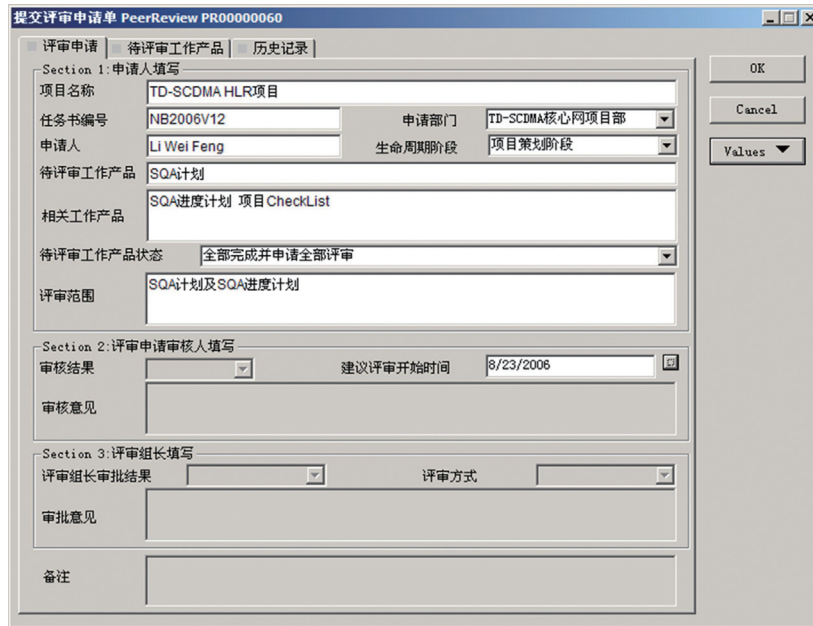
- 产品缺陷，尤其是在开发初期的缺陷缺少甄别手段，对于缺陷的识别多集中在编码后的测试阶段；
- 无法有效确保各个开发阶段输出的工作产品是否符合组织相应的标准和规范；
- 工作产品的完整性和正确性没有保证手段；
- 如何评估工作产品的可理解性和可维护性；
- 如何证实关键的或高风险的构件的质量；

如何解决以上问题，助力国防单位构建一个基于GJB5000成熟度第三级PeerReview(同行评审)关键过程域的评审平台呢？在诸多的同行评审方法中，基本上都会遵循下图所示的评审过程，只是根据评审方法的正规程度，有所裁剪。



我们可以利用ClearQuest完全自主定制的界面和工作过程引擎制定同行评审的处理过程，包括过程处理状态、过程涉及的数据以及过程涉及的表单布局及设计等正是它的擅长。例如，以下图所示国防单位同行评审的评审申请过程为例，ClearQuest能够从过程电子化到过程可视化借助ClearQuest提供的Designer方便地进行定制，并可以根据自己的需要进行界面的修改，满足自己的使用习惯。





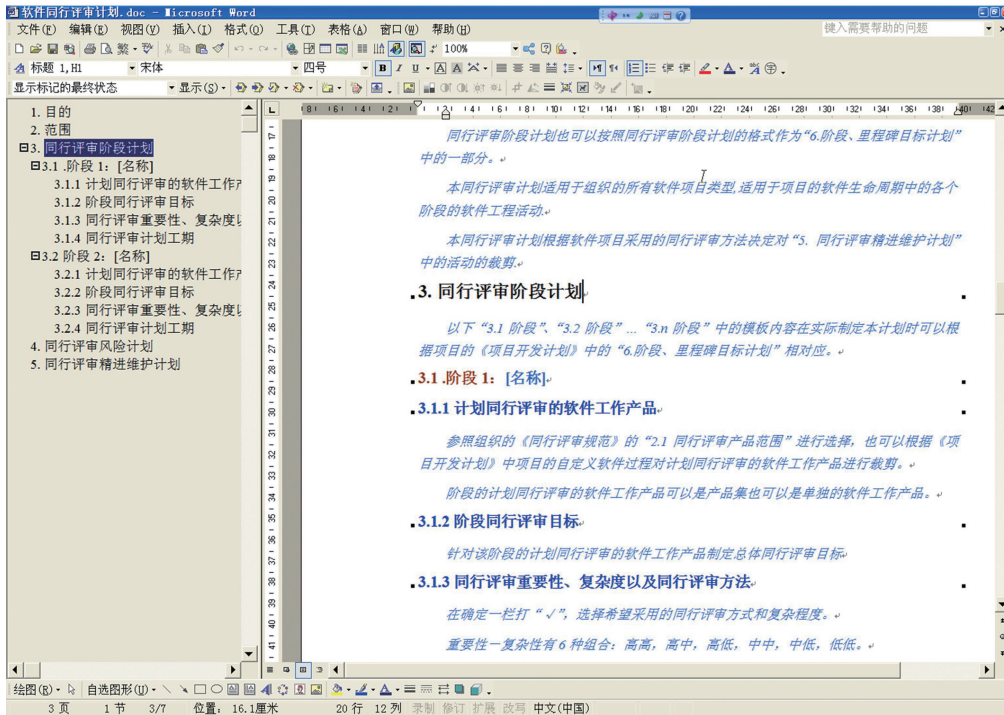
2.2.3.2 IBM Rational 同行评审平台助力国防单位GJB5000 ML3同行评审(Peerreview) KPA的实现

关键过程域	目的	目标	共同特征	关键实践	IBM SDP Solution			
					RUP 和 RMC	RPM	ClearQuest	ClearCase
PR	为了及早地和高效地从软件工作产品中消除缺陷。一个重要的伴随结果是对软件工作产品及可防止的缺陷得到更好的了解。	目标1同行评审活动是有计划的。	Co. 1 Ab. 1 Ab. 2 Ab. 3 Ab. 4 Me. 1 Ve. 1 Ve. 2 Ve. 3	活动1计划同行评审，并将计划写成文档。	SQA流程支持和方法支撑，参见章节 2. 1和3	结合软件开发计划对PR计划和活动进行管理 参见章节 2. 1	管理过程缺陷 参见章节 3	PR相关配置项的管理 参见全文
		目标2识别和消除在软件工作产品中的缺陷。		活动2按照已文档的规程进行同行评审。 活动3记录有关同行评审的执行情况和结果的数据。			管理过程缺陷 参见章节 2和3	

2.2.3.2.1 同行评审活动活动是有计划的

软件开发的每一个阶段都应该以同行评审作为检验标准,所以,同行评审计划是和软件开发计划紧密结合的验证活动,国防单位可以借助RUP裁剪并制定如下图所示的符合本所的评审计划,从评审对应的阶段以及采用的评审方式和对评审缺陷的管理等方面达到:

- 确定将经受同行审查的软件工作产品。
- 规定同行审查的进度。



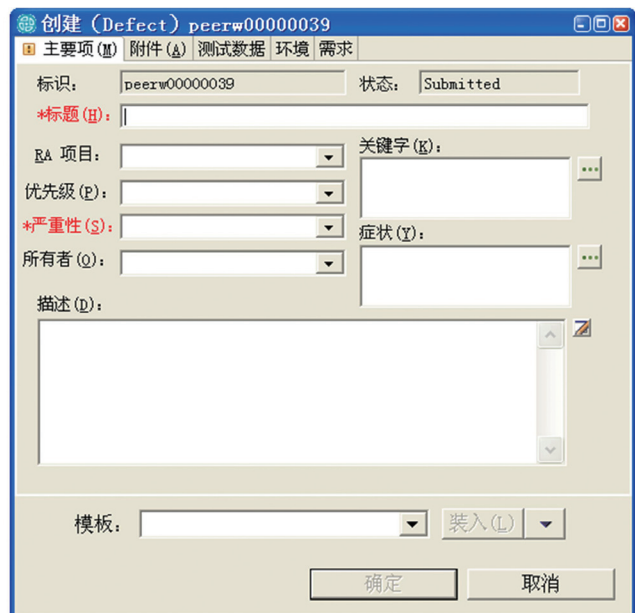
然后通过Rational RPM/ClearQuest和ClearCase的整合操作达到:

