

Rational Unified Process - 更高流程成熟 度的实现

法国 Q-Labs , Annie Kuntzmann-Combelles
加拿大 Rational Software , Philippe Kruchten

Rational Software 和 Q-Labs 白皮书

TP 178, 2/01

目录

摘要1
您是否足够高效？1
改进指南2
组织需要什么来启动 SPI 计划？2
CMM 模型 [6]2
IDEAL 模型 [4]2
流程2
Rational Unified Process3
ISO 文档3
使用 RUP 开发 SPI3
RUP 符合项目级别的 CMM 需求4
RUP 符合组织级别的 CMM 需求10
语用学：如何开始使用 RUP13
总结16
作者的联系信息17
参考资料17

摘要

本白皮书突出以下两点内容：一个成熟的组织（第 3 级开发单位）必须展示的关键概念，以及 Rational Unified Process（RUP）组件是如何满足这些需求的。

提出了高项目成熟度和组织成熟度。此外，第 4 部分还提供了一些关于 RUP 入门的好主意和早期采用者在各种环境中发现的一些主要优点的报告。

您是否足够高效？

现今，公司必须不断地寻找方法以达到更高的业绩。未来的挑战是在全球范围内的竞争。在逐鹿全球市场时，以下这些普遍迹象正变得愈加明显：

- 性能需求：在验证阶段发现过多的错误（bug），为了应付交付目标而将不稳定的产品版本投入市场
- 效率需求：项目超出预算和计划
- 失去市场份额：竞争对手的表现更好且交付的产品质量更高
- 缺乏称职的人力资源：人员更替率过高和 / 或雇佣新的工程师极其困难
- 需要以最小的风险集成不断发展的技术：支持产品的技术必须跟随趋势；团队不一定有足够的经验能够胜任这项挑战

一般而言，非常有必要对所开发产品的每个方面进行优化。因此，假设产品的开发流程对结果有着很深的影响（这一点已被广泛接受），那么最通常的做法之一就是提出改进产品的软件流程。

面临此类境况的组织或许能够通过制定以下解决方案取得成功。

- 解决方案的第一部分是了解软件的市场价值。如果软件始终是产品和项目的关键竞争优势，那么必须非常仔细地定义、记录并随着时间优化确定和实施它们的流程。我们观察到高级管理层仍然普遍深信软件会产生问题且出现太晚；尚未对软件执行附加值分析。
- 解决方案的第二部分是投入足够的资源用于解决流程中的主要问题：
 - 评估流程以确定改进的机会。
 - 不要进行重复劳动：而应当学习他人的经验以加快实现新的成果。 □ 对于任何项目，应同时从管理和技术角度进行考虑。
 - 管理今后需要的技能：学习、分享并成长。

有多种方式可以将组织转换成高产的团队；最佳的决策是改变您可以改变的，接受您不能改变的，借鉴任何您可以借鉴、已经过成功的软件社区验证的成果。例如，能力成熟度模型（CMM）[6] 和 Rational Unified Process（RUP）[11] 就是两个被广泛接受的强大工具，它们使您能够更快地制定出适合于您产品的开发流程。我们使用“工具”这一术语，这是因为当这两个组件是变更和改进软件实践的有效支持时，它们使您能够为您的组织环境定义最佳实践。。

下一部分将描述这两个组件（CMM 和 RUP）以及在改进流程和交付的产品时必须考虑的一些其他关键组件。

改进指南

在最近十年间，软件社区开始意识到流程，并认识到不能脱离生产产品的流程而单独评估软件产品。因此，许多组织创建了计划以改进他们的软件开发流程，这称为“软件流程改进（SPI）计划”。在多数情况下，软件开发集中于组织内的一个或多个单位或部门，并且被视为一个独立的规程。如果有 SPI 计划，情况就不同了，软件团队将与组织的其他规程更好的集成在一起。

组织需要什么来启动 SPI 计划？

- 一个启动该计划的激励条件和一个协调改进活动的关键人物；换言之，即建立一个具有远景描述和目标的改进项目。
- 一个参考模型，根据它评估项目和组织的实践并确定是否缺少什么部分而可能阻碍成功。美国软件工程研究院（SEI）开发的 CMM 模型是数千个组织使用的实际参考（标准），而 SPICE 框架（ISO 15504）[9] 则是它的取代者，可能很快就会成为官方标准，CMM 将符合该标准。
- 管理改进计划、使其成功结束的方法：SEI 中定义的 IDEAL [4] 框架已经证明了它的强大性。
- 适用于所有项目、使其成功的开发流程：RUP 就是一个现成的解决方案。

CMM 模型 [6]

美国软件工程研究院（SEI）的功能成熟度模型（CMM）是一个描述有效软件流程 [6] 中元素的框架。CMM 描述了一种演进式的改进方式，即从特殊、不成熟的流程演变为成熟、规范化的流程。它提出了多组涉及许多关键流程领域的推荐实践，这些实践证明能够增强软件开发和维护能力。CMM 可指导开发人员如何控制他们的开发和维护流程，以及如何逐步培养软件设计和管理优点。

IDEAL 模型 [4]

软件流程改进是一个系统性、协作性和长期的方法，用于对软件的组织和执行方式进行演进。改进方法中包括 IDEAL 方法，它是由 SEI 为 SPI 定义的一种集成方法。IDEAL 确定了五个阶段：初始、诊断、建立、操作和利用。每个阶段都以一个特定的活动为中心：

- 指定将要实现或支持的业务目的和目标（初始）。
- 根据相关标准或参考模型确定组织的当前状态（诊断）。 □ 开发计划以实施所选的方法（建立）。
- 集合所有可用资源（例如已有的工具、流程、知识和技能）以创建一个特定于组织需要的“最佳猜想”解决方案，并将该解决方案付诸实施（操作）。

