



在 CMMI 的规范下建立有效的需求管理

Telelogic 南中国区技术经理

吴 星



1. 介绍

关于本文

本文介绍了流程改进模型 - CMMI，着重描述 CMMI 对需求管理的要求，同时也提供了如何通过部署相关的工具使整个组织达到 CMMI 水平的要求。

什么是 CMMI？

软件工程学会（SEI），集成的能力成熟度模型（CMMI）是描述产品开发（包括系统工程和软件工程）的能力成熟度模型。SEI 把 CMM 描述成为包含一个或多个关键因素的有效流程，同时也描述了如何从杂乱的，不成熟的流程到规则的，成熟的具有更高质量和效率的流程。

CMMI 是对软件成熟度模型（SW-CMM），系统工程成熟度模型（SECM）和集成的产品开发成熟度模型（IPD-CMM）的最佳实践的建立和扩展。

难道流程改进不会耗费时间和金钱吗？它的回报是什么？

改进产品开发流程当然需要投资。但是，正确选择工具去支持这些流程能够加速流程实施和缩短产生回报的时间。企业运用 CMMI 或 CMMI 之前标准所收到的投资回报是有目共睹的。

在 2003 年 10 月份的报告中，SEI 发现所有使用 CMMI 的企业都受益匪浅，包括：

- 查找和修复缺陷的成本降低了 15%；修复一个缺陷的平均成本降低了 30%
- 推出新版本的时间缩短了 50%；软件开发能力提高了 30%
- 大大提高了系统的部署质量，只出现了 2%的错误
- 提高了客户满意度，相应的得到了更好的财务回报

CMMI 成熟度水平

CMMI 提供了级别式的和持续式的两种表示法。在本文中 ,将关注级别式表示法。CMMI 定义了五个级别（或水平）的过程成熟度（见图 1）。CMMI 鼓励企业先集中精力在那些可控制的过程域上，然后逐步将这些过程演变到更复杂的级别。本文将重点描述级别二和级别三中包含跟需求管理相关的过程域。



图 1：CMMI 的五个级别

过程域

CMMI 的过程域是一组相互关联，并且有一组可定义目标的最佳实践。图二表示了五个成熟度级别各自的过程域。

Level	Focus	Process
5 Optimizing	Continuous process improvement	<ul style="list-style-type: none"> Organizational innovation and deployment Casual analysts and resolution
4 Quantitatively managed	Quantitative management	<ul style="list-style-type: none"> Organizational process performance Quantitative project management
3 Defined	Process standardization	<ul style="list-style-type: none"> Requirements development Technical solution Product integration Verification Validation Organizational process focus Organizational process definition Organizational training Integrated project management Integrated supplier management Risk management Decision analysis and resolution Organizational environment for integration Integrated teaming
2 Managed	Basic project management	<ul style="list-style-type: none"> Requirements management Project planning Project monitoring and control Supplier agreement management Measurement and analysis Process and product quality assurance Configuration management
1 Initial		

图 2：CMMI 各级别的过程域

本文将关注 CMMI 第二级别中的需求管理和第三级别中的需求开发及相关技术解决方案。一旦付诸实施，它们相互紧密联系，并协同运行。

2. CMMI 级别二的需求管理

处在级别二的组织必须已经检查过一系列关键域，并建立起可重复性的流程。

过程域：需求管理

在该级别上，CMMI 对于需求管理过程域的目标是：

“需求是可管理的。项目计划和产品开发间的不一致性是可以识别的。”

所有产品的目标都是可以满足其责任人（Stakeholders）和客户（Customers）的需求，但它也必须遵守其内部功能上和质量上的约束。需求管理过程在这一点上起到了非常重要的作用。最近的 Standish 行业分析报告明确的指出，50%的成功项目把成功归功于建立了良好的需求管理过程。

为达到需求管理流程目标而必须建立的一系列最佳经验概括如下：

组织必须定义一组需求

CMMI 中建议：

“与需求提出者一起得出对需求的真正理解。”

“从项目参与者得到对需求的确认。”

任何需求管理过程的第一步都是确保所有的责任人（Stakeholders）理解项目的目标和目标设立的原因。因此，在组织级别上必须建立一整套流程可以涵盖所有责任人参与需求的定义，并最终对这些需求达成一致。在项目的进展过程中，组织能够从最终的结果反向追踪到最初的目标，以确保两者的匹配度。除此之外，任何针对需求的修改都需经过审核。

因此，为了建立和管理一整套具有良好定义的需求，需求管理工具应该能够让不同的用户查阅/修改需求文档，并可以识别需求的唯一性。用户也同样能够简便地建立起不同阶段的需求（例如，用户需求和产品需求）和项目的其他工件（例如设计和测试）间的可追溯性报告。需求管理工具同样也应该可以提供一套可审核的追踪过程，从而保证需求即使在发生变更的情况下仍然可以得到实现。

