**软件过程改进(SPI.Software Process Improvement)**

软件过程方法从上世纪90年代开始在软件开发中得到应 用，被许多软件开发组织所接受。并被认为是软件生产达到 工业化前的一个必须经历的阶段，是软件工程学科发展中的 一个重要里程碑，软件过程理论是现代软件开发人员和管理 人员必备的知识。

软件过程将技术、人和管理紧密地结合在一起，过程改 进是软件开发组织提高软件质量、提高生产率、降低成本的 一种有效方法。

软件过程改进已经形成了一套改进和评估的方法，代表 性成果有CMMI、ISO15504、ISO9000、6σ等。国内外众多软 件开发组织都以通过过程改进评估为手段，达到提高竞争力 的目的。

**一、名词解释**

**1.软件生存周期(Software Life Cycle)**

软件生存周期又称为软件生命期，生存期。是指从形成开发软件概念起，所开发的软件使用以后，直到失去使用价值消亡为止的整个过程。一般来说，整个生存周期包括计划（定义）、开发、运行（维护）三个时期，每一个时期又划分为若干阶段。每个 阶段有明确的任务，这样使规模大、结构复杂和管理复杂的软件开发变得容易控制和管理。SDLC的六个阶段：1. 定义及规划2.需求分析3. 软件设计4.程序编码5.软件测试6.运行维护

**2.项目(Project)**

项目是指一系列独特的、复 杂的并相互关联的活动，这些活动有着一个明确的目标或目的，必须在特定的时间、预算、资源限定内，依据规范完成。项目是为创造独特的产品、服务或成果而进行的临时性工作。项目参数包括项目范围、质量、成本、时间、资源。

**3.里程碑(Milestone)**

在制定项目进度计划时，在进度时间表上 设立一些 重要的时间检查 点，这样一来，就可以在项目执行过程中利用这些 重要的时间检查 点来对项目的进程进行检查 和控制。这些重要的时间检查 点被称 作项目的里程碑（Milestone）。

**4.基线(Baseline)**

基线是项目文档或者源码的一个稳定版本，它是进一步开发的基础。基线可以理解为项目版本中的一个’快照‘，它提供一个正式标准，随后的工作基于此标准，并且需经过授权才可以更改此版本。

**5.软件过程度量(Software Process Measurement)**

软件过程度量是指针对所指定的软件过程，以某种方式使其过程能力指标实现合理的量化，从而以一定的标准来衡量该软件过程的质量。过程度量是软件过程管理的重要内容，是进行过程的分析、控制和改进的基础。通过对过程进行合理度量，能够认知过程的现状，并在理解的基础上进行评价，然后分析过程产生的偏差并查找偏差原因，从而使已经或者可能将要失控的项目得到合理的控制，同时还能评价过程改进的效果，达到过程评价的认知目的。

**6.功能点分析(Function Point Analysis)**

功能点分析方法是最重 要也是最有效的软件 测量规模方法，它可以在项目早期就对软件项目进行测量，并在开发过程中不断地更新数据，从而实现一种持续一致的管理。从应用方面看，全球已经有成千上万个项目采用了功能点分析方法。从研究方面来看，功能点分析方法也已成为很多其他新型测量方法的基础。功能点分析是项目工作量估算的一种常 用方法，可用7个步骤概 括：1.确定功能点计算的类型；2.确定计算范围和应用程序边界；3.确定所有数据功能及其复杂性；4.确定所有事务 功能及其复杂性；5.得出未调整 功能点计数；6得出基于系统基本特征的值调整因子；7.计算已调整功能点计数。

**7.工作分解结构(WBS)**

WBS是项目管理重要的 专业 术语之一。WBS的基 本定义 ：以可交付 成果为导向对项目要素进行的分组，它归纳和定义了项目的整个工作范围每下 降一层代表 对项目工作的更详细 定义。 无论在项目管理实践中， 还是在PMP，IPMP考试中，工作分解结构（WBS）都是最重 要的 内容 之一。WBS总是处于计划过程的中心，也是制定进度计划、资源需求、成 本预算、风险管理计划和采购计划等的重要基础。WBS同时也是控制项目变更的重要基础。

**8.软件质量**

软件度量是对软件开发项目、过程及其产品进行数据定义、收集以及分析的持续性定量化过程，目的在于对此加以理解、预测、评估、控制和改善。

**9.RMMM计划(Risk Mitigation,Monitoring and Management Plan)**

软件项目风险管理是软件项目管理的重要内容。在进行软件项目风险管理时，要辩识风险，评估它们出现的概率及产生的影响，然后建立一个规划来管理风险。风险管理的主要目标是预防风险。软件项目风险是指在软件开发过程中遇到的预算和进度等方面的问题以及这些问题对软件项目的影响。软件项目风险会影响 项目计划的实现，如果项目 风险 变成现实，就有可能影响 项目的进度，增加 项目的成本，甚至 使软件项目不能实现。如果对项目进行风险管理，就可以最大限度的减少风险的发生。

