

第1章 概论

第1章 概论

- 1.1 软件企业
- 1.2 软件项目
- 1.3 软件过程
- 1.3 软件过程管理
- 1.4 软件过程模型
- 1.5 CMM/CMMI模型

市场线：市场策划、商务、产品销售、项目承接、跟进、收款

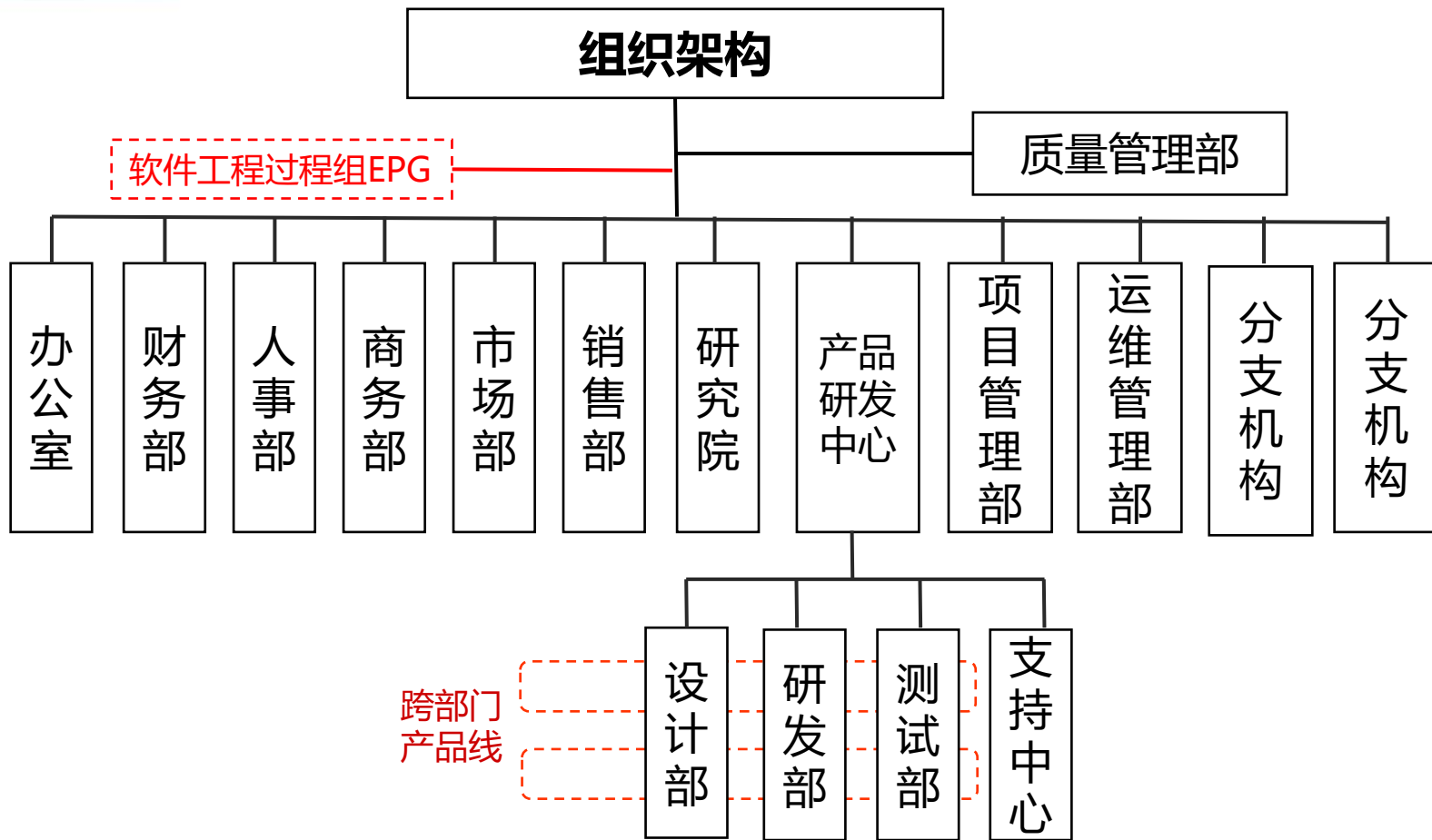
产品线：产品规划、设计、开发、测试、售前支持、售后支持

项目线：项目管理、项目实施、应用成效、项目验收、转运维

运维线：运维管理、运行保障、服务提升、应用推进、效能评价

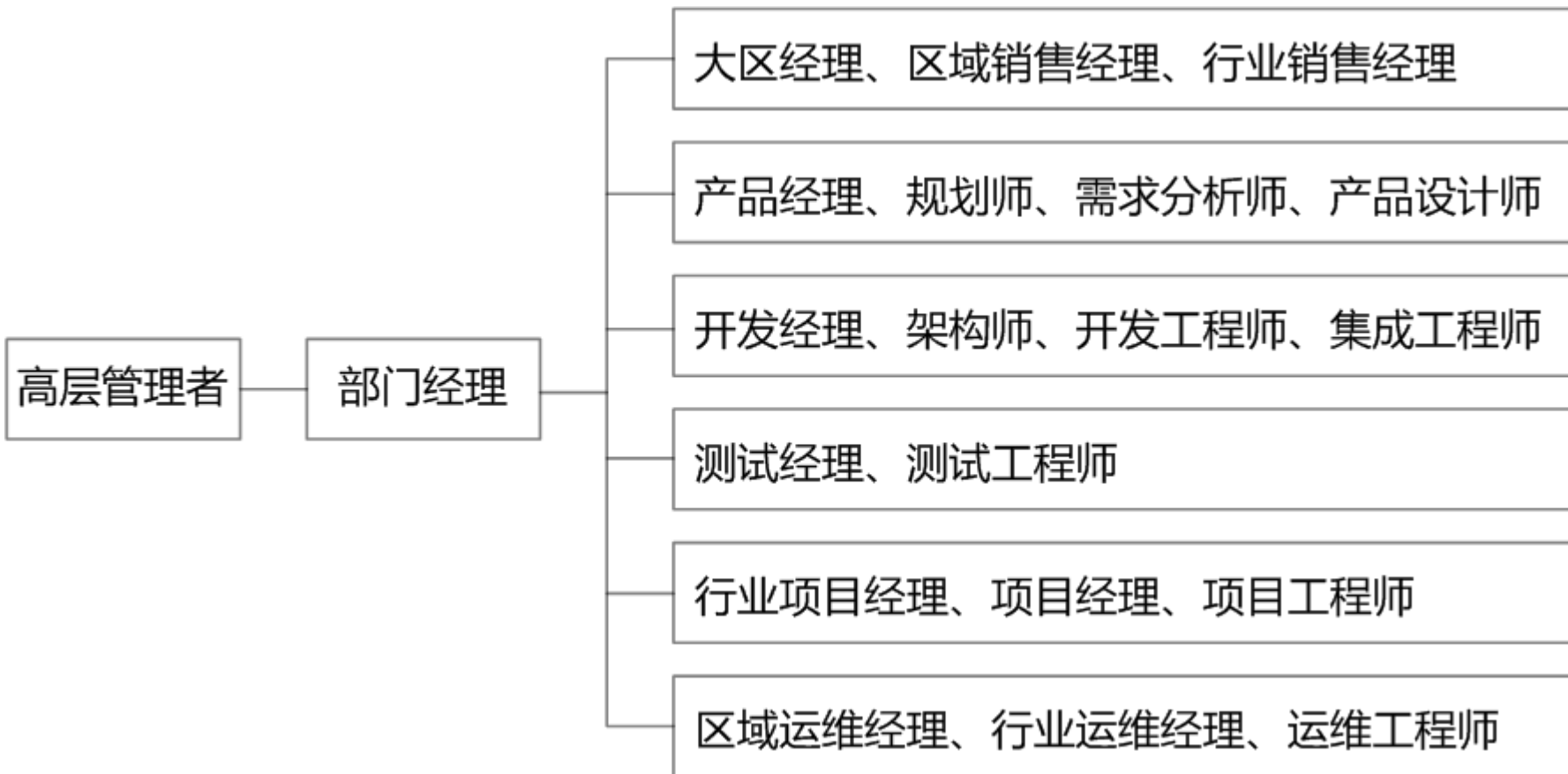
1.1 软件企业

— 组织架构



1.1 软件企业

— 组织岗位设置



领军人才： 深度研究领域发展方向、规划、方案

复合人才： 组织战略、业务、技术、管理、客户关系

担当人才： 有思路、有想法、执行力、快速打开局面

深耕人才： 主动融于企业，撑一片天，必有发展空间

1.1 软件企业

软件企业的生存法则

- **软件项目**是软件企业生存发展的唯一载体，软件项目实现软件企业和客户的共同价值；
- **软件产品**是软件企业的生命，高质量、高效率、低成本开发软件产品是企业追求的目标；
- **软件开发过程**是软件企业最复杂、最重要的业务流程
- 全面开展**软件项目**和**软件过程管理**是软件企业根基。