1. 同行评审计划的策划和编写是在整个项目策划的早期阶段并为之并行。
2. 通过RPM的工作流触发对同行评审计划的评审活动,用经批准的同行评审计划作为进行同行评审活动的基础。
3. 通过CQ完成对同行评审计划的评审活动和评审数据的记录以及缺陷跟踪。
4. 通过ClearCase对同行评审计划进行管理和控制。

2.2.3.2.2 借助Rational产品识别和消除在软件工作产品中的缺陷

2.2.3.2.2.1 整理、提交缺陷

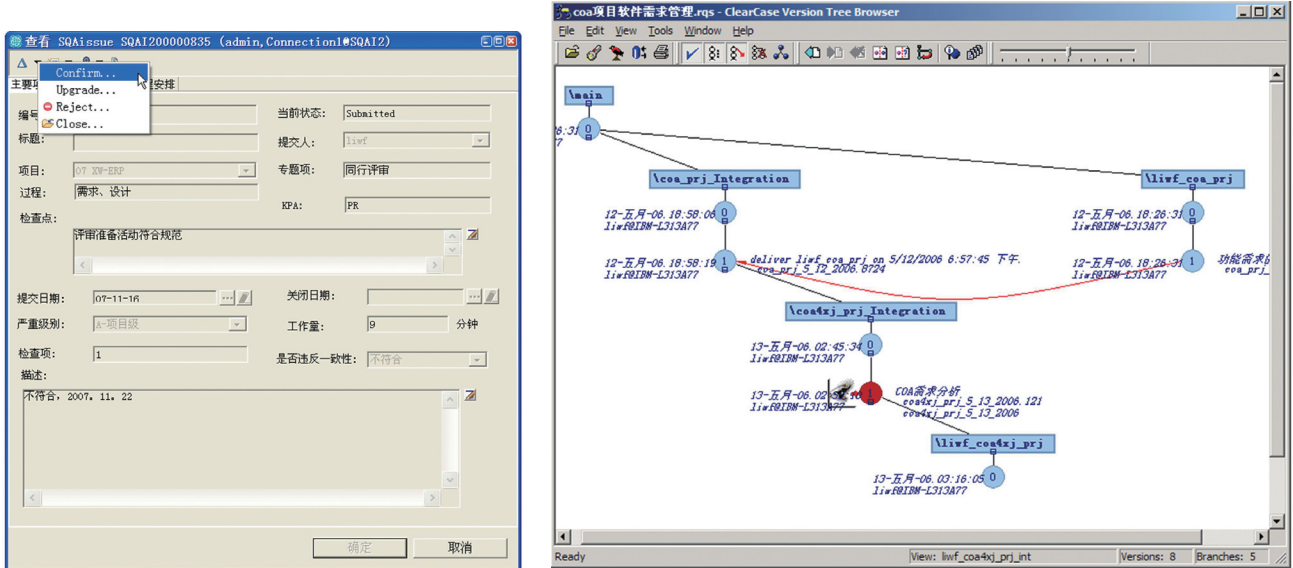
将所有通过评审发现的缺陷进行整理、归并、分级并确认纠正的优先级,并将得到了审查小组的认可的最终的同行评审记录进行提交,同时将缺陷数据登录到缺陷管理工具中进行跟踪管理。



2.2.3.2.2.2 跟踪、验证和缺陷统计分析

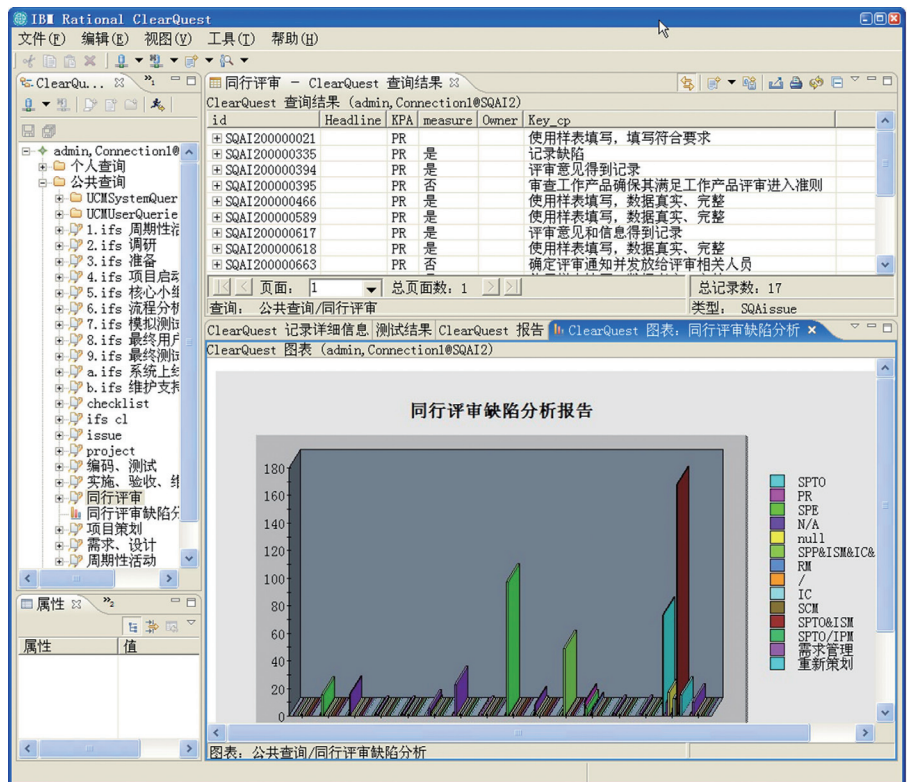
国防单位项目负责人借助运行在Rational ClearQuest下的同行评审系统验证修订后的修改的正确性，并在系统中闭合缺陷修订。

1. 项目负责人应对在审查会议中分配给他们的 ([1]) 问题进行跟踪，并将跟踪结果提交给缺陷管理系统。
2. 并且验证所有的已纠正的缺陷。开发人员对于已经纠正的缺陷要对同行评审记录在缺陷管理 ([2]) 工具 ([3]) 中进行更新。
3. 如果缺陷已经 ([3]) 完全修改，并且经过了回归测试后，由项目负责人通知SCM，并由其对工作产品作相应的配置状态变化。



2.2.3.2.2.3 项目负责人闭合同行评审总结，同时提交给SQA和SEPG

项目负责人在项目的生命周期阶段末期，或者在迭代周期的结束点，利用 Rational ClearQuest下的同行评审系统对该阶段发现的所有审查缺陷进行统计和汇总，形成缺陷统计分析报告，并对阶段的缺陷进行分析，能够确定出需要关注的重点，以及将采取的减少缺陷的措施。



2.3 技术类解决方案

2.3.1 软件测试

2.3.1.1 实施Rational软件测试解决方案能够解决国防单位目前存在的哪些问题

国防单位通过使用Rational软件测试解决方案,实施有效的测试管理和测试平台可以循序渐进地解决国防单位目前软件开发中的存在问题:

- 由于GJB5000成熟度第二级没有测试关键过程域,缺乏有效的测试最佳实践指导测试流程;
- 针对测试计划、执行、评估缺乏执行力度,导致测试不全面不充分;
- 没有自动化测试手段,导致测试人工成本较高;
- 没有建立测试需求与软件需求的追踪,无法了解测试计划覆盖的情况;
- 没有建立缺陷与软件需求的追踪,无法了解需求的缺陷率
- 缺乏边开发边测试的思想,测试工作在项目后期介入,使软件风险在项目后期才能降低,并加大开发人员查找bug和修改bug的成本;
- 缺乏手段查找软件内存bug,为系统使用留下隐患;
- 针对缺陷管理存在规范性不足,主要针对源码的缺陷管理,对于整个生命周期的质量保证无法满足;
- 对代码覆盖率缺乏有效的统计分析,易造成未测试到的代码流入到发布版本中;
- 对测试重视不够,缺乏有战斗力的测试团队;