**10.COCOMO模型**

结构性成本模型。它是由巴里·勃姆（Barry Boehm）提出的一种软件成本估算方法。这种模型使用一种基本的回归分析 公式， 本质上说是一 种参数 化的项目估算方法，参数建 模是 把项目的 某些特 征作为 参数 ，通过建立 一个 数字模型预测项目成本（类似于居住面积作为参数计算的整体的住房成本）。

**11.项目计划评审技术**

计划评审技术就是把工程项目当作一种系统，用网络图或者表格或者矩阵来表示各项具体工作的先后顺序和相互关系，以时间为中心，找出从开工到完工所需要时间的最长路线，并围绕关键路线对项目进行统筹规划，合理安排以及对各项工作的完成进度进行严密的控制，以达到用最少的时间和资源消耗来完成系统预定目标的一种计划与控制方法。

**12.软件质量模型**

一个软件的质量往往涉及到许多不同的质量属性,不同类型的软件所关注的质量属性也不尽相同。因此, 为了更好地理解、预测和评价软件和信息系统的质量, 人们建立了各种质量模型, 在软件生命周期的不同阶段对软件质量进行评测。常用的通用软件质量模型主要包括层次模型和关系模型, 它们在当前的软件开发中起到了一定的积极作用。

**13.基于时间的缺陷到达模式**

产品的缺陷密度、或者测试阶段的缺陷率是一个概括性指标，缺陷到达模式可以提供更多的过程信息，有时即使得到的整体缺陷率是一样的，但其质量差异可能较大，原因就是缺陷到达的模式不一样。越多的缺陷到达越早，则测试过程质量就越好。无论是从测试进展的观点，还是从用户重新发现(customer rediscoveries)的观点来看，缺陷的过程跟踪是非常重要的，开发周期里大量的严重缺陷将有可能阻止测试的进展，也必然直接影响软件产品的质量和性能。定性的分析比较容易，测试团队越成熟，峰值到达得越早，有时可以在第一周末或第二周就达到峰值。这个峰值的数值取决于代码质量、测试用例的设计质量和测试执行的策略、水平等，多数情况下，可以根据基线（或历史数据）推得。从一个峰值达到一个低而稳定的水平，需要长得多的时间，至少是达到峰值所用的时间的4-5倍。这个时间取决于峰值、缺陷移除效率等等。 在测试阶段初期，缺陷率增长很快。在达到峰值后，就随时间以较慢的速率下降，降低到最低点——零点。

**14.软件过程**

软件过程为一个为建造高质量软件所需完成的任务的框架，即形成软件产品的一系列步骤，包括中间产品、资源、角色及过程中采取的方法、工具等范畴。（课件） 人们在开发和维护软件机器相关产品时所涉及的各种活动、方法、实践和改革等。其中软件相关产品包括软件项目计划、设计文档、程序代码、测试用例和用户手册等。

**15.软件基本过程**

软件获取、供应、开发、运行和维护的过程，包括需求分析、软件设计、编码等过程。

**16.软件支持过程**

包括文档编制过程、配置管理过程、质量保证过程、验证和确认过程（测试过程）、评审过程等。

**17.软件组织过程**

对软件主要过程和支持过程的组织保证过程，包括管理过程、基础设施过程、改进过程和培训过程。

**18.过程框架**

通过在软件过程环境中定义软件过程架构、软件过程改进规划图、软件过程评估、软件过程改进计划四项框架活动来建立适用于目标软件项目的基本概念结构。

**19.软件能力成熟度模型**

软件能力成熟度模型是一种对软件组织在定义、实施、度量、控制和改善其软件过程的实践中各个发展阶段的描述形成的标准。CMM的核心是把软件开发视为一个过程，并根据这一原则对软件开发和维护进行过程监控和研究，以使其更加科学化、标准化、使企业能够更好地实现商业目标。

**20.统一过程**

统一过程主要分五个阶段：开启阶段(inception)，细化阶段(elaboration)，构建阶段(construction)，移交阶段(transition)，生产(production)。Rational Unified Process 是 Rational 公司开发和维护的过程产品。Rational Unified Process 是软件工程的过程。它提供了在开发组织中分派任务和责任的纪律化方法。它的目标是在可预见的日程和预算前提下，确保满足最终用户需求的高质量产品。

**21.过程模式**

一系列用来面向对象软件的通用技术、行为或各种任务、过程模式的一个重要特性在于，它只描述了一个软件开发人员应该做什么，而没有确切地说明应该做哪些细节。当过程模式能够被有组织的应用在一起时，它们就可以被用来为软件开发改进机构生成软件过程。因为过程模式并没有指定如何完成一个给定的工作，它们能够成为可复用的积木。软件开发人员可以据此来定制一个满足软件开发机构的特定需求的软件过程。

