

测试人员认证专家级大纲

改进测试过程

(实施改进与变更)

ISTQB 批准，发布
2010 年 1.0.2 版

国际软件测试认证委员会



ISTQB™

版权声明

在标注来源的情况下，可以全文复制或摘录本文档。

国际软件测试认证委员会版权所有 ©（以下简称“ISTQB®”）。

专家级大纲工作组：Erik van Veenendaal, Graham Bath, Isabel Evans, 2006-2009。.

中国软件测试认证委员会 (CSTQB)

修订历史

版本	日期	备注
V2009-Beta 版	2009 年 5 月 25 日	Beta 评审
V2009-Beta2 版	2009 年 7 月 2 日	Beta 评审的修订
V2009 发布 1.0.0 版	2009 年 10 月 23 日	发布
V2009 发布 1.0.1 版	2010 年 3 月 19 日	技术写作人员编辑
V2009 发布 1.0.2 版	2010 年 4 月 16 日	第 72 页第 14 行, 第 15 行打字错误, 交换 K5 和 K6 行

中国软件测试认证委员会 (CSTQB)

目录

修订历史.....	3
目录.....	4
致谢.....	7
1. 课程大纲引言.....	8
1.1 国际软件测试认证委员会.....	8
1.2 大纲编写目的.....	8
1.3 专家级软件测试认证工程师.....	8
1.3.1 知识水平.....	8
1.3.2 考试.....	9
1.3.3 授权.....	9
1.4 规范性内容和信息性内容.....	9
1.5 详细程度.....	9
1.6 大纲组织形式.....	9
1.7 术语和定义.....	10
1.8 学习目标 (LO) /知识等级 (K)	10
1.9 期望.....	11
2. 改进概要 285 分钟.....	12
2.1 为什么改进测试?	12
2.2 改进范围.....	14
2.3 质量观.....	14
2.4 通用改进过程.....	15
2.4.1 戴明环.....	15
2.4.2 IDEAL 改进框架.....	15
2.4.3 卓越的基本概念.....	16
2.5 改进方法概述.....	16
2.5.1 基于模型的改进方法概述.....	16
2.5.2 分析型改进方法概述.....	17
2.5.3 混合型改进方法.....	17
2.5.4 其他测试过程改进方法.....	17
3. 基于模型的改进 570 分钟.....	20
3.1 基于模型的方法简介.....	21
3.1.1 测试过程改进模型的理想特性.....	21
3.1.2 连续式和阶段式表示法.....	21
3.1.3 使用模型的假设.....	22
3.2 软件过程改进模型.....	22
3.2.1 CMMI.....	22
3.2.2 ISO/IEC 15504.....	23
3.3 测试过程改进模型.....	23
3.3.1 测试过程改进模型 (TPI®)	23
3.3.2 测试成熟度模型集成 (TMMi)	23
3.3.3 比较 TPI Next 模型与 TMMi 模型.....	24
3.4 基于内容的模型.....	24
3.4.1 STEP.....	24
3.4.2 CTP.....	25

4.	基于分析的改进	555 分钟	26
4.1	简介		26
4.2	原因分析		27
4.2.1	因果图		27
4.2.2	审查过程中的原因分析		28
4.2.3	使用标准异常分类		28
4.2.4	选取缺陷进行原因分析		28
4.3	GQM 方法		28
4.4	使用测量、度量和指标进行分析		29
4.4.1	简介		29
4.4.2	测试效力度量		29
4.4.3	测试效率/成本度量		30
4.4.4	交付周期度量		30
4.4.5	可预测性指标		31
4.4.6	产品质量度量		31
4.4.7	测试成熟性度量		31
5.	选取测试过程改进的方法	105 分钟	32
5.1	选取测试过程改进的方法		32
6.	改进过程	900 分钟	34
6.1	介绍		35
6.2	启动改进过程		35
6.2.1	确定改进需求		35
6.2.2	设置测试改进的目标		35
6.2.3	设置改进范围		37
6.2.4	启动阶段人员和文化的影响		37
6.3	诊断当前状况		38
6.3.1	计划评估		38
6.3.2	评估准备		39
6.3.3	开展面谈		39
6.3.4	初步反馈		39
6.3.5	分析结果		39
6.3.6	实施解决方案分析		40
6.3.7	建议改进活动		41
6.4	制定软件测试改进计划		41
6.4.1	设置优先级次序		41
6.4.2	开发可实施的方案		42
6.4.3	计划改进		42
6.5	采取行动, 落实改进		43
6.5.1	选择和执行试验		43
6.5.2	管理和控制实施		44
6.6	从已改进项目中学习		44
7.	组织, 角色和技能	465 分钟	45
7.1	组织		45
7.1.1	测试过程改进组		45
7.1.2	远程、离岸和外包团队的测试改进		46
7.2	个体角色		47
7.2.1	测试过程改进人员		47
7.2.2	评估人角色		47

7.3	测试过程改进人员/评估人员的技能.....	47
7.3.1	面谈技能.....	47
7.3.2	倾听技能.....	49
7.3.3	演示和报告技能.....	49
7.3.4	分析技能.....	50
7.3.5	记录技能.....	50
7.3.6	说服技能.....	50
7.3.7	管理技能.....	51
8.	管理变更 285 分钟.....	52
8.1	简介.....	52
8.2	变更管理的基本流程.....	52
8.3	变更管理过程的人为因素.....	53
9.	改进成功的要素 300 分钟.....	55
9.1	成功的关键要素.....	55
9.2	为改进建立文化.....	56
10.	适应不同生命周期模型 60 分钟.....	58
10.1	适应不同生命周期模型.....	58
11.	参考文献.....	60
11.1	标准.....	60
11.2	商标.....	60
11.3	书籍.....	61
11.4	论文和文章.....	63
11.5	参考网站.....	63
12.	附录 A – 培训机构的注意事项.....	65
12.1	培训时间.....	65
12.2	使用的标准.....	66
12.3	实践练习.....	66
12.3.1	课外实践练习的指导.....	66
12.4	培训机构有关标准的通用指南.....	67
12.4.1	专家级培训机构标准.....	67
12.4.2	专家级课程.....	67
12.4.3	讲师的准入标准.....	67
索引	69

致谢

本文档由国际软件测试认证委员会专家级工作组“改进测试过程”模块的核心团队编写。

Graham Bath（本大纲的负责人）

Isabel Evans

Erik van Veenendaal

参加 1.0.22010 中文本地化的专家如下（按姓氏拼音排序）

组长：崔启亮

成员：曹静、陈耿、贺妍、邵晓梅、宣以广、郑文强、朱少民

评审：周震漪、沈建雄

大纲的核心团队感谢提供建议和贡献的评审团队和加入 ISTQB 的各个国家委员会。

在本模块的专家级大纲完成之际，专家级工作组的成员如下（以字母顺序为序）：

Graham Bath	Beata Karpinska	Klaus Olsen
Rex Black	Caroline Molloy	Meile Posthuma
Isabel Evans	Silvio Moser	Maud Schlich
Matthias Hamburg	Thomas Müller	Erik van Veenendaal
Kari Kakkonen	Ingvar Nordstrom	

下列人员参与了评审，评论和投票（以字母顺序为序）：

Graham Bath	Beata Karpinska	Ingvar Nordstrom	Brian Wells
Rex Black	Kari Kakkonen	Klaus Olsen	
Sigrid Eldh	Marcel Kwakernaak	Meile Posthuma	
Isabel Evans	Judy McKay	Stuart Reid	
Cheryl George	Caroline Molloy	Maud Schlich	
Derk-Jan de Grood	Thomas Müller	Neil Thompson	
Matthias Hamburg	Silvio Moser	Erik van Veenendaal	

本文档由 ISTQB®大会于 2009 年 6 月 19 日正式发布。

1. 课程大纲引言

1.1 国际软件测试认证委员会

国际软件测试认证委员会（以下简称 ISTQB®）由多个国家和地区的委员会组成。关于 ISTQB 组织和成员的更多信息，请参考 ISTQB 官方网站。

1.2 大纲编写目的

大纲主要包含了专家级软件测试认证“改进测试过程”模块的基本内容，大纲的主要内容如下：

- 委员会成员可以将大纲翻成本国语言，供授权培训机构使用。委员会成员可以根据本地语言需求对大纲正文进行修订，对参考文献适当修订以满足当地出版要求；
- 考试组可以依据课程大纲开发符合本模块学习目标的本地化试题；
- 培训机构可以根据课程大纲开发培训课件，并采用适当的教学方法；
- 认证备考人员可以根据大纲做好考试准备（自行备考或作为培训课程的一部分）；
- 全球软件工程和信息系统工程领域的组织可以提高软件和系统测试的专业化，也可作为编写相关书籍和文章的参考资料。

任何以其他目的使用本大纲的组织 and 人员，需事先取得 ISTQB® 的书面许可。

1.3 专家级软件测试认证工程师

专家级资格认证主要面向在自身工作领域已经具备较高测试技能，且希望在专业领域进一步提高的人员。专家级模块涵盖了广泛的测试专题。

测试专家要有全面的测试知识，并且在某一测试领域有深入的理解。具有深入的理解意味着测试专家需要有足够的测试理论知识和实践经验，并且能够在组织或项目创建、实施测试过程时，起到关键性的方向指导作用。

参加专家级测试过程改进模块资格认证的人员，必须已经通过高级测试经理模块的认证。

除了通过资格认证考试，在获得专家级证书之前，考生还必须提供实际测试工作经验证明，特别要提供申请认证的专家级模块所在领域的工作经验证明。除了通过考试，还要符合以下要求：

- 至少 5 年的实际测试工作经验（需提交个人简历，包括 2 封推荐信）；
- 至少 2 年的专家级领域工作的经验（需提交个人简历，包括 2 封推荐信）；
- 至少发表过 1 篇文章，或在测试大会中进行过专家级模块相关的测试专题演讲。

符合上述标准的人可以收到正式的 ISTQB 专家级相应模块的认证证书，拥有 ISTQB 专家级证书的人员允许使用 CTEL（Certified Tester Expert Level）的缩写。

拥有测试专家级认证证书的人员需要定期更新该资格认证，可以按照 ISTQB-CEP 的要求实现最低要求的学分。想了解更多的细节，请参考官方网站[ISTQB 网站]。

1.3.1 知识水平

为了清楚地识别本大纲每一章的学习目标，在每章开头描述了本章的学习目标。

1.3.2 考试

所有的测试专家级资格认证考试都以本课程大纲为基础，同时考试也要基于高级大纲中的测试经理模块（尤其是第 8 章标准和测试改进过程）和初级大纲的相关内容。解答试题可能需要用到这些大纲中一个或多个章节的内容。

[ISTQB-EL-EXAM]. ISTQB 专家级考试指南中定义了考试的形式。

考试可以作为授权认证培训课程的一部分，也可以单独进行（例如，在授权的考试中心）。考试既可以进行书面形式的考试，也可以以电子方式进行，但所有的考试必须在考试认证委员会或者考试委员会授权人员的监督下进行。

1.3.3 授权

ISTQB 的成员委员会可以对课程材料符合本大纲要求的培训机构进行授权。授权指南可以从当地委员会或者其他团体获取。已授权的课程符合本大纲的要求并可以将 ISTQB 的认证考试作为课程的一部分。

1.4 规范性内容和信息性内容

大纲的规范部分属于考试范围，规范部分包括：

- 学习目标；
- 关键词；
- 必要的练习。

大纲的其余内容属于信息部分，是对学习目标的细化。

1.5 详细程度

本大纲的详细程度保持与国际教学和考试一致。为了达到此目的，本大纲包含如下内容：

- 专家级资格认证课程大纲总体目标；
- 每个知识领域的学习目标，描述认知学习成果和心态（这些是规范性的内容）。达到各知识领域需要达到的学习目标和知识水平；
- 教学内容列表，包括关键概念的阐述、已使用的文献或标准的来源、其他参考文献等（此为信息性的内容）。

本大纲的内容并不是对整个软件测试过程改进知识领域的描述；它仅覆盖了专家级认证培训课程中需要涉及的内容。

1.6 大纲组织形式

本大纲包含 10 个主要章节，每章开始部分显示了课程时间。例如，

2. 改进概要

180 分钟

说明第 2 章需要 180 分钟的授课时间。每章的开头都列出了特定的学习目标。

1.7 术语和定义

许多软件相关著作中的术语都在不断变更。本大纲中的术语定义来源于 ISTQB 发布的软件测试标准术语表。

本大纲每章开始部分的关键词定义源于 ISTQB 软件测试标准术语表。

1.8 学习目标 (LO) /知识等级 (K)

本大纲定义了如下学习目标。每个主题都应根据其定义的学习目标进行考核。

等级 1: 牢记 (K1)

考生可以识别、牢记和回忆的术语或者概念的内容。

关键词: 牢记 (remember)、回忆 (retrieve)、识别 (recognize)、认知 (know)。

例子

能够识别“失效”的定义:

- “不能向最终用户或其他干系人提供服务”或
- “组件或者系统的实际运行情况与期望的发布、服务或结果背离”。

等级 2: 理解 (K2)

考生应能够对大纲主题相关的内容选择原因或解释, 同事举出实例 (例如对比术语)、测试概念、测试过程 (解释任务流程) 进行总结、比较、分类并举例说明。

关键词: 总结 (summarize)、归类 (classify)、比较 (compare)、映射 (map)、对比 (contrast)、举例说明 (exemplify)、解释 (interpret)、翻译 (translate)、描述 (represent)、推断 (infer)、结论 (conclude)、分类 (categorize)。

举例

请解释测试应该尽早进行设计的原因:

- 在缺陷移除成本低的时候就发现它们;
- 尽早发现那些最重要的缺陷。

请解释集成测试和系统测试之间的异同:

- 相同: 需测试不同组件, 可以对非功能性的测试点进行测试;
- 不同: 集成测试关注于接口和交互, 而系统测试则关注从全系统角度进行测试, 如端到端的流程。

等级 3: 应用 (K3)

考生需要选择正确的测试概念或者技术, 并应用于给定场景中。K3 通常应用在流程知识方面。在评估软件应用, 或者为特定软件创建测试模型等活动时, 没有太多创造性工作。根据本大纲的模型, 从给定模型中按照流程步骤生成测试用例, 就是类似于 K3 的工作。

关键词: 实施 (implement)、执行 (execute)、运用 (use)、遵循流程 (follow a procedure)、使用流程 (apply a procedure)。

例子

- 可以识别有效边界值和无效边界值;
- 使用通用的测试用例生成流程, 从给定的状态转换图 (或一个测试用例集) 中派生出测试用例并覆盖所有的状态转换。

等级 4: 分析 (K4)

考生应可以将与流程或技术有关的信息分成不同组成部分，以便更好的理解，可以对现象和结论进行划分。典型的应用是对文档、软件、项目的情况进行分析，并提出合适的建议来解决问题或任务。

关键词：分析（analyze）、区分（differentiate）、选择（select）、结构（structure）、定位（focus）、特性（attribute）、解构（deconstruct）、评估（evaluate）、判定（judge）、监控（monitor）、协调（coordinate）、创建（create）、合成（synthesize）、产生（generate）、假定（hypothesize）、计划（plan）、设计（design）、构建（construct）、生产（produce）。

举例

- 分析产品风险，提供风险预防措施和缓解措施；
- 描述缺陷报告中的哪些部分是真实的，哪些内容是从测试结果中推导出来的。

等级 5：评估（K5）

考生需要根据标准和准则做出判断。检查过程或产品内的一致性或错误的地方，确定过程或产品是否具有内在的一致性，在实施过程时确定该过程的效力（比如，确定科学性结论是否遵从已观测到的数据）。

关键词：评估（Evaluate）、协调（coordinate）、检测（detect）、监控（monitor）、判断（judge）、批判（critique）。

举例

- 在给定的情况下，判断评审过程的有效性和高效性；
- 评估测试结果和问题报告，向干系人提出是否需要做进一步测试的建议；
- 评估给定的测试用例集是否达到了一定的覆盖率水平；
- 监控降低风险的活动，提出改进建议（包括整理结果）。

等级 6：创建（K6）

考生需要将各个知识点结合在一起，形成一个连贯的或功能性的实体。典型的应用方式是将知识点重新组合成一种新的模式或结构，设计过程以完成某些任务或者发明一种产品（例如，满足特定目的的构建习惯）。

关键词：生成（Generate）、推测（hypothesize）、计划（plan）、设计（design）、构建（construct）、生产（produce）。

举例

- 创建合适的风险管理过程，既包括严格执行的内容也包括非正式的内容；
- 创建测试方法，充分考虑公司政策、项目/产品、测试目标、风险和时间表，形成动态平衡的分析策略；
- 针对不同的评审类型要素，为组织构建有效的评审过程。

关于认知水平的学习目标的详细内容，请参考 [Anderson]

1.9 期望

本大纲的学习目标，旨在指导考生满足以下能力：

- 对提升测试过程提出建议；
- 确保测试的过程改进可以在组织内有效展开，并最大可能的取得成功；
- 在组织或项目内起到测试专家的作用。

考生通过了专家级的资格认证并不意味着马上就可以在测试过程改进领域成为世界级的专家。我们希望通过 ISTQB 专家级资格认证的人员，在他们组织内部进行测试过程改进的过程中或项目启动时，能够提供专家级的支持。

2. 改进概要

285 分钟

关键词:

戴明环(Deming cycle), EFQM 卓越模型(EFQM Excellence Model), IDEAL, 基于制造的质量(manufacturing-based quality), 基于产品的质量(product-based quality), 回顾会议(retrospective meeting), 软件生命周期(software lifecycle), 软件过程改进(SPI) (Software Process Improvement (SPI)), 标准(standard), 测试工具(test tool), 全面质量管理(TQM) (Total Quality Management (TQM)), 超越卓越的质量(transcendent-based quality), 基于用户的质量(user-based quality), 基于价值的质量(value-based quality)。

改进概要的学习目标

2.1 为什么改进测试?

- LO 2.1.1 (K2) 举例说明测试改进的典型原因。
- LO 2.1.2 (K2) 比较测试改进与其它改进的目标和措施。
- LO 2.1.3 (K6) 向所有干系人阐述建议测试过程改进的原因, 表明测试过程改进是如何与商业目标挂钩并解释测试过程改进与其他过程改进的关系。

2.2 改进范围

- LO 2.2.1 (K2) 了解能够改进的测试的不同方面和相关方面。

2.3 质量观点

- LO 2.3.1 (K2) 比较不同的质量观点。
- LO 2.3.2 (K2) 把不同的质量观点映射到测试中。

2.4 通用改进过程

- LO 2.4.1 (K2) 理解戴明环步骤。
- LO 2.4.2 (K2) 比较改进过程的两个通用的方法(戴明环与 IDEAL 框架)。
- LO 2.4.3 (K2) 从测试过程改进的角度, 列出各个卓越基本概念的例子。

2.5 改进方法概述

- LO 2.5.1 (K2) 比较基于模型的方法、分析型方法和混合型方法的特点。
- LO 2.5.2 (K2) 了解混合型方法的必要性。
- LO 2.5.3 (K2) 了解在人员技能方面的改进需求, 并解释在测试人员的配备、培训、咨询和指导等方面的改进。
- LO 2.5.4 (K2) 了解测试工具的引入如何改进测试过程的不同部分。
- LO 2.5.5 (K2) 了解改进的其他方法, 例如, 利用软件生命周期中的定期审查方法实施改进, 使用包含改进周期的测试方法(如在 SCRUM 中的项目回顾)实施改进, 通过采纳标准的方法实施改进, 以及通过聚焦于测试数据、测试环境等资源的方法实施改进。

2.1 为什么改进测试?