总结从用于实施 IDEAL 的流程获取的经验（利用）。

流程

有些组织没有流程；有些组织的流程基于他们的经验。有一些现成的流程可用，比如 OPEN 和 Rational Unified Process（RUP）。OMG Group 可能很快会提出一个通用流程模型。术语“流程”在软件工程社区中广泛应用，但是关于软件流程组件的设想却千差万别。让我们看一下 SEI 给出的定义：人们用来开发和维护软件及相关产品的

一组活动、方法、实践和变换。流程是确保项目与其定义的目标相符的关键元素。

此定义中包括活动、技术（即方法和工具）以及人员，注意到这一点很重要。这三个组件的重要性是相同的。

Rational Unified Process

RUP 是一个从长期的项目经验中获益的预存在流程框架示例 [10, 11]。RUP 强调尽早确定高风险区域，方法是快速开发出一个定义系统体系结构的初始版本。它不在项目的先启阶段采用的固定的一组严格需求，但允许您随着项目的演进不断优化需求。它期望并采纳变更。流程的重点并不在于文档或“仪式”，并且它使自己的许多与软件开发相关的繁重任务自动化。主要的重点仍然是软件产品本身及其质量，这是根据最终用户对软件产品的满意程度，以及该产品的投资收益率与目标之间的符合程度共同衡量的。

它具有足够的普遍性，可以定制成各种软件产品和项目（在规模和应用两个方面），它围绕三个领域：人员、流程和工具或方法 [10, 11]。

ISO 文档

最后，为了完成该全景，我们必须提到 ISO 文档，例如 ISO 9001, ISO 12207, ISO 15504 (SPICE) [9]，使用这些提供的参考，组织可以与其他组织比较他们的软件开发情况如何。它们形成了通用框架，它与先前提到的其他组件兼容。

ISO 15504（也称为 SPICE）是另一个用于分析软件流程的参考模型。它假定许多已完全实现的评估模型（CMM 就是其中之一）和许多的评估方法（由 SEI 定义的方法就是其中之一）能够与 ISO 15504 的标准化部分相映射。

在该白皮书的剩余部分，我们集中讨论 CMM 这一软件社区的实际标准。CMM 是一个完全实现的模型，它支持数百个软件流程改进计划。本文中讨论的任何内容都与 SPICE 兼容。

使用 RUP 开发 SPI ...

考虑到上一部分列出的关键元素 — CMM 参考模型、管理改进计划结束的方法（IDEAL）以及用于增强竞争力和迎接软件的全球挑战的自适应开发流程 - 现在我们将 RUP 概念映射到 CMM 需求并考虑 RUP 在实现特定的软件能力方面的潜力。

软件能力实际上对一个组织来说意味着以下任何一个方面：

- ☐ 在恰当的时候为后续项目做出正确的决策
- ☐ 业务的延续和生存
- ☐ 冒风险以及控制项目输出
- ☐ 获得良好的产品质量
- ☐ 在产品中集成最先进的软件技术

软件能力是以上所述内容的精密混合，但是有一些概念必须作为软件能力的一部分加以考虑：

□ 在项目级别上：

- **生命周期问题**：项目的生命周期不仅限于产品的实现和维护。有一个管理透视图用于处理业务、财务和项目策略。
- **流程选择和定制**：现成的和 / 或反映以往经验的标准流程并不能满足各种各样的项目目标。效率是通过为特定项目定义合适的开发流程取得的。
- **风险管理**：任何项目都面临风险。一个有能力的组织能预测风险并作出决定，以便重新配置项目而将出现风险的影响降至最低。
- **度量**：软件组织长期发展的一个关键是收集历史数据以便对软件质量和生产力进行分析。为每个项目收集某个历史数据是一项长期工作。关于工时、进度、程序规模和已实施的功能，以及缺陷数量的数据为计划未来的项目和提高可预测性奠定了坚实的基础。预测性能是成熟度的一个指标。

□ 在组织级别上 - 视为由项目组成的集合：

- **流程改进**：能力表示组织能够从过去（特别是从其他组织所犯的错误的）获取教训，并且能够获取的教训转化到流程演进中去。仅当定义了用于量化改进的度量值时才完成了改进迭代周期。

□ **资源和技能**：资源和技能：组织的能力与应用流程和开发产品的人员的能力紧密相关。

RUP 在项目级别上符合 CMM 需求

CMM 描述了达到特定的成熟级别的组织中项目的状态。

第 2 级组织中的项目已经安装了基本的软件管理控件。现实的项目的落实是基于对先前项目的观察结果和现有项目的需求。软件项目经理跟踪项目成本、进度和功能；一旦在落实中出现问题就可以立即确定。对软件需求以及为满足这些需求而开发的工作产品都设立了基线，它们的完整性受到控制。定义了软件项目标准之后，组织应确保切实遵守这些标准。软件项目与转包商（如果有）协同工作以建立强大的客户 - 供应商关系。

在已定义级别（第 3 级）中，组织中用于开发和维护软件的标准流程已被记录下来。项目定制组织的标准软件流程以开发他们自己定义的软件流程，该流程考虑到了项目的特点。这个定制的流程在 CMM 中称为项目定义的软件流程。定义的软件流程包含一组连贯、集成且经过良好定义的软件工程和管理流程。一个经过良好定义的流程的特点是包含准备条件、输入、用于执行工作的标准和过程、验证机制（例如同级复审）、输出，以及完成标准。因为已对软件流程进行了良好的定义，所以管理就可以深入了解所有项目的技术进度。

生命周期问题

在 CMM 中，生命周期问题包含在多个关键流程领域（KPA）中。在组织中第一次提出 CMM 概念时，人们通常会错误地理解术语“软件流程”，将它与“软件生命周期”混为一谈。项目生命周期中包含与开发问题同样多的业务和财务问题。第 2 级关键流程领域（KPA）以及一部分第 3 级关键流程领域（KPA）与项目生命周期有关。它们是：