**22.个体软件过程**

PSP个人软件过程 (Personal Software Process，PSP) 是一种可用于控制、管理和改进个人工作方式的自我持续改进过程，是一个包括软件开发表格、指南和规程的结构化框架。PSP与具体的技术（程序设计语言、工具或者设计方法）相对独立，其原则能够应用到几乎任何的软件工程任务之中。PSP能够说明个体软件过程的原则； 帮助软件工程师作出准确的计划；确定软件工程师为改善产品质量要采取的步骤；建立度量个体软件过程改善的基准；确定过程的改变对软件工程师能力的影响。

**23.团队软件过程 TSP**

团队软件过程是为开发软件产品的开发团队提供指导，TSP的早期实践侧重于帮助开发团队改善其质量和生产率，以使其更好的满足成本及进度的目标。TSP被设计为满足2~20人规模的开发团队，大型的多团队过程的TSP被设计为大约最多为150人左右的规模。 团队软件过程(TSP)加上PSP帮助高绩效的工程师在一个团队中工作，来开发有质量保证的软件产品，生产安全的软件产品，改进组织中的过程管理。通过TSP，一个组织能够建立起自我管理的团队来计划追踪他们的工作、建立目标，并拥有自己的过程和计划。这些团队可以是纯粹的软件开发团队，也可以是集成产品的团队，规模可以从3到20个工程师不等。TSP团队在广泛领域里可能运用XP, RUP或其它方法。TSP使具备PSP的工程人员组成的团队能够学习并取得成功。如果你的组织运用TSP，它会帮助您的组织建立一套成熟规范的工程实践，确保安全可靠的软件。

**24.过程规范**

过程规范就是对输入/输出和活动所构成的过程进行明文规定或约定俗成的标准。软件过程规范是软件开发组织行动的准则与指南，可以依据上述各类过程的特点而建立相应的规范，如软件基本过程规范、软件支持过程规范和软件组织过程规范。

**25.过程模型**

所谓软件过程模型就是一种开发策略，这种策略针对软件工程的各个阶段提供了一套范形，使工程的进展达到预期的目的。对一个软件的开发无论其大小，我们都需要选择一个合适的软件过程模型，这种选择基于项目和应用的性质、采用的方法、需要的控制，以及要交付的产品的特点。 常见的模型包括瀑布模型、螺旋模型、增量模型、迭代模型、V模型等。

**26.配置管理**

配置管理（Configuration Management，CM）是通过技术或行政手段对软件产品及其开发过程和生命周期进行控制、规范的一系列措施。配置管理的目标是记录软件产品的演化过程，确保软件开发者在软件生命周期中各个阶段都能得到精确的产品配置。

**27.配置项**

配置项指纳入配置管理范畴的所有项目。凡是纳入配置管理范畴的工作成果都是配置项（CI）；一个纯软件的CIs通常也称为软件配置（CSCIs）。配置项主要有两大类：属于产品组成部分的工作成果；项目管理和机构支撑过程产生的文档。 每个配置项的主要属性有：名称、标识符文件状态、版本、作者、日期等。

**28.预防性维护**

预防性维护是软件产品交付后进行的修改，以在软件产品中的潜在错误成为实际错误前，检测和更正它们。

**29.适应性维护**

计算机领域的各个方面发展变化十分迅速，经常会出现新的系统或新的版本，外部设备及其他系统原件经常在改变，而应用软件的使用时间，往往比原先的系统环境使用时间更为长久，因此，常需对软件加以改造，使之适应于新的环境。为使软件产品在新的环境下仍能使用而进行的维护，称为适应性维护。

**二、简答题**

**1.试给出在SEI和CMMI模型中采用过程评估和改进方法的两个优点和两个缺点。**

**优点：**过程域的层次划分便于以渐进方式进行改进，这会更为有效和持久。 过程域的划分使过程改进活动中心更为突出，也更容易管理。 有限的范围可以提供更精深的细节，并为与项目有关的过程提供指导。 CMMI目标和共性实践的描述方式使其可以适用于很多类组织和项目。 CMMI提供了成熟度级别(阶段式表示法)和能力级别(连续式表示法)作为确定和评价在过程改进上所取得的进步的方法。 CMMI对量度特别重视，这有助于确定过程改进活动的投资回报。

**缺点：**模型的规模可能导致读者的信息过载，或者引发“麻痹”，或者引发“盲目实施实践”。 独立过程域的数目之多使人们难以清楚地识别和理解它们的关系。 在某些时候，组织可能会开发出反映了这些过程域而没有反映业务运作的过程结构。 模型的精细程度可能会被理解成鼓励在一些简单过程就已经足够的情况下开发复杂而详细的过程。 对与项目无关的企业职能的支持非常有限。 特定实践和解释性指导倾向于以适用于大规模开发的方式措辞，对“微小改进”和“非开发”项目的解释上只提供非常有限的指导。 目标的意义并非总能得到理解，这会导致CMMI本身变成一个目标而不再是推动改进的手段。 实现CMMI目标的时间安排可能与短期业务循环无法保持一致。