针对以上存在的问题,国防单位如何对目前的测试状况进行改进?我们认为IBM Rational软件测试解决方案可以结合国防单位自身具体实际帮助我们测试管理、开发人员测试、测试人员测试、测试流程不同的角度解决在测试中的困惑,了解国际上先进的测试方法。

2.3.1.2 IBM Rational 软件测试解决方案助力国防单位测试能力的提升

2.3.1.2.1 方案目标

IBM将帮助国防单位建立基于IBM Rational产品的测试平台,该平台的导入并不是一个简单的工具安装,更重要的是引入IBM Rational的先进测试技术和软件测试管理流程。从而帮助国防单位更加顺利地基于测试基础平台的建设,IBM可向国防单位提供其专业的技术咨询服务,从以下几个方面来帮助国防单位建立完善的测试体系基础架构。

- 建立完善的软件测试基础平台

通过IBM Rational测试平台来统一整个测试团队,自顶向下覆盖软件测试流程规范、测试过程管理、软件测试自动化三个层次的工作需求,使国防单位的测试团队能够有序完成项目的软件测试任务。该测试基础平台不仅是指测试工具平台,也包括流程规范、工作指南等。

- 建设高水平的软件测试团队

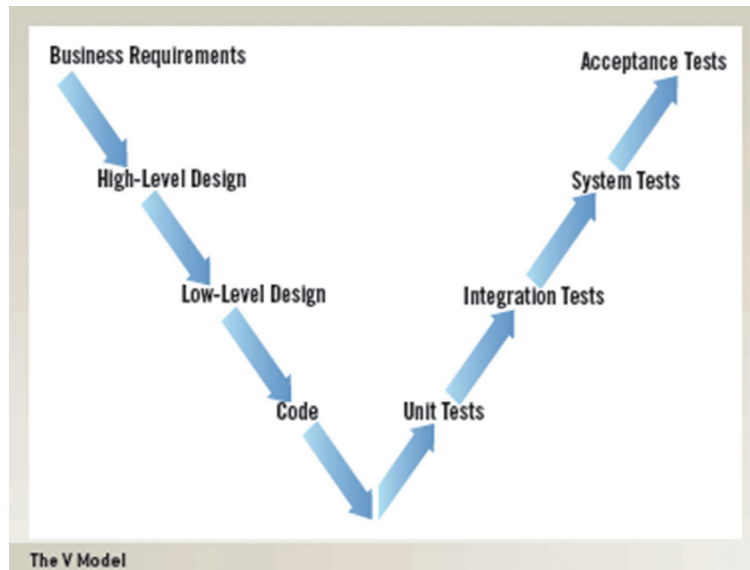
IBM Rational在多年的软件开发实践中积累了丰富的软件工程技术,并且将这些最佳实践经验总结在Rational Unified Process(RUP)中。我们将把RUP中的软件测试方法传递给国防单位的团队,帮助他们掌握业界最先进的软件工程技术,从而打造一支高水平的软件测试团队。

基于国防单位在测试方面所面临的问题,下面我们来了解一下IBM Rational提供的相应的解决方案平台,首先我们来看一下业界测试的最佳经验,这将有助于跳出我们现有的测试思路,提升我们针对测试的理解。

2.3.1.2.2 针对测试的最佳经验

2.3.1.2.2.1 统一的需求变更管理流程

为了保证项目进度,根据V-模型,在需求阶段完成之后,开发机构在业务需求的基础上进行软件开发,而测试机构则应该基于同样的需求准备验收测试(Acceptance Test)。



然而,在实际的开发过程中,业务需求会不断发生变化,如果测试机构不能及时了解这些变化,将有可能导致大量的测试人力资源浪费。因此,为了避免这一问题,需要在业务部门、开发就够与测试机构之间建立起一套统一的需求变更管理流程,以保证在需求变化之后,测试机构可以马上对测试工作做出相应的调整。

2.3.1.2.2.2 统一的、工具支持下的测试工作流程

一个有一定规模的团队,需要通过一个统一的测试流程来规范测试活动,明确每个角色的职责和所承担的任务,这样才能保证整个测试团队能有条不紊地完成测试项目。

由于实际测试中很可能会出现一个人同时测试多个子项目的情况,在这种情况下,如果没有适当工具的支持,是很容易出现错误的,也不利于测试进度的掌握和测试团队的管理。因此,最好将测试工作流程建立在成熟的工具之上建立相应测试平台,实现无纸化测试,即测试计划、测试用例、测试结果等文档的电子化流转。

2.3.1.2.2.3 测试必须在稳定的版本上进行

被测软件版本的稳定,是进行有效测试的重要保证,因此,除非特殊情况(如一个缺陷影响了后续测试工作的进行),否则,因在整个测试期间,保持被测版本的稳定,直到所有测试用例测试完成之后,再统一由开发中心提交包含了若干缺陷修复的被测版本,而不应频繁地更改被测版本,因为这可能导致新缺陷的引入。

2.3.1.2.2.4 科学地利用回归测试工具以提高测试效率

在测试的初期,大多数测试脚本都是由测试人员来手工测试的,但当被测系统相对稳定之后,可能的话尽量利用回归测试工具对某些测试脚本进行自动化改造,以期在随后的测试工作中提高测试工作的效率,然而,时机的选择非常重要,过早地引入回归测试工具也将导致浪费。

2.3.1.2.2.5 尽早进行、持续进行压力测试

压力测试往往被认为应该在项目的后期,由测试机构来进行。但实际上,在开发机构,应尽早进行压力测试以确认系统架构是否存在设计缺陷,如果这些缺陷拖到项目后期将往往会导致项目的彻底失败。另外,在开发的过程中,开发机构应通过持续的压力测试,来发现系统设计的变更对整个系统性能是否有显著的影响。

在测试后期,测试机构也需要进行压力测试,但此时的压力测试的目标将从应用程序扩展到整个系统,因此压力测试将更加全面和细致,尤其

应充分考虑各种不同的测试场景(如业务高峰、月底年底等等)的不同负载情况,同时,应紧密监控各种系统资源的应用状况。另外,在系统维护运行的过程中,如需进行硬件、操作系统、数据库或中间件升级,都需要测试组进行压力测试才能升级。

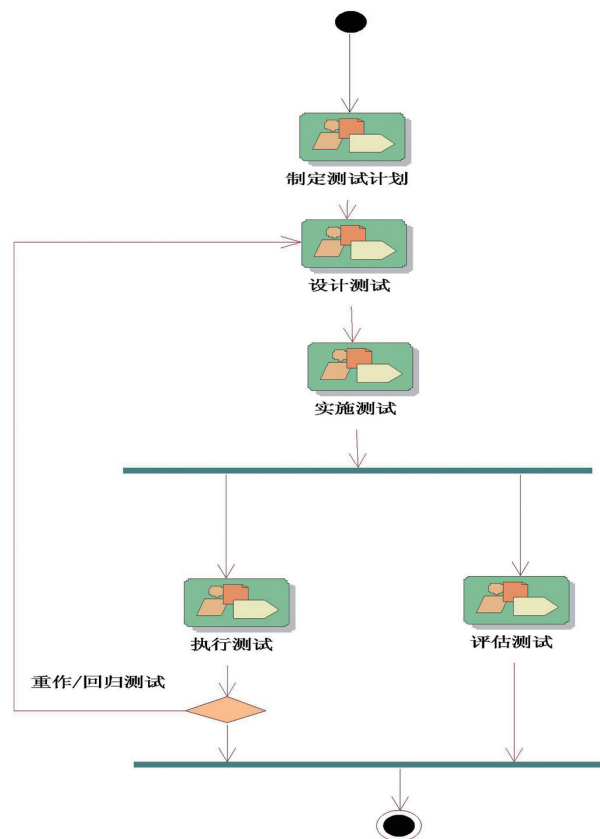
2.3.1.2.2.6 统一的、高效的缺陷追踪流程

在v-model的四中测试当中,开发中心的开发组应负责单元测试(Unit Test),测试组应负责集成测试(Integration Test)和系统测试(System Test),但是,如果任何一个测试团队发现了缺陷,都应通过统一的缺陷管理流程来将缺陷反映给开发团队,这可以提高测试组与开发组之间的沟通效率,提供对产品质量提供科学、准确的质量指标。