以软件为主导因素的系统构建越来越具有挑战性, 其在中发挥的作用也越来越重要。新的方法、技术和工具不断用于支持开发与维护任务。正因为软件在我们的日常经济和社会生活发挥了如此重要的作

用，迫使软件工程重点关注软件的质量问题。社会不再接受质量差的软件。软件的失效可能导致巨大的商业损失，甚至是灾难性的损失，例如，人员伤亡。

在下列情况，应采取测试过程的改进：

- 当前的业务和组织面临挑战；
- 当前已交付系统的维护面临挑战；
- 当前测试和质量保证面临挑战。

在这种情况下，作为可以采取的质量手段之一，测试学科的重要性迅速增长。通常，项目在测试上要花费大量的预算。

组织每天都要面临更严峻的业务目标，例如更短的产品上市时间，更高的质量和可靠性，以及更少的成本等。我们开发和生产的许多产品，其中大部分的开发成本都与软件相关。同时，现在软件开发可以选择外包或与其它地区合作开发的方式。随着软件开发朝着更多复用和平台架构的趋势发展，集成和测试正在成为直接影响产品质量以及有效和高效实施整个开发和生产过程的关键活动。测试人员可能的测试对象是软件产品，软硬件集成的组合产品，或者软件与其它不同媒介产品集成的组合产品。

软件的重要性越来越显著，规模不断增长。消费类产品的软件数量和专业产品复杂度大约每 24 个月翻一番。软件的复杂度直接影响到各个“单元”软件（例如，功能点）缺陷的数量。由于市场要求在更短的时间内以更低的成本开发和生产更好更可靠的产品，因此更高的测试效力和效率不再是可选项，而是成功的基本要素。

交付的系统包含软件代码，也包含其他产品和服务，可能也包括硬件、中间件和固件。在某些环境中，交付的服务可能包含新版本，工作量变更和新架构。这样导致测试的含义可能延伸为组织在业务过渡期的演练，直到组织新架构完全工作。

测试的范围不一定限于软件系统。此外，购买和使用软件的人们不仅仅需要代码，还需要服务和产品，例如业务流程、培训、用户手册和售后支持。测试的改进必须在更广泛的质量目标上进行 – 组织的目标，客户组织的目标以及 IT 组/团队的目标。

测试过程改进发生在各个商业/组织过程改进和 IT 或者软件过程改进中。

引起测试的业务改进的典型原因有：

- 商业组织的测试服务虽然提供了高质量的工程方法，但花费时间过长。如果目标是在保持高质量的同时缩短上市时间，那么测试改进工作可能集中于：
 - 不降低有效性的情况下提高效率来缩短测试时间；
 - 增加早期的测试活动（例如，静态测试）来缩短在生命周期后期解决缺陷的时间。
- 需要提高交付产品的质量，这种情况下，测试时间和测试成本的增加相对于改进的质量而言是合理的花费；
- 期望提高测试人员的能力以提供可预见性和更好的报告；
- 需要提供第三方服务的组织达到特定的能力要求；
- 需要通过降低测试成本来节约费用；
- 期望通过将测试融入到软件开发过程中来缩短整个项目的生产时间；
- 期望通过改进测试来减少失效的成本；
- 需要证明符合适用的标准（第 2.5.4.5 节）。

测试过程改进也可能发生在组织或者业务改进过程中。例如，可能需要通过下列方式进行管理：

- 全面质量管理（TQM）；
- 国际标准化组织 ISO 9000:2000；
- 卓越框架如“欧洲基础质量管理”（EFQM）卓越模型或其他类似的模型；
- 六西格玛。

测试过程改进还可能发生在 IT 以及软件过程改进过程中，可能需要通过下列方式进行管理：

- 能力成熟度模型集成 (CMMI) (第 3.2.1 节) ;
- ISO/IEC 15504 (第 3.2.2 节) ;
- ITIL [ITIL], [ITIL2] ;
- 团队与个人软件过程 [Humphrey]。

2.2 改进范围

软件过程改进 (SPI) 是针对产品质量、过程有效性和过程效率的持续改进, 最终实现软件产品的质量改进。

测试改进是在整个软件过程环境中, 对测试过程效力和/或效率的持续改进。这意味着测试改进可能超越测试过程本身, 比如扩展到基础设施、组织和测试者的技能。此外, 测试过程改进可能意味着需要对需求管理和开发过程其他部分的相关方面进行改进。反过来, 测试过程改进也可以被整体 SPI (软件过程改进) 所驱动。

测试目标必须始终与商业目标相一致。对于组织或者项目而言取得最高的测试成熟度并非总是最优选择。

2.3 质量观

在一个项目中可能使用质量的几个定义, 而项目全体成员对此并没有意识和承认。重要的是必须认识到没有“绝对正确”的质量定义。测试过程改进必须考虑最适合组织的质量观。本节将讨论这些质量观。

以 [Garvin Paper 84] 为基础, [Trienekens and van Veenendaal 97] 通过实例解释了 5 个软件质量维度。这 5 个维度分别是:

- 产品;
- 制造;
- 用户;
- 价值;
- 超越。

根据测试过程中可能使用的测试类型、测试级别和测试技术, 质量观可以通过使用角色和其代表性的观点在静态和动态测试活动中表述 [参见 Evans04 的例子]。

对于某个特定产品、服务或者项目, 如何定义“质量”取决于上下文。不同行业有不同的质量观。安全关键性产品需要在产品和生产质量定义方面进行特别强调。娱乐和游戏类产品需要用户定义质量, 可能还需要定义在其他行业中一般不会考虑的产品属性——比如将“令人心动”作为衡量可用性的一个方面。作为创新产品发布的软件需要基于价值定义, 因为如果投入更多时间以获得更优产品, 可能会错失市场。对于大多数商业或者定制软件来说, 平衡质量的各个方面将为客户提供最好的服务。对于这些产品, 应该这样自问: 能够提供的最大数量或最高等级特性 (基于产品) 支持用户完成他们的任务 (基于用户), 与此同时确保最优的投入产出比 (基于价值), 并且确保在项目管理中遵循可重复的, 有质量保证的过程 (基于制造)。

与这些质量观相关联的度量将在第 4 章分析方法和第 6 章改进过程中详细讨论。

2.4 通用改进过程

2.4.1 戴明环

持续改进涉及制定改进目标、采取措施达到目标，以及当目标达到后制定新的改进目标。持续改进模型就是为了支持这个概念而建立的。

戴明环是实现持续改进的有用通用框架，它包含以下步骤：

- **计划：**对质量属性、成本和服务等级设定目标。这个目标可能初始时被管理层作为商业改进目标，并先后分解到个人“控制点”，而且能够通过检查（见下面描述）这些“控制点”确定改进活动已被实施。定义的目标应该是可度量的（详情参见 4.4 章）。对当前实践情况和技能进行分析，然后才能制定改进计划改进测试过程；
- **实施：**制定计划以后，执行多个活动。包含人力资源的投资（比如培训和指导）；
- **检查：**在计划步骤确定的控制点通过特定的度量进行跟踪，并且识别出偏差。在特定时间段内可以预测每个度量的变化。将这些期望的变化与实际观察到的变化进行比较，提供实际与期望之间的差异信息；
- **行动（有时也被称为“分析/行动”）：**利用收集到的信息，识别改进绩效的机会并划分优先级。

在前 2 步（“计划”与“实施”）主要是确定重要问题的意识。在后 2 步（“检查”与“行动”）中，最常用统计方法与系统分析技术，以帮助确定统计学意义、依赖关系与进一步改进领域。

2.4.2 IDEAL 改进框架

IDEAL 框架 [IDEAL 96] 是上述戴明环的一个实例。它提供了可在改进测试过程中应用的涵盖以下阶段和子阶段的过程改进框架：

- **初始 (Initiating)**
 - 确定改进的原因；
 - 建立环境和得到支持；
 - 建立改进的基础架构。
- **诊断 (Diagnosing)**
 - 评价和描述当前实践状况；
 - 给出改进建议并记录阶段性成果。
- **创建 (Establishing)**
 - 制定策略和优先级；
 - 成立测试过程组 (TPG)（参见 7.1.1）；
 - 计划行动。
- **行动 (Acting)**
 - 定义过程与度量；
 - 计划与实施试点；
 - 计划、执行、跟踪工作准备就绪。
- **学习 (Learning)**
 - 分析经验教训并且文档化；
 - 修订组织方针。

2.4.3 卓越的基本概念

卓越基本概念被广泛应用于全球组织卓越模型中，使用以下 8 大标准衡量组织。欧洲质量管理基金会（EFQM）提供了一个实例，在本大纲的参考文献部分也提供了在欧洲之外使用的类似模型索引。

卓越基本概念如下（如[EFQM-Web]描述）：

- **结果取向：**“卓越依赖于所有相关人员的需求得到平衡和满足。（相关人员包括员工、客户、供应商、社会整体以及在组织中有财务利益的人）”
- **以客户为中心：**“客户是产品和服务质量的最终仲裁人，客户忠诚度、占有和扩充的市场份额最好通过清晰地聚焦于现有和潜在客户的需求得以优化。”
- **领导力与恒定的目标：**“在组织里和使组织和成员能达到卓越的环境里，组织的领导表现创造了统一明确的目标。”
- **通过过程和事实来管理：**“当所有相关的活动都被很好的理解并进行了系统化的管理，而且与当前运营有关的决策都已制定，那么组织运营将更加有效。使用可靠的信息，包括利益干系人的看法，实施改进。”
- **人员成长和参与：**“组织成员的潜力只有通过一致的价值观和信任与授权文化才能够完全释放出来，这种文化鼓励人人参与。”
- **不断学习，创新与改进：**“在不断学习、创新和改进文化中，基于管理和知识共享的组织绩效，组织绩效可达到最大化。”
- **合作伙伴关系的建设：**“当组织能够与合作伙伴保持双赢关系，建立在信任基础上，知识得到共享和整合时，组织的运作将更加有效。”
- **企业社会责任：**“组织及其成员的长期利益采用道德方法与超越社会大众的期望和规定时，能够得到最大保障。”

其他组织的质量和卓越做法，如六西格玛和平衡计分卡也提供了讨论组织目标、决定如何达到目标和度量目标，以及是否实现的方法。

2.5 改进方法概述

2.5.1 基于模型的改进方法概述

为了提高产品质量，软件领域致力于改进开发过程。已经广泛用于改善开发过程的方法是能力成熟度模型（CMM）。能力成熟度模型和它的后续版本能力成熟度模型集成（CMMI），以及 ISO/IEC 15504 往往被视为系统和软件过程改进的行业标准模型。

尽管事实上测试是项目成本的重要部分，但在各种软件过程改进模型中对测试的关注很少，例如，CMMI。作为解决方案之一，测试社区创建了与之对应的改进模型。

在 ISTQB 高级大纲中把测试过程改进作为测试行业的一个关键领域并确定了两种基于模型的改进方法：

- 过程模型定义了测试最佳实践的基本框架，并且定义了如何按部就班的对测试的不同方面进行改进。过程模型的例子有测试过程改进模型（TPI Next®）和测试成熟度模型集成（TMMi）。这些内容将在 3.3 中具体描述。ISTQB 高级大纲中也列出了应用较少的其他模型；
- 内容模型是非按部就班式的；不要求必须按一定的顺序进行改进。相反，定义了当妥善执行时会对测试过程有价值的特定活动。系统化测试和评估过程（STEP）和关键测试过程（CTP）是两个内容模型方法的基本例子。将在 3.4 中详述。

2.5.2 分析型改进方法概述

典型的分析型改进方法涉及分析特定的度量方法和标准评估测试过程的当前状况，决定应该采取何种改进步骤以及如何度量改进的影响。目标-问题-度量（GQM）方法是一种典型分析型方法，将在 4.3 章中详述。

第 4 章将更加详细的论述分析型改进方法。

2.5.3 混合型改进方法

混合型改进方法可应用在已达到更高过程成熟度级别的项目，（无论是基于模型的改进方法、分析型改进方法或者二者混合），可以应用于设置和度量其他项目的改进目标。这是一种通用的改进方法，混合型改进方法可在已实施改进的组织项目中向组织的其他项目推广，不需要完全依赖于定义好的最佳实践标准的过程模型。（参见 3.3 节）

2.5.4 其他测试过程改进方法

测试过程的改进还可以通过下述某些方面实现。注意，这些方面也已在 2.5.1 的模型中提到。

2.5.4.1 通过提高人员技能完成测试过程改进

测试过程改进也可以通过提高执行测试、管理测试或者基于测试进行决策的团队成员的测试理解、知识和技能来实现。可以是测试人员、测试经理、开发人员以及其他信息技术组成员、其他经理、用户、客户、审计人员和其他利益相关人员。

提升技能和能力可以通过以下方法来达到：培训、宣传讲座、指导、训练、与其他团队联系交流、利用知识库，阅读以及其他教育活动。

技能水平可能与职业发展道路和专业提升相关，例如，SFIA（信息时代技能框架）[SFIA 网站]

需要提升的技能和能力可以包括测试技能、其他信息技术技能、管理技能和软技能或领域技能。例如，

- 测试知识--测试的基本原则，技术，工具等；
- 软件工程知识--软件，需求，开发工具等；
- 领域知识—业务流程，用户特点等；
- 软技能—沟通，有效的工作方式，报告等。

测试过程改进人员所需要的技能将在 7.3 中详述。这些技能不仅仅测试改进组需要，整个测试团队都需要，特别是高级测试人员和测试经理。

改进团队的工作重点如下：

- 提高对开发过程和具体业务中测试活动/测试改进活动的效益和局限的认识；
- 提高知识和技能水平来支持在现有的或者改进过的测试过程中的活动；
- 提高个人能力，使他们能够开展活动；
- 建立明确定义的测试的角色和职责；
- 改善增加竞争力与奖励，表彰与职业发展之间的关系；
- 激励员工。

2.5.4.2 通过使用工具来实现测试过程改进

测试过程改进可以通过成功的引入工具来实现。这些改进可能是效率的提高，效力的增强，质量的提高或者所有以上这些。例如，

- 测试管理工具把编写测试用例的工作与记录缺陷的工作对应起来；
- 代码覆盖工具支持在组件级别实施出口准则。

开发测试工具的目的主要是提高测试效率，改进对测试的控制或者提高交付质量。实施测试工具的应用并非一件简单的小事，成功地运用取决于根据改进的需要选择合适的工具、以及通过合适的流程来运用工具。这些内容已经在 ISTQB 基础级和高级大纲中详述。

在过去的几年里测试工具的应用范围、多样性和应用领域剧增。当在测试过程的任何部分和整个软件生命周期内的任何要点中考虑潜在的过程改进时，测试过程改进组织应该考虑是否需要引入工具来支持改进。与 CASE 类似（计算机辅助软件工程 CASE），CAST（计算机辅助软件测试）也涵盖了很多现成从应用到平台的测试工具。对于这些工具的运用常常可以为测试过程的生产效率带来显著的提高。

过程改进人员也可以使用工具来协助收集、分析和报告数据，包括统计分析和过程建模。这些工具不一定是测试工具。

改进小组的重点是：

- 选择和应用工具辅助改进团队的工作，比如，统计分析和过程建模工具；
- 选择能够支持已经识别的改进工具，比如，在开发人员执行静态测试时，选择使用静态分析工具评估代码质量；
- 改进选择工具和应用工具的过程，比如，对工具实施试点发现的问题进行原因分析。

2.5.4.3 不同测试方法的测试过程改进

测试结束阶段是测试的主要阶段之一，该阶段可能进行项目回顾或总结经验教训。

使用持续改进循环是很多改进方法的核心。无论是顺序的还是迭代的生命周期都可能包含开发结束的事后回顾，阶段结束的回顾，经验教训总结会议，其他收集反馈和实施改进的机会。

较短迭代周期的迭代方法（比如敏捷方法）中，反馈循环会更加频繁发生，所以实施改进的机会更加频繁。比如，像 SCRUM 这样的敏捷开发生命周期模型中把持续改进的循环作为一般项目过程的一个输入部分，同时在每个迭代（“冲刺”）结束时进行项目回顾和过程改进（包括测试过程）。

在探索性测试中，每个测试阶段后都有对“接下来最好集中在哪里进行测试”的评估。这使得每个阶段结束时都进行一次改进循环。

在脚本化/结构化的测试中，制定测试策略/计划/脚本的工作可能会削弱在测试项目中过程改进的意愿。但是，仍然可能通过更频繁经验教训学习或者其他过程审查重新把注意力集中到测试改进上。尤其当遵循基于风险的方法时，基于风险的测试更加需要改变（改进），以应对新风险和变更风险，因为业务、产品和项目风险在整个系统生命周期中一直变化。

2.5.4.4 测试过程改进与采用的标准和法规相关

过程改进受标准和法规制约。例如，美国食品和药物管理局（FDA）或者美国的萨班斯-奥克斯利法案等领域规定的标准可能意味着在相应领域中必须执行某些特定的改进。

与标准的兼容可能是源于法律、法规，或者商业考虑，或者为了改进团队、组织、或者跨国界的沟通。标准还可能用作设置或度量内部过程以及通过其它组织设定基准进行改进。

测试过程改进组织可以通过选择合适的标准（参照高级大纲）进行改进，并且定义如何使用这些标准。标准可能用于以下目的：

- 为了满足审计程序的要求而实现兼容性；
- 为了与其他组织比较而作为度量基准；
- 作为改进思路的源头和例子来协助选择合适的改进；
- 在不断变化的合作伙伴网络中作为标准实践的来源，促进网络中的系统和流程交互更顺畅；
- 作为改变过程的框架。

2.5.4.5 针对特定资源的测试过程改进

测试环境、测试数据以及其他的技术资源的管理可能超出测试小组的控制。如果需要把这些领域作为改进重点，控制这些资源的团队也需要参与改进过程。

建立和管理测试环境所需的过程包括：

- 测试环境的需求定义；
- 测试环境的设计，建立和验证/测试；
- 测试环境的验收；
- 实施过程；
- 容量规划；
- 测试环境的配置管理和变更管理；
- 访问控制过程；
- 在团队之内/之间预定或调度测试环境；
- 拆除测试环境。

支持设计、采集和管理测试数据所需的过程包括：

- 测试分析和设计活动；
- 测试实施活动；
- 备份，恢复和归档过程；
- 在具体数据集合上配置和变更管理；
- 适用的数据安全程序（可能由法律规定）。

组织层面要求的环境和数据资源的改进可能包括要求节约成本和降低能源消耗，满足组织的社会责任和环保目标。这些改进可能着眼于提高测试环境的部署和利用效率，例如，测试环境的虚拟化。