第1章 概论



1.1 软件企业



1.2 软件项目



1.3 软件过程



1.3 软件过程管理



1.4 软件过程模型



1.5 CMM/CMMI模型

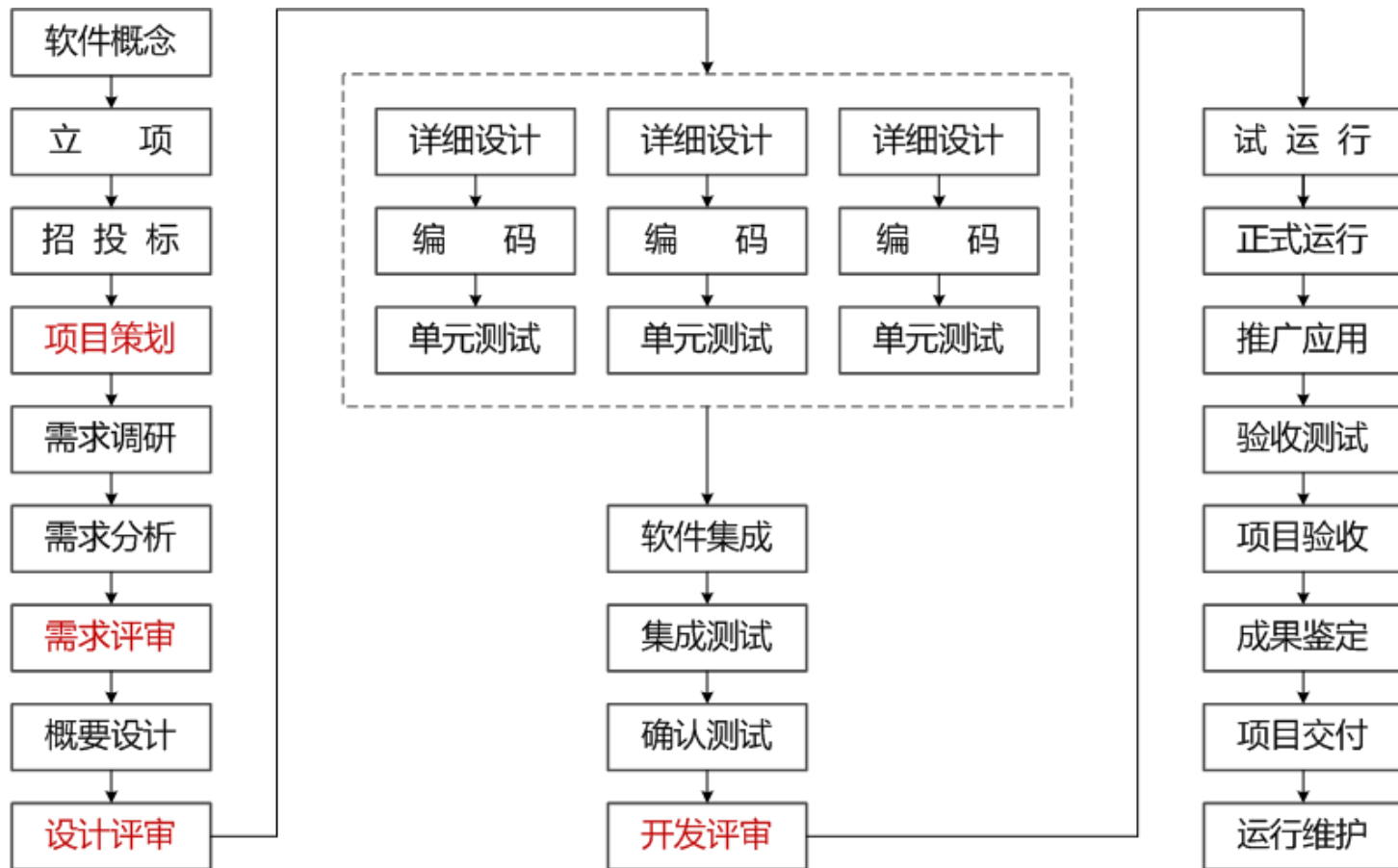
软件项目，也称为软件工程项目，它所涉及的范围覆盖了整个**软件工程过程**和**软件生命周期全过程**，包括立项、招投标、合同、启动、调研、需求分析、设计、开发、测试、试运行、正式运行、验收、运行维护等阶段。

软件项目可分为**开发项目**、**项目成果产品化**、**自主立项**、**产品项目**、**服务项目**、**研究项目**等类型。

1.2 软件项目

—— 什么是软件项目

开发项目



1.2 软件项目

—— 什么是软件项目

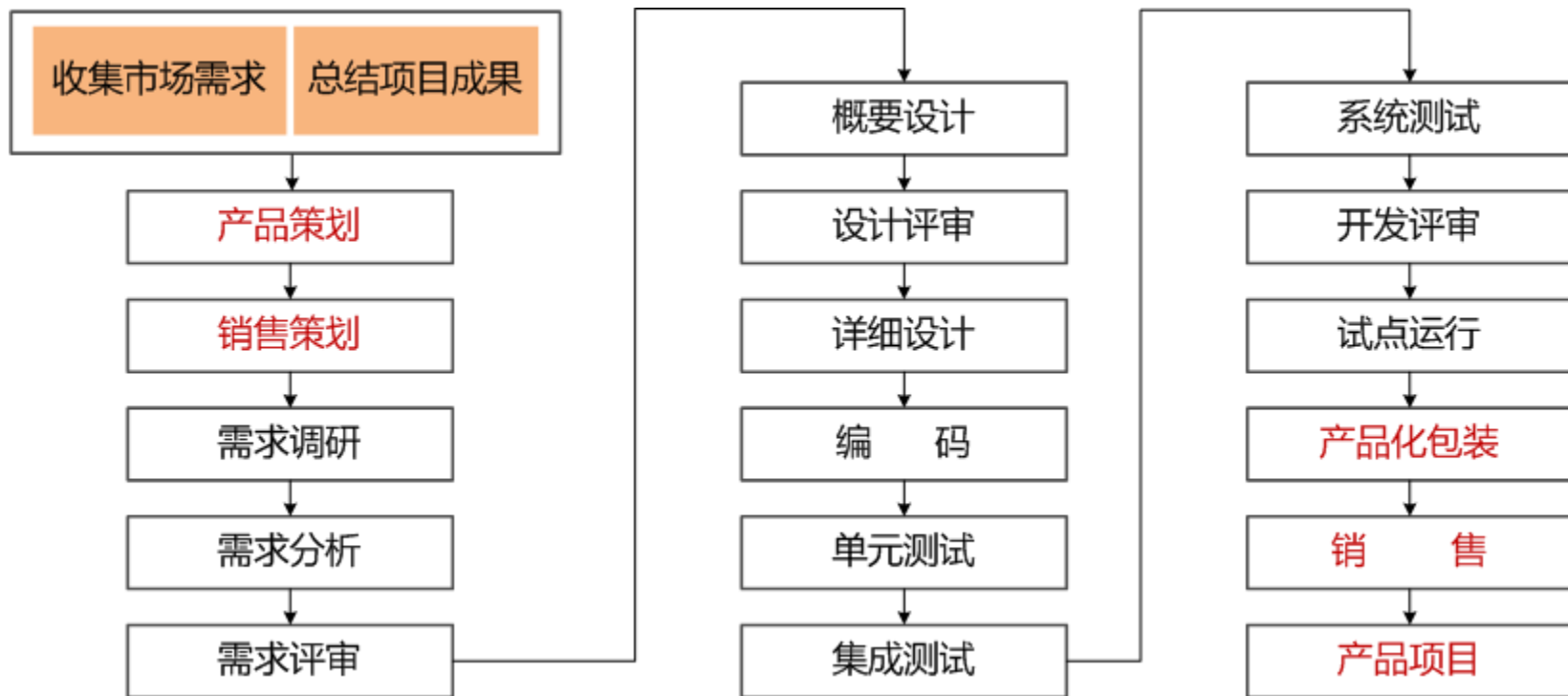
开发项目的主要阶段和产出物

软件概念	项目建议书
立 项	可研报告、效益分析
招 投 标	招标书、投标书、执行合同
项目策划	项目章程、项目实施方案、项目计划书
需求调研	需求报告
需求分析	需求规格说明书 (业务、流程、功能、数据、接口、性能等)
概要设计	概要设计规格说明书 (业务架构、技术架构、功能界面、数据结构、接口规范、集成方案等)
详细设计	详细设计规格说明书 (输入输出、处理逻辑、关键算法等)
测 试	测试方案、测试用例、缺陷记录与分析、测试报告
各类评审	会议纪要、不符合项跟踪表
试 运 行	试运行总结报告 (是否具备正式运行条件)
验收鉴定	验收材料、验收意见 (完成合同规定任务)、鉴定意见 (成果水平)、会议纪要

