# CMM/CMMI 不适合在当前软件开发当中应用的原因

## 1. CMM/CMMI 淡化了软件项目的独特性

### 1.1 不是所有的项目都是外包项目

**CMM/CMMI 的一个主要问题是它淡化了软件项目的独特性，让人产生一种错觉：软件是有严格规格的工业产品，在标准流程下，从一端输入原材料，在另一端获取软件产品。然而，在市场压力下，这种软件的标准化、规格化变得越来越不可能存在。**这也许和 CMM/CMMI 提出的背景和目的有关：*CMM 是美国国防部在 1984 年因当时该机构软件标案委外承作时，无法评估软件公司对软件标案的承接及执行能力，故委托美国卡内基美隆大学的软件工程学院所进行的一项研究成果，其目的是用来评估及改善软件发展公司的软件开发过程及软件开发能力，并且协助软件开发者持续改善软件流程成熟架构及软件品质，进而提升软件开发项目及软件发展公司的软件开发管理能力，达成软件发展的功能正确、缩短开发时程、降低成本及确保品质等目标。*[1] 可以看出，CMM/CMMI 实际是为评估软件公司对软件标案的承接及执行能力而产生的，它更像是安抚买家的安慰剂与赢得合同的催化剂，对于外包公司，当甲方项目需求类似（如某乙方公司只做财务类系统），CMM/CMMI 或许是有效的。然而，有很多公司不是外包公司而且有越来越多的公司成立自己的研发部门，这些公司只研发自己的产品、开发自己的需求，在市场压力下，这些需求场景越来越多样化、这些挑战越来越大，当他们应对这些场景与挑战时，CMM 还会是有效的吗？不会，而且当公司 CMM/CMMI 等级越来越高，在这种规范指导下，需要面对新场景与挑战的开发团队将会在组织的笼罩下将寸步难行。

### 1.2 过度渴望一致性

**CMM/CMMI 渴望一致性或者说保持稳定而平庸。这就像音频压缩算法，可以除去峰值，但是音乐中也就没有了高音与低音的美妙。**一方面，从项目产品的角度来看，稳定而平庸的公司很难有非连续性创新，其终将会被互联网的浪潮淘汰，因为这样的公司太多，用户有太多选择。另一方面，从工程师角度来看，在组织层级上要求 A+等级的团队与 C 等级的团队保持一致，这对 A+等级的团队或个人是极不公平的，这只表现出色、才华横溢的团队并没有受到表彰与推崇，反而为了流程受到排斥与限制，他们会对这种管理反感或许他们也会变得平庸。

## 2.CMM/CMMI 与敏捷方法冲突

**众所周知，CMM/CMMI 与敏捷是完全不同层面的东西，但它们实施起来仍然会在某些方面有冲突。近十几年来，敏捷方法在业界一直占据软件开发的主导地位，然而 CMM/CMMI 却与敏捷方法有着大量的冲突。人是敏捷开发的核心，敏捷开发总是以人为中心建立开发的过程和机制，而非把过程和机制强加给人，而 CMM/CMMI 恰恰相反。**

我们相信 CMMI 是在敏捷环境中改进流程的一种方法，通过检查敏捷方法对过程领域的覆盖，可以发现敏捷方法中的缺陷，从而改进敏捷方法。然而，使用 CMMI 的过程改进只能在一定程度上进行，因为有几个过程领域与敏捷原则相冲突，如果不牺牲一些敏捷的基础，第 3 级和第

4 级和第 5 级的大部分过程领域是不可能实现的，但这会让敏捷不再敏捷，消除了敏捷的优势。同时，这些行为也与 CMM/ CMMI 的目标相矛盾，即通过使敏捷方法尽可能好地改进过程，而不是将其转变为另一种不再敏捷的方法。

CMM/CMMI 与敏捷方法的冲突主要表现在那些明确处理过程改进的过程领域（组织过程焦点、组织过程绩效、定量项目管理、组织创新和部署）与敏捷方法相冲突。此外，通用实践“收集改进信息”明确地处理过程改进与敏捷方法冲突。“决策分析和解决”由于需要一个正式的评估过程而与 Scrum 和 XP 为代表的敏捷方法冲突。

## 2.CMM/CMMI 与当前企业管理方式冲突

**CMM/CMMI 与当前互联网企业管理提倡的自治化是冲突的。知识工作者必须要自我管理。他们必须有自主权。**[2]团队作为一个自治单位的概念已经大受欢迎，团队作为自治单位带来的好处得到了验证，也让企业有充分的理由选择这种管理方式。这种团队的典型特点有：首先，团队共同完成一个伟大的使命；其次，在信息流方面有着清晰的边界，与其他组织单位、资源或决策政策协同一致；第三，在这些边界内有权自我管理；第四，在某此合理安排的期限内保持稳定性。[3] 可见 CMM/CMMI 与自治理型团队提倡的以团队为单位的边界、自我管理是冲突的。举一个比较典型的例子：拥有几十个 scrum 团队的企业可能非常成功，尽管这几十个团队拥有不同的价值观、执行不同但适合团队本身的流程、交付不相同的产物，他们根据 CMMI 被判定为 2 级。如果想要达到 CMMI3，那么意味着流程需要被整个组织确定，而不能让团队自治、自己决定他们的工作方式。所有的团队都需要以相同的方式做事，并产生相同的结果。