一些过程模型（参见 3.3 节）明确在评估和建议中包含这些资源，但是如果使用了分析型方法，比如根本原因分析（参见第 4 章），与资源相关的改进就可以加入到改进建议中作为考虑因素。

对于改进团队而言，与资源相关的改进重点包括以下内容：

- 确定在测试团队控制范围以外的改进区域；
- 与控制团队合作，必要时通过管理方式提升级别；
- 与测试之外的其他改进团队合作完成改进；
- 在测试团队确定和实施改进，并且在整个测试团队中提供和管理资源。

3. 基于模型的改进

570 分钟

关键词:

CTP, CMMI, 连续式表示法 (continuous representation), GQM, 成熟度等级 (maturity level), 软件过程改进 (software process improvement), 阶段式表示法 (staged representation), STEP, TPI, TMMi, 内容模型 (content-based model), 过程模型 (process model)。

基于模型的改进学习目标

3.1 基于模型的方法简介

- LO 3.1.1 (K2) 结合基本通用属性理解测试过程改进模型的属性。
- LO 3.1.2 (K2) 比较连续式和阶段式方法, 包括它们的优缺点。
- LO 3.1.3 (K2) 对使用模型时所做的假设进行综合性总结。
- LO 3.1.4 (K2) 通过比较, 讨论基于模型的方法优缺点。

3.2 软件过程改进模型

- LO 3.2.1 (K2) 了解 CMMI 模型与测试的具体相关方面。
- LO 3.2.2 (K2) 将 CMMI 和 ISO/IEC 15504-5 中的测试过程改进适用性与专门为测试领域开发的测试过程改进 (TPI) 模型适用性进行比较。

3.3 测试过程改进模型

- LO 3.3.1 (K2) 总结 TPI Next 测试过程改进模型的背景和结构。
- LO 3.3.2 (K2) 总结 TPI Next 测试过程改进模型的关键领域。
- LO 3.3.3 (K2) 总结 TMMi 测试过程改进模型的背景和结构。
- LO 3.3.4 (K2) 总结 TMMi2 级的过程域和目标。
- LO 3.3.5 (K2) 总结 TMMi3 级的过程域和目标。
- LO 3.3.6 (K2) 总结 TMMi 和 CMMI 之间的关系。
- LO 3.3.7 (K5) 对于给定的场景从 TPI Next 或者 TMMi 选择合适的模型。
- LO 3.3.8 (K3) 使用 TPI Next 测试过程改进模型进行非正式评估。
- LO 3.3.9 (K3) 使用 TMMi 测试过程改进模型进行非正式评估。
- LO 3.3.10 (K5) 使用 TPI Next 模型或 TMMi 模型评估测试组织。

3.4 基于内容的模型

- LO 3.4.1 (K2) 总结基于内容的 STEP 模型的背景和结构。
- LO 3.4.2 (K2) 总结活动、工作产品和 STEP 模型的作用。
- LO 3.4.3 (K2) 总结基于内容的 CTP 模型。
- LO 3.4.4 (K2) 总结基于内容的 CTP 模型关键测试进程。
- LO 3.4.5 (K2) 总结基于内容的 CTP 模型度量的作用。
- LO 3.4.6 (K2) 比较在基于内容的方针中度量使用方法和在分析方针的度量使用方法 (第 4 章)。

3.1 基于模型的方法简介

3.1.1 测试过程改进模型的理想特性

测试过程改进模型具有以下特点：

- 易于使用；
- 公开；
- 可以得到顾问支持；
- 不是商业机构的营销工具；
- 被专业机构接受；
- 提供改进；
- 提供多种渐进式小改进；
- 基础牢固，具有实践性、经验性、理论性、发布性和合理性；
- 提供如何评估、确定并实施改进的细节；
- 可量化的改进；
- 可裁剪（针对具体项目）；
- 模型描述了将要开展改进活动的程度；
- 支持改进的顺序；
- 改进采用阶段式或连续式表示；
- 测试内容的详细程度；
- 特定测试问题的建议解决方案的丰富性和多样性；
- 评审所需的正式认可等级；
- 可以对组织进行认证。

3.1.2 连续式和阶段式表示法

过程模型通过使用阶段式或连续式表示法描述过程的成熟性。

阶段式表示法提供了系统性的“一次一个步骤”的方法来改进。该模型架构规定了组织必须通过的各个阶段，组织的测试过程有序改进。在迈向更高阶段之前，确保当前阶段已经建立充分的流程成熟度等级（在 TMMi 中，称为成熟度等级）。改进的重心是实现事先定义的过程领域（如 TMMi2 级中的测试计划和测试环境）的特定能力等级，该能力等级分配为某个成熟度等级（例如 TMMi4 级）。成熟度等级表示实现改进组织流程中的良好定义的演进高度。

阶段式模型的优势主要是概念简单。它提供了成熟度等级的评级，通常用于外部管理沟通和公司内部资质评估（例如，客户公司可能会要求所有的潜在供应商实现最低为 TMMi4 级的过程成熟度）。阶段式表示法的问题是灵活性有限。组织可能在所需过程领域的许多方面实现了相当高的等级，但仍无法实现整体成熟度等级。如果使用“要么拥有一切，要么一无所有”或“一劳永逸”的方法将无法实现预期的业务目标。

连续式表示法没有规定开发过程必须通过的成熟度等级。TPI Next 模型使用连续式表示法（见 3.3.1 节）。应用连续式表示法的组织可以根据希望达到的特定目标，从不同类别中选择具体领域进行改进。连续式表示法允许实现每个过程/关键域中的单个能力等级（例如，TPI Next 模型关键领域“测试策略”的成熟性可以在多个不断提高的等级中实现）。

连续式表示法的优点主要是灵活性，如下所示：

- 组织可以选择改进一个单独的问题点或者一组与组织业务目标紧密联系的进程/关键域；
- 能够以不同的速率改进不同的过程域；
- 可以避免阶段式模型的“要么拥有一切，要么一无所有”的缺点。

3.1.3 使用模型的假设

使用基于模型方法改进时，需要一些隐性假设：

- 模型描述了其作者认为的“最佳实践”。这个术语实际上可以更好的描述为“良好做法”，它可以为改进测试进程受益。对于测试过程改进人员而言，对于特定项目，必须判断这个“最佳”的真正含义；
- 使用基于模型的方法的假设前提是为了改进，必须符合“最佳实践”；
- 模型假设“标准”的项目和/或组织。因为所有的项目在某种程度上各自不同，基于模型的方法并不总是同样适用于所有项目。

使用模型不是纯粹机械地以检查表为基础的过程；为了获得最大效益必须正确应用经验和判断。通用模型需要解释以便考虑具体的项目因素。

下面的例子说明了可能需要模型解释的因素：

- 生命周期的应用（如传统的 V 模型或敏捷开发过程）；
- 使用的技术（如 Web、面向对象技术）；
- 系统架构（如分布式系统、SOA、嵌入式系统）；
- 风险级别（如安全关键系统与一般的商业系统）；
- 测试方法（例如，脚本式与探索式）；
- 组织部门内的应用适用性。

使用一般模型（并不仅仅是本大纲列出的过程改进模型）会有一些问题。这些问题包括：

- 缺乏模型由来的知识；
- 使用模型可能将原因和影响过于简单化；
- 在不当情况下应用模型；
- 使用该模型时没有考虑是否有用。仅把使用模型本身作为目标，而不是作为一个改进过程的工具；
- 用户容易认为遵循模型就可以理解模型；
- 缺乏使用模型的技能或经验。

3.2 软件过程改进模型

软件过程改进（SPI）模型通常只覆盖测试的一般性问题。为此，测试改进模型（如本课程 3.3 节内容）需要特地为测试过程进行开发。

3.2.1 CMMI

CMMI 可以通过两种方法和方式实施：阶段式表示法或连续式表示法。阶段式表示法有五个“成熟度等级”，每个等级都在前一个等级内的过程领域之上构建。连续式表示法组织可以集中精力改进其自身需要的主要领域，而不考虑其他领域。

阶段式表示法为组织层面提供了更多关注并在行业中取得最大收益。它还保持了与 CMM 的一致性，并且可以客观地评价组织的成熟度等级，而连续式表示法一般认为更加的灵活。

在 CMMI 中，确认和验证的过程域专门引用了静态和动态测试过程。过程域的技术解决方案和产品集成同样可以处理测试问题。

除了与测试相关的过程域，以下过程域也为实现更结构化的测试过程提供了支持：

- 项目计划；
- 项目监视和控制；
- 风险管理；

- 配置管理；
- 度量和分析；
- 原因分析和解决。

虽然 CMMI 模型中考虑了软件开发和测试的关系，但是专门的测试过程模型如 CTP、STEP、TMMi 和 TPI Next 模型提供了更详细的测试和测试过程。

CMMI 和测试的关系在 TMMi 模型中更加明确 [TMMi 基金网站]。

3.2.2 ISO/IEC 15504

ISO/IEC 15504-5 是国际软件过程改进的标准之一，它定义了许多过程类，包括工程、管理、组织、客户、供应商和支持。“支持”过程类（SUP）包括与测试相关的独立进程，包括验证和确认。过程能力等级通过预先定义的一组过程属性衡量，并且应用连续式表示法。

3.3 测试过程改进模型

2.5.1 节概述了各种类型的过程改进模型。本节和 3.4 节中将详细介绍在实践中应用的主要改进模型。

3.3.1 测试过程改进模型 (TPI®)

ISTQB 认证测试高级课程中介绍了 TPI 模型 [Koomen / Pol 99] 并陈述了与测试经理有关的学习目标。其继承版本于 2009 年发布：TPI Next [Sogeti 09]。

TPI Next 模型属于过程模型，它把测试过程的主要内容都区分开来。TPI Next 模型的核心要素是 16 个关键域，每个关键域都涵盖了测试过程的具体内容，如测试策略、度量、测试工具和测试环境。

每个关键域的成熟度等级都支持该关键域的透彻分析，每个成熟度等级都由特定的检查点来确定。多个关键域的检查点集合可以组成小改进步骤。使用集合减少了“一边倒”改进的风险（例如，一般不建议在缺陷管理和报告这两个关键域上达到一定的成熟度之前，要求度量关键域达到很高的成熟度）。因此，TPI Next 模型使用连续式表示法，但仍规定在其关键域中按一定的顺序进行一系列的过程改进。

可以通过涵盖全部关键域的成熟度矩阵总结和导出结果。当综合分析结果全部覆盖了关键域，则成熟度等级可以归结到整个测试过程。这些成熟度等级被命名为初始、控制、效率和优化。

大量反映优秀测试实践且具有优先级的改进建议可以协助测试组织定义合适的成长路径。成长目标及其实现的定义可以根据测试组织的需求和能力进行裁剪。

通用方法使 TPI Next 模型独立于任何 SPI 模型。它涵盖了测试工程方面和管理决策支持方面。

原始的 TPI 模型已经被调整为特定行业。例如，“汽车 TPI”定义了一个额外的关键域（“集成”）并被德国汽车制造商采用。

TPI Next 和软件过程改进模型（如 3.2 节提到的 CMMI 和 ISO/IEC 15504）之间存在着映射。

3.3.2 测试成熟度模型集成 (TMMi)

ISTQB 认证测试高级课程中介绍了 TMMi 模型并阐述了与测试经理有关的学习目标。

TMMi [TMMi 基金网站] 是过程改进的阶段式结构。它定义了以下几个等级或成熟度等级，组织的测试过程从混乱状态开始不断地演进：

- 初始;
- 受管理的;
- 已定义的;
- 管理和度量;
- 优化。

TMMi 中的五个成熟度等级规定了测试过程改进的成熟度层次和进化路径。每个阶段都确保足够的改进为下个阶段打下基础。TMMi 的内部结构具有丰富的测试实践，可以学习并系统地应用以支持增量式的软件测试改进过程。

TMMi 的每个成熟度等级都有一套过程域，组织需要把重点放在过程域上以实现该等级的成熟性。例如，在 TMMi 2 级中“受管理的”过程域是：

- 测试方针和策略;
- 测试计划;
- 测试监视和控制;
- 测试设计和执行;
- 测试环境。

TMMi 的模型结构主要基于 CMMI 的结构。这非常有利，因为很多人/组织都已经熟悉了 CMMI 的结构。CMMI 结构清晰地区分了要实现的必要（目标）实践或建议（具体做法、典型的工作产品等）实践。

3.3.3 比较 TPI Next 模型与 TMMi 模型

本节提供了大纲涵盖的两个测试过程改进模型的简单对比。根据具体项目情况和应遵循的改进目标，可以首选两个模型之一。

TMMi 和 TPI Next 之间的区别如下表所示：

类别	TPI Next	TMMi
类型	连续式模型	阶段式模型
测试方法	使用通用的 TMap (Next) 实践作为参考	独立于测试方法
术语	基于 Tmap [Pol.M & Van Veenendaal. E 98]	基于标准测试术语
SPI	与特定的 SPI 模型没有直接关系，但是可能存在某些映射关系	与 CMMI 高度相关
关注点	测试具体关注点有 16 个关键域。在整个测试过程的每个关键域都应关注	每个成熟度等级的过程域都具有有限的详细焦点，需要关注其他测试问题，如易用性评估、质量控制、缺陷预防和测试度量流程。
方法	彻底的、业务驱动和测试工程方法	强烈关注管理层的承诺

3.4 基于内容的模型

基于内容的模型允许测试过程通过提供良好的测试实践的结构化描述和应遵循的整体方法来改进。大纲描述的基于内容的模型是 STEP 和 CTP。

3.4.1 STEP

系统化测试和评估过程 (STEP) [Craig02] 不要求遵循特定的顺序进行改进。为此，STEP 评估模型可以与 TPI (Next) 模型混合。

STEP 方法的依据“测试是生命周期活动”的观点，该生命周期活动从需求制定开始，以系统废止结束。

高级大纲提供以下细节：

- 该方法的基本前提；
- 定量指标的例子；
- 定性因素的例子。

3.4.2 CTP

关键测试过程（CTP）[Black03] 评估模型是以某些至关重要的测试过程为基本前提。这些关键过程，如果表现良好，将支持成功的测试团队。该模型确定了 12 个关键测试过程。

CTP 评估确定哪些过程是强壮的，那些是薄弱的，并提供基于组织需要的优先改进建议。在 CTP 评估期间，通常审查多项定量和定性指标。

评估找出薄弱环节后，可以开始改进计划。尽管模型为各个关键测试过程 CTP 提供了通用的改进计划，评估团队仍需要进行重大调整。

中国软件测试认证委员会 (ISTQB)

4. 基于分析的改进

555 分钟

关键词:

原因分析 (causal analysis), 因果图 (cause-effect diagram), 因果图 (cause-effect graph), 缺陷发现率 (Defect Detection Percentage), 失效模式和影响分析 (Failure Mode and Effect Analysis), 故障树分析 (Fault Tree Analysis), 指标 (indicator), 审查 (inspection), 测量 (measure), 度量 (metric), 帕累托分析 (Pareto analysis)。

基于分析改进的学习目标

4.2 原因分析

- LO 4.2.1 (K2) 使用因果图理解原因分析。
- LO 4.2.2 (K2) 在审查过程中理解原因分析。
- LO 4.2.3 (K2) 理解原因分析标准的异常分类。
- LO 4.2.4 (K2) 比较各种原因分析方法。
- LO 4.2.5 (K3) 使用原因分析方法描述某一给定问题。
- LO 4.2.6 (K5) 基于原因分析的结果, 推荐并选择测试过程改进措施。
- LO 4.2.7 (K4) 使用结构化方法选取缺陷进行原因分析。

4.3 GQM 方法

- LO 4.3.1 (K2) 描述“目标-问题-度量 (GQM)”方法。
- LO 4.3.2 (K3) 应用“目标-问题-度量 (GQM)”方法, 能够从一个测试改进目标得出适合的度量点。
- LO 4.3.3 (K3) 为测试改进目标定义度量点。
- LO 4.3.4 (K2) 理解数据收集阶段的步骤和挑战。
- LO 4.3.5 (K2) 理解解释阶段的步骤和挑战。

4.4 使用测量、度量和指标进行分析

- LO 4.4.1 (K2) 能够举例说明不同度量的类别, 阐述在某个测试改进情景中如何使用这些度量类别。
- LO 4.4.2 (K5) 在特定的改进场景下, 为跟踪改进趋势推荐适合的度量和指标。

4.1 简介

分析方法是基于实际的问题和目标确定需要引入的改进项, 这与第 3 章提到的基于模型的方法或基于内容的方法不同, 它们是基于“最佳实践”的通用模型而得出改进项。

客观的测试过程改进必须进行数据分析, 对于纯粹的定性评估而言, 数据分析也是很有价值的支撑, 否则, 缺乏数据支撑的定性评估有可能给出不准确的建议。

应用分析方法改进过程, 包括对测试过程进行分析, 以识别问题域并设定特定的项目目标。关键参数的定义和度量, 是评估改进措施是否成功中的必要内容。

分析类的方法可以与基于内容的方法结合使用，以验证结果和提供多元化信息，类似与 CTP（关键测试过程）方法结合使用。同时，基于模型的方法，有时也会提到基于分析的方法，将其作为独立于模型存在的一个或多个其它关键域，这与在基于模型的方法中有时会提到 TMMi 道理相同。

一般而言，分析类的方法也可以用于测试管理。测试经理在项目级别使用分析类的方法，测试过程改进人员在过程级别使用分析类的方法。

4.2 原因分析

原因分析是对问题进行研究以识别可能引起问题的根本原因。所确定的解决方案将会消除问题产生的原因，而不仅仅针对直接表现的问题症状。若不使用原因分析进行测试过程改进，其尝试有可能会失败，因为没有触及到真正的根本原因，相同或类似的问题仍会发生。

很多软件过程改进模型都强调使用原因分析作为持续改进软件过程成熟性的手段。

本文以示例方式阐述如下几种系统化的原因分析方法：

- 因果图（石川鱼骨图）；
- 审查过程中的原因分析；
- 使用标准的异常分类。

注意：也可以使用其他原因分析方法（见高级大纲），某些情况下常见根本原因检查表可以作为原因分析的输入，比如基于缺陷进行根本原因分析。

4.2.1 因果图

因果图（也称为石川鱼骨图）由制造业和其它行业 [Ishikawa 91] 发展而来，已应用于 IT 领域 [Juran]。这些图表提供一种机制，针对各种情景，识别和讨论引起问题的根本原因。

根据 [Robson 95]，使用因果图的步骤如下所述：

1. 在图表右边写出结果。
2. 绘制鱼骨，并根据上下文打上标签。鱼骨的所有权是工作组而不是管理者。
3. 工作组头脑风暴的规则提示：
 - 没有批评 – 这个阶段接受所有的想法
 - 飞轮 – 随机的、疯狂的想法，建立在他人想法观点之上的观点一样受欢迎
 - 观点的数量 – 将产生大量的观点
 - 记录所有观点 – 包括随机的、疯狂的和重复的观点
 - 不要即时评价观点 – 经过头脑风暴，休息一下，然后再评价这些观点
4. 使用头脑风暴方法 – 对可能的原因进行头脑风暴，向图中添加鱼骨。对第一层次的各个原因，使用检查表识别根本原因（可能位于图中不同部分）。
5. 让这些观点孵化一段时间。
6. 分析因果图，寻找根本原因和症状的集群。使用帕累托思想（80%的收益来自于 20%的努力）识别待解决的候选集群。