1.2 软件项目

—— 什么是软件项目

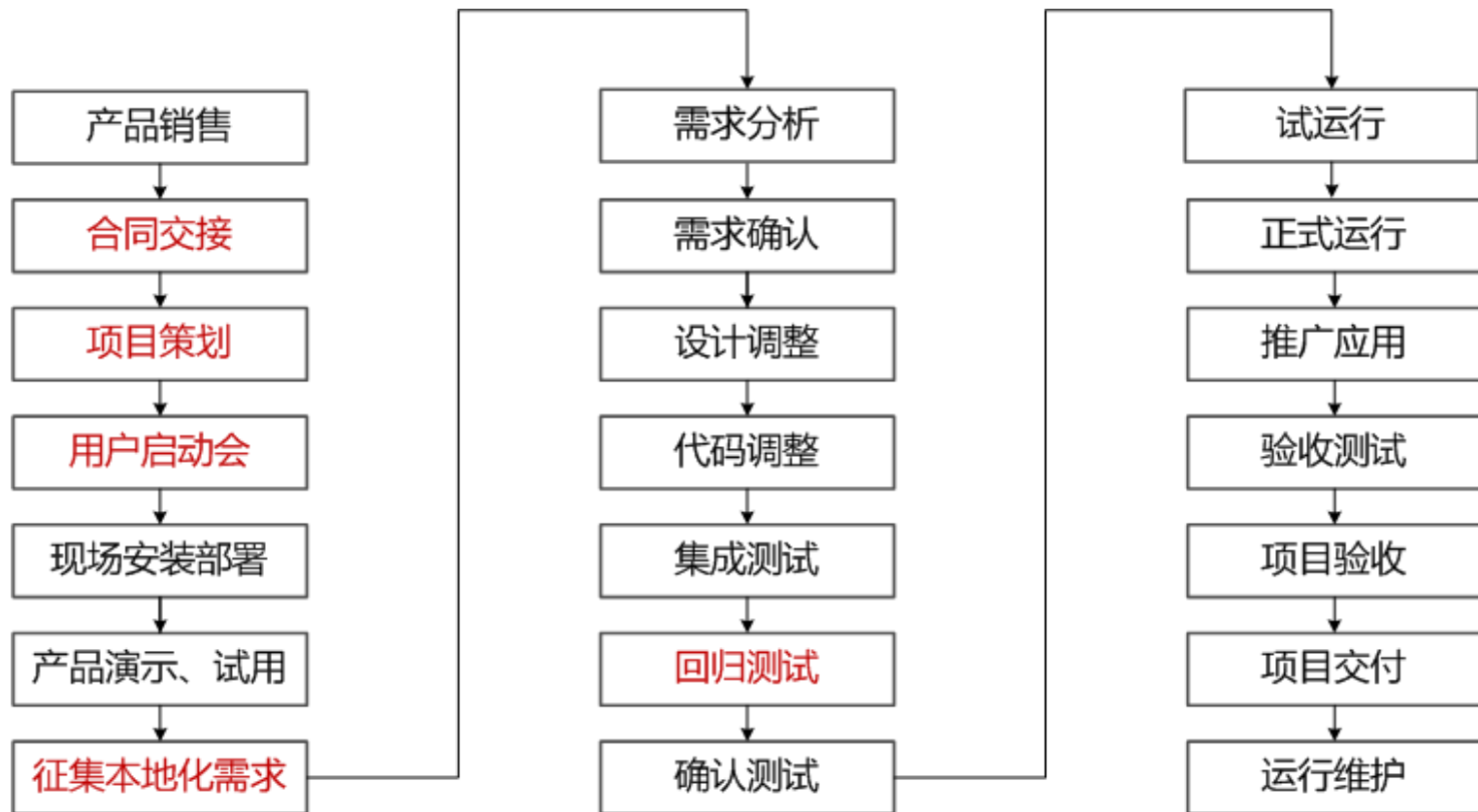
自主立项、项目成果产品化



1.2 软件项目

—— 什么是软件项目

产品项目



运维服务项目

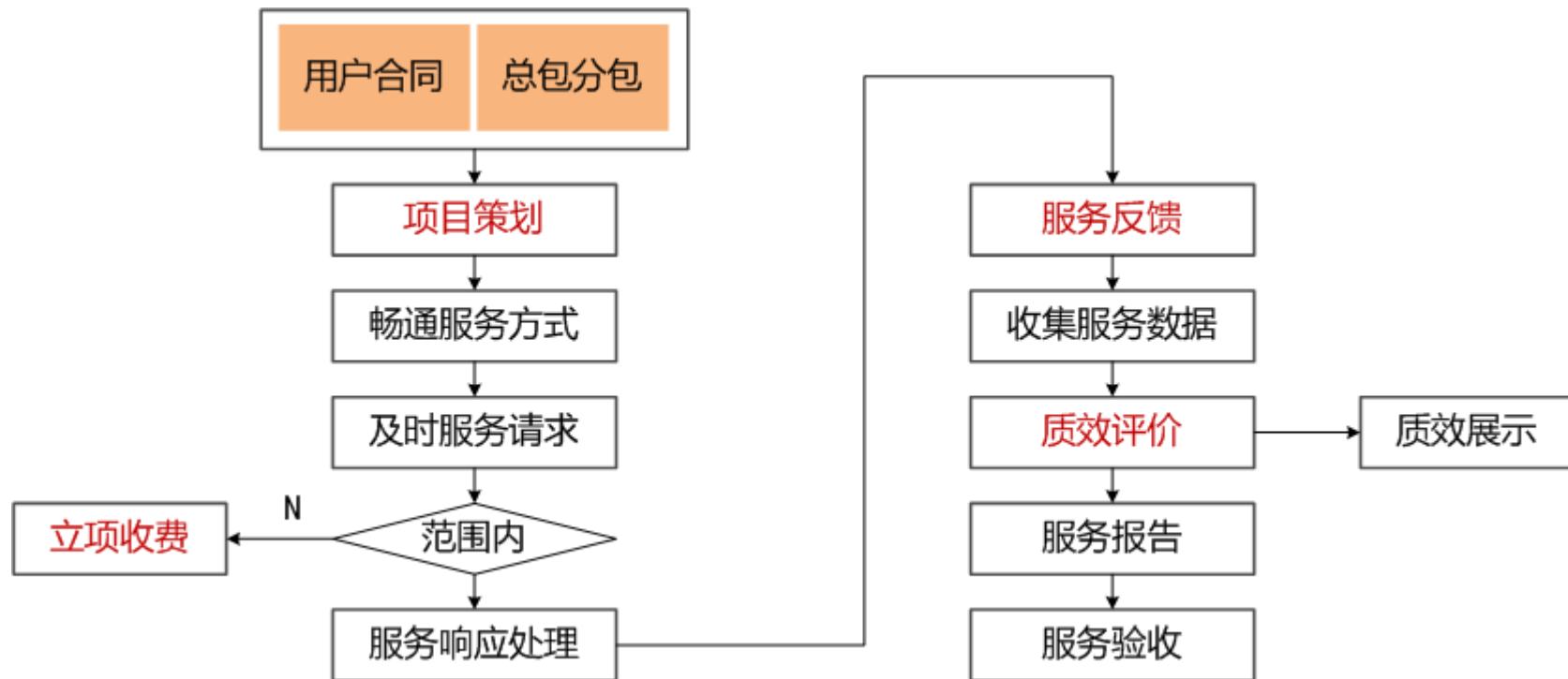
软件服务，通常有**免费质保期服务**、**质保期后有偿服务**、**驻场服务**、**一次性软件服务**等分类。

软件质保服务，主要包括解决软件运行过程中的bug、维护性修改、小规模의适应性修改和功能性能优化。较大规模的适应性修改或新需求，应通过升级、重构、新立项等项目方式实现。

1.2 软件项目

—— 什么是软件项目

质效型运维服务项目



1.2 软件项目

- 软件产品是一种“逻辑”产品。属于无形产品，难以通过望闻问切而直观感受，难以准确评价。
- 需求难以准确把握、且频繁变更。由于用户的成熟度或责任心的原因，用户一开始无法给出明确准确的需求。在开发和使用过程中，需求可能会经常变更，因此需要经常地修改程序与文档。

1.2 软件项目

- **难以在早期发现问题。** 需求不明确，加上后期修改可能没有进行全局性的考虑，产生的问题难以从早期的文档中直观地发现。
- **项目成员对文档的重视不够。** 符合用户需求的高质量软件需要依赖于大量准确规范的文档工作，但项目组成员对他并不感兴趣，很少愿意认真去做，直接影响了软件的质量。

- **劳动密集型 + 智力密集型。**软件开发过程需要大量高强度的脑力劳动，非常细致、高度复杂、使得软件的正确性难以保证。
- **技术发展快，经验失效快。**计算机技术日新月异，在应用架构、技术体系、实现路径上的选型，直接影响软件质量和系统稳定性。

- **软件项目过程复杂、可见性和可度量性差。** 软件项目开发、实施、维护过程比较复杂，过程可见性和可度量性差，进度、质量、成本、风险难以把控。
- **难于全面测试。** 更多的着眼于功能测试，而在覆盖、性能、压力、对接、三方嵌入、稳定性、可靠性等方面的测试难度较大。

- **实施和维护成本较高。** 强调在软件需求分析设计时，注重易实施、易维护的设计，而实际上由于需求、开发、测试等环节的限制因素，造成软件反复修改、实施和维护成本增高。
- **个人素养、团队能力和企业能力。** 项目成功取决于成员个人素养和职业精神、项目团队协作能力、以及企业的软件过程能力。

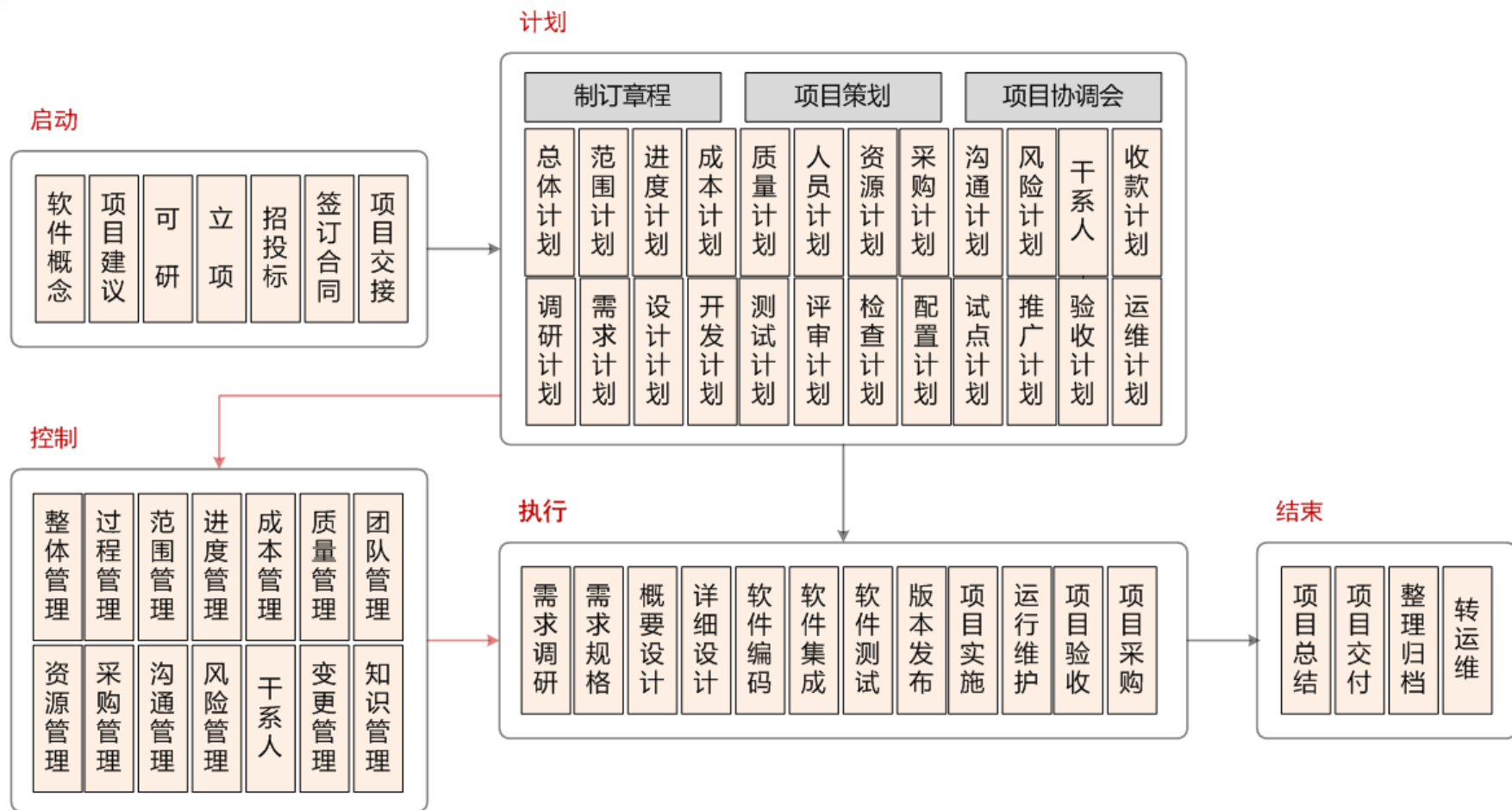
软件项目管理的提出是在20世纪70年代中期的美国，当时美国国防部专门研究了软件开发不能按时提交，预算超支和质量达不到用户要求的原因，结果发现70%的项目是因为管理不善引起的，而非技术原因。