维护需求的修改历史也同样重要，因此需求管理工具也应该支持对需求文本的基线化版本管理。

组织必须管理需求的变更

CMMI 中建议：

“ 在项目中进行管理需求变更。 ”

一旦需求建立被捕获或产生，所有的项目参与人员必须能够得到需求的最新信息。因此整个组织必须懂得：

- 需求变更的提出和产生。
- 这些变更将对整个项目有怎样的影响。
- 对于这些变更，相关责任人采取了怎样的一致行动。
- 这些经过审批的变更是否已经反映到最终的产品。

需求的变更是不可避免的，也是允许的，但必须加以管理和控制。相关工具必须提供一个可协作的，可配置的需求变更控制流程。用户在这个流程中必须能够：

- 针对需求提交变更请求。
- 在变更控制流程中，将单一需求项的多个变更请求作为一个独立的条目进行管理。
- 将相关的多个需求变更对应到一组需求上，以达到同步的更新。
- 将需求变更分派给相关评审人员。
- 能在线评审并评注需求变更，这样所有人都能够看见所有的评注。

- 迅速全面的评估对某个变更对于需求，设计和测试的影响。
- 根据评审人员的意见批准或拒绝变更请求。
- 自动地实施批准的变更以确保不发生任何的错误。

组织必须确保需求被满足

CMMI 中建议：

“ 必须维护/保持需求、项目计划和产品开发之间的双向可追踪性。 ”

只有完整的可追踪性才可以确保所有正确的需求都是以正确的方式实现的。因此，组织必须能够检查在开发过程的每一个阶段需求都被满足，并且说明最终产品的某一个具体特性是怎样反向对应到其需求的。

没有工具的强有力支持，创建、维护和报告需求的追踪性是极其耗时和容易出错的。因此，工具必须能够确保迅速，简单地建立起追踪连接。为了达到 CMMI 的要求，不仅要在各个需求之间，而且也要在需求和项目计划和产品开发（例如设计和测试）之间建立可追踪性。

需求管理工具必须能够收集/同步开发过程中所使用的其他工具中的信息，同时为了支持迭代和递增的开发流程，也必须在需求文本的不同版本间建立和维护可追踪性。一旦建立了追踪性，需求管理工具必须能够在一个窗口中查看需求之间，需求和计划，工件之间的关联状况。

组织必须确保项目计划，开发产品和需求是一致的

CMMI 中建议：

“ 必须识别出项目计划，开发产品和需求之间的矛盾(或不一致性)。 ”

识别这些矛盾可以确保组织的开发工作是正确的，这样所有工作都遵从最新的

相关信息并且没有忽视任何需求。

因此，组织必须建立起一套流程以确保所有的项目计划，开发产品和需求保持一致。这套流程中包括评审，分析，和管理所发生的需求变更。

项目计划和需求声明必须紧密关联，需求管理工具必须确保项目任务和需求数据是同步的，这样两者之间才可以建立起可追踪性。可追踪性的建立可以帮助我们更好地评估需求变更对项目的影响度。

3. CMMI 级别三的需求管理

在 CMMI 级别三中，组织需要建立、归档所有项目中所使用的公共的和一致的流程。

过程域：需求开发

需求开发过程将解释如何抽取责任人的需要、导出用户的需求声明，并把这些用户需求进一步分析/总结成为相应的系统需求。

CMMI 对需求开发定义了以下几个目标：

组织必须能够将收集来的责任人的需求转换成用户需求

CMMI 中建议：

“将收集来的责任人的需要，期望值，约束条件和接口定义转换成用户需求。”

在一般情况下，责任人的需要并没有被很好地定义和理解，这些需要甚至可能是不一致或相互矛盾的。一个责任人的典型代表必须参与产品开发的整个生命周期。在这个生命周期中，必须反复地定义，阐述和明确这些需求，最后总结出一组清晰的，被正确理解的用户需求。

以上过程非常类似于邀请客户代表进入项目队伍参与全周期的开发工作。此外，对需求采集和阐释的技巧也是极其重要的，目前较流行的是 Use Cases 方法。Use Cases 方法是一套在不同的使用场景下，从用户的角度出发，构造和记录功能性需求的方法。单个的 Use Case 只能记录简单的文本，最好的方式是使用 UML 中的 Use Case 视图对一组 Use Cases 进行描述，并表示它们之间的相关关系。