因果图也可用于从影响推导回根本原因，这在 [BS7925-2] 和 [Copeland 03] 中有所提及。

4.2.2 审查过程中的原因分析

ISTQB 基础级大纲描述了审查过程，并在 ISTQB 高级大纲中进一步得到阐述。软件审查过程 [Gilb & Graham] 是一种与原因分析不同的方法。

原因分析会议是有组织的讨论会议，一般持续两个小时，遵循一定的时间表和形式。

- 待分析的缺陷可以由审查组长或者由项目回顾会议选取。另外一种可选的方式是在分析会议的第一部分，首先识别要讨论的问题；
- 第一部分：缺陷分析（90 分钟）着眼于具体的缺陷和其特定的原因；
- 第二部分：通用分析（30 分钟），侧重于查明缺陷的大趋势。审查小组寻找缺陷的趋势、共性问题以及最近哪些进行的很好、哪些已得到改善或正在变得更糟糕。缺陷的分类（在高级大纲中引入）可以为这种分析提供宝贵的支持。通用分析也可以以动态测试中发现的特定缺陷为重点，为未来的测试策略提供基于风险的参考信息；
- 讨论必须保持所讨论的缺陷数量和严重程度与所投入的时间回报最大化。

缺陷分析中每个缺陷应包含如下内容：

- 缺陷描述 – 不是缺陷症状的描述而是对缺陷本身的描述；
- 原因分类 – 例如沟通、监督、教育、转录错误、过程；
- 原因描述 – 描述原因及任何导致原因的一系列事件；
- 创建缺陷的处理步骤 – 这与发现缺陷的处理步骤并不完全一样；
- 为消除根本原因建议采取的行动 – 必须具体可行。

4.2.3 使用标准异常分类

诸如[IEEE 1044]的标准提供了异常情况的通用分类，有益于理解故障引入时项目所处的阶段，检测到故障时的项目活动，纠正故障的成本，失效的成本，以及缺陷提出阶段和缺陷发现阶段的对比（也被称为缺陷遗漏）。

通用的分类方法可以在整个组织内对改进领域的统计数据进行分析。这种分类方法必须通过培训、工具和技术支持的方式引入，从而使任何人在使用事件管理系统时都明白如何使用和解释这些分类。分类信息可用于开发和测试过程的改进，以找出可产生经济效益的改进领域，并可跟踪改进措施直至成功。在整个生命周期的各个阶段，包括维护和运行阶段，都要记录所有被分析的缺陷。

4.2.4 选取缺陷进行原因分析

使用上述提到的方法，需要从大量的缺陷中选取具体待分析的缺陷。可采取以下多种方法相结合的方式选取待分析的缺陷：

- 帕累托分析（认为 20% 的缺陷代表了全体）；
- 统计数据的离群值（比如，使用统计方差值，即某一特定指标 $3-\sigma$ 值）；
- 项目回顾；
- 使用缺陷严重等级分类。

4.3 GQM 方法

GQM（目标-问题-度量）方法 [BasiliPapers], [Trienekens & van Veenendaal 97] 使用以下步骤确定合适的度量项：

- 设定特定的目标；
- 确定问题，对这些问题的回答将获得达成目标的相关信息；
- 确定度量项，对它们的测量将回答上述问题。

Basili 的 GQM 方法提供了一种反馈和评估的测量机制。GQM 使得度量：

- 围绕设定的特定目标；
- 应用于软件生命周期的产品、过程和资源的所有部分；
- 基于对组织状况、环境和目标特征及理解进行解释。

因此，适合于一个组织的度量不一定适合于另一个组织。定义合适的目标和问题，可以选择或定义合适的度量项，从而收集和分析合适的数据。

GQM 方法中的三个层次是：

1. 概念层 – 组织的“目标”，与产品质量、过程和资源（注：资源包括人、办公场所、硬件、软件）相关。这可能包括第2章中提到的关于质量的任何定义，生产力和精益求精的制造可以成为目标。可以用来定义目标。
2. 操作层 – 就其质量而言，描述产品、过程和资源的“问题” – Basili 将这些问题当做度量的对象。
3. 量化层 – “度量项”，可以是客观的（定量的、基于事实的），也可以是主观的（定性的，观点）。

目标可以引发一个或多个问题，问题可以引发一个或多个度量项。目标、问题和度量项可能与第 2 章中提到的质量的观点相关，或与第 2 章中提到的组织和 IT 框架要求的改善结果度量有关。

4.4 使用测量、度量和指标进行分析

4.4.1 简介

测量、度量和指标构成了改进计划的一部分。无论是正式或非正式开展这些改进均适用。无论数据是定量的还是定性的、客观的还是主观的，也同样适用。在改进过程中，受影响者的感受是衡量改进进展的有效手段。

测量、度量和指标在初始阶段有助于锁定目标域和改进机会点。在初始的改进行动中需要不断地查看以获取这些测量、度量和指标值，以便控制改进过程，确保变化朝着所希望的方向前进。

在软件生命周期的各个阶段（包括开发、维护和生产和使用阶段），都可以收集测量、度量和指标数据 [Nance & Arthur 02]。这些数据也可以用来获得其它的度量和指标。注意，在 4.4.2 节提到的与缺陷有关的所有项目中，区分缺陷优先级和严重级别很重要。测试经理可能使用特定的测量、度量和指标，尤其是在具体的测试估算活动和测试过程监控活动中。测试过程改进人员将在过程级别应用测量、度量和指标。

4.4.2 测试效力度量

4.4.2.1 缺陷检测率（DDP）

第一个指标，几乎可以使用于任何测试改进过程，也是大多数测试专家积极推荐的指标，称作缺陷检测率（DDP）。如果你已经在测试中发现了大多数（如果不是全部）的缺陷，而用户/客户在实际使用中只发现了很少的缺陷，这样的测试是良好的测试。

缺陷检测率定义为测试活动中发现的缺陷数除以已知的缺陷总数。可以按照每个测试阶段（例如集成测试、alpha 测试、beta 测试）计算 DDP，也可以计算所有测试阶段总的 DDP。DDP 是一个可以计算的度量项，计算 DDP 要在一个项目完成时，或者项目完成后的一段时间（例如 3 个月或 6 个月），在这段时间内有可能发现驻留的缺陷。

4.4.2.2 发布后缺陷率

这个指标定义为产品发布后一段时间内由用户发现的每千行代码的缺陷数。如果发布后缺陷率下降，那么用户的质量感受就会提升。

注意：这些缺陷度量项所提供的信息是与第 2 章中所讨论的从“制造”角度看待质量相关。

4.4.3 测试效率/成本度量

4.4.3.1 质量的组织成本

为了兜售测试改进的想法给别人，需要向他们展示质量改进的好处。为此，重要工作之一是通过度量测试成本和测试改进成本以及不测试成本（即失效成本），向别人展示测试活动以及测试改进的成本收益。

4.4.3.2 质量比率成本

“在静态测试（例如审查和评审）上的总投入”与“在动态测试（例如组件测试、集成测试、系统测试）上的总投入”之间的比值。如果这个比率增加，意味着提高了移除缺陷的效率。

4.4.3.3 早期缺陷检测

正如前面所述的效率指标，如果缺陷在过程较早的时候发现，将提高缺陷移除效率。这不仅可以应用于静态测试与动态测试的对比中，也可以应用于组件和集成测试与系统和验收测试的对比中。“早期缺陷检测”这个效率指标测量组件测试和集成测试阶段（早期的动态测试）发现的总缺陷数与动态测试阶段发现的所有缺陷数之比。

4.4.3.4 相对测试投入

基本指标之一是项目发布前总的测试投入（或成本）与总的项目投入（或成本）之比。发布后在维护和售后支持上的投入不计算在内。

4.4.3.5 测试效率

这个指标基于发现缺陷的数量和严重级别，与测试投入做比较（按照每个测试级别）。该指标用于判定测试投入是否聚焦于发现那些严重级别高的缺陷。测试效率也可能与测试任务的规模有关，比如测试点数量。

4.4.3.6 自动化水平

自动执行的测试用例的数目与执行的测试用例总数（包括手工执行和自动执行的）数目之比。

4.4.3.7 测试生产力

为产品准备的测试用例（或测试设计）的总数，与所需的测试总投入相关。当然这个效率指标也可以按照每个测试阶段测量。

注意：这些效率/成本度量项提供了从“价值”角度看待质量的信息（见第 2 章）。

4.4.4 交付周期度量

测试执行时的交付周期尤其重要，测试执行处于项目的关键路径上。交付周期定义为，标志一个或多个项目活动的开始和结束的两个里程碑之间的时间（天或周）。产品测试执行交付周期指标应当与产品的规模相关。这个指标可以按每个测试阶段度量，例如 **alpha** 或 **beta**。

4.4.5 可预测性指标

4.4.5.1 测试执行交付周期偏移

测试执行交付周期偏移是指估计的测试阶段所需要的测试执行交付周期与实际的测试执行交付周期之差。

为了达到改进的目的，需要测量相对于项目初始阶段估计值的测试交付周期偏移量，以及测量相对于测试执行开始阶段估计值的测试交付周期偏移量。

4.4.5.2 工作量偏移（或成本偏移）

工作量偏移是指估计的测试阶段所需要工作量（或成本）与实际工作量（或成本）之差。

4.4.5.3 测试用例偏移

测试用例偏移是指估计的测试阶段所需要的测试用例个数（或测试设计）与实际测试用例个数（或测试设计）之差。

4.4.6 产品质量度量

4.4.6.1 质量属性度量

可以用一些属性描述产品的质量（比如：功能、可靠性、可用性、效率、可维护性、可移植性）。这些都记录在[ISO 9126]以及它的后继者[ISO 25000]中。这些属性以及与此相关的指标在 ISTQB 高级大纲中已阐述。例如，与软件质量属性的可靠性有关的指标可以采取评价平均失效时间间隔（MTBT）和平均故障恢复时间（MTTR）。

软件测试过程是测量这些软件质量属性、获取信息的主要来源之一。通过测试过程发布有意义的、与产品质量相关的信息能力，可视为测试过程改进的潜在领域。

4.4.6.2 覆盖率指标

在测试过程中，通过测试获得的需求覆盖和代码覆盖可作为产品质量指标（假设产品质量的高低与覆盖程度相关）。

需求覆盖定义为已测试的需求数量与定义的需求总数量之比。通过区分已测需求数量和已验证通过的需求数量，可以将需求覆盖的定义进一步细化。如果覆盖率增加，测试将变得更好，预计产品质量也会变得更好。

代码覆盖定义为经测试执行过的软件代码与总代码百分比。ISTQB 基础级大纲中定义了不同级别的代码覆盖。

这些都提供了从“产品”角度看待质量的信息（见第 2 章）。

4.4.7 测试成熟性度量

这些度量依据所使用的模型，比如测试成熟度模型集成（TMMi）或测试过程改进模型（TPI Next），衡量组织级的测试成熟度等级。如果成熟度增加，意味着无法达成测试目标（与质量、交付周期和成本有关）的风险会降低。更多的细节可以参考高级大纲。

请注意这些度量在第 2 章提到的制造、产品和价值的角度描述质量，但是没有直接从用户的角度描述质量。因此测试经理可能希望采取特定的用户/客户满意度度量项、通过测量质量属性的可用性（尤其是与任务效率和效力相关的可用性）、或者通过定量的利益相关者意见的度量项来从用户角度衡量质量。

5. 选取测试过程改进的方法

105 分钟

关键词:

无

学习目标

5.1 选择测试过程改进的方法

LO 5.1.1 (K2) 选择最适合的测试过程改进方法的理由。

LO 5.1.2 (K5) 在某一特定的场景下以及在给定的改进范围内，提出测试过程改进方法的建议。

5.1 选取测试过程改进的方法

方法的选取依赖于以下因素：

- 考虑第 9 章中提到的成功关键因素；
- 考虑下面列出的一般指导原则。

下面列出的各项指导原则用于支撑决策过程。不要把这些当作必选的需求或者不可打破的规则。考虑 3.1.1 节提到的一般模型特征列表，有助于选择测试改进模型。

过程模型（例如 TMMi，TPI Next）最好应用于以下方面：

- 选择已经存在的单个测试过程，有助于建立新的软件测试过程；
- 类似项目之间建立对比或基准；
- 需要兼容软件过程改进模型；
- 公司的策略是达到某一指定的成熟度等级（例如，TMMi 3 级）；
- 需要定义良好的出发点，和定义好的改进路径；
- 需要测量测试成熟性，例如为了市场营销的目的；
- 组织内部尊重和接受过程模型。

内容模型（例如 CTP，STEP）最好应用于以下方面：

- 需要建立测试过程；
- 需要评估以识别当前测试过程的成本和相关的风险；
- 不需要按照 TMMi 或 TPI Next 所述，实施改进的顺序，而是根据业务需要确定；
- 需要裁减，以确保测试过程符合公司特定的背景；
- 期望或需求对已有的测试过程进行不连续的、快速的改变。

分析方法最好应用于以下方面：

- 需要针对具体的问题；
- 已提供、或者可以建立和采取测量和度量；
- 要求有证据表明需要测试过程；
- 关于改变变更的原因需要达成一致；
- 问题的根源不一定在测试过程所有者的控制或影响中；
- 需要有小规模的评估或有改进试点的预算；
- 需要试点以确定是否需要更大规模的调查和改进项目；

- 验证设想，收集关于问题以及改进建议的原因、现象和影响的证据；
- 相对于外部模型（参考或内容），组织文化更重视和信任根据本地证据开展的内部分析；
- 来自很多领域的利益相关者亲自参与分析（比如在头脑风暴环节）；
- 团队可以控制分析。

也可以使用**混合方法**，比如在过程模型或者内容模型中使用分析的方法：

- 在 TMMi 测试改进项目中使用根本原因分析；
- 在 STEP 测试改进项目中使用度量。

中国软件测试认证委员会 (CSTQB)

6. 改进过程

900 分钟

关键字:

活动 (acting)、评估报告 (assessment report)、平衡计分卡 (balanced scorecard)、企业仪表盘 (corporate dashboard)、诊断 (diagnosing)、创建 (establishing)、IDEAL、启动 (initiating)、学习 (learning)、过程评估 (process assessment)、测试改进计划 (test improvement plan)、测试方针 (test policy)。

改进过程的学习目标

应用个人技能以达到本章的学习目标，这些相关技能的学习目标在第 7 章已详细阐述。第 8 章的变更管理信息和第 9 章的关键成功因素也是必须的。

6.1 介绍

- LO 6.1.1 (K2) 总结测试方针的主要元素。
- LO 6.1.2 (K6) 建立测试改进方针。

6.2 启动改进过程

- LO 6.2.1 (K2) 概述 IDEAL 改进框架启动阶段的活动。
- LO 6.2.2 (K4) 分析商业目标（例如，使用企业仪表盘或者平衡计分卡）以获取合适的测试目标。
- LO 6.2.3 (K6) 创建针对给定方案的改进策略（包括测试过程改进的范围）。

6.3 诊断当前状况

- LO 6.3.1 (K2) 概述 IDEAL 改进框架诊断阶段的活动。
- LO 6.3.2 (K6) 应用特定的过程或者内容模型来计划和执行评估面谈，展示在面谈中需要运用的面谈风格和人际交往能力。
- LO 6.3.3 (K6) 创建并展示来评估的结论（基于对发现的分析）和评估裁决的总结。
- LO 6.3.4 (K2) 概述解决方案分析的方法。
- LO 6.3.5 (K5) 根据评估结果和已执行的 analysis 推荐测试过程改进活动。

6.4 建立测试改进计划

- LO 6.4.1 (K2) 概述 IDEAL 改进框架创建阶段的活动。
- LO 6.4.2 (K4) 根据给定的评估准则列表选择和划分推荐的优先级。
- LO 6.4.3 (K2) 比较自顶向下和自底向上改进方法。
- LO 6.4.4 (K2) 概述测试改进计划的典型内容。
- LO 6.4.5 (K6) 创建测试改进计划。

6.5 概述 IDEAL 改进框架实施阶段的活动

- LO 6.5.1 (K2) 概述 IDEAL 改进框架实施阶段的活动。
- LO 6.5.2 (K4) 从可能的列表中选择合适的试点项目。

6.6 从改进项目中学习

- LO 6.6.1 (K2) 概述 IDEAL 改进框架学习阶段的活动。

6.1 介绍

在组织层面的测试方针中，测试过程改进应作为明确的目标（关于测试方针的详细内容，请参考高级大纲）。组织层面的测试过程改进应该基于整体的测试方针。有效的测试过程改进需要系统化的过程。2.4 节描述戴明环和 IDEAL 过程改进框架时引入了通用的改进过程。本章中以 IDEAL 改进框架为基础详细描述了改进过程 [IDEAL 96]。该方法可以应用于各种生命周期模型。下面介绍的各部分都涉及 IDEAL 的各个主要活动：

- 启动；
- 诊断；
- 创建；
- 实施；
- 学习。

6.2 启动改进过程

启动过程改进在测试过程改进过程中可能是最重要的一个步骤。在启动阶段采取的活动直接影响改进过程的最终结果。低效的启动改进过程可能会导致令人失望的结果，并极大地降低进一步采取改进活动的机会。

模型针对启动阶段描述了下述概要性活动：

- 识别改进的触发点；
- 设置改进环境，并确定改进的支持者；
- 建立改进的基础架构（例如，组织）。

根据 IDEAL 模型的概要性活动，该阶段需要考虑下列因素：

- 确定实际改进需求；
- 基于商业需求定义和协调改进目标；
- 确定改进范围；
- 选择改进策略（参见第 5 章）；
- 考虑人员和文化的影响。

6.2.1 确定改进需求

作为确定改进需求的第一步，应该建立过程改进的认识。比较明显的测试过程改进需求来自于软件的主要失效，而其中的一些失效是由于低效的测试过程导致的。然而，改进的动机来自更为广泛的范围。可能来自下面一个或者多个利益相关者：

- 管理层/客户（例如，要求更有效的测试、生产过程中更少的次品）；
- 用户（例如，更好的易用性）；
- 开发者（例如，更好的缺陷分析支持）；
- 测试者（例如，建立更加系统的测试）；
- 维护者（例如，用更少的时间测试软件变更）。

以初始分析和利益相关者面谈为基础，获取和定义改进之处（面谈和分析需要的技能请参见 7.3 节）。作为分析的一部分，必须确定当前的一些指标，例如，基于产品失效总成本和测试总成本（参见 4.4.3.1 节组织质量成本和 6.2.2 节）确定的质量总成本。