软件项目管理是为了使软件项目能按照预定成本、进度、质量顺利完成，对人员、产品、过程和项目进行分析和管理的活动。

软件项目管理的对象是软件工程项目，涉及范围覆盖了整个软件工程过程。为使软件项目开发获得成功，关键问题是必须对软件项目的**工作范围、可能风险、需要资源（人、硬件/软件）、要实现的任务、经历的里程碑、花费工作量（成本）、进度安排**等做到心中有数。这种管理在技术工作开始之前就应开始，软件从概念到实现过程中继续进行，当项目结束时才终止。

1.2 软件项目

— 软件项目管理框架



第1章 概论



1.1 软件企业



1.2 软件项目



1.3 软件过程



1.4 软件过程管理



1.5 软件过程模型



1.6 CMM/CMMI模型

过程就是做事情的一种固有的流程和方式。

做任何事情都有过程存在，小到日常生活中的琐事，大到工程项目，软件项目，总是按过程一步一步往下**有序推进**的。

对于同样的一件事，有经验的人对完成这件事的过程会很了解，他会知道完成这件事需要经历几个步骤，每个步骤都完成什么事，需要什么样的资源、什么样的技术等等，因而可以顺利地完成工作。没有经验的人对过程不了解，往往有**无从下手**的感觉。

过程就是做事情的一种固有的流程和方式。

做任何事情都有过程存在，小到日常生活中的琐事，大到工程项目，软件项目，总是按过程一步一步往下**有序推进**的。

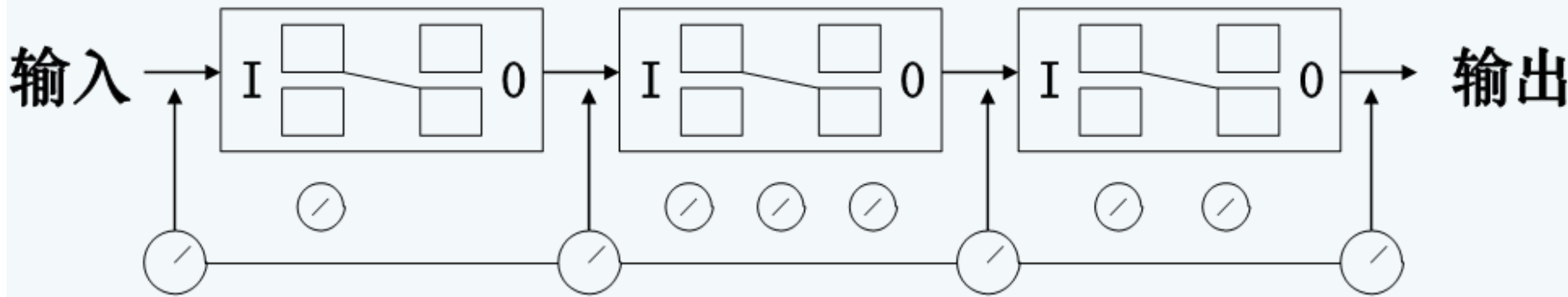
对于同样的一件事，有经验的人对完成这件事的过程会很了解，他会知道完成这件事需要经历几个步骤，每个步骤都完成什么事，需要什么样的资源、什么样的技术等等，因而可以顺利地完成工作。没有经验的人对过程不了解，往往有**无从下手**的感觉。

1.3 软件过程

——什么是过程

过程 (Process)：指一系列活动、任务、和它们之间的关系，它们共同把一组输入转换成所需要的输出。

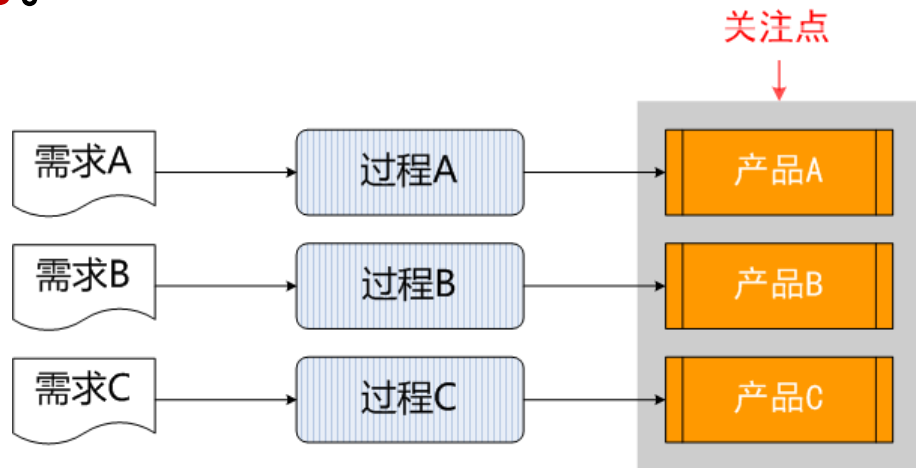
——ISO/IEC 12207; GB/T 8566



1.3 软件过程

- 过程由子过程、活动构成，每个子过程有不同的过程模型。
- 过程、子过程、活动是有序组织、有序演进的，其前序、后序、关联、紧前、紧后等关系是明确的。
- 过程、子过程、活动有明确的方法和规程、技术和工具、基础设施和环境、人力资源和技能、限制约束假设条件、入口准则、出口准则、产出物及检验标准、度量指标和绩效指数等要求。

将关注点只放在任务的最终结果上，而不关注任务的完成过程、不关心过程质量，那么不同团队或个人可能会采用不同的过程或方式完成任务，其工作效率、结果质量也可能不同，**完全**依赖个人的素质和能力。

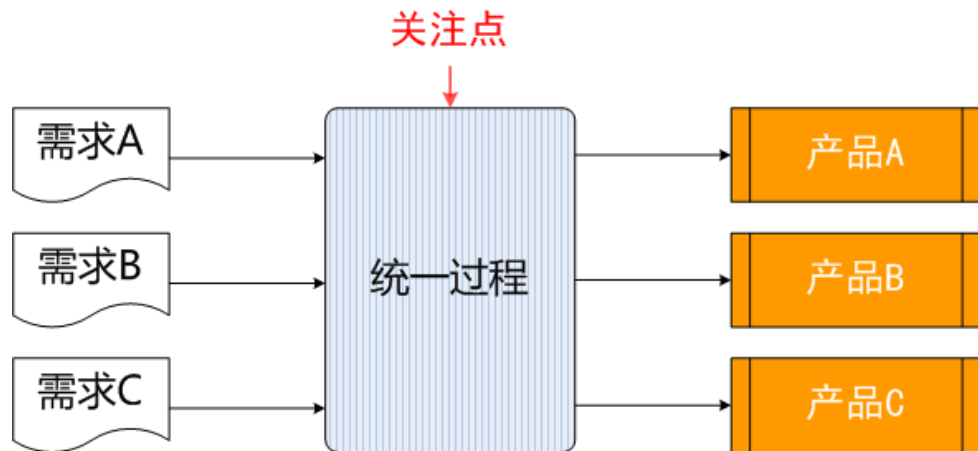


1.3 软件过程

— 过程思维

将关注点放在任务的完成过程上，采用统一过程，注重控制、测量和改进过程，即过程透明化。最终结果的一致性、质量会更好，依赖的是**企业过程能力**和统一行为，而不是个人能力。

产品的质量依赖于
企业的过程能力，而不
依赖于个人能力！



1.3 软件过程

——任务思维与过程思维

	面向任务的思维	面向过程的思维
注重	任务、人员、最终结果	总体目标、过程、各部分间的协调性、结果的一致性
特点	将工作任务分解，指派人员分头去完成	消除各部分间的冲突、提高总体效率、达到总体目标
过程规范	临时确定过程和活动	过程规范已成为行为准则，得到自觉贯彻实施
组织	职能与角色未必与过程需求相适应	职能和角色的确定均以支持过程为原则
管理	工作人员的表现用投入的工时、简单的任务完成量来衡量，与他们工作的过程成效无关。 管理人员经常救火式地处理问题和危机。	工作人员的表现以过程绩效和成果来衡量，衡量方法是明确的、一致的、大家都认可的。 管理人员关注过程质量和结果质量。
技能培训	培训是临时特定安排的，含有人员的个人因素	培训工作是计划之内的，目标是支持全过程需要的
工具技术	工具和技术是作为特定点问题的解决而获得，没有整体计划和策略	工具和技术均为支持全过程而选定的，并考虑过程活动的自动化

软件过程 (Software Process, SP) 是人们建立、维护和演化软件产品整个过程中所有技术活动和管理活动的集合, 是人们用来开发和维护软件及其相关产品的活动、方法、实践和改进的集合。其中相关产品是指项目计划、设计文档、代码、测试文档和用户手册等。