- 需求管理目标 1：分配给软件的系统需求受到控制以建立用于软件工程和管理基线。
- 软件项目规划目标 1：软件估计被记录下来以用于对软件项目进行规划和跟踪。
- 软件项目规划目标 2：软件项目的活动和落实已规划并记录。

- 软件项目跟踪目标 1：根据软件计划跟踪实际的结果和性能。
- 软件项目跟踪目标 2：对软件落实的变更征得了受影响团队和个人的同意。
- 软件配置管理目标 1：选择的软件工作产品经过确定、受到控制且可用。
- 软件配置管理目标 2：对已确定的软件工作产品进行变更受到控制。
- 软件配置管理目标 3：受影响的团队和个人小组能够获致软件基线的内容和状态。
- 软件产品工程目标 1：软件工程任务已定义、集成并始终如一地执行以产生软件。
- 软件产品工程目标 2：多个软件产品保持一致。

换言之，CMM 需要：

隐式地定义一个基线以考虑将要开发的应用程序所在的业务环境，为项目需求设定优先级并为项目的功能性和非功能性内容作出所有必要的决定。

项目计划已经建立，并且需要设置的活动、角色与职责、里程碑和工件是真实可靠的。项目计划将风险考虑在内。

项目计划受到监控，一旦功能内容或项目环境 / 组织发生大的偏差，项目团队能够作出反映。

能够确定和分析项目中的任何变更。作出进一步的决定并在基线和项目计划中报告。

选择并应用方法和工具以确保项目达到最佳性能。

Rational Unified Process 在先启阶段的主要任务中实现以下需求：

- 获得项目远景并通过匹配上一列表中的第 1 项来开发业务模型。
- 组织和计划项目
- 估计潜在风险。（这一项与上个列表中的第 2 项有关。）
- 部署统一变更管理（UCM）策略，它与先前列表中的第 3 和第 4 项相似。

先启阶段最重要的目标是实现在项目的生命周期目标中的所有项目干系人的并发。先启阶段主要对于新开发工作很重要；在该阶段中，有一些必须确定的重大业务和需求风险，然后才能继续进行项目。对于专注于增强现有系统的项目，先启阶段较简单一些，但确保项目值得进行并且可以进行仍然是重点。

在先启阶段的末尾，将对项目的生命周期目标进行检验，并决定是继续该项目还是取消它。

必须复审的工件是：

- 项目远景
- 业务案例
- 风险列表（请参阅“风险管理”部分）
- 软件开发计划

- ☐ 迭代计划
- ☐ 开发案例（请参阅“流程选择与定制”部分）

这些工件满足了在这一部分开头列出的“CMM 关键流程领域”的几个目标。

获取项目远景和开发业务模型

业务模型从业务工作者的内部观点定义业务用例。该模型定义了业务中工作的人员以及他们是如何处理和使用“业务的类和对象”- 应该如何彼此相关（无论静态还是动态）才能产生预期的结果。此模型还强调在业务区域中执行的角色以及他们的现行职责。同时，该模型的类的对象需能够执行所有的业务用例。

根据业务模型，Rational Unified Process 确定了一个称为“开发远景”的活动，该活动的目的是：

- ☐ 就需要解决的问题取得共识。
- ☐ 确定系统的项目干系人。
- ☐ 确定系统的边界。
- ☐ 描述系统的基本功能。

组织并计划项目

正如软件流程受到项目特征的影响，项目组织也是如此。此处提供的缺省结构（在下图 1 中阐述的）必须经过修改才能反映下列因素的影响：

- ☐ 业务环境
- ☐ 软件开发工作的规模
- ☐ 新颖度
- ☐ 应用程序类型
- ☐ 当前开发流程
- ☐ 组织因素
- ☐ 技术和管理的复杂性

这些因素在 RUP 中作为影响项目结构选择的流程判别式来考虑。项目结构主要根据以下两点来确定：

- ☐ 每次迭代的时间长度
- ☐ 迭代次数

迭代是相当完整的小项目，它经历所有主要的工作流，在多数情况下生成一个可执行但尚不完整的系统（称为发行版）。

确定迭代次数，根据风险、规模和复杂性，可能会有许多变化。

如果产品针对于某一全新的领域，您可能需要在先启阶段中增加一些迭代以巩固各种概念，向具有代表性的客户或最终用户显示各种模型，或对客户的标书作出可靠的答复。

如果产品很大且复杂，并且开发时间较长，则需要在构造阶段规划进行三次或更多的迭代。

在项目生命期内，组织将不断发展以支持项目计划中所包括的工作细分结构。这展示在图 1，[7] 中。

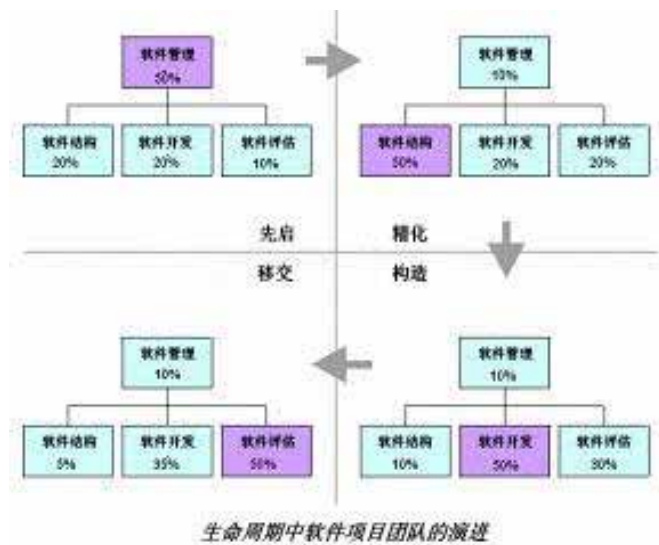


图 1：软件项目团队评估

这种演进在每个阶段都强调一组不同的活动：

- 先启团队：一个专注于进行规划的组织，得到其他团队的充分支持以确保计划代表了各方面的一致意见。
- 精化团队：一个专注于体系结构的组织，其中项目的驱动力量存在于软件体系结构团队中，并在必要时由软件开发团队和软件评估团队提供支持以实现稳定的体系结构基线。
- 构造团队：一个平衡的组织，其中多数活动存在于软件开发团队和软件评估团队中。
- 移交团队：一个专注于客户的组织，在其中，客户的使用反馈推动了部署活动。