**2.考虑你所在机构中所用的软件过程类型。使用 SEI 模型找出了多少个关键过程域？根据该模型，你所在机构的过程成熟度等级如何划定？**

（1）考虑我在机构所用的软件过程类型，使用SEI模型找出了如下关键过程域：

【过程管理】： ①OPD：组织级过程定义。建立和维护有用的组织过程资产。 ②OPF：组织级过程焦点。在理解现有过程强项和弱项的基础上计划和实施组织过程改善。 ③OT：组织培训管理。增加组织各级人员的技能和知识，使他们能有效地执行他们的任务。 【项目管理】： ④PP：项目计划。保证在正确的时间有正确的资源可用。为每个人员分配任务、协调人员。根据实际情况，调整项目。 ⑤PMC：项目监督与控制。通过项目的跟踪与监控活动，及时反映项目的进度、费用、风险、规模、关键计算机资源及工作量等情况，通过对跟踪结果的分析，依据跟踪与监控策略采取有效的行动，使项目组能在既定的时间、费用、质量要求等情况下完成项目。 ⑥SAM：供应商协议管理。旨在对以正式协定的形式从项目之外的供方采办的产品和服务实施管理。 ⑦IPM：集成项目管理。根据从组织标准过程剪裁而来的集成的、定义的过程对项目和利益相关者的介入进行管理。 ⑧RSKM：风险管理。识别潜在的问题，以便策划应对风险的活动和必要时在整个项目生存周期中实施这些活动，缓解不利的影响，实现目标。

【工程管理】： ⑨RD：需求开发。需求开发的目的在于定义系统的边界和功能、非功能需求，以便涉众(客户、最终用户)和项目组对所开发的内容达成一致。 (10)REQM需求管理。需求管理的目的是在客户和软件项目之间就需要满足的需求建立和 维护一致的约定。 (11)TS：技术解决方案。在开发、设计和实现满足需求的解决方案。解决方案的设计和实现等都围绕产品、产品组件和与过程有关的产品。 (12)PI：产品集成。从产品部件组装产品，确保集成产品功能正确并交付产品。 (13)VAL：确认。确认证明产品或产品部件在实际应用下满足应用要求。 (14)VER：验证。验证确保选定的工作产品满足需求规格。

【支持管理】： (15)CM：配置管理。建立和维护在项目的整个软件生存周期中软件项目产品的完整性 。 (16)PPQA：过程和产品质量保证。为项目组和管理层提供项目过程和相关工作产品的客观信息。 (17)MA：测量与分析。开发和维持度量的能力，以便支持对管理信息的需要。作为改进、了解、控制决策。 (18)DAR：决策分析与解决。应用正式的评估过程依据指标评估候选方案，在此基础上进行决策。 （2）成熟度等级分为：初始级、可管理级、已定义级、量化管理级、优化管理级。 我所在企业的成熟度等级为第三级已定义级。

**3.若过程改进中包括度量人在过程中的工作，并对过程进行彻底的变更，这样的项目是否是不人道的？对过程改进会发生哪些抵触行为？**

（1）该项目不是不人道的。

（2）流程优化如同技术革新，是管理创新的必然趋势，任何人只能去顺应这种趋势。 在过程改进中找到新的定位；就像我们不能评价技术创新不人道一样，我们不能认为过程改进不人道。过程改进：必然到来新的组织、岗位、职责调整，必然伴随着人员调整，或人员分工调整。这必然会影响既得利益者的抵触；新的分工，需要新的技能支撑，或新的职责支撑，需要付出培训成本，磨合适应的成本；需要所有系统接口部门，重新熟悉新的规则；需要建立全新的管理评价体系来支撑运转。

**4.给出 SEI 的 CMMI 不能适用的两个领域，并说明理由。**

（1）CMMI模型太重，不适合小项目及轻量级开发 CMMI的开发模型，一般以重量级为主，也有螺旋和快速原型等经典的软件工程开发模式。CMMI注重过程和文档等诸多原因，让CMMI模型比较重，比较适合外包或大规模团队的分工协作。

（2）CMMI适用于成熟市场，不适合探索性研究领域 CMMI强调流程管理，领导要求高，不能很好地适应环境变化，重视计划，重视外部力量对项目的控制，管理成本一般比较高。诸多CMMI特性让CMMI适用于已经具备稳定模式的领域，不适用于还处于探索型的研究。

**5.如何将现有的软件开发向敏捷开发方法转换？期间会遇到哪些困难，如何解决？**

如何转换：将要引入的敏捷开发方法，培训给所有的开发人员，进行知识普及并宣扬敏捷开发的好处。进行实际的试转换，看实际效果。 困难：开发人员已经习惯了原有的开发方法，不愿学习新的开发方法；公司的决策层不确定实施敏捷后能够真正提高团队的开发效率。 如何解决：引进每日站立会议，检查每个成员的进度和问题，并设法解决，打消开发人员的顾虑；实施敏捷开发后，进行数据度量统计，和以往传统开发进行对比，并和高层做沟通。