当一个组织逐步走向成熟时, 软件过程的定义也会日趋完善, 其内部的过程实施将更具一致性。

SEI-CMM将过程定义为：为实现预定目的而执行的一组实践；过程包含工具，方法，材料及人员。

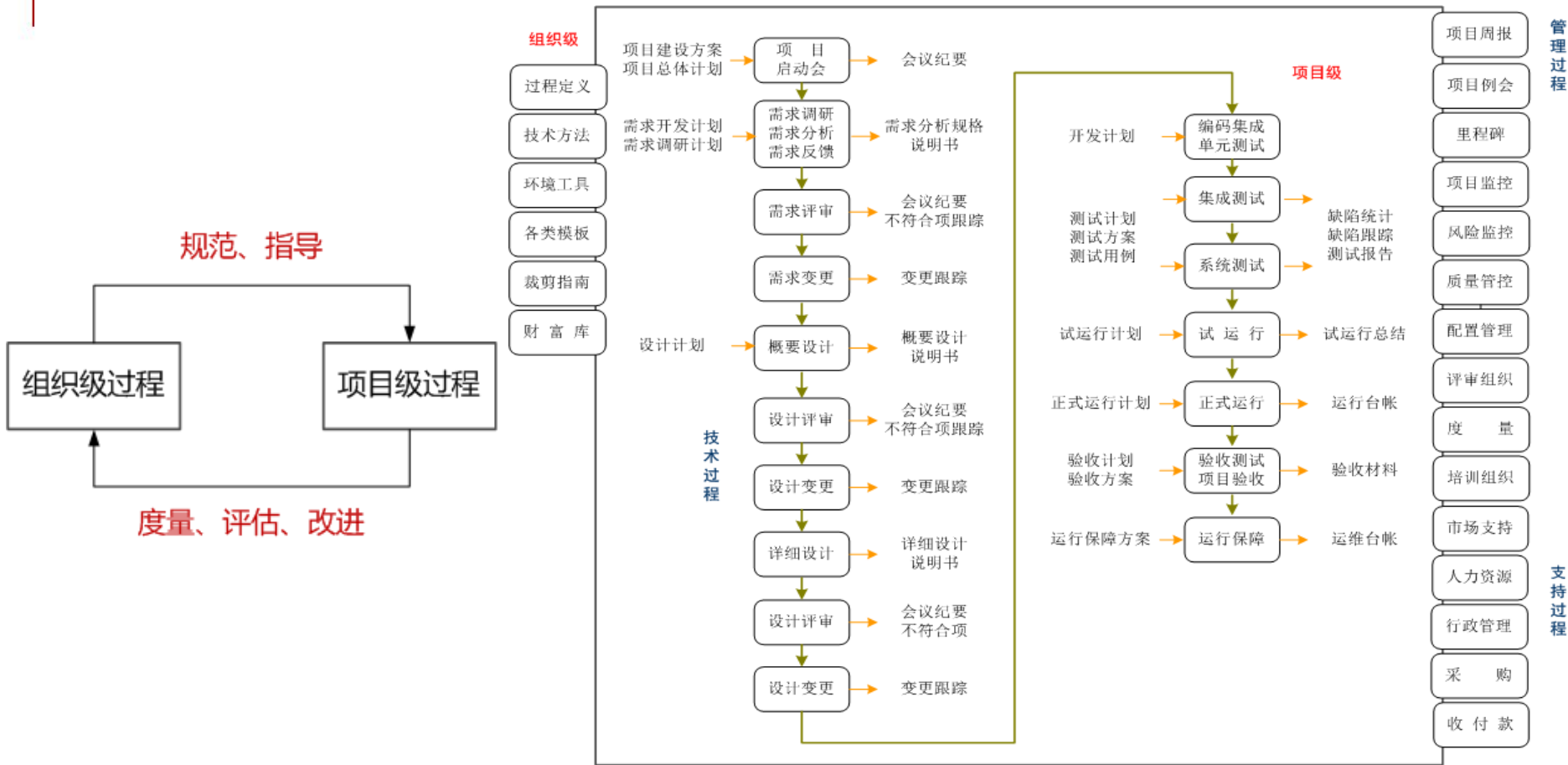
- 规定了实现预定目的或成果所要完成的一系列活动/任务-
做什么；
- 执行活动相关的角色和职责- 谁来做；
- 完成活动所要采用的技术、方法和步骤- 如何去做

1.3 软件过程

- 工程类：是构成软件生存周期主要部分的那些过程，包括获取、供应、开发、操作、维护等过程。
- 支持类：可穿插到基本过程中提供支持的一系列过程，包括文档开发、配置管理、质量保证、验证、确认、联合评审、审计、问题解决等过程。
- 组织类：组织（企业）用来建立、实施一种基础结构、并不断改进该基础结构的过程，包括管理、基础、改进、培训等过程。

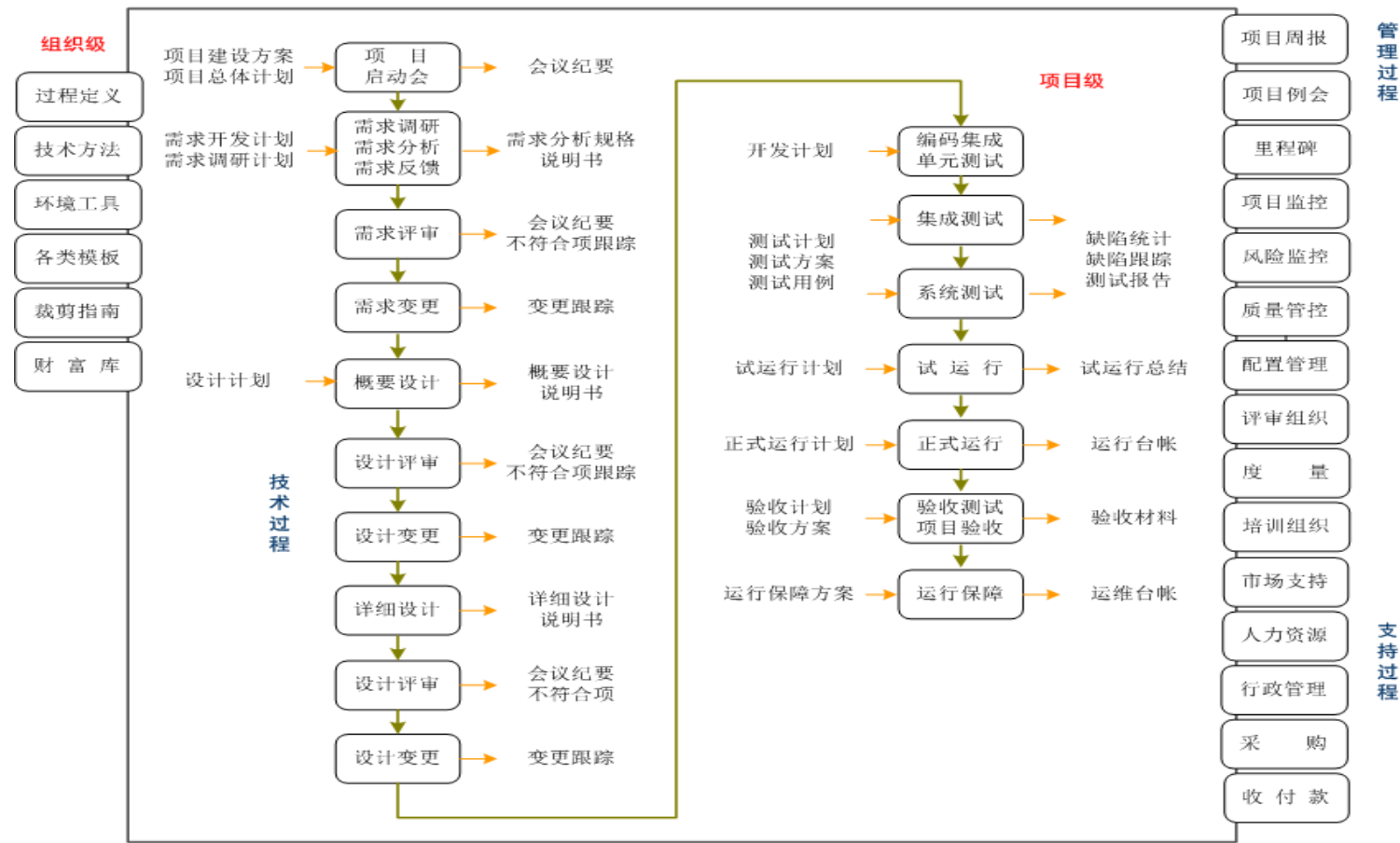
1.3 软件过程

—— 组织级与项目级过程



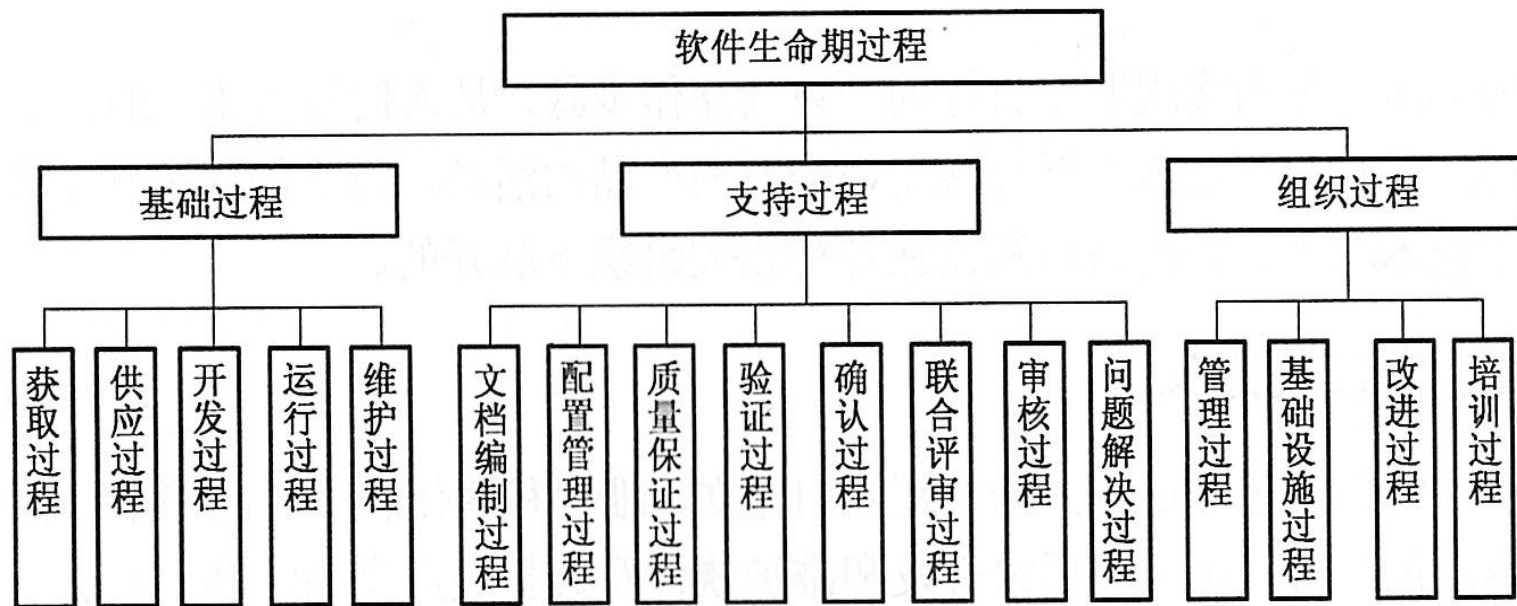
1.3 软件过程

— 技术、管理、支持过程



1.3 软件过程

— ISO/UEC12207过程标准



ISO/UEC12207 软件生命周期过程标准

1.3 软件过程

— ISO/UEC12207过程标准

基础过程	获取过程	立项、招投标、签订合同。
	供应过程	获取需求、需求确认、干系人沟通、项目周报、项目通报、阶段检查、项目验收、项目交付、项目评价、满意度调查、收款。
	开发过程	需求规格、概要设计、详细设计、编码、集成、测试、评审、确认等软件工程活动。
	运行过程	项目实施、安装部署、数据迁移、系统调试、操作培训、操作指导、运行保障。
	维护过程	请求-处理-反馈, bug和简单问题快速处理, 新增、变更和优化需求分析, 前后台服务, 数据库、软件、应用环境和接口监控, 数据质量, 移植, 主动服务, 应用推进。