此演进期内的团队间迁移确保了知识和能力得以保留在项目中。

部署统一变更管理（UCM）策略

在软件开发中，统一变更管理（UCM）是 Rational Software 用于管理变更（从需求转化为发行版的过程中）的方法。UCM 跨越整个开发生命周期，它定义了如何管理对需求、设计模型、文档、组件、测试用例和源代码所作的变更。

UCM 模型的一个关键方面是：它将用于规划和跟踪项目进度的活动和正在进行变更的工件进行了统一。UCM 模型是通过流程和工具这两者来实现的。Rational ClearCase® 和 Rational ClearQuest® 是 UCM 的基础技术。ClearCase 可管理由软件项目生成的所有工件（包括系统工件和项目管理工作件）。ClearQuest 可管理项目的任务、缺陷和改进请求（通常称为活动），并可提供跟踪项目进度所必需的图表和报告工具。

流程选择和定制

成熟级别 3 包含三个针对整个组织的 KPA 以及四个针对项目组织、管理和工程的 KPA。管理和工程活动的软件流程都被记录、

标准化并集成成为组织范围内的软件流程。所有的项目都使用同一个经记录并核准的组织流程版本来开发和维护软件。组织流程重点（OPF），组织流程定义（OPD）以及训练程序（TP）KPA_s 确保组织确定项目的最佳级别 2 并将它们作为组织的标准来记录。他们还关注基于项目所确定的需要的技能管理、从过去项目中吸取的经验教训以及项目未来的任务和远景。然而，如果为特定项目的成功完成、管理风险和改进性能采用标准流程时没有任何指南，成熟度就不能够满足第 3 级需求。

因此，对于大多数的前导审核员和 CMM 社区，第 3 级由集成软件管理（ISM）KPA 来刻画特征，它的目的是将软件工程和管理活动集成为一致的、已定义的软件流程。该已定义的流程是标准流程的最适合的定制，有助于满足客户需要和约束、市场需要以及业务策略。

ISM 由 CMM 模型中的两个目标定义。第一个涉及定制，而第二个则需要项目管理。ISM 中的活动 10：“项目的软件风险是根据已记录的过程确定、评估、记录和管理”的是第 3 级组织的第二个关键指标。风险管理在第 2 级的组织中是必需的，但是第 3 级的组织已经能够非常成功地预测风险并能够展示企业决策系统。以下的“风险管理”部分详细描述这些方面。

在 RUP 中，“开发案例”和“环境工作流”的概念满足了这些需求。开发案例是 Rational Unified Process 产品的一个静态配置；换言之，它是针对特定项目、产品和组织定制的软件工程业务流程。开发案例的重点在于做什么和怎么做，它还概述了要遵循的流程，每个人都可以了解它。

Rational Unified Process 列出了流程的组成部分，可以在给定的开发案例中对这些组成部分进行修改、定制、添加或禁止。

- **核心流程工作流** - 软件项目很少完全跳过某个核心流程工作流，例如“分析与涉及”、“实施”等。
- **工件** - 项目更有可能因它们必须生成、更新和交付的工件而异。
- **活动** - 至少有两个原因可能使活动不同：使用工件作为输入和生产或更新工件作为输出的活动将会受到这些工件修改的影响。尤其是如果不再需要某个工件或工件的某个信息元素，可能要取消相应的步骤或做重大修改。也可对活动进行修改以引入属于特定应用领域或开发专业的特定技术、方法和工具，例如设计步骤、编程语言、自动代码生成工具、度量技术等等。

流程工程师负责配置流程、决定开发流程的外观，在开发组织（团队、项目或公司）中“安装”开发案例，并培训开发人员如何使用它。

一旦流程工程师设置了开发案例，项目经理就针对给定的项目对其进行实例化并执行。这通常被称为流程制定。

随着流程的展开，人们会在流程本身进行期间获取经验，然后流程工程师就使用获取的经验作为反馈来改进流程。

风险管理

在 CMM 中广泛涉及风险管理，主要是在第 2 级中和更深的第 3 级中。列出了特定的风险活动以满足关键流程领域软件项目计划、软件项目跟踪和监管以及集成软件管理的目标。CMM 需求遵循许多组织中观察到的实践：在第 2 级的组织中，项目通常定期确定并评估风险。不过也有很低级的预活动。在第 2 级上，风险管理尚未普及 / 受到理解，因此项目的后期和早期的风险确定就会有问題。

相反，在第 3 级的组织中，会对风险进行确定、评估并减轻，并且团队具有很好的预测能力。在最高级的团队中，有大量措施用于决策。

我们在基于 CMM 的评估和软件流程改进方面的长期经验，使得我们对软件单位在管理风险方面面临的困难有着更清楚的认识。尽管对项目的控制和度量投入了所有的精力，这仍然是一个弱项。

使用 RUP 和迭代方法非常有助于成功地管理项目风险。整个 Rational Unified Process 是由风险管理驱动的。

风险驱动迭代计划；迭代是围绕解决特定风险而规划的，具体方法是尝试限制或减轻风险。会对风险列表进行定期复审以评估风险减轻策略的有效性；反过来，风险减轻策略又会驱动对项目计划和随后的迭代计划进行修改。