6.2.2 设置测试改进的目标

建立测试过程改进的主要目标必须与质量、成本、时间和商业价值联系在一起。需要三个主要步骤：

- 对改进的未来建立全面的愿景；
- 设置具体目标；
- 测试改进与商业目标协调一致。

测试改进目标通常记录在组织的测试方针中。

6.2.2.1 对未来建立全面的愿景

测试过程改进必须关注改进发起人（利益相关者）的利益以及将要达到的总体目标的愿景。

原因如下：

- 发起人需要在投入资源之前确信可以获得投资回报；
- 必要的变更管理必须与商定的目标相关。

当无法定义通用愿景时，可能会导致测试改进的提案无法通过（参见第 9 章），其中的原因包括：

- 未正确定义的目标可能会潜藏未解决的利益冲突；
- 可能关注于不合适的或者无法达到的目标，从而导致资源浪费或改进失败。

6.2.2.2 设定具体目标

任何测试过程改进都必须具有具体的良好定义的目标。这样有利于：

- 采取合适的行动；
- 定义改进努力的成功（或失败）。

可以各种方式对目标进行描述：

- 以合适的尺度或问卷支持的定性目标（例如，从最差到最好，或者变好到变坏）；
- 具有度量方法的定量目标。例如，目标、问题、度量（GQM）方法（参见 4.3 节）有利于测量标准的定义，并且可以以可度量的结果的方式和目标进行关联；
- 以成熟度级别的方式描述目标。假如基于过程模型开展测试过程改进，也可能以问题的方式描述目标。通常包括定义需要达到的具体成熟度级别，目标既可以是针对整个测试过程，也可以是测试过程的某个领域。

6.2.2.3 测试改进与商业目标协调一致

测试过程改进需要与下面的因素达成一致：

- 商业目标；
- 正在开展的软件过程改进；
- 组织改进（参见第 2 章）；
- 组织结构。

组织可以采用企业仪表盘（Corporate dashboards）和平衡计分卡，促使测试改进与组织改进目标一致，例如，

- 财务目标 – 例如，生产率改进、改进的收入、改进的利润、改进的财务周转，同样也和质量价值观保持一致；
- 改进的产品质量 – 与产品、制造和用户对质量的要求一致；
- 客户目标 – 例如，改进的市场份额、改进的客户满意度、改进的风险管理过程，也与用户对质量的要求一致；
- 内部目标 – 例如，更好的项目输出可预见性、软件开发阶段降低的缺陷/失效、减少项目研发时间和降低工作量/成本，与质量的制造要求一致。离岸或者外包也可以作为内部目标，它们在降低测试过程成本和确保业务的核心竞争力方面非常有效；
- 创新和改进目标 – 新的市场/产业、更多的新产品入市、更快的投放市场，以及过程/架构/标准认证（例如，CMMI 或者行业标准）与质量价值观保持一致

- 人员目标 – 例如，工作满意度、员工流动，以及减少病假和其他原因的缺席，可以和任何质量观保持一致，但它们同样会影响质量的超然视角（信任、名誉）；
- 社会参与度/政治目标 – 例如，组织对环境的影响、名誉和公共形象，可以和任何质量观保持一致，但它们同样会影响质量的超然视角（信任、名誉）。

6.2.3 设置改进范围

为了设定测试过程改进范围，需要阐述清楚一些问题，包括：

- 通用过程范围 – 其它非测试过程的范围；
- 测试过程范围 – 涉及测试过程的部分；
- 测试级别 – 改进计划范围内的测试级别；
- 项目范围 – 项目或者组织是否在此范围之内。

6.2.3.1 通用过程范围

改进范围可能包括通用软件开发过程的各个方面，例如项目管理、需求管理和配置管理。需要明确是否通过改进测试过程就可以达到规定的改进目标，还是其他过程（例如，服务管理、开发和在[ITIL]中描述的其他支持过程）必须同时改进才能成功。假如是后者，那么可能需要更多的资源和技能。

6.2.3.2 测试过程范围

仅关注测试过程中的单个领域进行改进，相对于考虑所有可能的领域比较划算。有时期望改进测试过程的所有方面，而有时只关注特定方面（例如，测试计划）。

假如测试改进的范围限定在特定领域，应该注意考虑其他相关方面。改进限定在特定领域可能会导致次最佳化。这样是否有道理？例如，将改进工作量关注在建立测试度量规划上，但实际中可能并不适用（例如，在测试报告中不使用）。

6.2.3.3 项目范围

测试过程改进规划可能会以项目为中心或者以组织为中心的方式进行组织。

以项目为中心的改进关注点是单个项目或者一组项目（例如，程序）。评估，也称为审计，通常在项目测试人员中开展（例如，测试人员、测试经理）。测试过程改进范围常常限制在相对较小的过程相关的任务之上，例如，测试计划或者测试技术。这些高度关注的改进项目通常具有较好的成本效益，假如，按照 6.2.3.2 节的提示，仔细选择测试过程范围。

以项目为中心的改进会产生相对较快的结果，但可能使组织层面的问题得不到解决。组织层面的改进需要更长的持续时间，也会产生更广泛的收益，但是通常需要花费更多的时间来达到目标，需要更多的成本来实施。

以组织为中心的改进关注点在测试组织、部门或者团队。除了评估单个项目，组织作为整体也是它的关注焦点。适用于所有项目的测试过程领域是需要特别关注的焦点（例如，培训、组织架构）。

在 IDEAL 模型的启动阶段，需要建立过程改进的基础架构[IDEAL 96]。需要考虑测试改进项目的结构化组织。7.1 节更详细地阐述了这些问题。

6.2.4 启动阶段人员和文化的影响

人员方面的因素会影响改进目标的实现。启动阶段的关键成功因素如下：

- 知识级别；
- 组织文化；
- 人员文化；
- 易接受程度。

第 7、8、9 章阐述了其他信息，关注改进团队所需要的软技能和变更管理和组织文化方面的人员因素影响。

测试改进建议对人员问题很敏感，需要根据组织人员的不同风格、文化和需求，推荐不同的改进策略。

6.3 诊断当前状况

模型描述了诊断阶段的下列概要性活动：

- 评价和描述当前的实践；
- 提出建议并文档化阶段结果。

此阶段的最终结果通常是一份测试评估报告。

根据 IDEAL 模型概要性活动，此阶段需要考虑的因素包括：

- 计划评估；
- 评估准备；
- 开展面谈；
- 给出反馈；
- 分析结果；
- 实施解决方案分析；
- 建议改进活动。

此阶段的活动依赖于测试过程改进所采用的方法（参见第 5 章）。

如果采用基于分析的方法（参见第 4 章），可以应用各种原因分析技术（参见 4.2 节），对度量、测量和指标进行分析。

如果采用基于模型的方法（参见第 3 章），需要计划和执行评估活动。6.3.1 节到 6.3.3 节详细覆盖了这部分内容。

6.3.1 计划评估

计划评估可以清晰地识别以下活动并安排时间：

- 评估准备：
 - 初步分析；
 - 准备评审材料，例如，检查表；
 - 收集已有的测试件资料（例如，测试计划、测试规格说明）。
- 与测试过程涉及的不同角色进行面对面的交流：
 - 测试人员；
 - 测试经理；
 - 开发人员；
 - 项目经理；
 - 企业所有者；
 - 业务分析人员（领域专家）；
 - 环境经理、缺陷经理、发布经理、自动化专家。
- 在每次评审中需要覆盖的具体领域；
- 评估完成之后需要提供初始的反馈（数据、格式、期望结果）；
- 为受访者提供后续步骤的信息。

根据在启动阶段达成的一致目标（6.2.2 节）和范围（6.2.3 节），评估计划必须确保覆盖了所有的测试领域。对于某些特定领域，建议在不同受访者之间都覆盖该部分内容以确保进行一致性检查。

6.3.2 评估准备

开展正式的评审之前通常会对文档进行初步分析。假如评估范围是组织级别，测试方针、测试过程描述、所用的模板和主测试计划都应该是评估的重点。假如评估范围是项目层面，应该选择测试计划、测试规格说明和测试报告为重点。来自其他利益相关者（开发、商业等）的文档如果在评估的范围之内（参见 6.2.3 节）也可能相关。

分析的目的：

- 在评审之前获得当前测试过程的信息；
- 为评审准备特定的问题；
- 开展正式的不需要讨论的评估活动。例如，检查文档以确保全面并符合标准。

在开展评审之前，建议先营造合适的环境。环境应该：

- 舒适；
- 不被打扰；
- 专有。

6.3.3 开展面谈

根据评估计划（参见 6.3.1 节）实施面谈。面谈的内容通常由采用的模型决定（参见第 3 章）。

面谈通常面向个人开展，这样有利于受访者可以自由的表达观点，并且确保“安全”的环境。面谈员在开展面谈之前应该向受访者和信息提供者经常强调保密的重要性，尤其在预计有敏感问题的情况下。基于这样的原因，建议面谈活动在受访者主管不在场的情况下开展。

可以鼓励受访者诚实地提供信息，假如：

- 确保保密；
- 认可给出的改进理念；
- 不惧怕惩罚或者存在失败；
- 知道和理解他们提供的信息会被如何使用。

成功开展面谈需要具备各种技能。这些在 7.3.1 节更加详细的描述。

6.3.4 初步反馈

面谈之后可以给出初步的反馈，典型的反馈是简短的报告或者走查。这非常有助于和被评审者一起确认初始评估结果，目的是为了澄清可能的误解，或者向利益相关者提供关键点的概览。需要注意的是任何时候都需要遵守保密的规则，并避免对识别的任何问题进行指责。

6.3.5 分析结果

假如改进采用了基于分析的方法（参见第 4 章），通过应用下面的概念来分析当前的状况：

- 系统思考 [Weinberg 92]；
- 临界点 [Gladwell]。

系统思考可以帮助分析不同系统（过程）组件之间的关系，并将这些关系以稳定循环或者加强循环的方式呈现。加强循环可能会有些负面影响（恶性循环）或者正面影响（良性循环）。

临界点可以帮助识别系统中的特定点，其中微小的、良好关注的改进可能会打破恶性循环，并引发进一步改进的连锁反应。

假如测试过程改进采用了基于模型的方法（参见第 3 章），需要对当前状况的过程成熟度和在初始阶段定义的目标之间进行比较。

假如可以获得合适的基准，评估结果应采用这些基准。在可能的情况下，使用下面的一些基准：

- 基于公司层面，代表整个组织
- 基于行业层面，可能涉及相同的业务领域
- 基于项目层面，与满足目标的特定项目进行比较

假如已经创建了主要性能指标（参见 4.4 节和 6.2.2 节），这些指标也应纳入分析。例如，假如缺陷密度百分比 DDP 低于要求的级别，产品过程中发现的缺陷分析应该用于评估它们的来源。

评估的结果应该提供足够的信息，用于定义建议以及支持规划过程（参见 6.3.7 节和 6.4 节）。

6.3.6 实施解决方案分析

解决方案分析用来识别潜在的问题解决方案，并从中进行合理的选择。任何已选择的改进点或者解决方案都可以通过下面的方法确定：

- 正确改进解决方案的预想 – 已经确定针对问题的解决方案，例如，所有的测试都应该自动化。这意味着由于预想的解决方案并不能解决问题的根本原因或者会导致情况更糟，而没有真正的开展这个方案；
- 建议的解决方案可能以关键实践/领域引入到评估中应用的模型，它们有外部的证据说明所具备的优点。此处的缺陷是模型中采用的后续实践可能没有解决当前环境中引起问题的根本原因，甚至导致问题更糟；
- 来自客户或者利益相关者的需求，例如，客户可能要求所有提供商具备 ISO9001:2008 资格，这样的优点是目标和关注点明确，缺点是过程中所需的变更可能不能提供改进或者与其他计划的改进冲突；
- 根据问题收集的信息进行解决方案分析，其优点是对每个建议的解决方案的负面结果和正面结果都进行了讨论，因此选择的解决方案肯定有正面的收益，同时副作用最小。缺点是分析过程费时、费资源和费钱；
- 对解决方案裁剪的方法，是上述几种方法的混合体，例如，利益相关者要求采用 TMMi 级别 4 和开展成本-收益分析，得到的结果是部分采用 TMMi 级别 3。该方法的优点是关注解决方案本身分析，缺点是分析过程费时、费资源和费钱。

解决方案分析过程包含了下面的一个或者多个因素，依赖于采用的方法：

- 识别问题优先级并分析根本原因，以选择开发哪个解决方案；
- 识别成本优势和其他收益，包括不实施解决方案的风险；
- 识别实施解决方案的成本、风险和负面影响；
- 识别实施解决方案的限制；
- 识别冲突，例如可能会相互否定或不兼容的解决方案；
- 分析反馈循环（不管是恶性循环还是良性循环）；
- 评估解决方案并设定优先级；
- 基于早期活动中收集的信息和指标执行差距分析；
- 执行成本-收益分析以提供投资收益估算；
- 执行反向鱼骨图分析，以鱼骨为基础进行反向的根本原因分析，针对早期识别的特定根本原因的同一个鱼骨通过头脑风暴法获取解决方案。也可以增加其他的限制，例如，预算、资源、时间限制。

6.3.7 建议改进活动

评估报告必须将结果与特定的测试改进目标关联起来。在评估结束之后应该尽快发布报告，可能作为初步的版本，它之后会有完整版本。

评估报告至少包括：

- 参考愿景陈述的管理摘要（参见6.2.2.1节）；
- 范围和目标陈述；
- 分析的结果，包括：
 - 积极方面；
 - 待改进方面；
 - 存在的问题。
- 改进建议列表。

如果可能，建议应该作为改进需求进行考虑（例如，为缺陷跟踪系统提供工具支持），可以用不同的方法实现（例如，使用工具 XYZ 和提供培训）。6.5 节覆盖了实施任务的内容。

改进建议应该包括下面的信息：

- 唯一的标识符（为了可跟踪性）；
- 建议对一个或者多个已经声明的目标的影响（如果可能，采用尺度来表示完成的程度，例如，最少、部分、全部或者百分比值）；
- 成本和收益的估算；
- 实施的时间跨度（例如，短期、中期、长期）；
- 实施的风险（例如，变更过程中的概要性阻力、达到特定改进的风险）；
- 依赖与假设（例如，假设另一个建议也在实施）。

为了帮助规划和跟踪概要性建议，它们在可能的情况下应该分解为微小的有着明确结果的改进步骤。

有些过程模型，例如，TPI Next，包含了特定的改进建议用来帮助创建改进建议的任务。

6.4 制定软件测试改进计划

IDEAL模型描述了“创建”阶段的三个高层次活动，包括：

- 设置优先级；
- 开发方法；
- 规划行动。

通常，此阶段的最终结果是输出测试改进计划。

在此阶段，基于 IDEAL 模型的高层次活动，需要考虑以下几个方面：

- 设置优先级次序；
- 开发实施方法；
- 规划所需改进的行动。

6.4.1 设置优先级次序

建议在“创建”阶段，应该根据优先级列表中的标准来设定优先级次序。每项标准都应该考虑改进要求和所涉及的利益相关者的加权。

至少应考虑如下内容：

- 改进持续时间—应该在短期（“Quick-win”）和长期改进之间取得平衡。短期改进的优势是，能够迅速呈现投资回报并且能够对实施团队产生强大的激励作用。长期改进会引发测试过程的根本性改进，包括文化和组织问题；
- 实施改进的风险—许多改进需要改变现有的测试实践。每项改进都有失败的风险。因此，需要在实施改进之中考虑如下多个因素：
 - 一旦改进不得不放弃，应具备能够返回到目前状态的能力；
 - 关注改进过程中对整体系统有影响的关键改进活动，尤其注意某些改进活动会依赖于这些活动的成功实施；
 - 具备能够真正落实改进规划的能力。是否有足够的资源可供使用？实施改进活动的团队的主要成员是否有可能安排其他任务？改进过程中的风险能否识别，例如对某些变化的抵制？
 - 合适的改进成本收益比（可能作为投资回报的价值表现）。
- 目标关联—能否对提议的改进和业务既定目标之间做出明确的关联？
- 杠杆作用—所实施的改进会对具体目标产生多大程度的影响（例如，高、中、低）？

6.4.2 开发可实施的方案

利益相关者考虑所提议的改进并设置优先级。具体的实施改进策略应在初始化阶段选定，应选择与改进范围密切相关的方案。两种主要的方案如下：

- 自顶向下；
- 自底向上。

自顶向下方案的特点：

- 改进的范围通常可覆盖几个项目或整个组织；
- 改进过程一般是在整个团队中展开；
- 需要对结果进行详细的分析，以找出不同项目的相同点（良好或不良实践）；
- 表达和沟通能力往往与能否对目标和提议达成一致意见相关；
- 以下几点与取得成功密切相关（详见第9章）：
 - 对人员有效的管理；
 - 获得赞助；
 - 管理期望。

自底向上方案的特点：

- 改进范围通常只能覆盖一个项目，最多两个项目；
- 所选择的方法往往不很正式。例如，某个项目可能会选择采用分析法而不是更正式的基于模型的方法；
- 改进过程一般是在一个项目中展开；
- 此方案典型的目标是对一些指定的改进进行尝试，以获得经验并构建支持；
- 自底向上法非常适合测试改进项目资金有限的情况，或者项目组首要反映成本收益比；
- 自底向上方案的结果会用来验证采用自顶向下实施的改进效果。

6.4.3 计划改进

主要的改进计划活动如下：

- 对基于提议的明确方面达成一致；
- 建立必要的性能指标（见第2、4和6章）；
- 确定改进活动的优先级，并把相关联的改进合并为一组活动（逐步改进）；
- 把需改进的问题与建议相结合，把性能指标与任务成果相结合。例如通过建立数据库来实现；
- 选定合适的过程改进方案（参见6.4.2节）；

- 安排变更；
- 确立小组或团队以实施改进（参见7.1.1节）；
- 分派任务；
- 改进计划中应记录以上提及的相关点。

IDEAL 模型描述了两种可能的实施计划：

- 战略上的实施计划，特点如下：
 - 3到5年的时间跨度；
 - 覆盖整个组织的测试过程改进，并把这些改进与其他的软件过程改进、已计划或正在进行的TQM（全面质量管理）集成在一起。
- 战术上的实施计划，特点如下：
 - 相对较短的时间跨度，一年或更短时间；
 - 专注于测试过程小组（负责改进实施的团队）活动计划的细节。更详细的内容参见7.1.1节。

6.5 采取行动，落实改进

模型描述了“行动”阶段中四个高层次的活动：

- 创建解决方案；
- 试验/测试解决方案；
- 完善解决方案；
- 实施解决方案。

基于 IDEAL 模型的高层次活动，在此阶段需要考虑以下两个方面：

- 选择和执行试验；
- 管理和控制变更。

6.5.1 选择和执行试验

对合适的改进进行试验是降低失败风险、获得有用经验、构建支持和减少实施改进失败风险的有效途径。对于涉及工作实践中重大变更或需要占用大量资源的改进，这一点尤为重要。

试验方案的选择应该权衡以下多个因素：

- 实用主义—试验能否代表真正的解决方案？切勿选择仅仅提供了特别快的或容易实现的机会的试验；
- 解决方案的可扩展性—试验的结果是否可应用到所有的情况？如果试验不能够代表实际项目的复杂度和大小，那么实施的改进由于存在风险而不能大规模应用；
- 对当前项目的影响—试验不应该在当前项目中进行，除非可接受影响。需要额外注意的是，现行的实践被试验进行期间的改进实践所取代的情况。更好的解决办法是新试验与现行项目完全同步执行，虽然这可能引入资源问题（并不希望因为试验导致项目员工对同样的任务执行两次）；
- 失败的风险—即使使用的试验是降低风险的措施，也要对试验失败的风险进行评估。在试验的评估中此因素是非常重要的因素。试验评估也应该考虑对整个项目改进失败的财务和激励风险。