2.3.1.2.3 测试流程规范的定制

软件测试管理需要一个稳定的流程来支持。因为软件开发本身更多的是人的活动,团队的活动。这个团队包括了业务人员、需求人员、分析设计人员、管理人员、测试人员和开发人员等多角色的参与,因此在开发中提倡的是整合化的项目管理,需要通过流程来将各个方面的人员根据项目的实际需要整合到项目活动中。流程是为了各个角色之间能够更好的协作。实践证明,只有基于稳定的流程的团队协作才是高效的,也只有基于稳定的流程所产出的软件工件的质量才是有保证的。因此,在软件行业中,有这样一句话,“流程的稳定性等同于产品质量的稳定性”,实际上也和GJB5000的思想是一致的。因此,对于国防单位来讲,建立稳定高效的测试管理流程,有利于内部的管理和部门间或不同开发组间的协作,一旦过程中出现问题可及时发现是哪个阶段的问题,将解决问题的成本降至最低。这个流程实际上是由一系列的规程组合而成的,例如:需求的管理规程、测试项目管理规程、缺陷修复规程、变更控制规程、产品交付规程等。所以开发的其他阶段实施了GJB5000的KPA,有助于与测试的衔接,规范的工作。

规范的背后实际是由一系列业界最佳实践的方法论来支撑,IBM Rational Unified Process(以下简称RUP)提供了一套完整的测试流程方法论的框架,软件测试团队可以以它为基础,根据业务发展的实际要求,定制符合团队需要的软件测试流程。RUP中的软件测试流程如图所示:



每个测试环节的具体阐述如下:

- 制定测试计划的目的是确定和描述要实施和执行的测试。这是通过生成包含测试需求和测试策略的测试计划来完成的。可以制定一个单独的测试计划,用于描述所有要实施和执行的测试类型,也可以为每种测试类型制定一个测试计划;
- 设计测试的目的是确定、描述和生成测试过程和测试用例;
- 实施测试的目的是实施(记录、生成或编写)设计测试中定义的测试过程。输出工件是测试过程的计算机可读版本,称为测试脚本;
- 执行测试的目的是确保整个系统按既定意图运行。系统集成成员在各迭代中编译并链接系统。每一迭代都需要测试增加的功能,并重复执行以前版本测试过的所有测试用例(回归测试);
- 评估测试的目的是生成并交付测试评估摘要。这是通过复审并评估测试结果、确定并记录变更请求,以及计算主要测试评测方法来完成的。测试评估摘要以组织有序的格式提供测试结果和主要测试评测方法,用于评估测试对象和测试流程的质量。

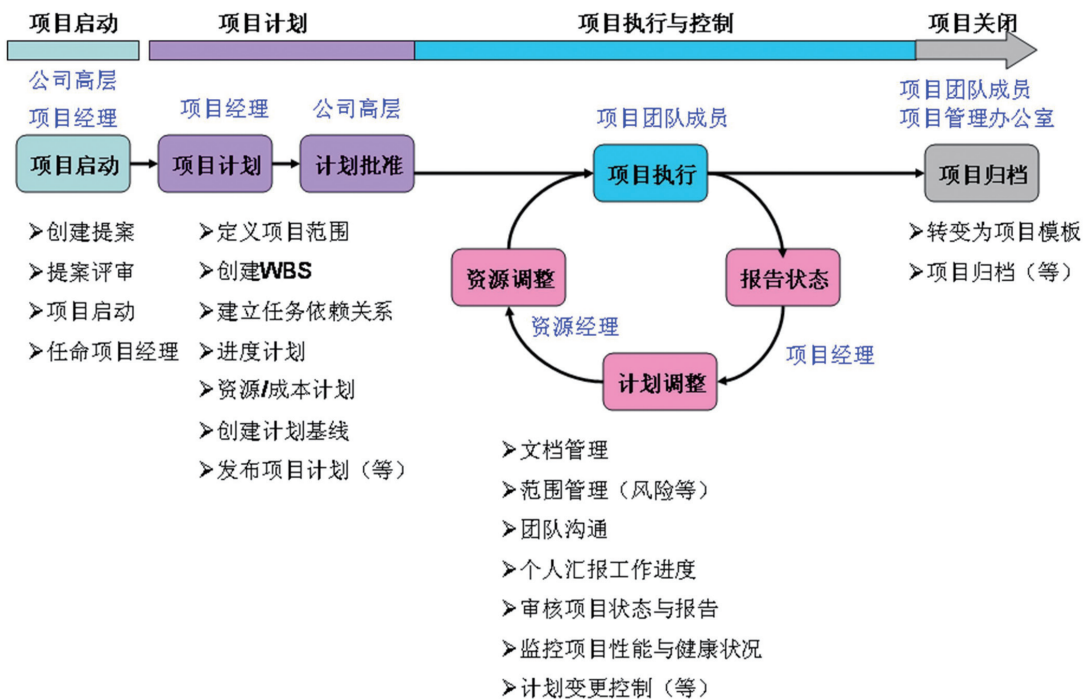
针对RUP提供的流程通常是标准的, IBM提供工具Rational Method Composer(简称RMC)可对此流程进行定制, 这样国防单位最终可通过RMC形成符合单位需要的测试流程规范。并可发布成网页的形式供团队所有人员参考。

2.3.1.2.4 测试管理平台

建立自动化管理平台来进行测试管理。由于软件开发中所产生的工件是细碎的、复杂的, 并且质量是不直观可见的, 非常难于评价, 这样, 就需要通过自动化平台来进行管理和控制。自动化管理平台, 能够稳定地执行我们所设定的流程中的相关规程, 为使用者提供使用流程中的相关规程的简化的操作, 同时, 也能够强化工作流程的定义, 并且在规程执行的过程中能够自动地收集全部的信息, 并且能够自动化部分的质量保证活动, 从而能够以较低的管理成本确保管理效果, 做到简化流程操作, 稳定流程执行, 明确分工职责, 明确操作轨迹, 并且提供必要的审计和控制能力。通过自动化管理平台的实施, 可以极大地简化流程的实施, 从而确保流程能够被使用, 也就确保了各个规程相关活动的质量。同时, 软件资产能够得到妥善的保护, 项目状况也能够通过工具平台提供的统计、分析功能准确反映。对于每个交付, 也能够提供准确的控制。测试管理平台由以下几部分组成:

2.3.1.2.4.1 测试项目管理平台

测试作为项目的一部分会随着项目的进展不断的变化, 这样项目经理或测试经理也需要随时了解项目的测试进度情况以及在某一时期的状态和风险等等。IBM提供项目管理工具Rational Portfolio Manager简称(RPM)来进行测试项目的管理, 并结合项目生命周期的项目启动、项目计划、项目执行、项目控制和项目关闭五个阶段进行管理。下图显示了一个完整的项目管理生命周期, 并给出了每个阶段的一些比较常见的活动和任务。