**6.分析比较 CMMI、ISO15504 和 6sigma 之间的共同点和区别。**

ISO15504与CMMI，都共同着眼于质量和过程管理，两者都为了解决同样的问题。从一方面说他们是相互联系、相互补充的。两者都吸收了现代质量管理理论，都以“过程思维”为指导。ISO15504中的质量要素都可以对应到CMMI中关键过程区域特征上，而CMMI在生产过程中的管理重点，又弥补了ISO15504在微观管理上的不足。但是它们的基础是有差异的：ISO15504确定一个质量体系的最少需求，而CMMI模型更在注重持续过程改进。而且，ISO15504只建立了一个可接受水平，而CMMI是一个具有五个水平的评估工具。所以，在建立企业标准时，可以综合考虑ISO15504和CMMI的质量管理要求，使两者都能更好的发挥各自的优势。 ISO15504和六西格玛之间无论经营观念、管理体系，还是管理决策，都不可替换，对于组织质量管理工作而言，所起的作用也是各有千秋。首先，ISO15504族标准为组织的质量管理工作提供了一个基础平台，而六西格玛管理法给组织的质量管理工作带来了一个新的、垂直的方法体系。其次，通过ISO15504认证只能证明该组织已经具备保证本组织生产或提供服务达到国际基本标准的能力，但能否长期保持，还需采用一些有效的质量管理方法，以确保组织质量得到持续改进。而六西格玛管理就是一种非常优秀的方法，可以说二者是互相补充的。 CMMI和六西格玛有许多相似之处，但也有重要的差别。首先，CMMI是一个只应用于软件过程的特殊的质量活动，而六西格玛是在整个公司上实现并用来改进所有过程的。在降低偏差、量化性能、改进过程方面，CMMI可以看作是六西格玛的一个子集。其次，由于六西格玛强烈的以客户为中心，更加强调协同工作和基于事实做决策，所以可以更好的保证处理问题的正确性。

**7.软件过程为什么必须进行改进？**

凡是活动，都存在过程；凡是过程，都存在改进；凡是改进，都没有终点。软件过程改进的必要性主要表现在以下几点： 经过一段时间后，过程的性能区域降低； 客户有越来愈高的需求； 组织的过程是逐步成熟的； 组织的目标可能是变化的； 组织的环境是不断变化的； 竞争对手在改进。

**8.软件工程中引入软件过程的作用和意义是什么？**

软件过程是软件生存期中的一系列相关软件工程活动的集合。每个软件过程是由一组工作任务、项目里程碑、软件工程产品和交付物、质量保证点等组成。 软件过程是为了获得高质量软件所需要完成的一系列任务的框架，它规定了完成各项任务的工作步骤。软件过程的优劣决定了软件质量的高低，好的过程是高效高质量的前提。软件项目总是超出预算，落后于进度表，质量不可靠。 过程要经过不成熟到成熟的发展历程。成熟的过程才能发挥应有的作用。 过程的成熟要经过三个关口：僵化、固化、优化 SPIN（software process improvement network）专家人们，SPI的第一步是SPC(软件过程创建 Creative)。此时应该降低过度的灵活性，即僵化。在这个阶段，障碍并未彻底消除。过程的创建只有借助行政的力量推进才能得以完成。 固化阶段解决的正是消除障碍的问题，对应于障碍消除ABO阶梯的B阶段（Awareness, Buy-in, Ownership 即：了解—接受—拥有）在这个基础上，改进才真正开始，也就是优化的开始。

**9.软件过程改进中如何管理变革？**

伙伴策略：依赖于个人之间的关系；利用研讨会、午餐会和活动来宣布和讨论需要改变的事情以及如何改变。

政策策略：试图通过影响官方和非官方的领导人，使强大的组织结构发生变革‘寻找和说服那些最受尊敬、有众多支持者的人。

经济策略：相信金钱具有最好的说服力；基于假设-人们的原动力的经济刺激。

对抗策略：基于假设-如果能够唤起并调动起人们对当前问题的不满和愤怒，他们会愿意改变；更多地依赖于策略家的说服力，是人们感到现存的问题，但不提倡暴力。

学术策略：假设如果你提供给人们足够的信息和正确的事实，他们会接受变革；通常产生雇主、专家和咨询的委员会的研究报告。

工程策略：假设工作性质发生变化，许多人员不得不改变；对组织结构方面问题的强调，导致对环境很敏感。

军事策略：依赖于严酷的武力或无知；时常被军队、警察、学生政治压力集团、政党所用重点是学习在斗争中使用武器；需要力量和敏捷，遵守纪律将受到奖励。

**10.软件过程改进的框架的构成是什么？每个构成部分的作用是什么？**

软件过程改进的框架包括软件过程基础设施、过程改进路线图、软件过程评估方法和软件过程改进计划。

每部分作用是：

(1)软件过程基础设施 它包含组织管理基础设施和技术基础设施，可为软件过程改进的活动提供必要的条件和支持。

(2)过程改进路线图 它应提供表明有效软件过程特征的模型，以及逐步达到有效软件过程的途径，软件组织依靠路线图的指引可以朝着有效软件过程前进。事实上，CMM（软件能力成熟度模型）和SPICE提供的成熟度等级都属于这种路线图。当然，软件组织从自身的实际情况出发对这些模型所作的裁剪版本，只要是适用的也应看成是过程改进路线图。