1.3 软件过程

— ISO/UEC12207过程标准

支持过程	文档编制过程	各类文档的编制、评审、审核、版本控制、发布、一致性、管理与利用。
	配置管理过程	配置项标识与控制、建立和发布基线、配置审计与状态报告。
	质量保证过程	组织相关检查评审审核活动，收集质量数据并客观评价过程、工作和产出物，缺陷分析，通报不符合项并确保跟踪解决。
	验证过程	通过测试和评审，提供客观证据验证规定要求已得到满足。如单元测试验证是否复合模块详细设计、集成测试验证是否符合概要设计、系统测试验证是否符合需求规格。

1.3 软件过程

— ISO/UEC12207过程标准

支持过程	确认过程	通过测试和评审，提供客观证据对特定的预期用途和应用要求已得到满足的认定，考虑的是产品最终用途和应用要求是否可以实现，通常采用实际或模拟的使用环境。
	联合评审过程	强调用户、相关干系人、同行、专家共同参与的软件评审活动，如需求评审、设计评审、开发评审、运行评审。
	审核过程	技术、管理、支持等各类过程的规程、计划、实施、产出物的审核、决策和批准。
	问题解决过程	分析问题的根本原因、论证对策和解决方案、确保问题得到解决。

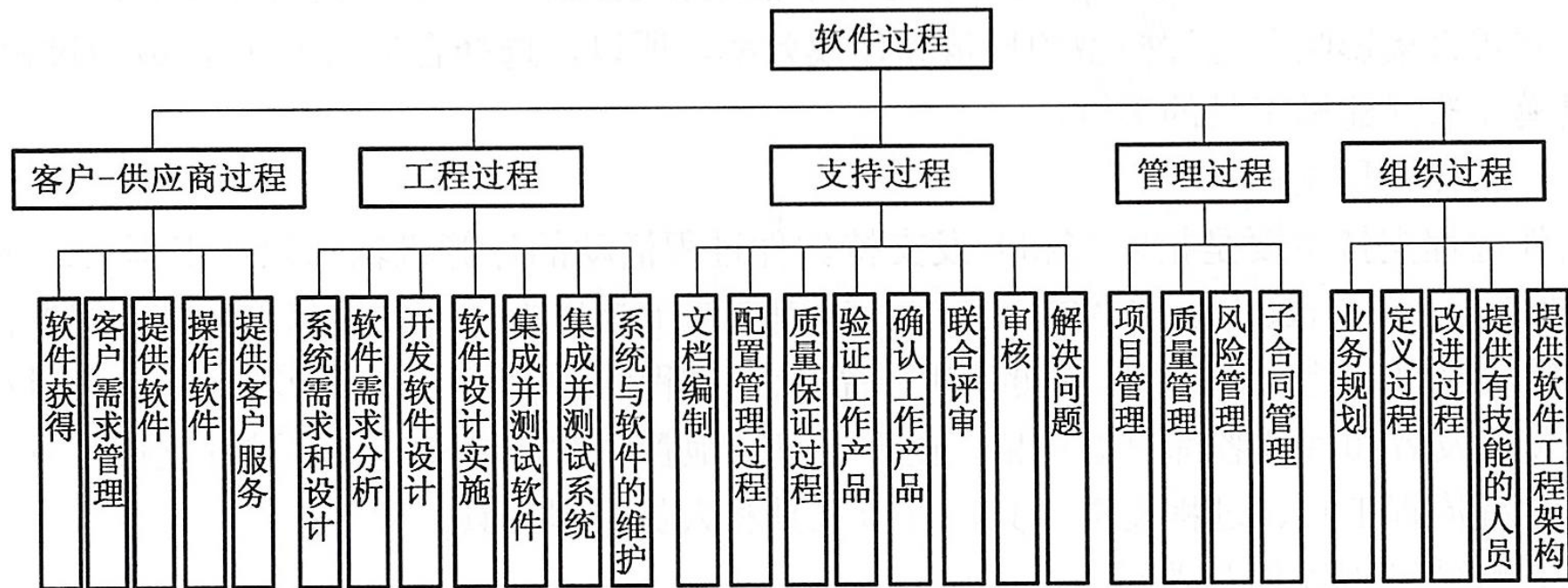
1.3 软件过程

— ISO/UEC12207过程标准

组织过程	管理过程	策划、计划、监督与控制、沟通、度量、绩效评价、信息通报等过程。
	基础设施建立过程	硬件、软件、工具、技术、方法、规程、标准等基础设施资源的建立、维护和评价。
	改进过程	过程度量、评价过程绩效、提出过程改进目标、实施过程改进活动； 总结技术、管理、支持等项目和过程方面的好经验、好做法，上升为组织级标准过程。
	培训过程	针对组织级过程、项目级过程的需要，开展组织级、项目级的规程、方法、技术、技能和管理等方面的培训，提高全员素质和技能，充分开发人力资源。

1.3 软件过程

— ISO/IEC15504 评估标准



软件过程的主要研究方向之一：软件过程分析和建模。

软件过程建模方法是软件过程技术的起点，其中形式化和半形式化建模方法有基于规则，或基于过程程序的。

软件过程分析和建模对于保证过程定义的质量、建立全面而灵活的过程体系具有重要作用。

使用过程建模语言（Process Modeling Language,PML)描述和定义过程，建立过程模型。

软件过程的主要研究方向之二：软件过程支持。

主要是指研究和开发支持软件过程活动的计算机辅助软件过程工具，以便很好地支持、管理并规范化软件过程，使得软件过程透明度更好，为项目的软件过程提供指导，使开发者和管理者都有据可依，可更加有效地管理软件过程。

软件过程支持工具主要包括软件过程流程工具、过程文档工具、评审工具、人员管理工具等。

软件过程的主要研究方向之三：软件过程评估和改进。

软件过程评估和改进是指根据某种模型对现有软件过程进行考核和评价，找出其中的不足之处，然后加以改进。改进对生产高质量软件产品和提高软件生产率的重要性已被越来越多的软件开发组织所认同。

软件能力成熟度模型CMMI除了用于软件过程评估外，还向软件组织提供了指导其进行软件过程管理和软件过程改进的框架。

第1章 概论



1.1 软件企业



1.2 软件项目



1.3 软件过程



1.4 软件过程管理



1.5 软件过程管理模型



1.6 CMM/CMMI模型

过程管理的目的是最大限度地提高软件产品的质量和软件开发过程的生产率。**过程、人、技术**是产品质量和生产率的三个要素。因此，过程管理是提高产品质量的重要环节，为此需要建立过程管理的标准，为软件企业构建合理高效的过程体系。

过程管理需要强大的管理层支持和持续努力，是“**一把手工程**”。

对于一个软件企业的过程管理来说，应善于总结项目经验及一些完善的做事过程，称之为**最佳实践(Best Practices)**。

过程管理就是对最佳实践的有效积累，形成**可重复**的过程，建立企业财富库，逐步提高其软件过程能力。

过程管理包括过程定义与过程改进。**过程定义**就是总结最佳实践，形成一套稳定的可重复的软件过程。**过程改进**就是针对过程实践中的偏差和不切合实际的地方进行优化。

1.4 软件过程管理

—— 主要管理内容

1	企业级的工程过程、管理过程、支持过程的定义和维护，包括标准、规程、规范、技术、技能、方法、工具、模板、裁剪指南、度量指标、财富库等。
2	企业级软件项目和过程的宣传、培训、推动、指导、贯彻实施。
3	企业级软件项目和过程的能力评估、改进过程。
4	软件生命周期定义、软件开发模型、技术框架、技术体系。
5	需求分析、软件设计、编码集成、软件测试等软件工程技术、方法和工具。
6	软件项目启动、策划、实施、控制、变更、扫尾的整体过程管理。
7	软件项目的范围、进度、成本、质量、资源、团队、沟通、风险、干系人、采购等要素的管理。
8	配置管理、过程和产品质量保证、度量和分析、原因分析流程、决策分析等支持过程。

第1章 概论



1.1 软件企业



1.2 软件项目



1.3 软件过程



1.4 软件过程管理



1.5 软件过程模型



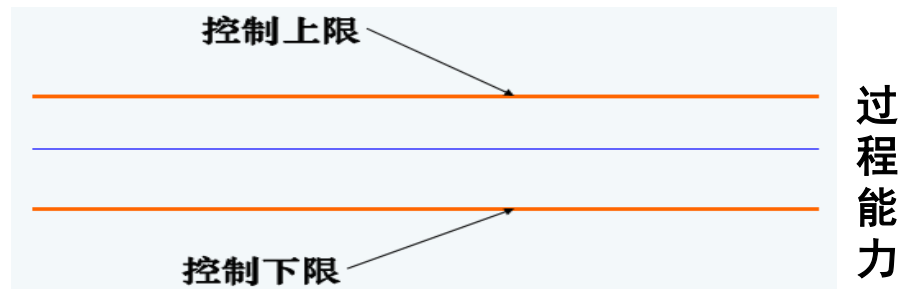
1.6 CMM/CMMI模型

1.5 软件过程模型

— 软件过程能力

软件过程能力 (Software Process Capability, SPC) 描述了在遵循软件过程后能够实现预期结果的程度。是软件过程本身具有的按预定计划生产产品的能力，或者说是得到预期结果的界限范围。一个组织的软件过程能力为组织提供了预测软件项目开发的数据基础。

软件过程能力是企业能力，
而不是个人能力。

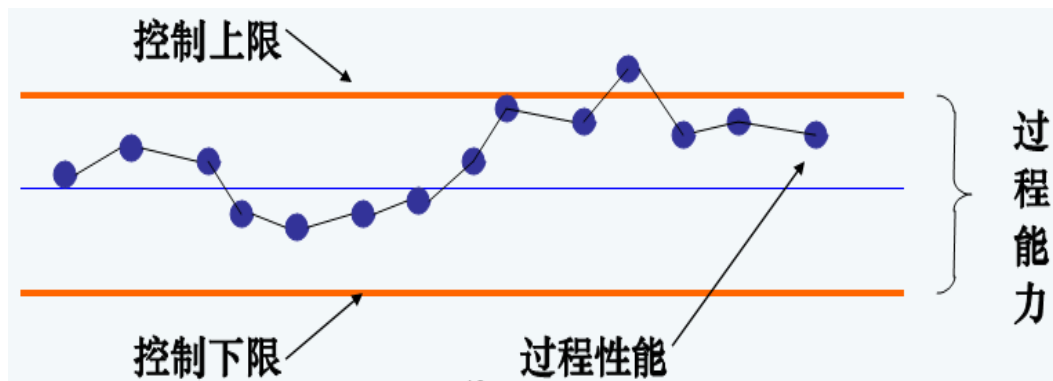


1.5 软件过程模型

— 软件过程性能

软件过程性能（Software Process Performance）遵循软件过程所得到或者软件过程执行的实际结果；过程性能体现了过程的实际能力，是过程能力的具体体现。