管理风险的关键不是等到风险成为事实（并成为问题或故障）后才决定如何处理。就像在洲际飞行中失之毫厘则谬以千里一样，较早地管理风险几乎总是比事后清理代价更低，麻烦更小。

这里确定了四类风险：

- 资源风险
 - 组织
 - 拨款
 - 人员
 - 时间
- 业务风险
- 技术风险
 - 范围风险
 - 技术风险
 - 外部依赖性风险
- 进度风险

风险管理计划和风险列表是由 RUP 的“项目管理”工作流确定的两个工件。它们建议您确定并评估风险。

度量

度量是 CMM 的一个常用功能。它是第 2 级至第 5 级中的每个关键流程领域都具有的一个常用功能，用于指示将实践制度化的时间。度量非常重要，以至于在 SW-CMM 的新版本（即嵌入 CMMI 框架中的版本，“集成 CMM 系统工程 + 软件工程”）中，每个关键流程领域（KPA）都有一个基于度量的目标。

CMM 的关键是度量流程以确定它们在第 2 级中的充分性和在第 3 级中的有效性。从报告度量值转换到对度量值进行分析和操作通常是困难的，但是有一点很清楚，那就是项目越来越成熟。mature。从数量上管理项目的一些较小的早期成功，促使了对度量的接受和理解。

除了这些一般需求之外，某些 KPA 还对诸如软件项目计划、软件项目跟踪和监管以及集成软件管理等度量有特定的需求。它们与实现项目估计和项目控制的项目数据有关。

RUP 提供使用度量值的指南。度量计划和度量工件必须在先启阶段生成。the Inception phase.

项目度量工件存储着项目的度量数据。一旦进行了度量或度量可用，就会对该数据进行更新。它还包含从原始数据计算得来的派生度量值，并且还存储如何获取派生度量值的过程和算法等信息。关于项目状态的报告 - 例如达到目标（功能、质量等等）的过程、支出以及其他的资源消耗 -

是项目度量生成的。通过使用 Rational Project Dashboard 方法，可以频繁甚至几乎持续地显示项目状态；例如，自动软件数据收集代理程序提供项目状态实时显示。

Rational 文档编辑了一组初始度量值，它们相当简单，新手可以藉此入门。项目某些方面的度量值包括：

- ☐ 进度的规模和复杂度
- ☐ 需求或实施、规模或复杂性方面的变更率稳定性 ☐ 变更范围的模块性
- ☐ 错误数目和类型方面的质量
- ☐ 错误频率的成熟度
- ☐ 项目支出的资源与计划支出的对比

RUP 符合组织级别的 CMM 需求

第 2 级组织的软件流程能力可以总结为已规范化，因为软件项目的计划和跟踪很稳定，且可以重复早期的成功。通过遵循基于先前项目执行的现实计划，项目的流程处于项目管理系统的有效控制之下。

在已定义级别（第 3 级）上，组织中用于开发和维护软件的标准流程被记录下来，这包括软件工程和管理流程两个方面，然后这些流程被集成为一个统一的标准。该标准流程在 CMM 中称为组织的标准软件流程。在适当的时候，将使用并更改在第 3 级上建立的流程，以帮助软件管理者和技术人员更有效地工作。在标准化组织的软件流程时，组织将开发有效的软件工程实践。有一个负责组织的软件流程活动的团体，例如，软件工程或 SEPG。进行一次组织范围内的培训活动以确保员工和管理者具有履行各自角色所必需的知识和技能。

第 3 级组织的软件流程能力可以概括为标准且一致的，因为软件工程和管理活动都是稳定且可重复的。在已确定的产品线内，成本、进度和功能都处于控制之下，且软件质量受到跟踪。该流程能力基于对已定义的软件流程中的活动、角色及职责的普遍和组织范围的理解。

流程改进

能力表示一个组织能够从过去（特别是从错误中）和其他组织那里吸取经验教训，并能够将它们转化到流程演进中去。只有定义了用于量化改进的度量时，才完成了改进迭代周期。

当组织的成熟度提高时，标准流程会发生改变。从第 2 级到达第 3 级表示项目内的所有优秀实践都已制度化，并且有一个相应的评估流程将帮助确定项目范围内的最佳实践，这些最佳实践将被记录在组织标准软件流程（OSSP）中。这些是组织流程重点（OPF）和关键流程领域（KPA）的需求。然后，将根据从组织内项目获取的经验教训来改进 OSSP。由于项目使用相同的标准，因此很容易增补吸取的经验和教训的存储库，且 OSSP 本身也可以从各种经验中获益。

RUP 环境工作流开发了一个类似的方法。“在组织中实施流程”概念说明了如何在开发组织内以组织级别实施流程和工具。

可以使用 RUP 的先启、精化、构造和移交这四个阶段来描述如何在软件开发组织中实施新流程。

在组织中实施流程和工具的步骤

CMM 的 OSSP 是组织中的新流程，它的定义可以遵循 RUP 的四个阶段。

关键是决定是否要开发一个组织范围内的开发环境，使每个软件开发项目通过必要的修改都可以使用该环境，但是必须要有一定的成熟度。

如果您决定开发一个组织范围内的环境，您必须启动一个项目以开发组织的开发环境。如果您决定启动这样一个项目，必须清楚此项目团队与软件开发项目团队的合作将会很紧密。RUP 还建议您把它看作一个特殊的项目。这里又实现了 CMM 需求。

流程实施项目分成许多阶段，每个阶段中都执行所有四个步骤，直到项目就绪且流程和工具已部署完成，整个组织可以成功地使用它们为止。

一个流程实施项目可以分成多个阶段

四个阶段为：

- 阶段 1：将流程实施项目出售给赞助商。 □ 阶段 2：处理主要风险。
- 阶段 3：完成所有工作 - 开发案例的模板、指南和示例均已就绪，且培训课程已安排妥当。
- 阶段 4：将它部署到整个组织中。