(3)软件过程评估方法 它是评估软件组织现行和现用的软件过程、做法和基础设施的方法和技术。评估通常要对照过程改进路线图，评估的结果要能表明，从提高过程有效性方面看哪些是强项，哪些是弱项。改进措施应能导致过程成熟度沿着改进路线图提高过程成熟度。过程评估方法可以是公开适用的标准方法，例如SEI评估方法即基于CMM估价的内部过程改进CBA—IPI(CMMfbased appraisal for internal process improvement)或BOOTSTARAP方法，也是按SPICE规定的准则进行内部评估。

(4)软件过程改进计划 评估后把发现的问题转化为软件过程改进的行动计划。这包括为改进过程基础设施以及提高其有效性必须采取的措施，过程改进应能使改进的过程规范化并提高过程的有效性。

**11.描述在软件设计过程中的主要活动以及这些活动的输出。使用一个实体--‐关系图（E--‐R 图），说明在这些活动输出之间可能存在的关系**。

主要活动：体系结构设计、接口设计、组件设计、数据库设计。

活动输出：系统体系结构、接口描述、组件描述、数据库描述。

**12.论述度量在软件过程改进中作用。**

（1）阻碍软件过程改进的主要障碍是高层经理不愿意在软件过程改进上投资。这是需要证据证明SPI能带来投资效益（ROI），度量可以提供必要性的证据。

（2）正确的、持续的进行软件的度量时，产品和过程的质量属性的数据就为实施和管理过程改进活动提供有效的基础。

（3）有效度量的作用在于能帮助软件组织认清自己的能力，并进一步为他们的生产和服务制订出可行的计划。 （4）度量还能使人们找到变化的趋势和预测出现的问题，因此可以更好地控制成本、减少风险，改进质量，确保实现商业目标。

**13.什么叫集成化过程改进？它的意义是什么？**

在软件过程的各种现代版本中，为适应各组织的学科，创建了不同的过程改进模型，开发了多种语言。这种多样性对于沟通问题产生不利影响。而集成化过程改进就是用来改变这种情况；通过提供一种单一的语言，使多种学科能够共享过程改进活动并关注一个统一的过程改进目标。

意义：

1) 成本效益被理解，过程改进所获得的成本节省非常可观。相对于采用多个模型，一个连续改进的组织如果采用了公共模型，就可以减少以下费用：采用模型和评估方法所需的培训费用；在相同组织或人员执行各种评估需要的费用；在数据仓库中维护冗余数据的过程资产；维护或采用多种模型的专业知识。

2) 重点明确，集成化过程改进计划可以弄清楚各种活动的目的和商业目标。通过大范围的各种过程改进活动的集成，更容易把实践人员和主管的队伍团结在过程改进的目标下。有改进重点，能统一和加强思想，高效的安排和使用匮乏的资源，并为跨越不同学科的过程改进提供一种共同语言。特别是一个其有公共术语和公共评估方法的单一模型提供这类重点。

3) 过程集成，过程集成和精益组织。集成过程改进的一个不太明显的收益是它对组织产生的“集成”影响。当过程的定义跨越了组织和学科的边界时，通常会产生新的理解和相互学习，从而使关键工作统筹化，并消除了冗余的不必要的活动。“烟囱式”的过程改进通常假定组织的接口是有效的。在跨部门进行改进过程时，该组织可另外得到过程重构的效果。这种简化持精益概念，即努力消除产品中的浪费，为客户提供。

4)灵活性，集成所带来的最后一个效益是适应和利用业务式工程环境变化的能力。

**14.制定软件过程改进计划的流程是什么？解释其中的主要活动的作用和目的？**

(1)过程分析 考察和理解现有的过程，在一些情况下需要对过程的某些环节进行度量和定量分析，利用取得的数据来表明过程的状况。同时，这些数据可以用来与过程改进后的状况进行对比。

(2)确定改进 利用过程分析的结果，找出原有过程中质量、进度和成本的瓶颈。针对发现的问题，制定过程改进方案，提出需要采用什么规程、方法和工具的建议。

(3)过程变更 实施过程变更，把新的规程、方法和工具安置于合适的过程环节上，并且与其他的软件过程活动集成起来。

(4)培训 没有培训的过程变更在大多数场合注定要失败。有的单位在培训工作不够充分的情况下，强制推行过程变更，这样做不会收到好的效果。

(5)调整过程变更 在初步实施过程变更后，不可能立即收到圆满的效果，在过程修改后还可能会出现一些小的问题，这就需要进行适当的调整。此外，同时引入太多的变更是不切实际的。