在项目启动阶段, 和测试的关系并不明显, 我们直接来看项目计划阶段。项目计划阶段, 项目经理定义项目范围, 测试计划作为开发计划的一部分会被定义, 并创建和测试相关的WBS, 建立测试任务以及测试与开发任务之间的依赖关系, 计划测试的进度、资源和成本, 最终创建项目计划基线, 以便今后监控项目执行性能; 最后项目经理向测试团队发布项目的测试计划。

项目执行与项目控制是相互迭代和关联的两个阶段。在测试执行过程中, 项目经理需要不断地对项目的测试状态进行监控, 管理遇到的问题、

风险以及缺陷,从而根据发现的状况调整项目的开发计划和相应的资源计划;如果项目计划有比较大的调整,还需要获得高层的审批,并重新建立项目基线。

2.3.1.2.4.2 测试资产管理平台

在软件开发过程中所产生的全部的代码、设计文档、数据文件、配置文件、模型等,我们统称为软件资产。为什么我们要提出对测试资产的管理,实际上在测试过程中会产生大量的配置项,包括测试计划文档、测试设计文档、测试脚本、测试数据等等,而且这些配置项会随着开发的版本变化而变化,需要开发和测试的版本保持一致才不会发生混乱。良好的测试资产也能够提供测试项目间的信息共享,减少相关测试项目的工作重复性。IBM通过ClearCase来实现测试资产的管理,并可有效解决以下软件开发测试中的常见问题:

- 开发人员未经授权修改代码或文档;
- 人员流动造成企业的软件核心技术泄密;
- 找不到某个文件的历史版本;
- 无法重现历史版本;
- 使维护工作十分困难;
- 因一些特性无法按期完成而影响整个项目的进度或导致整个项目失败;
- 已修复的测试脚本的Bug在新版本中出现;
- 配置管理制度难于实施;
- 分处异地的测试团队难于协同,可能会造成重复工作;

2.3.1.2.4.3 测试过程管理平台

测试管理平台需要能够提供完整的测试管理能力,并且能够与主流的测试工具进行集成;能够支持对于测试计划的定制,能够支持对于测试用例的定义,能够支持对于测试用例的配置,能够支持对于测试计划的自动调度执行,并且能够支持对于测试执行结果的自动收集,在没有主流的自动化测试工具应用的条件下,依然能够支持手工的测试管理,并且能够方便地将测试相关信息与需求进行关联,以便评估测试计划的完整性,从而管理验证和确认活动的完整性和有效性,保证系统质量。

IBM通过ClearQuest实现了对整个软件测试生命周期的管理,可以帮助软件测试团队快速建立软件测试平台和测试管理流程,使软件测试团队快速拥有以下能力:

- 提供测试管理的核心平台,整合了从测试需求、测试计划、测试设计、测试实施、测试执行到测试结果分析、测试报告的自动生成等整个测试生命周期的管理活动。同时,统一组织各种Test Suite, Test Case, Test Script,方便地进行回归测试
- 遵循RUP标准测试流程,使测试人员能够在统一的测试管理平台上、遵循统一的测试管理流程,完成对包括产品的功能性、可靠性和性能等全方位的质量测试。
- 作为一种整合解决方案,测试管理平台与Rational其它工具一起,提供从测试需求、到整个软件测试流程管理、缺陷追踪、测试结果评测的可追踪性,方便测试管理人员进行软件测试过程监控和有关软件质量的各种量化指标的采集、分析。
- 缺陷追踪能力。在实际测试中,往往测试发现的问题成百上千,对于这些问题如何保证最终都将被解决,这就需要建立缺陷追踪。通过缺陷追踪能够有效地支持缺陷的流程化管理,并且能够将缺陷与需求和测试有效关联,从而了解哪些需求存在缺陷,哪些测试发现了缺陷,方便对于项目、测试工作的客观评价。

2.3.1.2.5 测试工具平台

工欲善其事,必先利其器。有了良好的测试管理平台,好的测试工具能使测试人员在测试过程中提高效率,并保证测试的完整性。IBM提供业界优秀的测试工具,包括:功能测试工具(Rational Functional Tester,简称RFT)、性能测试工具(Rational Performance Tester,简称RPT)、白盒的运行时分析工具(Rational PurifyPlus)、嵌入式系统测试工具(Rational Test RealTime)和应用程序安全测试工具(Rational AppScan)。

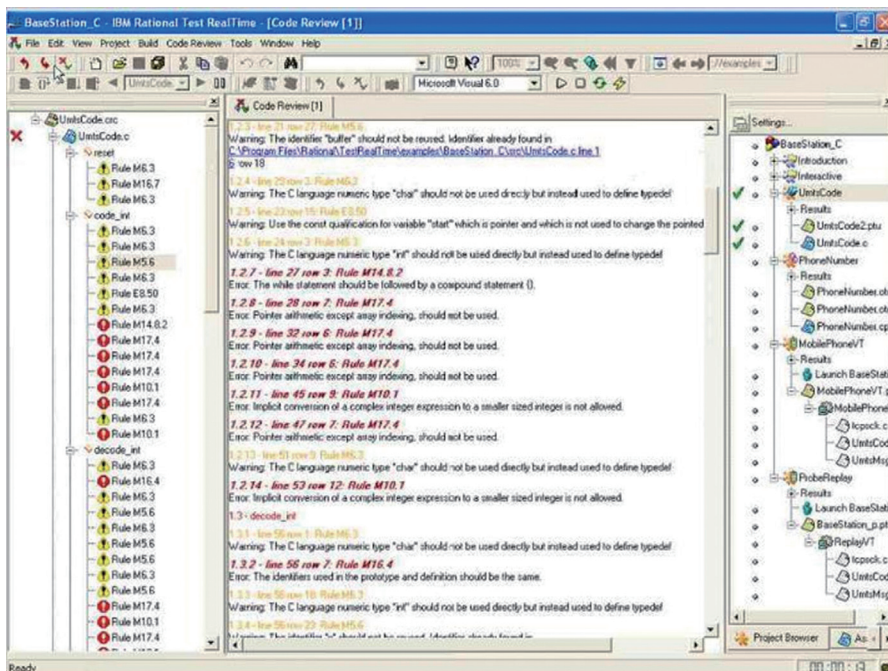
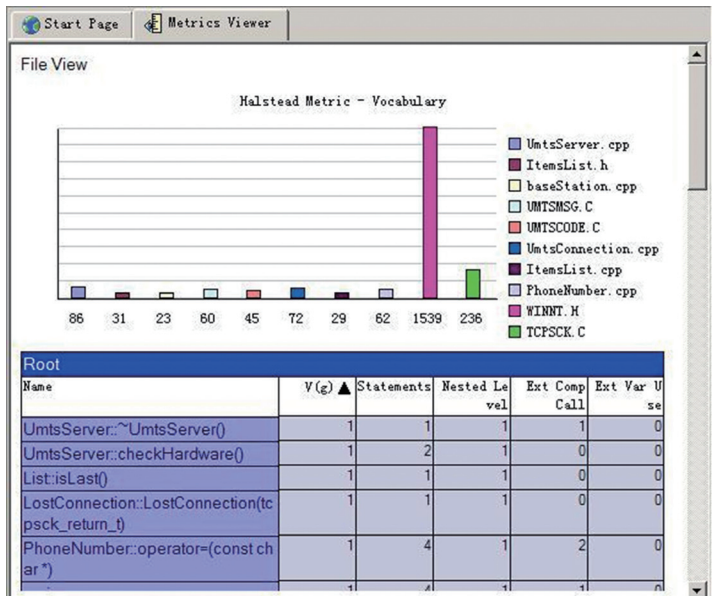
根据我们前面提到的测试最佳实践的V-模型(参考2.2.1),我们将测试分为开发人员的测试和测试人员的测试两类。开发人员的测试主要集中在单元测试阶段,包括嵌入式系统测试工具和白盒的运行时分析工具的使用;集成测试和系统测试由测试人员来完成,包括功能测试工具和性能测试工具的使用。