因此，需求管理工具必须能够支持在文本说明的需求声明中插入相关的 UML 视图，并可将文本和视图同时显示在一个文档中。

组织必须可从用户需求分析出系统需求

CMMI 中建议：

“对用户需求进行推敲和细化以开发出对系统需求。”

对用户需求进行彻底的分析，识别出所有已暗示但没有被清晰表述的需求，再结合所开发目标产品的技术架构细节，可得出该产品的系统需求。系统需求根据产品的规模和复杂度，可细化成系统需求、子系统需求、模块需求等。在这个过程中，必须同时建立起不同需求之间的可追踪性，为后续的决策指定提供参考，并支持需求变更影响度分析。

根据 CMMI 要求，在技术架构的基础上，通过对用户需求的分析可推导出系统需求。因此，需求管理工具需要支持实现这种推导，这种推导的支持不仅包括在不同层次的需求之间创建可追踪性，还包括提供可分析上层需求的技术和相关方法。在 UML 建模方法中，可通过 Use Case 视图，Activity 视图和 Sequence 视图等有效的方法来帮助分析和导出新的需求，上层的技术架构通过 Architecture 视图（也称为复合结构）进行描述。这些方法都要求在需求文档中建立模型，用来记录分析和决策过程，从而为下一步提出开发技术方案提供指导。

组织必须能够验证需求

CMMI 中建议：

“需求必须可以被分析和验证，同时必须开发出对“必需的功能性”的定义。”

当那些非正式的需要转化为正式的需求时，需求的分析和验证对于确定责任人需要、用户需求、系统需求等是否可行，是否可以在预算范围内达到，是否跟当前系统运行环境相匹配都是非常必要的。

用于需求分析和验证的技术包括按时间顺序的使用场景验证需求，并提推导新

的需求。场景方法是一种有效的采集，阐述和推导需求的方法，在这个方法中我们常用 UML 中的 Activity 视图或 Sequence 视图来表示场景。

“必需的功能性”的定义是通过功能性分析建立起一整套功能性的架构。功能性分析描述了系统的行为，时序活动，输入输出以及其他所有对该系统应用的描述。功能性架构则从逻辑上描述了一组功能(或服务)和需求是怎样相互满足的。UML 的 Activity 视图能够从工作流的角度描述系统的活动行为，并且可以识别负责这些工作流的相应组件，这些视图在描述功能性分析，以及将需求进一步分配到子系统级别时非常有用。因此，需求管理工具应该能够支持图形化模型和文字性的需求说明存在于一个单一的文档形式中，并且这样文档形式可以支持不同层次间需求的追踪关系（无论这些需求是以文字形式还是以 UML 的图形化模型表示）。

过程域：技术解决方案

在这个过程域中，我们需要关注的是评估和选择设计方案，进行详细设计，最终实施。那么这些和需求管理又有什么关系呢？组织必须首先确保他们可以持续地开发出满足需求的解决方案，一种方法就是通过验证所产生的设计来确保初始方案是正确的。在这个过程中，组织应该能够发现那些不切实际的，或因定义不够充分而无法实现的需求。同时，组织应该允许需求在整个设计过程中不断地充实变化，但这一点的前提必须保证这些变化是可控制的。所以，可追踪性不仅体现在不同层次的需求之间（用户需求，系统需求，子系统需求等），也应该体现在需求和解决方案之间。

我们应该保留从责任人要求到系统和子系统需求的分析过程和分析方法，以便我们在设计和编码阶段仍然可以遵循这些过程和方法。因此，需求管理工具应该能够让后续的系统设计者和开发者可以浏览、创建和维护最初的需求和他们设计之间的可追踪性，并以此检验后续设计/开发工作对需求的依从性，以及对需求变更的影响度估计。由此，需求分析员和项目经理应该具有从系统责任人要求到系统具体设计/开发/测试间的可追踪性有一个完整概念，这个可追踪性可以让他们做到基于需

求的项目过程监督，并可以保证所有的开发和测试工作与需求保持一致。

4. 总结

对于那些有大型复杂的产品或系统开发项目的组织来说，CMMI 是评估和改进开发流程的必由之路。许多大型公司见证了这条必由之路，如：洛克希德·马丁 (Lockheed Martin)，波音 (Boeing)，诺斯罗普·格鲁曼 (Northrop Grumman)，通用汽车以及 JP 摩根等。

在 CMMI 2 级和 3 级中，重点要求达到的是有效的需求管理和需求分析实践。在这两个阶段，全面的工具支持对于帮助整个组织理解、定义和实施 CMMI 描述的最佳经验是必不可少的。通过在开发生命周期中提供对需求采集/定义、需求分析以及需求变更追踪的全面支持，开发组织才能够从这些过程改进中得到最大收获。