**15.简述 CMMI--‐DEV V1.3 中每一成熟度等级所包含的过程域。**

成熟度等级2（7个）：需求管理、项目计划、项目监督和控制、供应合同管理、度量和分析、过程和产品质量管理、配置管理

成熟度等级3（11个）：需求开发、技术解决方案、产品集成、验证、确认、组织级过程焦点、组织级过程定义、组织级培训、集成化项目管理、风险管理、决策分析和解决方案

成熟度等级4（2个）：组织级过程性能、项目定量管理

成熟度等级5（2个）：组织级改革和实施、因果分析和解决方案

**三、论述题**

**1.论述计划驱动开发方法与敏捷开发的方法优缺点。**

答：计划驱动开发起源于系统工程和质量规范，建立系统工程的原则，协调大量需要精确协同工作的组件。通过从需求到已完成的代码等一系列代表用来推动软件开发的过程，计划驱动开发非常精确低依赖于明确的步骤。计划驱动开发的关键是过程的定义和管理，和过程改进联系在一起，强势在于标准化所带来的可比较性和可重复性。过程需要进行定义、标准化需要逐步改进以提供控制。管理其操作所需的数据。 优点：适应大型产品和团队；适应应对高安全性的产品；项目初期需要高素质人员和专家，项目平稳进行期间，对人员的素质要求降低；清晰的政策和规程定义了人们的角色，使人们感到舒适，有权利。靠秩序繁荣。 缺点：详细设计和庞大的预先设计非常适用于高度稳定的环境，对于高度动态的环境导致返工。 敏捷开发的方法有Scrum、XP极限编程、ASD自适应软件开发、FDD特征驱动软件开发、Pair programming、pragmatic programming 优点：适应小型产品和团队；整个过程都需要高素质人员和专家参与；更高的自由度，使人们感到舒适，有权利。靠混沌繁荣。 缺点：没有经过安全关键性产品的考验；简单的设计和缺乏文档有潜在问题；简单设计和持续重构非常适用于高度动态的环境，但对于高度稳定的环境会导致潜在的返工。

**2.在什么情况下产品质量可能决定于开发团队的质量？举例说明什么类型的软件产品特别依赖于个人的天赋和能力。**

（1）使用敏捷开发的情况下，开发团队质量决定产品质量。因为敏捷开发始于简单的设计，并缺乏全面的文档和计划，整个过程都需要高质量的团队，才能保证高质量的产品。

（2）高度动态环境下的产品开发或探究性的产品的开发，对人的能力和天赋依赖度高。例如金融类产品，随着监管环境变化要快速响应。例如区块链产品，创新性探究，需要人的能力和天赋。

**3.ISO 9001:2008 标准中的 PDCA 循环，又叫戴明环，是美国质量管理专家休哈特博士首先提出的，由戴明采纳、宣传，获得普及，从而也被称为 “戴明环” 。它是全面质量管理所应遵循的科学程序。论述它在评估软件项目质量管理中的作用和意义。**

戴明环为企业实施质量改进提供了一个周期模型，这个模型影响着当前大部分质量与过程改进方法。作为一个通用模型，软件企业同样可以采用它作为质量过程改进的方法，通过对问题的定义，确定改进机会，建立改进计划和目标，根据问题产生的原因实施改进，在改进实施中收集数据进行分析评价，从而进行持续的改进。通过采用戴明环方法，软件企业可以指定自己的质量过程改进路径。

**4.结合 CMMI 的实施，论述软件过程改进过程中主要阶段的作用。**

(1) 远景阶段：该阶段中设置与前进的方向，拼出了前进的道路。它考虑到了未来的发展和在技术、经济、政策、市场和商业方面的变化。

(2) 策略阶段：策略描绘出达到的远景的途径和达到远景目标必须经过的实践步骤。

(3) 调动阶段：团结一致的调动当前的人力和资源，设计并实现相应的行动计划。

(4) 推动阶段：保持推动力，监控策略的成就，强化同盟关系。

(5) 实现阶段： 实现已制定的策略，否则过程改进会成为一种幻想。软件过程改进本身也是一个改变的过程，也应该经历上面提到的类似阶段

**5.复用的关键障碍之一是使软件工程师考虑利用现有的构件，而不是重新开发新构件，请建议 3 到 4 种软件组织可以用来激励软件工程师进行复用的方式。为了支持复用，应该采用什么技术？**

（1）激励方法：

　　①提供便捷的针对可复用组件的审核、发布、下载流程和工具；

　　②用奖励方式鼓励编写可复用组件；

　　③开发通用框架，指定专职高级研发定期对框架进行维护，将项目/产品中可复用的代码进行封装，添加到通用框架中。

（2）为了支持复用，采用技术：

　　①模块化：模块化用来分割，组织和打包软件。每个模块完成一个特定的子功能，所有的模块按某种方法组装起来，成为一个整体，完成整个系统所要求的功能。在系统的结构中，模块是可组合、分解和更换的单元。模块化是一种处理复杂系统分解成为更好的可管理模块的方式。它可以通过在不同组件设定不同的功能，把一个问题分解成多个小的独立、互相作用的组件，来处理复杂、大型的软件。