2.3.1.2.5.1 开发人员测试

开发人员测试更多是在单元测试阶段,又称模块测试,是针对软件设计的最小单位 - D程序模块进行正确性检验的测试工作。其目的在于发现各模块内部可能存在的各种差错。单元测试需要从程序的内部结构出发设计测试用例,多个模块可以平行地独立进行单元测试。通常“单元”不是一个完整的程序,它需要有上层调用,并调用下层模块。为使其正常执行,需要设计驱动程序并用打桩的办法模拟下层模块。

在这个阶段主要考虑的测试包括:

- 静态分析: 在不执行源代码的情况下,对代码中组件关系和方法调用,代码复杂度量以及代码的编写方法进行分析。支持圈复杂度度量(cyclomatic complexity metrics)和Halstead度量标准,度量元包括代码行、注释行、空行、注释率、被调次数、变量被应用次数、测试时间、圈复杂度最大值、平均值、离散值、嵌套的最大值、平均值等。
- 代码规则检查: 支持MISRA-C 2004代码编程规则检查,支持代码编程规则的选择定制,能自动验证应用软件是否遵循了所选择的编程规则集,报告所有违反编程规则的代码并以文本和图形方式显示。
- 组件测试: 对独立的组件编写测试驱动程序,并执行被测试组件,从而验证其功能

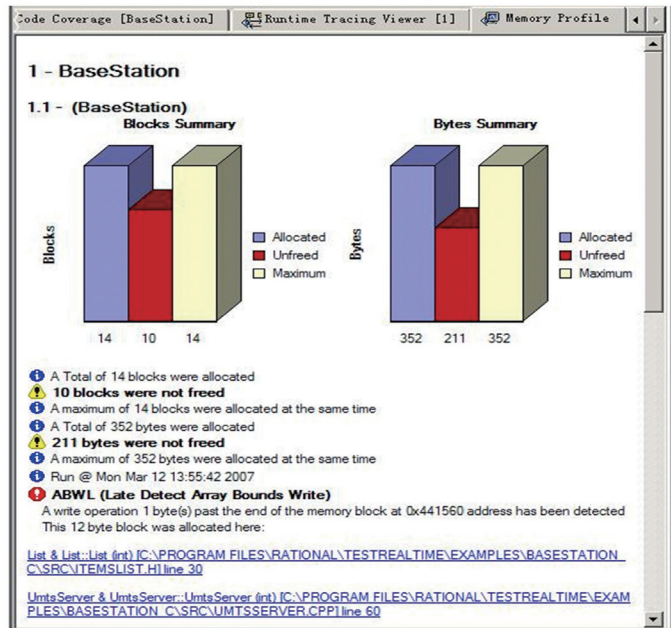
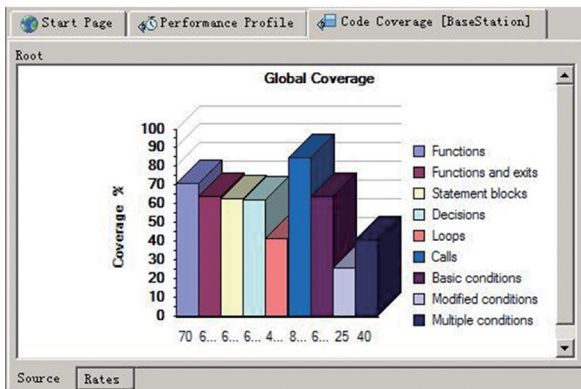


- 运行时分析: 通过分析在软件执行时所收集的数据来理解软件的内部运行状况, 从而发现如下问题:

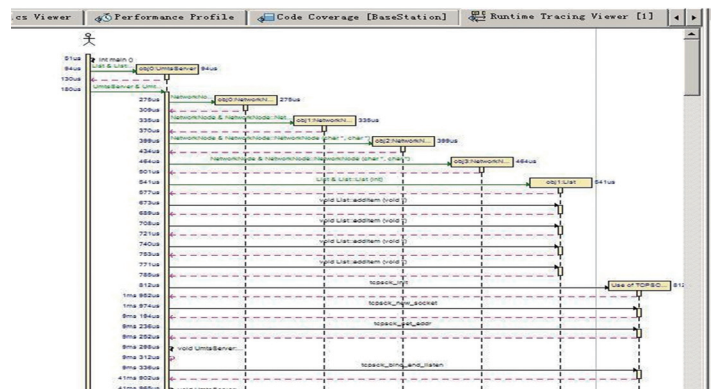
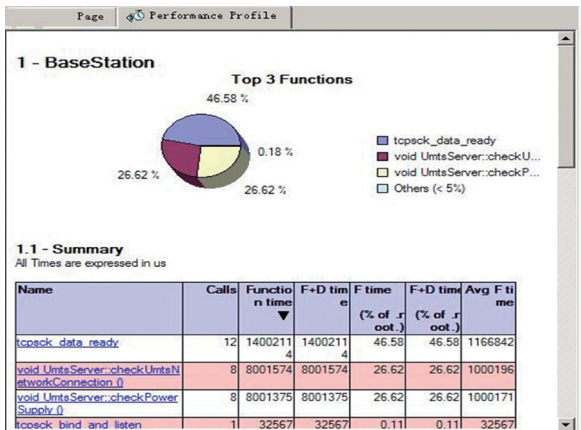
```

datapool2.ptu | filter3D.c | rtrtDataInLine.c | datapool.ptu
SERVICE sinsin
SERVICE_TYPE extern
- Tested service parameters declarations
#float x,y;
- By function returned type declaration
#float z;
TEST 1 LOOP at_1loop
FAMILY nominal
ELEMENT
VAR x,  init = atof(readVarf("Xarray")),  ev ==
VAR y,  init = atof(readVarf("Yarray")),  ev ==
VAR z,  init = 0.0, min = atof(readVarf("inf")), max = atof(readVarf("sup"))
#z = sinsin(*PI, y*PI);
END ELEMENT
END TEST - TEST 1
END SERVICE - sinsin
    
```

- 代码覆盖
- C/C++应用的内存访问错误和内存泄漏
- 性能分析
- 运行时跟踪和线程分析



2.3.1.2.5.2 测试人员的功能测试



2.3.1.2.5.2.1 功能测试方案工作方式

功能测试主要围绕Windows图形界面、字符终端和Browser界面进行测试。客户端可以为Java、VC、VB、PB、Delphi等编制的软件、各种字符终端软件或者运行浏览器Microsoft Explorer和Netscape, 通过自动录制形成测试脚本实现自动化的功能/回归测试。

软件功能测试方法如下所示:

测试需求:	验证业务功能和业务规则
方法:	<p>利用有效的和无效的数据来执行各个用例、用例流或功能, 以核实以下内容:</p> <ul style="list-style-type: none"> • 在使用有效数据时得到预期的结果。 • 在使用无效数据时显示相应的错误消息或警告消息。 • 各业务规则都得到了正确的应用。
完成标准:	<ul style="list-style-type: none"> • 所计划的测试已全部执行。 • 所发现的缺陷已全部解决。
工具实现:	<ul style="list-style-type: none"> • 实现功能测试脚本和测试数据的自动生成和回放, 测试执行自动化, 测试结果比较的自动化和回归测试等

1) 明确功能测试需求

功能测试应该侧重于可以被直接追踪到用例或业务功能和业务规则的所有测试需求。这些测试的目标在于核实能否正确地接受、处理和检索数据以及业务规则是否正确实施。这种类型的测试基于黑盒方法, 即通过图形用户界面(GUI)与应用程序交互并分析输出结果来验证应用程序及其内部进程。

2) 利用ClearQuest TestManager制定功能测试计划和测试用例

基于系统的需求, 利用ClearQuest TestManager可以帮助制定测试计划和测试用例。测试计划应明确测试目标、所需完成测试用例、测试里程碑和主要评测手段、所需资源和提交结果。测试用例应明确每个测试用例的测试输入、执行条件和期望结果。