软件这些阶段针对使用 Rational Unified Process 的软件开发项目，则它们可以分别称为先启、精化、构造和移交阶段。

每当为了增强性能(基于特定项目中获取的经验和教训或根据特定技术)而必须使标准流程演进时，都将定义相同的项目阶段。

RUP 还定义了“管理组织变更”的概念，这是软件流程改进的总背景。对成功实施流程变更的建议如下：

- 以各种级别确定组织内的变更代理。
- 以小的、合理的并且可度量的步骤计划变更。
- 使用适当的基态语言将变更传达到组织级别。

这些建议与 IDEAL 方法给出的建议类似，它们受到了该方法的启发。

最后，RUP 定义了一个特别的角色称为“导师”(Mentor)。导师的工作就是对项目团队进行讲授和指导，让他们了解需要什么以及何时需要。典型的指导方式有：

□ 研讨会领导者

有些活动最好在小组中进行；例如，在用例建模过程中查找参与者和用例。在此类活动的整个过程中，有必要让一位流程专家来担任建模领导者。

□ 流程专家

流程专家是项目的现场支持人员。流程专家的任务是帮助开发人员尽可能有效地使用流程和模型。

□ 项目经理支持

流程专家可以帮助项目经理计划和控制项目。有时，项目经理对所涉及的流程只有少许经验或者毫无经验。

□ 复审员

一种节省成本的知识传授方法就是让流程专家对每个阶段的结果进行复审。流程复审员还在复审由项目执行的任何流程修改时带来价值。

在 CMM 中，导师是否与 SEPG（软件工程流程小组）扮演不同的角色？不是，他们是相同类型的角色。

资源和技能

在组织级别，CMM 确定了一个“培训计划”关键流程领域，最好称为“技能管理”因为在此处，后者才是真正相关的问题。一个成熟的组织必须确定并计划中长期的技能需求，并对它们进行管理，以便在需要的时候可以使用它们。

培训计划的目的是拓展个人的技能和知识，以便他们可以有效且高效地执行他们的角色。培训是组织的职责，但是当项目的需求比较特别时，软件项目需要确定必要的技能并提供必要的培训。RUP 中的管理工作流在为项目配备人员时会考虑这些问题。

关键流程领域培训计划有三个目标：

- 目标 1：规划培训活动。
- 目标 2：为拓展执行软件管理和技术角色所必需的技能 and 知识提供培训。
- 目标 3：软件工程小组和软件相关小组中的成员接受执行他们的角色所必需的培训。

RUP 的多个方面满足这些需求。准确地确定角色并定义其能力。角色可由个人或作为团队协作工作的一组人实现。项目团队成员通常扮演许多不同的角色，就像一个人可以戴许多帽子，一个人也可以扮演许多不同的角色。

角色不是指个人，而是说明个人在企业中应该有怎样的行为以及这些个人应该承担什么职责。

虽然大多数角色由组织内的人员来实现，但是开发组织外部的人员也扮演重要的角色，例如，所开发的项目或产品的干系人。

每种类型的角色由团队成员应该具备的技能和知识仔细定义。根据这些定义，组织能够在培训中派生出缺少的必需技能和特点，使这些角色可以有效地工作。

RUP 指南构成了一组很好的规则，以提供作为培训。课程开发人员是所列出的角色类别中的一个。

RUP 唯一没有完全涵盖，却是 CMM 所需的方面是为将来发展新技能而确定、计划、交付培训。设想一下，往后的一年中，组织的战略是将活动集中于电子商务，该目标必须进一步分解，并且可能会影响到所执行开发的类型以及所使用环境的类型。团队必须为此变更做好准备，并且必须定义、开发或从外部获得必要的培训。第 3 级的培训关键流程领域需要仔细管理、跟踪并记录这些问题。

- 本白皮书的第三部分提供了一部分成熟度概要文件与 RUP 功能的详细映射。此白皮书系统地比较了 CMM 应用和 RUP 相应的指南的每个目标，我们有意不这样做而是选择关注高成熟度的关键标志。
- 在 RUP 内部一方面通过测试活动另一方面通过开发质量保证计划工作流来处理软件质量。然而我们的表现突出了我们对高成熟度的真正含义的经验和观察：第 3 级组织与和第 1 级或准第 2 级组织的不同之处。

下一部分针对这些组合技术的早期采用者的一些反馈。

语用学：如何开始使用 RUP

最常提的问题是：以前是否有人这样做过？

当向管理员和工作人员提供先前的概念时，将可能不只一次被问这个问题。变得更加成熟、更加面向流程、管理风险并获得成功对于任何软件公司都具有吸引力。然而，在环境不断变迁，需求迅速变化，且团队需要应用他们未受过培训的技术的情况下，这是否可能呢？

Rational 客户库中包含了 Computing Devices International 的经验：Computing Devices 是一家领先的国防电子和信息公司。几乎在每个行业中，信息始终是关键的竞争优势。对于 Computing Devices International，它不仅是信息 - 他们的任务是提供端到端的解决方案，该解决方案随时随地根据请求提供至关重要的信息。

他们需要确保可靠、按时地交付高质量的解决方案。他们经验的主要优点包括以下几方面：

- 通过迭代设计方法加快开发流程
- 将开发时间从 3 年减少到 19 个月
- 确保可靠及时交付的能力
- 客户满意度提高
- 开发成本降低 33%

Computing Devices 为其客户开发的精密系统高度依赖于为最迫切的需求所开发的软件。Computing Devices 首先将注意力集中于编写对安全至关重要的软件。

以前，根据结构化的设计或“瀑布”方法，组织内的不同业务单元使用不同的工具。结果，开发流程很慢且没有一致的方法。这将导致错过截止日期、高成本及软件质量始终不能满足他们的高标准。Computing Devices 意识到他们需要立即采取行动。