　　②组件化：组件化是一种高效的处理复杂应用系统，更好的明确功能模块作用的方式。为了解耦：把复杂系统拆分成多个组件，分离组件边界和责任，便于独立升级和维护。

　　③服务化：服务化架构是第五代移动通信系统（5G）的重要特征，结合移动核心网的网络的特点和技术发展趋势，将网络功能划分为可重用的若干个“服务”。“服务”之间使用轻量化接口通信。其目标是实现5G系统的高效化、软件化、开放化。

**6.论企业软件过程改进的实施。请围绕 “企业软件过程改进的实施” 论题，依次从以下四个方面进行论述：**

（1）叙述软件过程改进实施的主要活动。

（2）概要叙述你参与实施的企业软件过程改进项目以及你所担任的主要工作。

（3）论述该企业实施软件过程改进项目中如何根据企业的实际情况采用模型标准以及实施的主要方法和步骤。

（4）具体阐述该企业在实施软件过程改进的活动中所发现并解决的主要问题和效果。

**7.在当今 “3C” 的环境下，持续的改进是企业生存发展的永恒主题，其运用的工具不是单一的。某企业拟针对“某项服务顾客投诉率高”进行改进，在不同的阶段可采用哪些工具。**

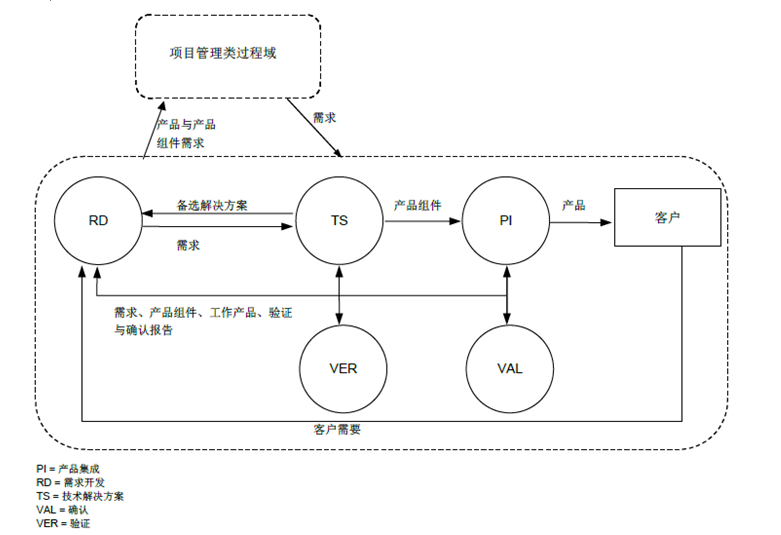
（1）过程流程工具在各个阶段划分活动，然后划分可操作的检查项，通过清晰的记录在实施软件过程中发生的所有活动及活动执行者严格管理软件过程。

（2）过程文档工具为文档提供编写模板，可以依据文档模板进行裁剪，定制每个项目的过程文档。使软件企业内部的各个项目质检能遵照一个标准化的文档编写流程并对文档进行管理。

（3）评审工具提供标准的评审规程，项目可对之裁剪，形成本项目的评审规程，软件人员可使用它来进行评审并记录评审情况。

（4）人员管理工具对实施过程的人员进行管理，人员执行某项活动后记录人员的相关信息，明确各个阶段和活动的执行人和责任人，便于责任的分析和处理。

**8.根据下图，分析说明 CMMI--‐DEV V1.3 中，五个工程类过程域之间的互动关系。**



（1）“需求开发”过程域提供需求至“技术解决方案”过程域，在此处需求被转换为产品架构、产品组件设计与产品组件。需求也被提供给“产品集成”过程域，在此处产品组件被组合起来，接口得到验证，以确保其符合“需求开发”过程域所提供的接口需求。

（2）“技术解决方案”过程域开发产品组件的技术数据包，用于“产品集成”过程域或“供方协议管理”过程域。备选解决方案被考察，并基于已建立的准则选择最优设计。这些准则可以因产品不同而有显著区别，这取决于产品类型、操作环境、性能需求、支持需求以及成本或交付进度。选择最终解决方案的工作会用到“决策分析与解决”过程域中的特定实践。 “技术解决方案”过程域依赖于“验证”过程域中的特定实践，以在设计过程中并在最后的构建之前，执行设计验证与同级评审。

（3）“验证”过程域确保选定的工作产品满足规定的需求。“验证”过程域选择工作产品与验证方法，用于对照规定的需求对工作产品进行验证。一般来说验证是一个增量式的过程，从产品组件的验证开始，通常以对完全组装了的产品进行验证为结束。

（4）“确认”过程域的范围包括对产品、产品组件、选定的中间工作产品与过程的确认。这些已确认的要素常常需要再次验证与再次确认。确认中发现的问题往往在“需求开发”过程域或“技术解决方案”过程域中得到解决。

（5）“产品集成”过程域在实施产品集成过程时使用了“确认”过程域与“验证”过程域中的特定实践。“验证”过程域的实践在产品集成之前验证了产品组件的接口与接口需求。接口验证是集成过程中的必要事件。在操作环境中进行产品集成时，就要使用到“确认”过程域中的特定实践。