3) 利用RFT录制测试脚本

基于测试计划和测试用例, 利用RFT录制测试脚本。在脚本的录制过程中应充分考虑到脚本的重用和可维护性, 灵活的组织和利用测试脚本, 是成功实现软件的自动化功能测试的关键。

4) 利用数据池(datapool)技术实现测试数据变化

利用RFT的数据池(datapool)功能按需求形成测试数据, 实现大批量数据驱动的软件功能测试和测试流程中数据的前后相关性。

5) 利用ClearQuest TestManager执行测试用例

按照测试计划的要求, 组织测试用例并和录制形成的测试脚本关联, 形成便于回归测试的各种测试套件。之后, 执行各种测试套件或相关测试用例。并监视测试执行情况, 分析测试执行结果, 改进测试用例。

6) 评估测试报告

ClearQuest TestManager会自动记录功能测试的结果并生成各种类型的测试报告, 测试人员可以通过这些测试数据来生成各种测试报告和测试结果。

2.3.1.2.5.2.2 功能测试解决方案的特性

IBM Rational的功能测试解决方案的目标, 是使功能性测试变得更简单、有效并可重复执行, 为快速提升功能测试能力。它主要具有以下特点:

- 1) 能够方便的对各种环境(IDE)中开发的应用程序, 完成包括测试计划、测试设计、测试实施、测试执行和测试结果分析等全部测试流程。
- 2) 能够方便的录制或编写各种功能测试脚本, 实现自动化的功能/回归测试。
- 3) 利用数据池方便地解决大批量数据驱动的功能测试;
- 4) 能够方便地完成分布式功能测试, 可以一次测试多种测试平台;
- 5) 能够自动完成功能测试需求覆盖, 确保应用程序满足产品规格说明和测试计划的每一条业务需求。

2.3.1.2.5.3 测试人员的性能测试

2.3.1.2.5.4 性能测试方案工作方式

性能测试是用负载生成工具模拟实际用户使用应用软件的过程, IBM Rational性能测试方案的系统架构如图所示, IBM Rational的性能测试工具安装在Master主机上, 控制脚本运行和整个负载加载过程。根据负载模型的要求, 首先将测试脚本下载到Agent机器, 然后在Agent机器运行脚本, 模拟多个虚拟用户, 分别加载到服务器。同时在整个测试过程自动收集测试数据, 由测试主机统一处理, 生成测试报告及各种报表。



具体性能测试步骤如下:

- 1) 明确性能测试需求: 在进行性能测试之前, 需要根据系统的性能需求明确描述性能测试的目标。
- 2) 确定系统的负载模型: 性能测试的关键是模拟系统的真实负载。为此, 我们通过如下手段获得系统的负载模型:

从最终使用人员获得操作情况, 如经常进行的业务类型、业务操作的频率

根据系统日志, 可以获得每日所进行的各种业务类型和业务量

通过和系统测试人员、操作人员、系统架构师充分沟通和配合, 形成如下的系统负载模型。

使用IBM Rational的测试解决方案, 可以使用以下负载模型建立技术:

- 建立虚拟用户组, 模拟不同类型的系统用户, 执行不同的系统业务流程;
- 基于Scenario模拟不同的业务操作流程;

- 基于Selector模拟人的随机操作或选择的行为;
- 基于Synchronize Point和Delay等协调不同脚本或不同的业务操作间的执行顺序, 建立基于时间表的负载模型;
- 基于Event和Event Dependence, 建立基于事件触发确定脚本执行顺序的负载模型;

3) 利用RPT录制测试脚本

利用RPT录制负载模型中列出各种业务, 自动形成测试脚本。

4) 和利用数据池(datapool)技术实现测试数据变化

利用RPT的数据池(datapool)功能按需求形成测试数据, 并且不用编写任何代码就能实现在测试执行时不同用户使用不同的数据访问系统, 保证性能测试的真实性。

5) 利用RPT创建测试执行计划

测试执行计划是基于录制形成的测试脚本, 模拟各种业务, 实现负载模型的过程。

6) 测试执行

RPT可以根据指定的总用户数自动调整每个角色的用户, 并可以监视执行情况。

7) 评估测试报告

RPT会自动记录性能测试的结果并生成各种类型的测试报告, 测试人员可以通过这些测试数据来分析系统的性能瓶颈所在。

2.3.1.2.5.5 性能测试方案的特性

IBM Rational性能测试解决方案可以方便灵活地模拟各种负载模型, 完成以查找响应时间瓶颈、系统吞吐量、最大并发虚拟用户等为目地的各种要求的性能测试。包括:

- 能够完成对性能测试的测试需求、测试计划、测试设计、测试实施、测试执行和测试结果分析等整个测试生命周期的管理;
- 能够方便的完成性能测试对负载模型的各种要求, 包括:
 - 能够建立复杂的Scenario模型;
 - 能够准确模拟复杂负载的时序控制;
 - 能够进行基于Transaction的负载分析;
 - 能够建立面向目标的事务负载模型, 例如: 100事务/秒
 - 响应时间精确到1/100秒;
 - 能够模拟不同虚拟用户的不同IP地址;
- 能够方便地完成性能测试过程中各种指标的观测;
- 能够方便地完成性能测试结果分析和各种结果报告的生成;

2.3.1.2.5.6 测试人员的应用安全性测试

目前, 企业开发很多转向B/S结构, 不管是内部站点还是外部站点, 网站的应用的安全性摆在我们面前。也许很多企业使用了网络扫描工具、使用了防火墙或使用SSL加密数据认为自己的网站是安全的, 但这些都不能彻底提高WEB应用安全性。据Gartner统计, 2/3的Web应用是脆弱的, 75%的信息安全攻击都来自web应用的层次上, 这些攻击包括:

- 跨网站脚本攻击
- SQL注入

- 缓冲区溢出
- Cookie毒药
- 隐藏域
- 篡改参数
- 暴力浏览
- 等等…

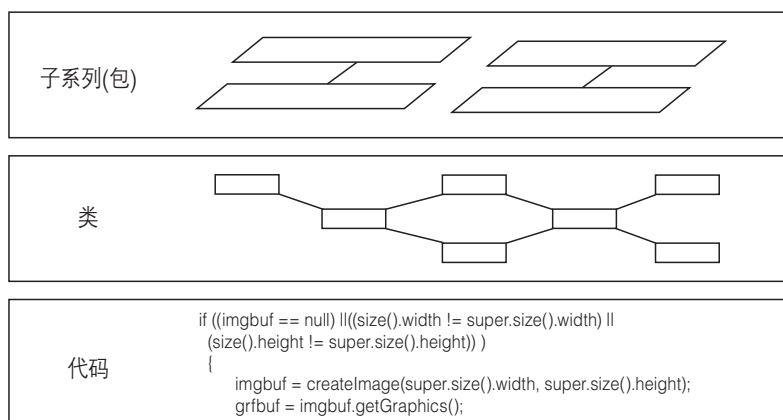
为了有效对抗来自对Web应用的攻击,应用安全测试工具帮助测试人员以类似于防病毒软件的使用方式扫描WEB应用,发现可能有的安全漏洞,并对相应的漏洞提出修改建议供开发人员解决,以多种报告的形式展现测试结果,最终保证Web应用的安全性。

2.3.2 可视化建模

2.3.2.1 有效管理系统复杂度

面向对象方法最基本的原则就是抽象,把一类具有相同属性和行为的实体抽象成为一个类(Class),再通过把类实例化成对象(Object)来映射现实世界中的某一个具体实体。对象通过操作(Operation)来对外提供相应的服务,在对象模型中我们只需要描述对象所实现的功能,而封装了操作实现的细节。与软件代码相比,对象模型描述的也是同一个系统,但它展示的是系统结构中最关键的元素以及它们之间的关系,所有的编码细节都已经被忽略掉了,从而有利于开发人员把握理解整个系统。