过程性能度量包括两个方面，一方面是对过程的度量（如工作量、评审效率、缺陷移除等），另一方面是对产品质量的度量（如故障数、缺陷密度等）



软件过程成熟度是指软件过程被明确地定义、管理、度量、评价、控制和产生实效的程度，意味着软件过程能力持续改善和提升，表明企业实施软件过程的实际水平和增长潜力。

软件过程成熟度通常用等级描述，定义每个等级的关键过程域的目标和实践，每一个等级为上一个等级提供基础和前提。

1.5 软件过程模型

—— 软件过程成熟标准

- (1) 软件过程能力高;
- (2) 软件过程性能可预见性
- (3) 软件过程规范化
- (4) 过程的一致性
- (5) 过程的丰富性
- (6) 过程的可视性
- (7) 过程的稳定性
- (8) 过程的不断改进

1.5 软件过程模型

——成熟过程与不成熟过程比较

	不成熟过程	成熟过程
角色与职责	没有明确规定角色和职责 每个人在做他认为要做的事 常会发生重叠和不清楚的所属关系和责任	角色和职责已有明确规定 相互关系无重叠 有明确的目标和测量方法 能够体现持续过程改进的机制
处理变更的方式	每个人都按自己的想法做事	人员遵循一个规划好的文件化过程 可分享取得的经验
对发生问题的反应	无秩序的混乱现象随处可见 “救火”方式解决出现问题的情况经常发生 每个人都想当英雄	根据已有的知识和专业规则对发生的问题进行分析处理
可靠性	延迟交付产品或超出预算 如有估算也不可靠	估算准确 项目得到有效的控制和管理 目标一般能够达到
对工作人员的奖励	奖励的对象是“救火队员”	奖励那些生产高质量产品团队，没有或少有失效 奖励那些防火者而不是救火者
预见性	质量不可把握，依赖于个人 进度和预算不能根据以往的经验确定 过程不透明	项目的进度和产品的质量均可预见 进度和预算根据以往项目的经验确定，符合实际 过程是透明的

1.5 软件过程模型

—— 典型软件成熟度模型

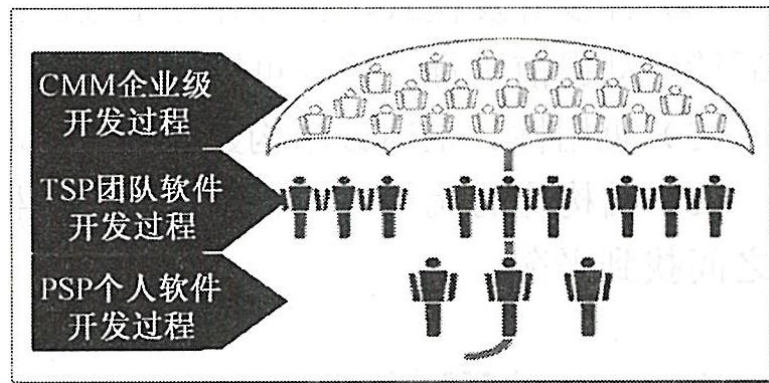
CMM/CMMI(Capability Maturity Model for software)

PSP/TSP(Personal/Team Software Process)

敏捷模型、极限编程XP(Xtreme Programming)

IPD(Integrated Product Development,IBM)

ISO9001



第1章 概论



1.1 软件企业



1.2 软件项目



1.3 软件过程



1.4 软件过程管理



1.5 软件过程模型



1.6 CMM/CMMI模型

CMM/CMMI原本是当年美国军方为了评估自己的软件产品供应商的过程质量水平，而委托美国卡内基梅隆大学软件工程学院（SEI）开发的一套过程评估体系。

后来美国卡耐基梅隆大学软件工程研究所组织全世界的软件过程改进和软件开发管理方面的开发而来，并在全世界推广实施的一种软件能力成熟度评估标准，主要用于指导软件开发过程的改进和进行软件开发能力的评估。

- CMM (Capability Maturity Model) 是对软件组织定义、实施、度量、控制和改善其软件过程实践的各个发展阶段的描述。
- CMM是把软件开发视为一个过程，对软件开发和维护过程进行过程监控和研究，使其更加科学化、标准化、使企业能更好的实现商业目标。

- CMM被用来确定一个机构的软件过程的成熟程度以及指明如何提高该成熟度的参考模型
- CMM描述了软件过程从无序到有序、从特殊到一般、从定性管理到定量管理、最终到达可动态优化的成熟过程，给出了该过程中五个成熟阶段的基本特征和应遵循的原则、采取的行动，以帮助软件机构改进其软件过程

CMM/CMMI给软件企业提供一个软件过程**框架模型**、**路线图**，只要按照这个框架模型、路线图不断进行实践和改进，就可以逐步提高软件过程能力成熟度，可以解决软件生产中的困难。

CMM/CMMI是目前国际上最流行、最实用的一种软件生产过程标准，得到了众多国家及国际软件产业界的认可，成为当今企业从事规模软件生产不可缺少的一项内容。

CMM/SW-CMM：软件能力成熟度模型

P-CMM：人力资源管理能力成熟度模型

SA-CMM：软件获取能力成熟度模型

IPD-CMM：集成产品开发能力成熟度模型

SE-CMM：系统工程能力成熟度模型

SSE-CMM：系统安全工程能力成熟度模型

CMM族，在CMM族的情况下，如果一个组织涉及到多个领域，就要同时实施几个模型，而这些模型之间又存在一些差异，会给组织造成困难、分散精力和资源，最终每个都难于取得成效；同时在人员培训、过程的评估和改进方面付出多重代价，增加了组织的负担，提高了过程改进的成本。

卡内基.梅隆大学软件工程研究所SEI研究CMM族的集成整合问题，并发布了集成的软件能力成熟度模型**CMMI**。

CMM/CMMI只关注“应该做什么”，而不关注“应该怎么做”，未提供实现各过程域所需要的知识和方法，为了解决上述问题，CMU-SEI在CMM1.1基础上提出了PSP/TSP。

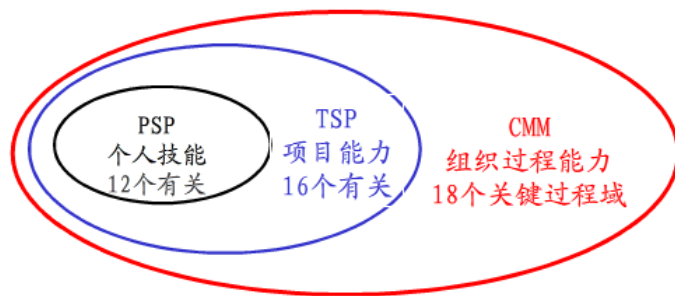
PSP是一种可用于控制、管理和改进软件工程师个人工作方式的自我改善过程，是一个包括软件开发表格、指南和规程的结构化框架。对于一个要改善软件质量和过程质量的企业来说，规范软件工程师的开发过程是解决软件质量问题的必要前提。

1.6 CMM/CMMI模型

— CMM、PSP、TSP

TSP注重团队的高效工作和产品交付能力，使软件工程师将个体过程结合进小组软件过程，并通过指导管理层如何支持和授权项目小组，坚持团队的高质量的工作，并且依据数据进行项目管理，以达到生产高质量产品的目的。

CMM、PSP和TSP三者之间的关系

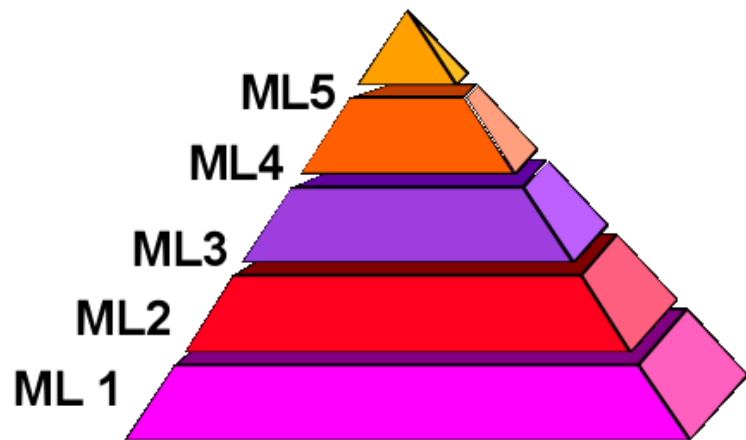


CMM/CMMI 使用**级别**为一个要改进其软件开发过程的组织描述了一种推荐的演变路径。级别也可以是评估中评级活动的产出结果。

CMM/CMMI支持两种使用级别的改进路径。即**能力级别**和**成熟度级别**。能力级别路径使组织能够渐增地改进由组织选定的、一个（或几个）单独过程域的过程。成熟度级别路径使组织能够通过渐增地解决连续的过程域集来改进相关的过程集合。