6.5.2 管理和控制实施

对于既定的改进目标，需要执行、检测和跟进测试计划行动。应该收集测试改进计划中的措施、测量标准和指标，以与既定目标形成对比。

如果执行试验的经验教训分析得出了积极的效果，那么很有可能决定推广此改进到组织的其他部分和（或）其他项目中。改进的推广遵循已定义的过程，尤其是整个组织会受此变更的影响。第 8 章将致力于测试改进过程的这个关键方面。

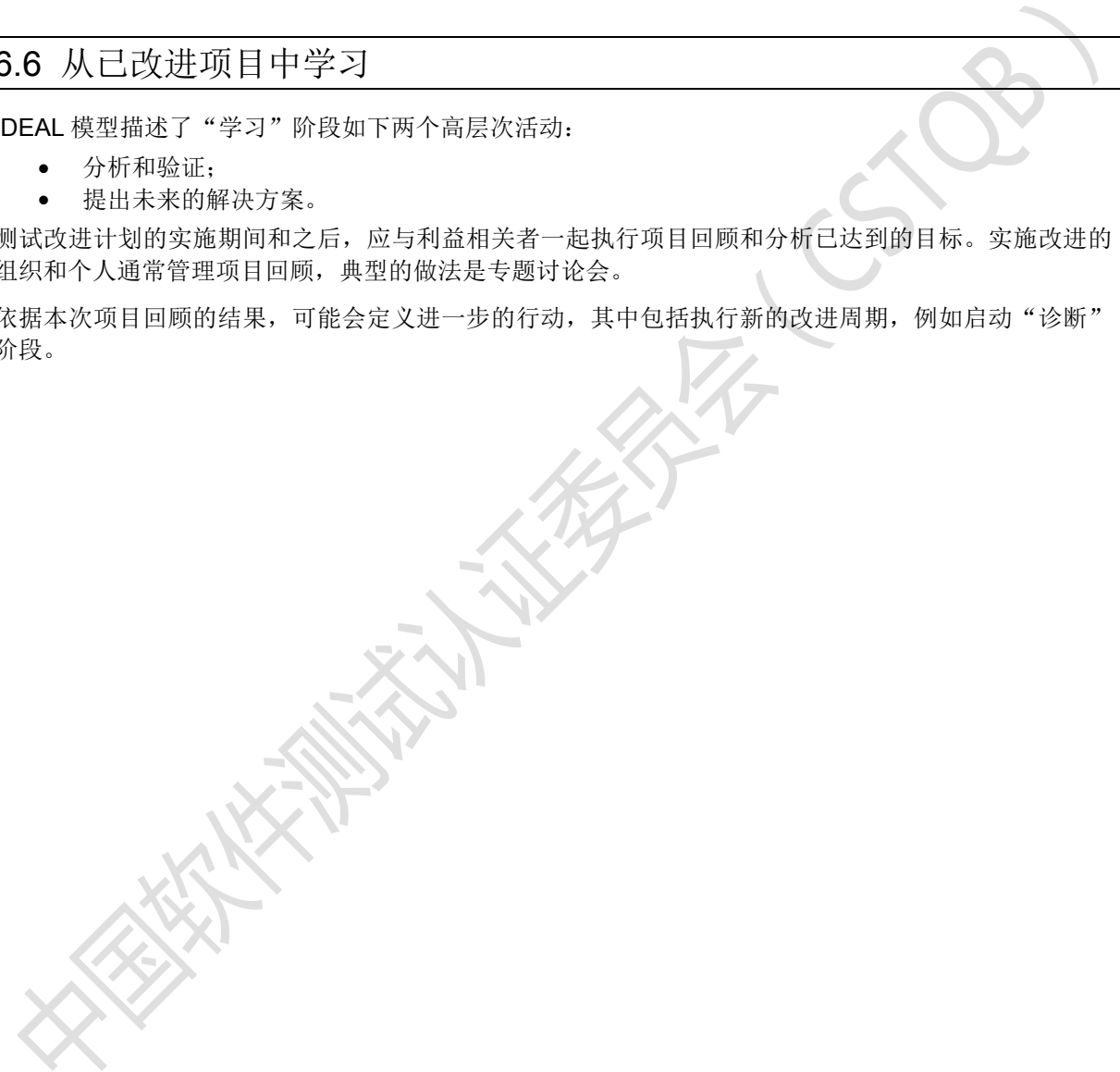
6.6 从已改进项目中学习

IDEAL 模型描述了“学习”阶段如下两个高层次活动：

- 分析和验证；
- 提出未来的解决方案。

测试改进计划的实施期间和之后，应与利益相关者一起执行项目回顾和分析已达到的目标。实施改进的组织和个人通常管理项目回顾，典型的做法是专题讨论会。

依据本次项目回顾的结果，可能会定义进一步的行动，其中包括执行新的改进周期，例如启动“诊断”阶段。



7. 组织, 角色和技能

465 分钟.

关键词:

评估人 (assessor), 相互依赖行为 (codependent behavior), 情商 (emotional intelligence), 评估组长 (lead-assessor), 思维导图 (Mind-Map), 测试过程组 (TPG), 测试过程改进人员 (test process improver), 事务性分析 (transactional analysis)。

改进团队的角色和技能的学习目标

培训机构应该考虑基于技能的学习目标 (第 7.3 节) 和第 6 章的学习目标。

7.1 组织

- LO 7.1.1 (K2) 理解测试改进项目中 TPG (测试过程组) 的角色、任务和职责。
- LO 7.1.2 (K4) 评价不同组织结构以组织测试过程改进项目。
- LO 7.1.3 (K2) 理解开发活动外包或离岸对组织测试改进项目的影响。
- LO 7.1.4 (K6) 为测试过程改进项目的指定范围设计组织结构。

7.2 个人角色

- LO 7.2.1 (K2) 理解测试过程改进项目中的个人角色。

7.3 技能

- LO 7.3.1 (K2) 理解执行评估的必要技能。
- LO 7.3.2 (K5) 评估测试专业人员 (例如 TPG (测试过程组)、技术工作组的潜在成员) 在开展评估活动所需的基本软技能的不足。
- LO 7.3.3 (K3) 在评估过程中应用访谈技能、倾听技能和记录技能, 例如在“诊断目前现状”的访谈环节应用这些技能。
- LO 7.3.4 (K3) 在评估过程中应用分析技能, 例如在“诊断目前现状”的结果分析中应用分析技能。
- LO 7.3.5 (K2) 理解测试过程改进项目中的演示和报告技能。
- LO 7.3.6 (K2) 理解测试过程改进项目中的说服技巧。

7.1 组织

通过创建一个组织来确保改进动作的正确实施, 并对改进过程负责, 可以更有效地实施测试过程改进 (参见 9.1 节)。对于实施组织级的过程改进项目尤其如此。而对于较小规模的改进项目必须权衡一下设置独立改进组织的成本收益。

[Burnstein 03] 和 [IDEAL 96] 提供有关测试改进组织的有用信息。

7.1.1 测试过程改进组

[Burnstein 03] 中描述 TPG (测试过程组) 由一些个体组成, 该组织可能与软件工程过程组 (SEPG) 等其它质量相关组织进行合作。

TPG 不要临时设置, 而应该设为组织的常设部门。主要原因如下:

- 临时设置的改进组的工作范围通常局限于特定的项目或者单个问题领域。常设的测试改进组具有更大的工作范围，更易于识别和提出影响组织的改进建议；
- 常设的 TPG 可以更有效地控制提出的改进措施的实施。临时改进组具有在改进完全实施前被解散的风险。缺乏足够的控制可能导致商定的目标未能实现或未被报告；
- 执行有效测试过程改进所需的各类技能在专门的组内更容易得到培养；
- 常设的 TPG 可以作为测试过程的“所有者”，为主要利益相关者提供交流的重要渠道；
- 许多成熟度模型（例如 TMMi）包括常设的测试过程组（TPG），作为更高测试过程成熟度的指标。

[Burnstein 03] 提出了有效的测试改进组可能包括具有广泛背景的受人尊敬的人员（例如，管理，开发，测试），他们被认为是实践者。他们不仅提供针对复杂测试过程的富有价值的见解，而且所提出的改进建议也更容易被接受（受人尊敬的实践者通常比纯理论家更好接受）。

IDEAL 模型[IDEAL 96]描述了过程改进组织的设置，组成如下：

- 执行委员会（EC）
 - 应用在大型组织；
 - 处理战略和方向问题。
- 管理督导组（MSG）
 - 由组织的现有管理架构的高层经理组成；
 - 设定目标，成功条件和优先级；
 - 指导组织内的实施活动；
 - 提供资源；
 - 设置过程改进特定方面的技术工作组。
- 技术工作组（TWG）
 - 只在需要完成特定目标时存在此组；
 - 调查问题，向管理督导组提出解决方案；
 - 执行原型活动；
 - 依据从原型获得的教训修正战术行动计划。

IDEAL 模型包含上述组织各项的样例章程。

测试过程改进组织主要关注过程，但是也应当承担培训的责任；长期的过程改进也需要人员不断提升才能实现。过程改进组织的目标、成员、角色、职责以及与组织内其他部分的主要接口应该清楚地说明，并且要符合组织的测试方针（参见高级大纲）。

7.1.2 远程、离岸和外包团队的测试改进

当改进所涉及的团队工作于同一个组织并且在同一地点时，各部分的改进工作要比在其他组织或地点更容易组织和开展。

软件开发生命周期中工作外包或离岸开展时，问题沟通、根本原因、解决方案、变更的证据和计划都可能受到影响。此种情况下改进团队的重点应当在以下方面：

- 需要改进的政治、文化或协议的误解或理解，这可能阻塞或妨碍改进建议，或者意味着改进方案不被理解或触及敏感情绪；
- 在通知人们改进的顺序方面需要注意沟通的时间，也需要考虑本地假日，时区等；
- 各个阶段的信息要通知和涉及各方，从改进初期到信息收集、方案选择、试点项目及滚动/变更；
- 从各方征求反馈，无论是改进项目成功或其他方面。

下列情况下将产生潜在问题：

- 在岸测试团队与离岸/外包测试团队相比具有不同过程成熟度级别、效率和效力时；
- 在过程、沟通和质量文化方面，各个组的期望存在较大差异时；
- 设定的过程不遵守时；
- 在任何一方尝试改进或变更引起文化和交流冲突时。

7.2 个体角色

7.2.1 测试过程改进人员

测试过程改进人员必须具有按照本大纲执行任务的能力。

测试过程改进人员能够获得的改进是有限度的。例如，如果开发经理或者客户控制它们自己的测试，这已经超出了测试过程改进人员的影响范围，对测试过程的所需变更可能超出了测试过程改进人员的工作范围。

测试过程改进人员应当理解能够改进的范围，需要时从管理和其他组请求支持。在执行他们的角色职能时，测试过程改进人员可能仅限于对过程改进提出建议，而不是去实施这些改进。在组织的改进项目中，测试过程改进人员可能向总体变更经理报告。

7.2.2 评估人角色

由于执行评估需要许多技术和软技能（参见 7.3 节），需要指派一个特定评估人角色。

如果采用基于模型的方法，实施评估的人员需要具备特定模型知识。在一些情况下（例如，TMMi, CMMI, ISO/IEC 15504, EFQM 评估人员），评估人员需要获得正式培训和认证。例如，TMMi 基金组织划分评估组长和评估人员（有关评估人员的角色和成为认证评估人员的更多信息，参见 [TMMi 基金网站]）。

7.3 测试过程改进人员/评估人员的技能

成功执行测试过程改进依靠广泛的技能。它们既可以是技术技能，例如分析的技能，也可以是非技术技能，例如人际技能，通常也称为“软技能”。下面是与测试过程改进人员和评估人员相关的技能：

- 面谈技能；
- 倾听技能；
- 演示和报告技能；
- 分析技能；
- 记录技能。

下列附加技能与测试过程改进人员特别相关：

- 说服技能；
- 管理技能。

执行评估的任何人都需要这些技能。TPG/技术工作组的成员也可能需要其他技能。

7.3.1 面谈技能

面谈由下列流程组成：

- 开始—阐明目的和目标；
- 提出问题；

- 倾听—收集信息（参见第 7.3.1.4 节）；
- 总结—对于连贯的信息定期进行总结；
- 检查—在总结当前问题的基础上询问新问题，加深面谈人员之间的深层理解；
- 结束—解释下一步工作。

面谈技能是获得信息和成功执行评估的基本技能。良好的面谈风格如下：

- 询问开放式的问题，不要询问“是/否”的问题；
- 不要逐字逐句读出调查问卷；
- 访谈人员使用人际沟通技巧展开讨论（例如本节提到的内容），引导对话覆盖要点。

访谈者和受访者之间的互动通常较为复杂，缺乏必要的技能，将会产生误解、信息隐瞒甚至获取了错误的信息。访谈者不必是心理学家，但他们需要良好的人际沟通技能，这些技能来自于对以下概念的共同：

- 情商 [Mayer 04]；
- 事务分析 [Wagner 91]；
- 相互依赖行为 [Copeland Paper 01]。

7.3.1.1 情商

情商（EI）可以“帮助人们意识和导引社会环境” [Mayer 04]。Salovey 与 Mayer 提出的基于能力的模型提出个体具有不同的处理情感信息的能力。对于测试过程改进人员而言，这种能力在进行访谈时很重要，因为事实信息通常只是所传达的信息的一个方面。

该模型提出的情商包含四种能力：

- 情绪感知—从被访谈者面部和声音检测和解释其情绪以及识别自己情绪的能力；
- 运用情绪—利用情绪促使思考和分析问题的能力。高情商测试过程改进人员为了获取受访者某一特定方面的信息或者观点，能够在访谈时捕捉受访者情绪的变化；
- 理解情绪—鉴别情感之间复杂的关系，以及随着时间推移情感如何发展；
- 管理情绪—调节个人和他人情绪的能力。高情商测试过程改进人员可以管理情绪以达到面谈目标。

可以使用基于能力的模型度量情商，但这不是测试过程改进人员预期将要拥有的技能。

7.3.1.2 事务分析

[Wagner 91] 描述了在组织或业务范围内事务分析的观点，认为每个人由六个“自我”组成。这些“自我”进一步分类为三个“有效的自我”和三个“无效的自我”。

与“有效的自我”交流通常认为是积极的和有建设性的。

- 普通儿童自发行动，表达感受，需要认可，指导和激励；
- 成人富有逻辑性和理性；依靠事实而不靠感觉处理事情；
- 养育孩子的父母对别人比较严厉，但他们也能理解，敏感和关怀。

与“无效的自我”交流通常认为无助于获取信息。

- 挑剔的父母使用身体语言、手势和音调疏远他人，可能是讽刺，竖手指或提高嗓门；
- 叛逆的孩子易怒并持续生气，非常负面，不听话，可能故意忘记事情或拖延；
- 顺从的儿童责怪自己，使用软声细语，低泣，谨慎和自我保护。

测试过程改进人员要理解事务分析，能够改进面谈技能，可以从面谈对象得到的可疑信息中区分信息，可以建议复述问题或（在极端情况下）修正面谈方法。

7.3.1.3 相互依赖行为

“相互依赖”一词描述了人际交往的不良方式，其中一个人的缺点或弱点由另一人补偿。Lee Copeland [Copeland Paper 01] 的文章指出“我们做的一切错事，都有正确的理由”。我们不正确对待别人的需要，甚至可能开始承担责任或“掩盖”他人的行为。

察觉相互依赖对于测试过程改进很重要，因为这些不健康的相互依赖可以屏蔽特定问题的真实原因，甚至是问题本身。

Lee Copeland 提供了软件开发领域的一些相互依赖事例：

- 开发人员没有清楚地理解用户需要，就同意实现系统；
- 如果管理人员给测试人员不合理的工作进度表，测试人员在规定时间内尽最大能力进行测试。

在本例中存在不正确的行为方式（来自开发和管理）和相互依赖的响应。这些响应的短期影响可能是有利的，但长期结果可能是破坏性的。因为发出了错误的信息“我们不关心用户”或“我们的测试估计总是夸张”。

面谈时测试过程改进人员必须清楚典型的相互依赖的一些信息：

- 包含“我尽最大努力”（即使我知道这是错误的）的响应；
- 包含“没关系”（这需要纠正，但是我装作是正确的）的响应；
- 否定风险（这将带来灾难，但是我能承受这个风险，既然我提到了）；
- 受访人员试图以某种“正常”方式说服面谈人员显然是不正确的行为的响应。

从长远来看，相互依赖行为的人有可能成为受害者，他们为不断地接受他们明知是错误的状况而愤怒。他们会萌生辞职的想法，开始容忍不正常、不健康、不适当的行为。

测试过程改进人员应该明白软件测试专业人才应是有帮助的。与此同时，他们应该仔细比较相互依赖行为的短期利益和长期可能出现的困难。改进建议可能需要把重点放在长期问题上，并帮助相互依赖人员走出困境。

7.3.2 倾听技能

倾听技能有助于从谈话中提取信息，以及准备可能的回答。“积极聆听”是一种技术，其重点是以正在发言的人为重点，提供聆听和响应的结构化方法。

7.3.3 演示和报告技能

演示和报告在下列情况下比较重要：

- 获得测试过程改进的支持；
- 向利益相关者清楚地说明结果；
- 具体的改进建议。

运用管理摘要的技能，有助于突出适当抽象层次的关键点，无需太多的非必要细节。有效应用这些技能需要访谈者注意以下方面：

- 具有选择性，仅选择一些关键想法；
- 你的或者客户的想法（或客户）在特定背景下是否可行；
- 改进的时间表是否现实；
- 使用管理者语言；
- 希望提出问题。

测试过程改进人员在展示和报告信息的时候，应该了解他们的听众。2.3 节提到的质量观点可以指导信息展示的类型和深度。

具体的演示和/或报告技能（参见[Few 08]，[Tufte 90] 和 [Tufte 97]）包括如下：

- 信息设计；
- 为听众选择合适媒介的能力；
- 理解适当使用度量和统计作为证据；
- 理解正确使用图、图表和图形；
- 有效的对公众讲话；
- 从观众发现反馈，了解观众的反应。

7.3.4 分析技能

第4章提到了很多与技术相关的分析改进方法。应用这些技术所需的通用分析技能总结如下：

- 总结所收集的信息的能力；
- 识别信息趋势和模式的能力；
- 转换信息到其他形式的的能力，例如文本到过程流图，思维导图到幻灯片演示等；
- 掌握何时统计分析合适，何时不合适；
- 分析和报告数据时，了解和区分：（1）原因和结果（2）相关性（3）巧合；
- 理解统计信息的应用和分析。