复杂的对象模型往往由成千上万个模型元素构成,UML中的包(Package)可以很好地管理这些模型元素,我们在设计过程常用的一些概念如子系统、模块等概念也可以用包来表示。所以整个系统的对象模型就呈现出一种层次结构,当你想要了解系统的整体结构时,你看到的是子系统之间关系;你了解子系统内部的设计时,你又可深入到子系统内部观察它的内部结构。在这种层次结构中,每一层所包含的信息量都是有限的,以便于每一位开发人员的理解;但随着工作的需要,开发人员随时可以了解更深层次的设计细节。



2.3.2.2 增强团队的沟通

对象模型同时也作为软件设计的蓝图,记录了开发人员的设计思想。对于设计者而言,对象模型提供了一个工具来帮助他来整理设计思路,整个的设计过程都可以被记录下来;同时,也避免开发者在整个系统架构明确之前就陷入编码的细节之中,对于模型的调整修改相对于代码的改动要简单得多。

另一方面,对象模型也使得设计的结果很容易被其他人所理解,设计者的设计意图可以被完整的传递而不发生信息的失真。可视化建模采用的是标准的统一建模语言UML,所有的开发人员都应该采用这种统一建模语言来进行系统的设计,从而保证大家工作的结果是所有人都可以理

解的。这也是UML语言的设计目的之一，即使用UML来统一整个开发团队的沟通手段。

2.3.2.3 提高系统设计的可重用性

面向对象技术最基本的原则就是抽象，即把整个系统的功能尽可能地分配到多个类中去，每个类应该只做并且做好一件事情。因为每个类实现的功能比较单一，所以可以有更多的机会被重用。同时尽量利用构件化的思想把关系比较紧密的类组合成构件，构件具有定义明确的功能并且以接口的形式对外提供服务。基于构件的架构具有最大的可重用性，一方面可以重用现有的商业构件来搭建系统，另一方面当前系统中的构件也可以被其他的系统所重用。

2.3.2.4 增强系统架构的灵活性

好的系统架构应该是具有最大的灵活性，即它不仅能满足系统目前的需求，更重要的是它还可以满足系统将来的需求。可视化建模技术可以从以下几个方面增强系统架构的灵活性。

- 把系统中易变的功能和不变的功能分开，把跟需求相关的功能和通用的服务功能分开。这样当系统需求发生变化的时候，我们可以使大部分的系统结构保持不变，变动地只是局部。
- 尽可能使用接口来调用另一对象所提供的服务，而不是为了一些所谓的“效率”而直接访问对象内部的数据结构。当必须修改某一对象的功能时，只要该对象实现的接口不发生变化，该对象的使用者就不需要做任何的变化。

2.3.2.5 建立以构件为基础的软件架构

在应用可视化建模技术进行系统分析设计的过程中，一个重要的实践经验就是建立基于构件的架构(Component-Based Architecture)。构件技术将为软件系统提供最大的灵活性，构件的特点是：构件内部的元素之间高内聚、构件之间低耦合，整个系统的功能由一组构件相互协作而完成，构件之间通过定义明确的接口来使用相互的服务。构件体现了面向对象技术封装(Encapsulation)和模块化(Modularity)的思想，既能够适当隐藏不需要关心的细节，又可以在需要的时候随时展现设计的详细结构，从而有效管理系统的复杂度。

另一方面，系统功能是由一组相互协作的构件提供的，系统功能的增加修改只需要增加新的构件或改动相关的构件，使得系统的扩展性大大增强，能够迅速地适应业务快速变化的需求。构件技术也大大提高了系统开发的重用水平，不但可以重用自己开发的构件，也可以重用其他的或商业化的构件

2.3.2.6 保证模型和代码之间的一致性

在传统的软件开发过程中，一个最常见的问题是很难保证系统设计和代码之间的一致性。开发人员在进入到编码阶段后，为了满足编程工具的约束或基于非功能性需求(性能、可靠性、可维护性等)的考虑，必然会对系统的设计进行一定的修改，但这种修改很少会反映到系统设计中，而直接体现在系统代码中。这就造成了系统设计和代码之间的脱节，设计文档已经不再精确反映系统的实际情况，使得系统在日后难于维护。在迭代化的开发过程中，这种问题会更加显著，因为每一次迭代都是在上一次迭代的基础上进行的，这种设计和代码的不一致性会严重影响到下一次迭代的开发。

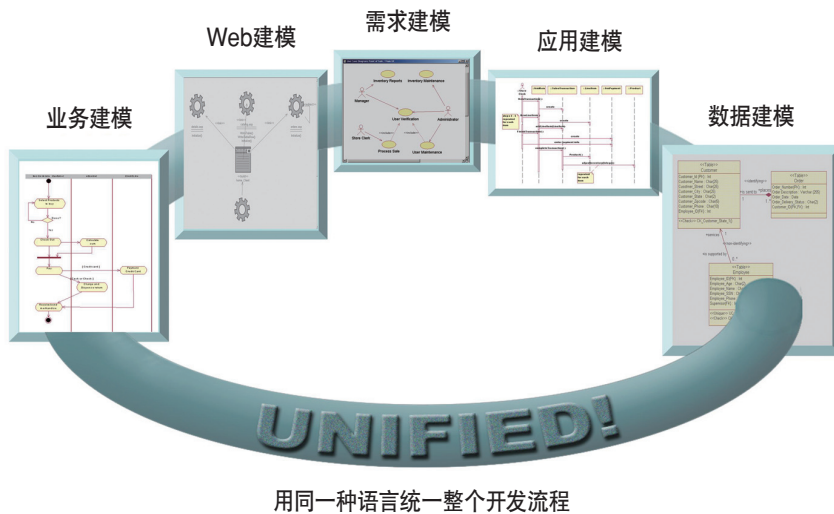
由于传统设计方法在语义上缺乏严格的定义，模型和代码之间的一致性必须由手工来维护。在可视化建模技术中，由于UML语言的语义没有二义性，可以由工具来实现这种模型和代码之间的同步，从而很好地保证模型和代码之间的一致性。下文中将要介绍的IBM Rose和XDE都具有这样的同步功能，这种功能包括了从模型自动生成代码的正向工程(Forward Engineering)技术，以及根据代码变化来自动地更新模型的逆向工程(Reverse Engineering)技术，统称为双向工程RTE(Round-Trip Engineering)。

2.3.2.7 使用UML统一软件开发生命周期

在项目开发的开始阶段, 开发团队应该对客户未来的业务有一个全面地了解。我们可以通过业务建模(Business Modeling)来达到这一目的, 通过UML来建立完整的业务过程描述, 使得开发团队和业务部门能够对软件系统所处的业务背景有一个全面而一致的认识, 在此基础上再进一步抽取具体的系统需求。

对于很多涉及到数据库的应用软件, 以往我们都是用实体关系图(ER图)来描述数据库的内部结构。现在利用UML我们可以更好地描述数据库内部的表结构, 并利用双向工程技术来从现有的数据库或数据定义语言DDL中抽取数据库表结构并建立模型, 或是根据定义好的数据模型在数据库中自动产生相应的表结构, 这种技术称为数据建模(Data Modeling)技术。

所以软件开发的整个生命周期(包括业务建模、用例建模、应用建模、数据建模)都可以用可视化建模技术统一起来。在传统的开发技术中, 这些步骤是由不同的技术完成的, 如业务模型是由IDEF语言来描述, 分析设计由数据流图来表示, 数据库结构是用ER来定义等等。在可视化建模技术中, 所有的这些开发活动都可以由同一种语言UML来描述, 这样可以大大增强团队的沟通, 提高开发效率和软件质量。





Rational. software