## 4.CMM/CMMI 与当前开发技术、架构冲突

**CMM/CMMI 与当前流行的微服务架构是冲突的。***微服务架构，是一种软件架构方式，它将应用构建成一系列按业务领域划分模块的、小的自治服务。微服务架构有很多重要的优点：首先，它解决了复杂性问题，它将单体应用分解为一组服务，虽然功能总量不变，但应用程序已被分解为可管理的模块或服务，微服务架构强化了应用模块化的水平，因此，微服务开发的速度要快很多，更容易理解和维护；其次，这种体系结构使得每个服务都可以由专注于此服务的团队独立开发，只要符合服务 API 契约，开发人员可以自由选择开发技术，这就意味着开发人员可以采用新技术编写或重构服务，由于服务相对较小，所以这并不会对整体应用造成太大影响；第三，微服务架构可以使每个微服务独立部署，开发人员无需协调对服务升级或更改的部署，这些更改可以在测试通过后立即部署，所以微服务架构也使得 CI／CD 成为可能。最后，微服务架构使得每个服务都可独立扩展，我们只需定义满足服务部署要求的配置、容量、实例数量等约束条件即可。*一方面，微服务在管理要求项目团队独立开发以保证系统独立性与专注度，以降低运维、变更成本，而 CMM/CMMI 在第三级及以上要求流程本整个组织确定，因此项目团队必须服从组织流程而不能独立自治的开发项目，这不符合微服务架构的开发实践；另一方面，我们从康威定律（Organizations which design systems are constrained to produce designs which are copies of the communication structures of these organizations.）可知组织形式等同系统设计、组织沟通方式会通过系统设计表达出来，当我们符合高级别的 CMM/CMMI 等级时，组

织架构变得更加非扁平化、团队之间的沟通交流变得更多以保持一致性，从而导致我们的系统设计变得耦合度更大，而非更加独立，这与微服务架构设计的初衷是违逆的；最后，微服务强调快速迭代、快速开发、持续集成、持续部署的概念，而在 CMM/CMMI 的种种限制下，这很难实现。

## 6.CMM/CMMI 导致人才流失

人才是当前企业竞争的力量源泉，一个拥有一群富有激情与才华人才的公司在市场中具有极大的竞争优势。

### 6.1CMM/CMMI 贬低个人的价值

**CMM/CMMI 贬低了个人的价值。**软件开发是一种技能游戏，而无形的技能是一种巨大的差异，认为开发人员是可替代的劳动者的观点是无稽之谈。在当前市场压力下，软件开发像一场充满挑战的足球比赛，面对新的挑战，球队获胜的关键因素是球员而非老套的战术，当球员个人能力被老套的战术束缚，他只有一个选择——退役。程序员也一样，每个人都希望自己能在项目中发挥自己独到的作用，而 CMM/CMMI 限制了这种可能，为了降低风险，它让每个程序员发挥着同样的作用，或者说它让每个开发人员都成为可替代的劳动者。这无疑降低了个人在项目中的归属感与价值感，导致人才流失。

### 6.2CMM/CMMI 让人失去激情

**CMM/CMMI 奉行“把所有经验都记录下来，然后按照脚本来做”的方式，让工作变得无聊。**很难想象，这样的公司里的工作人员能有工作的激情；也很难想象，这样的公司能吸引优秀的人才来这里工作；但不难想象，公司里原本那些富有才华和激情的人，要么被磨灭、要么离开。

## 6.CMM/CMMI 没有得到市场的选择

**如果 CMMI 有不可替代的优势或它是适用于当前软件软件开发的，那么市场将会选择它，但事实证明即使已经存在了三十年，它仍然没有成为赢家。**事实上，在过去三十年里，主要是由与 CMMI 核心原则背道而驰的框架和流程所主导，2000-2020 时代显然一直由敏捷方法主导的。我们可以看到申请 CMM/CMMI 认证的公司多为外包公司，广大的互联网公司并没有选择进行 CMM/CMMI 认证，而是选择了 scrum、kanban 等于 CMMI 核心选择违逆的敏捷方法，从图 1 中可以看到接受调查的公司中只有百分之三十没有实用敏捷方法。

# 参考文献

[1] wensonlee. CMMI 是什么[EB]/[OL].
https://blog.csdn.net/wensonlee/article/details/83127814.2018-12-04

[2] （美）德鲁克．21 世纪的管理挑战[M]．美：机械工业出版社，2009-9-1

[3] Peter Hundermark,Sigi Kaltenecker，什么是自组织团队？[EB]/[OL].
https://www.infoq.cn/article/what-are-self-organising-teams

[4] （美）德鲁克．21 世纪的管理挑战[M]．美：机械工业出版社，2009-9-1