7.3.5 记录技能

这些技能可以有效地捕捉相关信息。例如在进行面谈时可能特别重要。

思维导图 [Buzan 95] 在准备和控制面谈，建立各个主题之间的联系以及记录笔记方面，是一种自然的和易于获得的技术。使用可视元素（“随手涂鸦”）和文字之间的关联提供以下收益：

- 有效捕捉“大框架图”；
- 更快更好的组织记录；
- 便于总结的演示；
- 易于将内容与您的观点关联；
- 易于回忆要点。

请注意，使用思维导图应作为访谈者个人的选择。一些访谈者会抑制在访谈期间使用思维导图以防止过早的分析。可以使用以下替代方法：

- 使用自然语言保存实际访谈结果；
- 使用关键字等简短方式的文字；
- 使用图和流程图。

使用笔记本电脑记录笔记有时可能成为访谈者和受访者的障碍，甚至无法实现倾听和面谈技巧的目的。做出电子记录的决定需要谨慎并且取得受访人员同意。

有些访谈人员使用音频或视频和抄写的方式记录。这可能是有用的，但必须获得受访人员的许可，并对如何使用有关记录的信息，以及何时/如何销毁记录达成一致。记录时受访人员可能无法自由面谈，即使同意这种记录方法。

在某些情况下，例如小组研讨会，可以公开地收集和显示记录，例如，使用挂图、即时贴或标识卡片。在房间里的每个人都可以看到和贡献记录，更容易达成协议，从而使这些记录发挥效益。

7.3.6 说服技能

针对特定改进需要说服利益相关者和需要设置未来愿景时，这些技能对测试过程改进人员很重要。例如，变更开始时有些阻力或利益相关者没有足够的时间理解问题和做出决策，此时说服技能就很重要。

[Frank 90] 描述了应用说服技能的有用技术。该简单技术包含以下步骤：

- 设定目标；
- 选择听众（除非非常明显）；
- 选择方法；
- 使用一种物件获得注意；
- 明确主题；
- 询问目标（或获得目标的下面步骤）。

[Cialdini] 描述的说服技术也是销售和市场技术的一部分，[Burnstein 03] 警告和反对这种技能。

7.3.7 管理技能

测试过程改进人员需要具备特定测试改进任务相关的广泛管理技能，其中的详细内容超出了本大纲的范围。这些技能包括以下：

- 计划；
- 估算；
- 做出决定；
- 风险管理。

中国软件测试认证委员会 (CSTQB)

8. 管理变更

285 分钟

关键词:

变更管理 (change management)

管理变更的学习目标

8.2 变更管理的基本流程

LO 8.2.1 (K2) 变更管理的基本流程概述。

LO 8.2.2 (K6) 考虑变更管理相关事宜, 创建测试改进计划, 包含步骤和实施方法。

8.3 变更管理过程中的人为因素

LO 8.3.1 (K2) 概述变更管理过程中的人为因素。

LO 8.3.2 (K4) 分析人们对变更的态度并且与 Satir 模式关联起来。

LO 8.3.3 (K5) 推荐能够让相关人员接受变更的方法。

8.1 简介

没有变更管理, 过程改进将不可能成功; 在改进过程中投入最大精力的是部署。本章将以一系列的步骤和活动来描述变更管理。管理变更表现为一系列的步骤和实施活动。

8.2 变更管理的基本流程

如果过程改进不在管理变更的背景下进行, 注定要失败。在[Kotter & Rathgeber 05]中提出的八个变更步骤适用于任何 IT 学科, 包括测试过程改进。

阶段设置

步骤 1. 创造紧迫感

- 以客观的度量数据, 辅以风险 (例如, 不按计划实施变更的风险) 的描述, 确立改进需求 (参见 6.2.1 节);
- 明确何种变更将在何时发生, 并给出总体时间表;
- 获得明确的管理支持和资源。

步骤 2. 集合指导小组 (例如, 测试过程小组, 参见 7.1.1 节)

- 将乐于接受变更者转变为变更拥护者;
- 让这些人起到“倍增器”的作用 (例如, 第一批支持者可以将他们了解的信息传递给他人, 并激励他们)。

确定需要做什么

步骤 3. 明确变更的愿景和策略

- 管理预期目标 (目标清晰, 清楚哪些是要达到的, 哪些不是-“我们不是在改变世界”);
- 制定策略 (参见 6.2.2 节)。

确保实施到位

步骤 4. 进行沟通，使人理解并引入变更过程

- 提供以下相关信息（可以通过演示、路演、新闻短讯等方式）：
 - 变更的措施如何与组织设定的目标对齐（例如，战略、政策、目标等）；
 - 这些方法如何对员工有益并改善他们的工作；
 - 之前的成功和失败之处，表明这次有什么不同。
- 初期思想（在一个低风险的项目上采用自下而上的变更策略，有可能会支持初期思想）；
- 激励所有的变更影响者（详细内容请参见 [Maslow] 中的“需求的层次”）。

步骤 5. 授权他人行动

- 为实行变更提供管理支持（比如，帮助解决阻碍变更的因素）；
- 为变更影响着提供反馈机制，基于文化的原因，该机制可以匿名方式。

步骤 6. 展示短期成功

- 快速的成功—公布奖励—使他们获得动力和激励；
- 优先安排合理预期的速赢措施，这些速赢措施不会在不久的将来因为更广泛的考虑因素而撤销。

步骤 7. 持之以恒

- 确保自上而下的改进策略，由独立的改进小组支撑，该改进小组对改进过程负责，并最终确保商定的变更按照改进计划得以实施。（例如，测试过程小组，参见 7.1.1 节）。

坚持不懈

步骤 8. 创建新的文化

- 以增长的方式逐步的展开变更，尽可能避免大的变更；
- 确定引入的变更是否起到了改进的作用，将每次的成功看成机会，分清哪些正确，哪些可以进一步改进；
- 公开目标成就（做好事情并与别人讨论），可在定量度量项或者定性指标下进行（例如，基于类似“事情有没有改善？”）；
- 如果没有达到目标，需要分析原因，并从中吸取教训；
- 如果采纳变更中遇到了问题，确保能得到管理层的支持；
- 建立持续改进的文化氛围。

8.3 变更管理过程的人为因素

本节讨论对于变更的态度以及学习的需求，个人或团队对变更的反应依赖于他们之前实施变更的经验、对变更的态度、对组织的信任程度以及团队与变更的相关程度。

在制定改进实施计划时，变更管理过程必须允许认识变更、讨论变更、允许存在对变更具有不同态度。

对于变更中人为因素有价值的信息，在 [Karten 09] 和 [Kübler-Ross 07] 中有论述。

[Karten 09] 中所涵盖的材料主要论述萨提亚模型（Satir Model）和在 IT 项目中变更的人为因素。Satir 模式描述了变更对个人或小组工作表现的影响，对生产力的影响，变更中的各方面因素组成如下：

- 历史状况与目前的正常状态；
- 干扰事件（“外来因素”）；
- 混乱与干扰事件的反应；

- 转变思路与走出混乱；
- 实践与整合与变更调整；
- 新状况与新的正常状态。

伊丽莎白库伯勒 - 罗斯模型 [Kübler-Ross 07] 考察围绕丧亲之痛和预期的丧亲之痛的悲伤阶段，作为业务变更的比喻，论述人们如何处理在实际工作中的变更。最近，亚当斯加入更多的阶段（用*标记）。这些阶段内容如下：

- 缓解*—至少现在知道发生了什么；
- 震惊和惊讶*—难以置信的感觉；
- 否定—不接受所有变更，可能就是向自己证明变更没有发生，或希望它消失；
- 愤怒—经历愤怒和沮丧；
- 谈判—试图不让变更发生；
- 压抑或抑郁—情绪低落，表现为冷漠或悲伤；
- 接受—接受现状；
- 试验*—对接受状况有很深的内在认识之后，会产生可能将来还会发生事情的想法；
- 发现*—发现事情没有开始想象的那么坏。

注意：虽然萨提亚模式和库伯勒-罗斯模型都描述了变更过程的阶段，但这些阶段并不是线性的过程。人们在面对变更时不一定会按这个顺序经过这些阶段，可能有些阶段会重复，甚至有些阶段根本就没有。这些“阶段”只是人们对变更情感反应的一些简要描述。

[Honey&Mumford 02], [Kirton web], 和 [Myers&Briggs 95] 提供了如下的有关信息：

- 个性类型（例如：麦尔必瑞斯类型（Myers-Briggs））；
- 变更和改善的动机；
- 变更是否受欢迎或阻碍；
- 团队是否迟早都会成为变更接受者；
- 他们是否做好试验准备并可以接受变更失败，还是他们在没有完美的解决方案之前不准备变更；
- 每个人都有自己喜欢的学习方式，因此变更建议的接受和参与方式需要多样化[Honey & Mumford]；
- 采用不同的方法促使人们接受并喜欢变更，适应现有的方法或创造新的方法[Kirton Web]。

9. 改进成功的要素

300 分钟

关键字:

改进成功的要素（critical success factor），测试过程改进宣言（test process improvement manifesto）

本章学习目标:

9.1 成功的关键要素

- LO 9.1.1 (K2) 解释因未考虑成功的关键要素带来的风险。
- LO 9.1.2 (K5) 评估测试改进项目的关键成功要素。
- LO 9.1.3 (K5) 推荐适当的措施以降低项目改进风险。

9.2 为改进建立文化

- LO 9.2.1 (K2) 理解建立改进文化所包含的因素。
- LO 9.2.2 (K6) 创建考虑文化因素的测试改进计划。

9.1 成功的关键要素

第 8 章引入测试过程改进并取得成功的关键因素是描述和确定变更管理过程。在这一章中，我们将会讨论很多其他的因素，并将其分为两组。

“着手开始”

第一组成功因素主要针对改进项目的启动阶段，并可与理想改进框架中的“启动”阶段和“诊断”阶段相结合（参见 2.4.2 节）。这些成功因素包括：

- 对改进过程设置清晰的、可度量的和现实的目标；
- 管理层的承诺和合适的支持；
- 测试改进计划作为正式的项目来组织；
- 项目人员有足够的时间参与项目；
- 目标与（开发）组织的成熟度相呼应；
- 建立变更管理过程（参见第 8 章）。

“着手落实”

第二组成功因素主要针对改进项目的实施阶段。这些因素包括：

- 清晰的改进时间跨度和反馈周期长度的定义。这些时间跨度都不太大，因此可以保持这一势头；
- 每个周期的改进目标都是清晰的、可度量的和现实的；
- 确定和组织改进过程的所有权；
- 控制和监视变更管理过程的所有步骤（参见第 8 章）；
- 定义和实施改进之中需要测试专家参与；
- 当问题出现在测试学科之外时，其他的利益相关者需要参与进来，例如，说明规格的质量、变更和发布管理过程；
- 受控的抵制和营销执行，例如，抵制的水平将会取决于之前过程改进的成功或失败；

- 尽可能使用已有的实践。不为了改变而改变。如果现有的可用的实践未被使用，首要的事情是调查清楚不使用它的原因；
- 组织稳定的项目团队协同工作并且欢迎变更；
- 考虑应用支持和（或）启动测试改进的工具（参见2.5.4.2节）；
- 考虑加入具有合适的知识和技能的人员。这不仅覆盖了一般的测试，而且覆盖了改进过程的相关部分和所采用的改进方法的使用技能（例如，特定的模型和分析技术）；
- 考虑人力因素，如学习方式、性格特点和态度；
- 在需要时，可以请外部顾问参与。例如，为了特定的知识和技能，但不应该让他们对改进项目担负全部责任；
- 应意识到那些可能是强制性的外部标准，例如，FDA对医疗行业的约束；
- 整体过程和术语应预先定义，以确定它们在改进策略的各个组件中是一致的，并且属于整体框架的一部分；
- 构建与所有受影响的相关者的关系，例如与软件过程改进人员、质量保证人员和人力资源人员；
- 清晰的进度展示；
- 内部批准和（或）服从监管程序；
- 确保与其他改进措施相一致；
- 开发和测试的成熟度水平保持大致上的步调一致，以避免潜在的过程不一致。

9.2 为改进建立文化

改进需要置于公司文化背景之中，举例如下：

- 管理文化（指挥和控制，咨询，团队驱动）将会影响提议的策略的可接受度；
- 组织所处的地理位置（例如，一些模型和改进策略在美国能够获得更好的接受，而另一些可能亚洲人更容易接受）；
- 对于改进的目标、策略和战略以及态度（例如，在组织中是否有正在使用的改进策略或是已经获得成功的策略）；
- 各部门间的关系，例如，当两家公司合并时，是否会出现对那些被认为是来自于其他组织的改进过程变更的抵触；
- 正在使用的生命周期模型（线性模型、迭代模型、敏捷模型、本土（Home grown）模型、非过程模型）将会影响那些对项目来说是可接受的改进的频率；
- 正在使用的测试方法（自动化测试、手工测试、脚本化测试、探索性测试、混合测试、随机测试）将会影响已被提议的改进类型的可接受程度。

测试过程改进宣言[van Veenendaal Paper 08]是一个策略的例子。它以敏捷宣言为模型，并建议考虑下列五个要点：

“**灵活性胜过详细过程**”建议组织需要应对变化，并且能够处理这些变化带来的风险。对灵活性的需要，正是对测试工作人员是知识性人员的认知。他们需要思考、适应和应用那些根据项目的具体情况而定的改进。改进过程的灵活性和自由度源于对员工的信任，并能够激励他们主动改进。

“**最佳实践胜过模板**”表明模板的可用性，但范例能更好的展示如何使用模板。最佳实践范例没有必要作为事实标准。它们只是在特定的情况中从众多的实例中挑选的最好实例。

“**以部署为导向胜过以过程为导向**”表明构建过程容易，但改进的挑战在于部署（实施）流程，使之真正采用。所有过程改进都与变更管理相关；改进投资的主体是部署投资。

“**评审胜过质量保证（部门）**”表明项目的成功离不开交流与反馈。如果应用得当，这正是同行审查。但质量保证人员并不能给出及时有效的反馈，本团队内快速的反馈将会起到更好的作用。

“**业务驱动胜过模型驱动**”表明，改进应与业务收益而不仅仅与外部标准一致。

如果与组织中团队驱动方式、敏捷软件开发和探索式测试同步，那么这一改进方法将是可接受的。如果在命令和控制的管理风格之中，或是在强烈依赖于过程细节之中，或是在脚本化的测试中，这一方法很难被接受。

中国软件测试认证委员会 (CSTQB)

10. 适应不同生命周期模型

60 分钟

关键词:

敏捷 (Agile), 敏捷测试 (agile testing), 极限编程 (extreme programming), 生命周期模型 (life cycle model), SCRUM, 项目追溯 (project retrospective), RUP。

适应不同生命周期模型的学习目标

10.1 适应不同生命周期模型

- LO 10.1.1 (K2) 了解改进的组织方式的影响因素并且这些因素总是与环境相关。
- LO 10.1.2 (K2) 总结敏捷环境中的测试改进方法。
- LO 10.1.3 (K2) 总结迭代环境中的测试改进方法。
- LO 10.1.4 (K2) 举例说明改进模型何处需要调整以适应敏捷和/或迭代生命周期。

10.1 适应不同生命周期模型

前面教学大纲里描述的改进方法并不针对特定的生命周期方法。然而,改进总是需要在一个特定范围内进行设置,例如:

- 组织的管理文化(命令和控制型、协商型、团队驱动型)会影响所建议的方法被接受的程度;
- 正在使用的生命周期模型(顺序、迭代、敏捷、本土模型或无过程)会影响项目对过程周期变更频率的接受程度。
- 正在使用的测试方法(自动、手工、脚本、探索、混合方法、随机测试)会影响建议的变更类型的接受程度。

例如采用敏捷软件开发生命周期是指:

- 强调自我管理团队,可以按需要改变自己的流程;
- 需要考虑与精益方法的关系,尤其是在组织层面。

这些因素可能使我们偏向于选择那些大家青睐的改进方法。精益管理,如短戴明周期和因果图的使用。这并不意味着这些方法不能用于其他生命周期模型,也不意味着以模型为基础的方法不能运用在使用敏捷生命周期的组织或项目上。生命周期的选择不应该决定改进方法的选择。

任何软件过程模型或测试过程模型都可以用作关于如何切实提高已被判定为重要改进的参考点。与强调顺序的传统生命周期模型相比,在敏捷或迭代情况下,很多可选择的想法会导致不同的改进路径。

在一个迭代环境下,人们可能会使用 RUP 中的想法,或在敏捷情况下使用 SCRUM 中的想法,如在每个迭代(Sprint)结束时的追溯会议中提供快速反馈和每隔几天就进行过程改进的机会。在敏捷开发环境下,许多强调顺序的、基于内容模型的改进路径需要进行程度地剪裁。在 SCRUM、极限编程或其他相关开发测试实践中所定义的敏捷测试,可以提供更适合敏捷过程的测试结构。

顺序模型(如 V 模型)在所有的阶段结束时具有反馈回路来检查产品和过程的一致性和适用性(验证和确认)。基于这些阶段结束时的评审,测试过程(静态和动态测试)的改进建议可使用本章所述的任何方法来实现。

如果选中一个特定的成熟目标,例如目标是一个 CMMi 等级,这并不妨碍使用测试过程改进的任何特定方法,也没有规定任何具体的生命周期模型。

改进小组的重点是:

- 确定选择的生命周期模型是否会诱发特定的改进过程的选择；
- 根据情况确定合适的过程改进方法；
- 根据情况确定合适的测试结构和活动。

中国软件测试认证委员会 (CSTQB)

11. 参考文献

11.1 标准

编号	参考标准
[BS7925-2]	BS 7925-2 Software Component Testing
[IEEE 1044]	IEEE Std 1044™ IEEE Standard Classification for Software Anomalies
[ISO 25000]	ISO/IEC 25000:2005 Software engineering. Software product quality requirements and evaluation (SquaRE). Guide to SquaRE
[GB/T 25000.1-2010]	《软件工程 软件产品质量要求与评价(SQuaRE)SQuaRE 指南》
[ISO/IEC 15504]	ISO/IEC 15504-5 ISO 15504 – SPICE (Software Process Improvement and Capability dEtermination), Part 5, Assessment Model [1998]
[ISO 9126]	ISO/IEC 9126-1:2001, Software Engineering – Software Product Quality
[GB/T16260-2006]	《软件工程 产品质量 第1部分：质量模型》
[ISTQB-Glossary]	ISTQB Glossary of terms used in Software Testing, Version 2.1, available from [ISTQB-Web]

11.2 商标

本文当中使用了以下注册商标和服务标识：

CMM®, CMMI®, EFQM Excellence Model™, TMMSM, TMMi®, IDEALSM, ISTQB®, TMap®, TPI® and TPI Next®