Skandia-IT 是 Rational 资料中的另一个示例。该组织在短短 12 个月的时间内，按时或提前交付了九个大型的保险系统。

他们通过使用先进的开发流程吸引开发人员和顾问加快新人才的聘用，通过用例来获取迫切的需求、模型系统和直接开发来确保客户满意度，并通过合并旧系统并将它们作为组件使用来迅速开发出灵活的三层体系结构。。这是 Skandia-IT 遇到的一些挑战。

Skandia-IT 的经理解释道：“我们改变整个流程有多方面的原因。最重要的是我们重新组织了整个业务。我们在改变开发流程的同时，将面向产品的组织转变为面向客户的组织。我们的新系统使客户可以通过电话或万维网自己处理许多事情。我们在 Skandia 的员工可以集中更多的时间变成保险专家，而不是学习如何使用系统。节省下来的时间可以投入到客户关系上。”

Skandia-IT 已经在新流程的安装上投入很多。这项投入已经获得收益。更好的质量、可预测的交付时间和副作用，并且可以更容易地聘请到有经验的项目领导者和开发人员。

还有其他关于公司关注于流程并开始使用 RUP 加速部署的例子。Q-Labs 与 Lysis 合作，后者是一家发展迅速，从事因特网广播软件开发的公司。在一开始的环境中，侧重于高级别的流程成熟度的流程技术和方法可能并不合适。该类型的组织主要是快节奏的、活跃的和具有创新性的，对速度的需求与最大程度地进行流程控制之间的反差产生了以下问题：流程主题能够多大程度地包容观点多样性？

- 此类公司的特点是：
- 它们是迅速发展的软件单位。它们的开发速度很快（少于六个月）。
- 它们使用领先的对象技术交付更好的产品。 □ 它们适应动态的市场。

此环境和其它软件行业没有实际的不同（除了时间刻度完全不同），可预测性与生存和灭亡等价，团队必须迅速改变。几乎没有学习的时间，最佳的选择是从现成的经验，经过良好验证的实践入手。在 Lysis 采用 CMM 指导他们的成长并采用 RUP 作为安装流程的背景时，的确就是这么做的。

毫无疑问需要定义软件实践。它提供了了解、沟通、执行和管理。它还形成了阐明与其他组织的关系（无论是内部还是外部）的基础。在 Lysis 的案例中，目标是与市场营销和销售以及与潜在的软件转包商的关系。已定义的流程必须保持灵活：在人员、基础结构、发行时间表和采用的技术方面可能发生迅速的变化。开发流程需要容纳它们。必须迅速及时地作出响应。建议使用迭代周期短的迭代流程、面向风险的方法以及结构化管理。能够在生产线的所有组件中重复流程是必需的。流程定义有助于使随公司发展而新加入的员工了解信息。

在 CMM 中，流程定义（成熟度级别 3）遵循流程可重复性（成熟度级别 2），但是在使用像 RUP 这样的标准流程开始的公司中，这对于促进并建立可重复性从而获得重大收益很有帮助。

Lysis 使用的方法从按照 CMM 需求评估开发实践入手。这使得编程、目标技术和工具环境最感兴趣的团队作出关于流程的好的“买入”。基于这些观察，建立了一个具有严格时间限制的操作计划用于定义流程。由顶级管理为支持 SPI 而设置的业务目标包括：

- 缩短开发流程以抢占目标市场。 □ 加速从项目模式向产品模式的转换
- 管理公司的成长
- 使人员满意
- 保持启动反应
- 提高可预测性和可见性（对于 IPO，这是必需的季度结果） □ 量化产品质量的改进
- 管理成本和潜在利润
- 优化客户满意度
- 确定并管理与合作者的联系（对于转包部分的工作）

SPI 计划本身已经使用 RUP 框架建立为一个项目，其中：

- 先启 = 评估
- 精化 = 项目计划 + 基于 Web 存储库定义和定型
- 构造 = 已定义的流程 + 实施和工具使用
- 移交 = 试点项目 + 部署

工作包是针对 CMM 关键流程领域（KPA）而定义的，它们中的每一个都包含目标、要执行的活动、可交付件和复审流程。例如，软件计划的目标是：

操作的目标是：

- ☐ 记录软件计划的最新优秀实践
- ☐ 为项目定义一般组织集
- ☐ 为计划定义角色和职责
- ☐ 为项目级别定义落实流程
- ☐ 为一组标准的项目流程定义一个一般项目生命周期并定制指南
- ☐ 定义标准工作细分结构（WBS）
- ☐ 列出可交付件标准列表
- ☐ 为开发计划定义模板和指南
- ☐ 编写项目计划流程
- ☐ 为计划定义必需的技能 and 培训

选择从 RUP 入手将加快这些操作中的一部分并将确保组织从强大的经验中获益。虽然 SPI 计划尚未完成，但是已经可以观察到一些可见的成功因素：新产品的开发被更好地结构化；具有计划且定期复查。采取了纠正措施，并且软件团队外部的可见性已大大提高。

Rational Software 报告了一个与 E-corporation 有关的相似案例。E-corporation 是一家欧洲交互式架构设计公司，该公司在全球范围内指导公司走向数字业务的前沿。E-corporation 将策略、艺术和软件技能结合起来构思、创建并实施因特网和电子商务投资。

E-corporation 从部署 RUP 所获取的收益有：

- ☐ 在团队成员之间提供了更好的沟通，这提高了他们的效率。
- ☐ 更准确地满足了客户期望。
- ☐ 提供了一个用于满足高质量和高速度要求的框架。
- ☐ 减轻了迭代开发的风险，并使用了基于专家知识和行业认可的最佳实践的经过验证的流程。

