

基于基线化的迭代开发和风险管理策略 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/226/2021\\_2022\\_\\_E5\\_9F\\_BA\\_E4\\_BA\\_8E\\_E5\\_9F\\_BA\\_E7\\_c41\\_226618.htm](https://www.100test.com/kao_ti2020/226/2021_2022__E5_9F_BA_E4_BA_8E_E5_9F_BA_E7_c41_226618.htm)

IPMS ( Integrated Project Management System ) ，是一个为上海贝尔阿尔卡特某移动事业部门定制化开发的多系统集成项目管理系统，其目标是通过对该部门整个开发管理工具环境的集成来达到对项目的实时跟踪和管理，以及数据的采集和度量。然而，一个客观事实是，国内大量的中小型项目中的绝大多数都存在着需求不明确性的问题，伴随而来的是大量需求变更所带来的开发风险。那么面对这样的客观事实，是否意味着项目经理根本无法应对这样的问题呢？真正优秀的项目经理又会如何挑战需求不明确和需求变更所带来的开发风险呢？

[案例背景] 背景描述1：随着阿尔卡特与朗讯科技的合并，该移动事业部门也将与原两家公司的多个部门进行合并。合并后的新部门将采用全新的产品线定义，并实施新的项目管理流程。在这样的情况下，以前的IPMS系统已经无法满足该事业部门对项目的流程管理，IPMS系统三期开发势在必行。

背景描述2：IPMS系统作为一个集成化的管理系统与众多功能强大的软件系统集成，其中包括ClearQuest，一个强大的事务状态管理软件。原先IPMS与ClearQuest通过一组ClearQuest软件预先定义的API进行数据通讯。

[提出问题] 由于合并的时间过于仓促，以及部门之间的项目数据定义和管理流程的不同，导致IPMS三期开发的需求极不明确，整合后的新部门内部还在为部门的项目管理的流程定义和系统可能的功能修改争论不休，在这样的情况下，项目进展十分缓慢。IPMS三期的第

一个迭代中有这样一个非功能性需求，即ClearQuest的版本升级，随之而来的一个风险便是，伴随着ClearQuest的版本升级，没有人知道原版本中与IPMS数据通讯的API接口是否可以向下兼容。[解决方案] 1.基于现实的问题，即该事业部门确实无法在很短的时间内协调和构思出统一的系统需求，以及部门希望尽快将部分功能使用起来的愿望，项目组采取了基于需求基线化管理的迭代开发。项目组采用了合适的开发步骤，首先基于目前的需求不稳定性项目组决定了以一个月为开发的迭代周期；然后在策划期提取出一组客户的高端需求，通过对需求的简要分析和工作量估算，并依据一组参数（比如重要性，紧急程度和需求稳定程度）规划出一个月内（一次迭代期）的项目需求；对这些需求建立基线，一旦需求的基线确立，那么在这个迭代周期内的需求应该是稳定的；最后项目组在一个迭代周期内，通过依据CMMI的流程对系统进行三期开发。通过这样的方式，能够尽可能的降低需求不明确所带来的开发风险，增强需求的可管理性，但是如果在迭代内部发生了需求的变更又该如何处理呢？ 2.项目组通过对风险可能发生时间点和阈值的控制，来管理风险的危机。项目组首先识别出了ClearQuest升级后的API兼容性风险；同时，通过讨论，项目组确定了这个风险可能转化为问题的时间点，即一旦完成三期对项目注册功能的修改后，必须在测试时确定IPMS中注册的项目数据是否可以正确的被保存入ClearQuest中，我们可称那个时间点为A时间点；当项目的开发活动快到达A时间点之前，项目组安排人力对新版本ClearQuest中的相同API进行了简单的功能原型测试，测试结果表明确实存在兼容性问题；项目组立刻决定修改项目计

划，并在原先的风险预留的时间段内增加了对API兼容性问题的处理事务，从而解决了ClearQuest版本升级可能导致的系统无法正常注册项目的重大问题，使得项目得以平稳开发 [案例评析] 在软件项目开发过程中，对于开发模型的选择，需要在项目定义过程中明确。CMMI V1.2 For dev中，过程域IPM[注：4] 的SP1.1 Establish the Project 's Defined Process有明确要求。在上述案例中，提到的迭代式开发模型为众多开发模型中的一种，而在项目中具体要使用哪种模型可从组织项目定义过程中进行选择。软件开发模型通常有以下几种：瀑布型，迭代型，原型等，具体选择何种，需要视项目的特点而定；上述案例的主要特点是需求的不稳定性，选择迭代式开发模型无疑是一种较好的选择。可以看到，上文中提到“基线化”一词，有实施CMMI的企业或参加过CMMI过程改进活动的个人对这个词一定不陌生。CMMI模型中，要求对过程的产出物进行配置管理。过程域CM[注：5]中，SP1.3 Create or release baselines 要求建立并发布基线。基线是经过评审并通过的一系列产出物，基线建立以后，后续的开发工作需以此作为基础。上述案例中，之所以提出基线化一词，意在强调阶段性地需求需要经过评审并确定之后，以此指导后续开发工作。此外，越来越多的人关注软件项目开发过程中风险管理环节。风险管理过程是用于识别潜在的问题，并策划应对策略，在需要时实施相应动作以消除不利影响。在CMMI模型中，有专门一个PA对风险管理进行描述和要求。上述案例中，正是识别到由于ClearQuest升级而带来的API不兼容性风险，并针对于风险采取了利用阈值控制等措施。软件开发过程中，我们会遇到各种各样的风险，而且这些风险一旦发生，

会给项目的顺利进行带来严重威胁，因此在项目计划时，就要制定一个严密的风险管理计划，并且对于风险情况进行严格的跟踪，这样才可能把风险对项目所带来的影响降低，至最小。注：1：基线化：在配置管理系统中，基线就是一个CI（配置项）或一组CIs在其生命周期的不同时间点上通过正式评审而进入正式受控的一种状态，而这个过程被称为“基线化”。2：迭代：迭代就是为了完成一定的阶段性目标而所从事的一系列开发活动，属于开发模型中的一种。3：风险管理：风险管理指对项目风险进行识别、分析、并采取应对措施的系统过程。它包括尽量扩大有利于项目目标事项发生的概率与后果，而尽量减小不利于项目目标事项发生的概率与后果。4：IPM：集成项目管理，为CMMI中一过程域。5：CM：配置管理，为CMMI中一过程域。100Test 下载频道开通，各类考试题目直接下载。详细请访问 [www.100test.com](http://www.100test.com)