CMM 和 CMMI 由卡耐基梅隆大学在美国专利和商标局注册

EFQM Excellence Model™ 是欧洲质量管理基金会(European Foundation for Quality Management)的商标

IDEAL 是卡耐基梅隆大学软件工程研究所(SEI)的服务标识

ISTQB® 是国际软件测试认证委员会的注册商标

TMM 是伊利诺理工大学的注册服务标识

TMMi 是 TMMi 基金会的注册商标

TPI 是荷兰 Sogeti 公司的注册商标

TPI Next 是荷兰 Sogeti 公司的注册商标

11.3 书籍

编号	参考书籍
[Adams et al]	Adams, Hayes and Hopson, "Transition: Understanding & managing personal change", 1976
[Anderson 01]	Anderson, L. W. and Krathwohl, D. R. (eds) (2001). "A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives", Allyn & Bacon.
[Atwater 81]	Eastwood Atwater, "I Hear You". Prentice-Hall, 1981, ISBN: 0-13-450684-7
[Bernstein]	Bernstein A. J., "Emotional Vampires: Dealing With People Who Drain You Dry" McGraw-Hill Professional; ISBN: 978-0071381673
[Black03]	Rex Black, "Critical Testing Processes", Addison-Wesley, 2003, ISBN: 0-201-74868-1
[Burnstein 03]	Ilene Burnstein, "Practical Software Testing", Springer, 2003, ISBN: 0-387-95131-8
[Buzan 95]	Tony Buzan, "The Mindmap Book", BBC-Books, 1995, ISBN: 0-563-37101-3
[Cialdini]	Cialdini, R, "Influence: The Psychology of Persuasion", Harper Business ISBN: 978-0061241895
[Craig02]	Craig, Rick David; Jaskiel, Stefan P., "Systematic Software Testing", Artech House, 2002, ISBN: 1-580-53508-9
[Copeland 03]	Lee Copeland, "A Practitioner's Guide to Software Test Design", Artech House, 2003, ISBN: 1-58053-791-X
[Evans04]	Isabel Evans, "Achieving Software Quality through Teamwork", Artech House, 2004, ISBN: 978-1580536622
[Few 08]	Few, S., "Information Dashboard Design: The Effective Visual Communication of Data"; ISBN: 978-0596100162
[Frank 90]	Milo O. Frank, "How to get your point across in 30 seconds or less", Simon & Schuster 1990, ISBN: 0-552-13010-9
[Gilb & Graham]	Gilb, T., and Graham, D., "Software Inspection", Addison Wesley 1993, ISBN: 0-201-63181-4
[Gladwell]	Gladwell, M., "The Tipping Point: How Little Things Can Make a Big Difference"; ISBN: 978-0349113463
[Honey&Mumford 02]	Honey, P. and Mumford, A., "The Learning Styles Helper's Guide", Peter Honey Publications, 2002, from Peter Honey Publications Limited, 10 Linden Avenue, Maidenhead, Berks SL6 6HB or website [Honey-Web]
[Humphrey]	Humphrey, W. "Introduction to the Team Software Process", Massachusetts: SEI, 2000 "Introduction to the Personal Software Process", Massachusetts: SEI, 1997
[Huff 93]	Darrell Huff, "How to Lie with Statistics", W.W. Norton, 1993, ISBN: 0-393-31072-8
[ISTQB-CEP]	ISTQB Certified Tester Expert Level, Certification Extension Process, [ISTQB-Web] 版本 1.0, 2008 年 6 月 17 日. 可以从[ISTQB 网站]下载
ISTQB-EL-EXAM	ISTQB 专家级考试指南, 版本 1.0, 可以从[ISTQB 网站]下载

- [ITIL] ITIL, Best Practice for Service Support, Office of Government Commerce, 2002
- [ITIL2] ITIL, Best Practice for Service Delivery, Office of Government Commerce, 2002
- [IDEAL 96] Bob McFeeley/Software Engineering Institute (SEI), "IDEAL: A User's Guide for Software Process Improvement", 1996, CMU/SEI-96-HB-001
- [Ishikawa 91] "What Is Total Quality Control? The Japanese Way", Prentice Hall Business Classics, ISBN:013952441X
- [Juran] Quality Handbook (McGraw-Hill International Editions, Industrial Engineering Series) ISBN: 0071165398
- [Karten 09] Naomi Karten, "Changing How You Manage and Communicate Change: Focusing on the Human Side of Change", IT Governance Publishing, 2009, ISBN: 978-1905356942
- [Kotter & Rathgeber 05] John Kotter and Holger Rathgeber, "Our Iceberg is Melting", Pan Macmillan, 2005, ISBN: 978-0-230-01420-6
- [Koomen/Pol 99] Tim Koomen, Martin Pol, "Test Process Improvement", Addison-Wesley, 1999, ISBN: 0-201-59624-5
- [Kübler-Ross 07] Elisabeth Kubler-Ross & David Kessler, "On Grief and Grieving: Finding the Meaning of Grief Through the Five Stages of Loss", Scribner Book Company; Reprint edition, 5 Jun 2007, ISBN :978-0743266291
- [Maslow] Maslow, A H,
"Toward a Psychology of Being", ISBN: 978-0471293095, Wiley 1998
"Maslow on Management", ISBN: 978-0471247807, Wiley, 1998
- [Mayer 04] Mayer, J.D., "Emotional intelligence: Key readings on the Mayer and Salovey model", 2004, ISBN: 1-867943-72-2
- [Myers&Briggs 95] Myers, Isabel Briggs (1980). "Understanding Personality Type". Davies-Black Publishing; Reprint edition ,1995, ISBN: 0-89106-074-X
- [Nance & Arthur 02] "Managing Software Quality: A Measurement Framework for Assessment and Prediction", Springer, 2002, ISBN: 1852333936
- [Page 08] Page, A, Johnston, K, Rollinson B, "How we test software at Microsoft", pub Microsoft, 2008, ISBN: 978-0-7356-2425-2
- [Pol.M & Van Veenendaal. E 98] Pol.M and van Veenendaal. E, "Structured testing of information systems: an introduction to Tmap[®]", Kluwer, 1998, ISBN: 90-267-2910-3
- [Robson 95] "Problem Solving in Groups", Gower, 1995, ISBN: 0-566-07415-x
- [Satir 91] Satir V., Banmen, J., Gerber J., Gomori M. "The Satir model: Family therapy and beyond", Science and Behavior Books, Inc. 1991, ISBN 978-0-831400-78-1
- [Sogeti 09] Sogeti, "TPI Next – Business Driven Test Process Improvement", UTN Publishing, 2009, ISBN 90-72194-97-7
- [Trienekens & van Veenendaal 97] Trienekens and van Veenendaal, "Software Quality from a Business Perspective", Kluwer, 1997
- [Tufte 90] Tufte, E., "Visual Explanations", Graphics Press, 1990, ISBN: 0-961-39214-2
- [Tufte 97] Tufte, E., "Envisioning Information", Graphics Press, 1997, ISBN 0-961-39211-8
- [Wagner 91] "The Transactional Manager", Spiro Press, ISBN: 978-185835496, 1996
- [Weinberg 92] Weinberg, G, "Quality Software Management, Vol. 1", (92), Dorset House,

1992, ISBN: 0-932633-22-6

11.4 论文和文章

编号	参考文献/文章
[Basili Papers]	<ul style="list-style-type: none">• V.R. Basili, "Software Modeling and Measurement: The Goal Question Metric Paradigm," Computer Science Technical Report Series, CS-TR-2956 (UMIACS-TR-92-96), University of Maryland, College Park, MD, September, 1992.• V.R. Basili, H. D. Rombach, "The TAME Project: Towards Improvement-Oriented Software Environments," IEEE Transactions on Software Engineering, vol.SE-14, no.6, June 1988, pp.758-773.• V.R. Basili, R.W. Selby, "Data Collection and Analysis in Software Research and Management," Proceedings of the American Statistical Association and Biomeasure Society, Joint Statistical Meetings, Philadelphia, PA, August 1984.• R. Basili, D. M. Weiss, "A Methodology for Collecting Valid Software Engineering Data," IEEE Transactions on Software Engineering, vol. SE-10, no.6, November 1984, pp. 728-738.
[Copeland Paper 01]	Lee Copeland, "When helping doesn't help", SQE Magazine, January 2001
[Garvin Paper 84]	Garvin, D., "What does product quality really mean?", Sloan Management Review, Vol. 26, No. 1, 1984
[Hackman and Oldman Paper 76]	J. R. Hackman, G. R. Oldham (1976). "Motivation through design of work". <i>Organizational behaviour and human performance</i> 16 : 250–279.
[van Veenendaal Paper 08]	van Veenendaal, E., "Test Process Improvement Manifesto", Testing Experience, Dec 2008

11.5 参考网站

在出版专家级大纲时，ISTQB 检查了下列参考网站，但是 ISTQB 不承担这些网站都可以访问的责任。

编号	参考网站	链接
[ISTQB 网站]	国际软件测试认证委员会网站，参考了本网站最新的 ISTQB 术语和大纲。	www.istqb.org
[EFQM 网站]	欧洲质量管理基金会的网站 说明：EFQM 国家/洲际特定模型的网站提供链接，基于为全球卓越模型组的成员提供的卓越基本概念，包括北美，拉丁美洲，澳大利亚和新加坡的成员。	www.efqm.org

EFQM 模型仅作为使用这些概念的一种体现，本文作

编号	参考网站	链接
	为一个例子引用。	
[Honey-Web]	Peter Honey 的网站	www.peterhoney.com
[Kirton-Web]	McHale, J., Innovators Rule OK-or do they?	www.kaicentre.com
[SEI-Web]	软件工程研究所的网站，发布 CMM，CMMi 和其他软件 工程。	www.sei.cmu.edu
[SFIAMWeb]	SFIA 基金会的网站（信息时代的技能框架）	www.sfia.org.uk
[TMMi- Foundation-Web]	TMMi 基金会的网站	www.TMMifoundation.org

中国软件测试认证委员会 (CSTQB)

12. 附录 A – 培训机构的注意事项

12.1 培训时间

本大纲的各章按分钟分配时间。目的是指导认证课程各个部分的相对时间比例，同时对讲授各个部分给出了最少时间。

培训机构可以用更多的时间进行授课，考生可以花费更多的时间阅读和研究。课程表可以与大纲的章节顺序不同。

不需要在连续的时间段讲授课程

可以使用下面的指导时间表：

学习目标 知识级别	分钟（平均）
K2	15
K3	60
K4	75
K5	90
K6	90

下表给出了各章讲授和练习的时间指导，以及在课外练习的时间（全部时间以分钟为单位）。注意课外练习的时间也可以作为课堂练习的一部分（参见下面的 12.3.1 节）。

Nr. 编号	章名	课程讲授和练习时间	课外练习时间	全部时间
1	简介	60	0	60
2	改进概要	285	0	285
3	基于模型的改进	390	90	570
4	基于分析的改进	465	90	555
5	测试过程改进的选择方法	15	90	105
6	过程改进	465	435	900
7	组织，角色和技能	375	90	465
8	管理变更	195	90	285
9	改进成功的要素	120	180	300
10	适应不同生命周期模型	60	0	60
	合计	2520	1065	3585

下表列出了课程总计的天数，按照每个工作日平均 7 小时计算。

课程组成	天数	小时	分钟
课程讲授和练习	6	0	0
课外练习	2	3	45
合计:	8	3	45

12.2 使用的标准

本大纲引用了一些既定的标准（参见 11.1 节），需要在准备培训材料时使用。各个使用的标准必须是本大纲引用的当前版本。也可以使用 and 参考其他的出版文献，模版或者本大纲没有引用的标准，但是不作为考试内容。

12.3 实践练习

实践工作需要涵盖学员所希望应用知识（K3 或更高学习目标）的全部内容。讲座和练习需要基于大纲内容的学习目标和主题描述。

12.3.1 课外实践练习的指导

执行课外实践练习可以实现特定的学习目标。下表列出了课外练习覆盖的学习目标：

主题领域	相关学习目标
基于模型的改进	LO 3.3.10 (K5) 使用 TPI Next 或 TMMi 模型评价测试组织
基于分析的改进	LO 4.4.2 (K5) 推荐适当的度和指标，在特定改进情境下跟踪改进趋势
测试过程改进的选择方法	LO 5.1.2 (K5) 在特定场景和给定改进范围中推荐测试改进的方法
过程改进	LO 6.3.2 (K6) 使用特定的流程或内容模型计划和执行评估面谈，演示面谈风格和人际技巧认知
	LO 6.3.3 (K6) 在成果分析的基础上创建并提交总结以及评估的成果
	LO 6.3.4 (K2) 总结解决方案分析的方法
	LO 6.3.5 (K5) 在评估结果和执行分析的基础上，推荐测试过程改进的行动
	LO 6.4.5 (K6) 创建测试改进计划
	LO 6.5.2 (K4) 从可行性列表中选择合适的试点工作
组织，角色和技能	LO 7.3.2 (K5) 评估测试的专业性（例如潜在的 TPG/技术工作组的成员），关注执行评估所需的主要软技能的不足之处。
管理变更	LO 8.2.2 (K6) 创建测试改进计划，考虑变更管理问题，合适的步骤和行动
改进成功的要素	LO 9.1.2 (K5) 评估测试改进项目的关键成功因素
	LO 9.1.3 (K5) 推荐合适的措施，减轻识别的项目风险

下列指南适用于：

1. 培训机构需要在上课前发布课外实践练习的要求；

2. 培训机构需要在实践练习前批准学员提交的建议；
3. 培训机构需要确保在学员做实践练习前已经讲授了相关知识；
4. 培训机构和学员之间必须交流回答问题和检查进度；
5. 实践练习的结果必须提交给培训机构。建议介绍这些结果或者至少其他学员可以看到。

12.4 培训机构有关标准的通用指南

12.4.1 专家级培训机构标准

培训机构需要向 ISTQB 的国家委员会（注：在中国是 CSTQB）提交一份申请表。国家委员会授予培训机构根据下列条件开展 ISTQB 专家级模块课程培训的权利：

- 通常培训机构也被授权提供下列 ISTQB 课程：
 - 初级（基础级）；
 - 高级，测试经理模块。
- 如果向委员会充分说明理由，可以允许替换方案；
- 培训机构最少具备国家委员会批准模块的两位专家级讲师。如果仅有一位讲师需要 ISTQB 酌情批准；
- 培训机构具有被认可的测试地位（作为一家培训机构或其他），特别在“改进测试过程”领域。

12.4.2 专家级课程

培训机构需要向国家委员会提交一份具备授权课程材料的申请表。国家委员会根据下列条件批准课程材料：

- 符合大纲的专家级模块课程材料；
- 培训时间符合 12.1 节“培训时间”中的规定值；
- 课程中的练习讨论和根据工作分配的时间不少于 12.1 节“培训时间”给出的时间；
- 每个班最多 10 人（使讲师以学员为中心）；
- 需要使用 ISTQB 当前定义的术语。

12.4.3 讲师的准入标准

专家级需要讲师的正式认可。讲师需要向国家委员会提交一份申请表，说明以前所教的 ISTQB 级别，也能够表明具有相应的经验，以有效的方式向学员传达，使学员能够学会。可以根据以下条件判断：

- 讲师至少通过了准入条件定义的 ISTQB 规定的高级模块或同等模块（例如，ISEB Practitioner 1.1 版—2001 年 9 月 4 日）；
- 强烈建议讲师拥有全部 ISTQB 高级证书；
- 强烈建议具有大学学位，教学经验，是公认的专家并且具有丰富的经验。由于全部达到这些条件是不现实的，各个条件之间可以互相补充。因此，讲师需要满足以下附加条件中的至少两项：
 - 讲师具有丰富的讲课经验（5 年讲课经验，每年讲授 3 个班，共计 5 天），并且具有专项领域的讲课经验（讲授至少 3 个班的专家级模块所覆盖专题领域）；
 - 讲师是公认的测试行业带头人，会议主旨发言人，作者或同等资历；
 - 讲师具有测试相关的高级大学学位（例如，学士，硕士或博士）；

- 讲师具有至少**6**年实际工作经验，**2**年模块所属领域的专业知识。

如果不符合条件，只要讲师通过了 ISTQB 高级大纲的全部认证，满足附加条件列表的至少一个条件，ISTQB 可以任命讲师作为专家级课程的见习讲师。在与专家级讲师合作 **3** 年后，见习讲师可以再次发送申请获得 ISTQB 认可的正式讲师地位。

中国软件测试认证委员会 (CSTQB)

索引

- CMM, 16
- CMMI, 22
- CTP, 16, 23, 25
- EFQM, 13
- FDA, 18
- GQM 方法, 28
- IDEAL 改进架构, 15
- IEEE 1044, 28
- ISO 9000, 13
- ISO/IEC 15504, 14, 16, 23
- ITIL, 14
- SCRUM, 18, 58
- STEP, 16, 23, 25
- TMMi, 23, 24
- TPI/TPI Next, 23, 24
- TQM, 13
- V 模型, 58
- 临界点, 39
- 事务分析, 48
- 倾听技能, 49
- 六西格玛, 13, 16
- 内容模型, 16, 32
- 分析技能, 50
- 分析方法, 17, 26, 32
- 卓越的基本概念, 16
- 发布后缺陷率, 30
- 变更管理, 52
 - Satir 模式, 53
 - 人为因素, 53
 - 悲伤阶段, 54
 - 过程, 52
- 启动改进过程, 35
- 原因分析, 27
- 因果图, 27
- 团队与个人软件过程, 14
- 培训时间, 65
- 基于模型方法, 16
 - 假设, 22
 - 理想特性, 21
- 大纲期望, 11
- 实践练习, 66
- 审查过程, 28
- 工作量偏移 (或成本偏移), 31
- 工具, 17
- 平衡计分卡, 16
- 建议改进活动, 41
- 开展面谈, 39
- 情商, 48
- 成功的关键要素, 55
- 战术实施计划, 43
- 战略实施计划, 43
- 戴明环, 15
- 技能, 17
- 改进概要, 12
- 敏捷, 58
- 文化改进, 56
- 早期缺陷检测, 30
- 测试执行交付周期偏移, 31
- 测试改进, 14
- 测试改进计划, 41
- 测试效率, 30
- 测试生产力, 30
- 测试用例偏移, 31
- 测试过程改进人员
 - 技能, 47
 - 角色, 47
- TPG, 45
- 混合方法, 17
- 演示和报告技能, 49
- 相互依赖行为, 48
- 相对测试投入, 30
- 管理变更, 52
- 管理技能, 51
- 系统思考, 39
- 组织, 45
- 缺陷检测率 (DDP), 29
- 自动化水平, 30
- 萨班斯-奥克斯利法案, 18
- 覆盖率指标, 31
- 解决方案分析, 40
- 计划评估, 38
- 记录技能, 50
- 评估
 - 准备, 39
 - 分析结果, 39
 - 计划, 39
- 评估人角色, 47
- 诊断当前状况, 38
- 说服技能, 50
- 质量属性度量, 31
- 质量比率成本, 30

质量的组织成本, 30

质量观, 14

软件过程改进 (SPI), 14, 22

过程模型, 32

连续式表示法, 21

适应不同生命周期模型, 58

阶段式表示法, 21

面谈技能, 47

中国软件测试认证委员会 (CSTQB)