E-corporation 从对所有项目使用一个共同流程所获得的最大收获在于可以有效地重用组件、经验和活动。当组织从各种项目中取得经验之后，就可以不断地对 Rational Unified Process 进行微调。由于遵循一致的结构和语言记录所有的活动，因此许多在先前项目中取得的经验可以带到新的项目中去。

E-corporation 计划在所有的项目中应用 Rational Unified Process 以进一步利用团队的统一性。Rational Unified Process 为其开发团队提供了应用经验证的规程的能力，从而使这些团队能够优化所有的工作、简化沟通并确保满足所有需求的高质量解决方案。

这些是各种组织如何观察到通过结合 Rational Unified Process 和 CMM 指南的优势。我们目前正在同其他的组织合作，这些组织将 RUP 作为通用结构应用于适合的组织流程资产中。

总结

我们对映射 RUP 和 CMM 所作的描述同样适用于 ISO 15504。这两个引用的底层概念是相似的，我们着重说明了成熟度概要文件支持 ISO 15504 的能力度量。

该白皮书阐明了 RUP 概念能够在多大程度上满足高成熟度组织的目标。因此，采用 RUP 是成功实现 CMM 需求的一个方法。

最近由 Telcordia Technologies 发布的一篇文章给出了一些业务上高成熟度的量化结果。Telcordia，最近被评为成熟度级别 5，报告称：

- ☐ 自从启动 SPI 计划，领域错误密度减少率达到 94%。
- ☐ 从 1995 起，超过 98% 的软件发行版是按时交付的。
- ☐ 测试一行代码的成本降低了 64%。
- ☐ Telcordia 的质量历程花费了六年的时间达到这一成熟度级别。

对于电子商务和电子商业领域的软件密集型公司，交付时间不允许花费五六年的时间来优化流程。因此，达到保持竞争力的所需成熟度的唯一可行方法是：

- ☐ 了解软件流程的价值
- ☐ 采用已有的框架，比如 RUP
- ☐ 使框架适应市场环境的特点
- ☐ 衡量它对业务的影响
- ☐ 学习并加强定制的流程模型

由于 RUP 模型是通用的，因此您不要迷失自己的方向，这一点很重要。因此，Rational 确定了“基本 RUP”，其中包含采用模型 [11, 12] 时需要重点考虑的方面。可以使用缺省的 RUP，但是一开始不需要实施所有内容。重要的功能为：

- ☐ 业务案例
- ☐ 进度安排
- ☐ 远景文档
- ☐ 风险
- ☐ 体系结构
- ☐ 变更请求以及如何处理缺陷
- ☐ 测试
- ☐ 软件产品本身
- ☐ 用户支持文档
- ☐ 项目评估

执行 RUP 本身并不是目标；但它有助于解决小组或团队中棘手的问题，并且有助于满足竞争目标。

作者的联系方式

Annie Kuntzmann-Combelles , Executive VP
Q-Labs France
28 Villa Baudran 94742 Arcueil cedex
电话 +33 (0)1 49 08 58 00
akc@objectif.fr

Philippe Kruchten, Rational Fellow
Rational Software Canada
pbk@rational.com

参考资料

- [1] *A Quantitative Approach to Software Management— The AMI Handbook* , K. Pulford , A. Kuntzmann-Combelles 和 S. Shirlaw , Addison Wesley Longman , 1995 年
- [2] Anchoring the Software Process ,Barry W. Boehm ,1996 年 , *IEEE Software* ,1996 年 7 月 ,第 73-82 页
- [3] *A Rational Development Process* , Philippe Kruchten , 1996 年 ,选自 *CrossTalk* , 9(7) , 1996 年 7 月 , 第 11-16 页
- [4] *IDEAL: A User's Guide for Software Process Improvement* , Robert McFeeley , 1996 年 , Software Engineering Institute , Pittsburgh , PA , CMU/SEI-96-HB-001
- [5] *Software Project Survival Guide*. Redmond , Steve McConnell , 选自 Microsoft Press , 华盛顿州 , 1997 年。
- [6] *Capability Maturity Model for Software, Version 1.1* , Mark Paulk 等 , Software Engineering Institute , 匹兹堡 , 1993 年 , PA SEI-93-TR-024
- [7] *Software Project Management: A Unified Framework* , Walker Royce , Addison Wesley Longman , 1998 年
- [8] Reaching CMM Levels 2 and 3 with the Rational Unified Process white paper ,Jas Madhur 等 ,Rational Software ,1998 年
- [9] *the Theory & Practice of Software Process Improvement* , Alec Dorling 等 , SPICE , IEEE Computer Society , 1999 年
- [10] *The Rational Unified Process—An Introduction, 2nd ed* , Philippe Kruchten , Addison Wesley Longman , 2000 年
- [11] Rational Unified Process, version 2000.02.10 , Rational Software Corporation
- [12] *The Ten Essentials of RUP , The Rational Edge* , Leslee Probasco , 2000 年 , 2000 年 12 月 , <http://www.therationaledge.com>.



两家总部：

Rational Software
18880 Homestead Road
Cupertino, CA 95014
电话：(408) 863-9900

Rational Software
20 Maguire Road
Lexington, MA 02421
电话：(781) 676-2400

免费电话：(800) 728-1212
电子邮件：info@rational.com
Web：www.rational.com
全球网址：www.rational.com/worldwide

Rational、Rational 徽标和 Rational Unified Process 是 Rational Software Corporation 在美国和 / 或其他国家或地区的注册商标。Microsoft、Microsoft Windows、Microsoft Visual Studio、Microsoft Word、Microsoft Project、Visual C++ 和 Visual Basic 是 Microsoft Corporation 的商标或注册商标。其他所有名称均仅用于标识目的，它们是其相应公司的商标或注册商标。ALL RIGHTS RESERVED.

Copyright 2006 Rational Software Corporation.
如有更改，恕不另行通知。