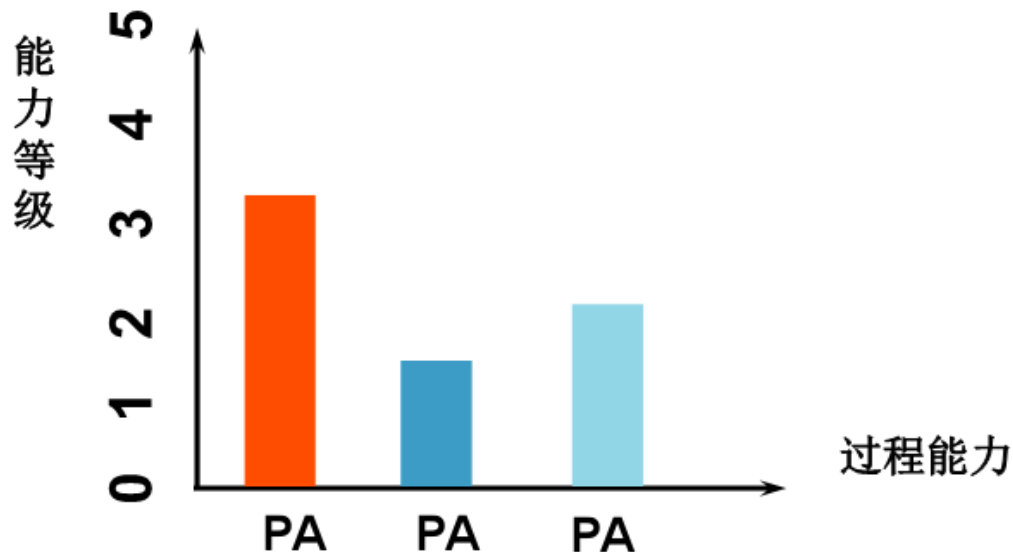
1.6 CMM/CMMI模型

— CMMI的表示方法



组织的成熟度等级

阶段型表示

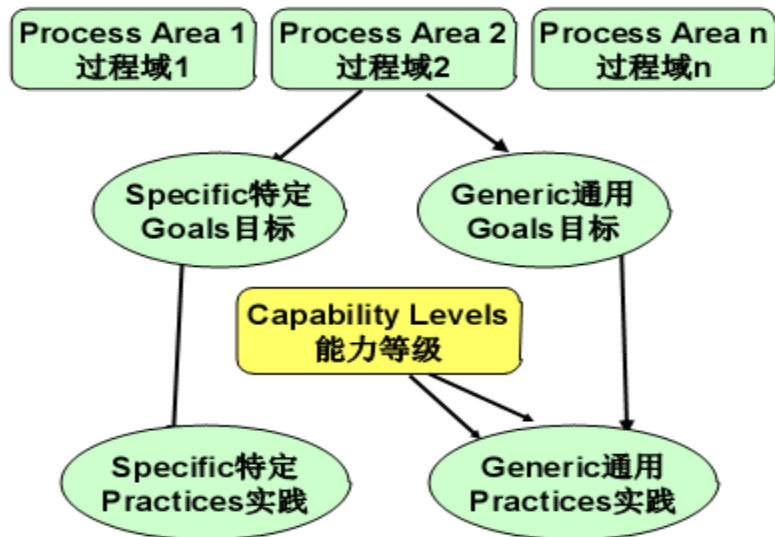


连续型表示

1.6 CMM/CMMI模型

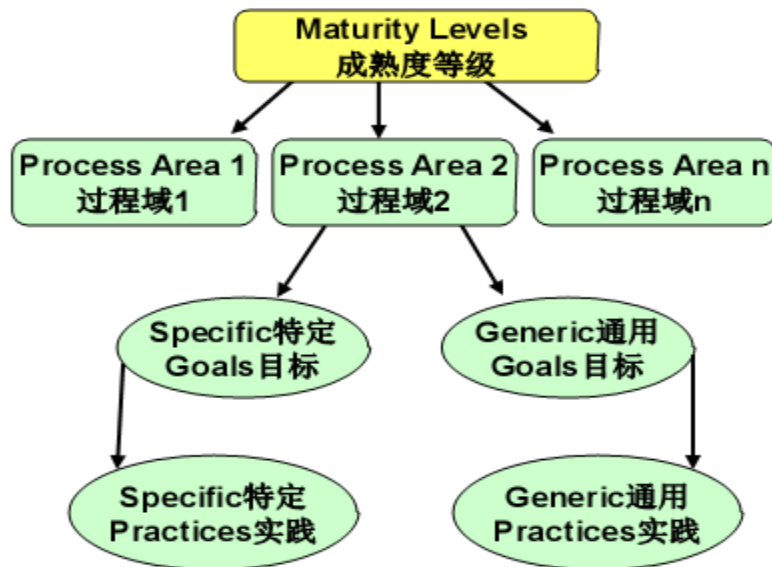
— CMMI的表示方法

Continuous连续式



Similar to SE-CMM 与SE-CMM类似
6 Capability Levels (CL) 6个能力等级
Flexible selection of process areas to improve
可灵活选择待改进的过程域

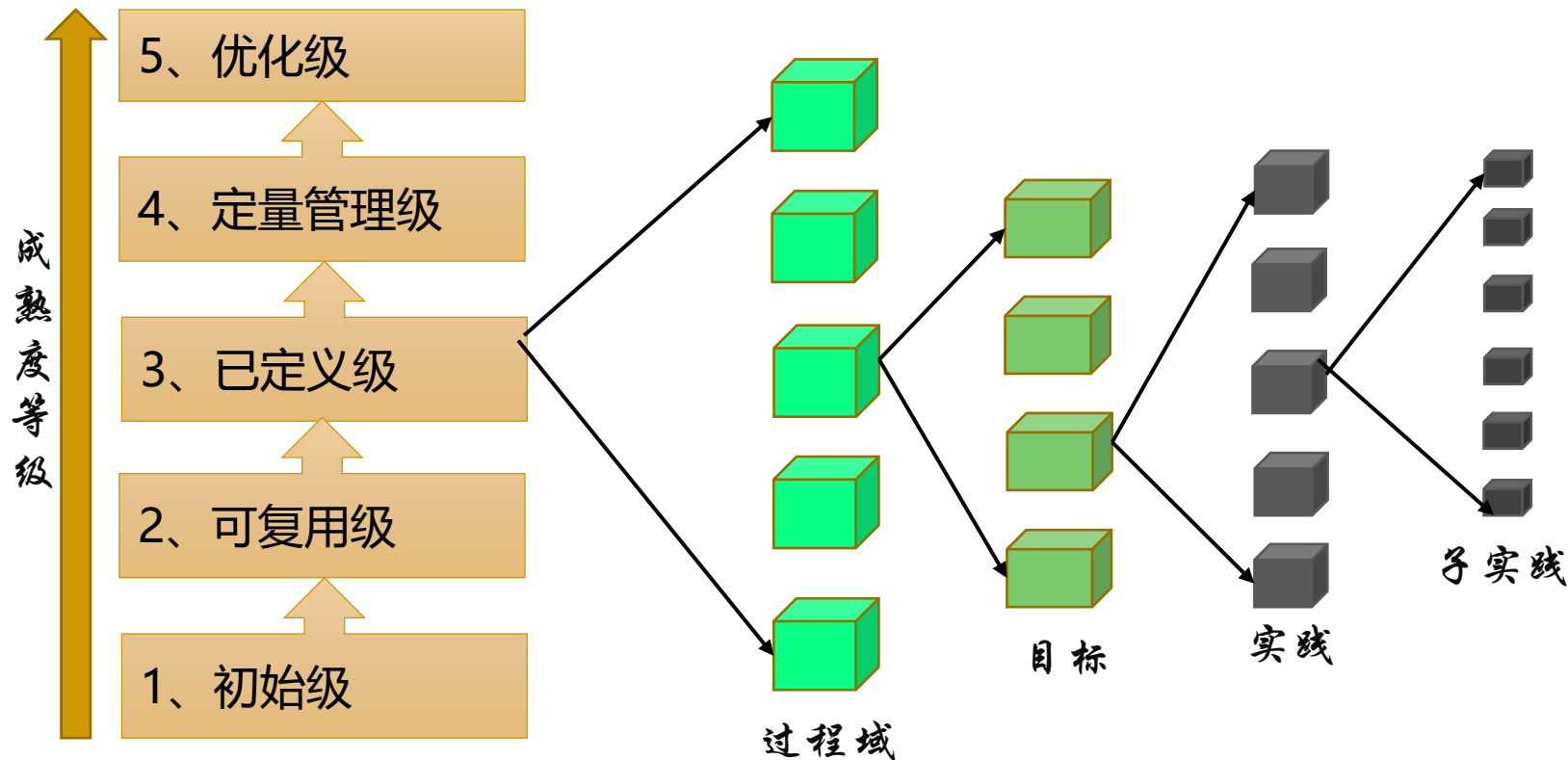
Staged阶段式



Similar to SW-CMM 与SW-CMM类似
5 Maturity Levels (ML) 5个成熟度等级
A set of pre-defined process areas at each ML
在每个成熟度等级上预定义了一组过程域

1.6 CMM/CMMI模型

— CMMI阶段表示方法



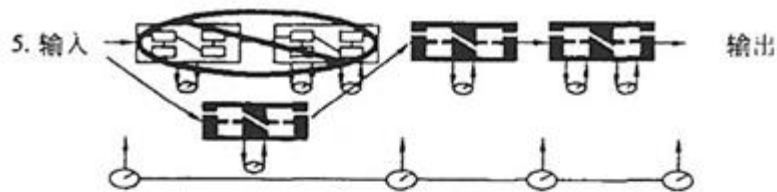
1.6 CMM/CMMI模型

— CMMI的阶段表示方法

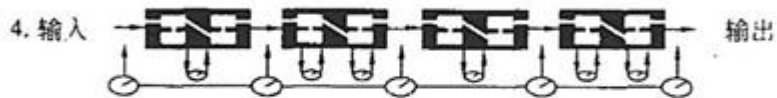
1	初始级	没有稳定的软件过程规范和软件工程环境，项目进展难以预测，项目行动大多是救火式驱动，项目的成功依赖于有经验的人员或小组。
2	可重复级	基于以往类似项目经验和自身项目特点，制定了每个项目主要过程的管理规范，且可重复可预期，但未上升到组织级的、标准的、一致的项目过程管理规范。
3	已定义级	制定了组织级的标准过程文件，建立了组织的软件工程过程组EPG和软件过程基础设施，实施了全员过程知识和技能的培训，指导具体项目工作。
4	已管理级	为产品和过程建立了量化指标，开展有效的过程度量，建立组织财富库，范围、进度、成本、质量等要素和过程能力、过程绩效已受控。
5	可优化级	积极吸收项目过程中的优秀实践、技术方法和管理方法，上升为组织级软件过程标准文件，并积极推广应用，建立了高效的软件过程改进机制。

1.6 CMM/CMMI模型

— CMMI的阶段表示



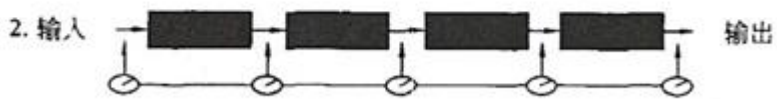
建立了软件过程改进机制，不断吸收先进实践成果，软件过程绩效和过程能力持续改进提升。



过程定量管理与控制，过程绩效和过程能力受控。



具有组织级标准过程，并指导所有项目工作。



每个项目都有基本的过程规范，部分过程可重复可预测，但没有组织级的一致性、标准的软件过程。



软件过程是无定形的、不稳定的，过程不透明。

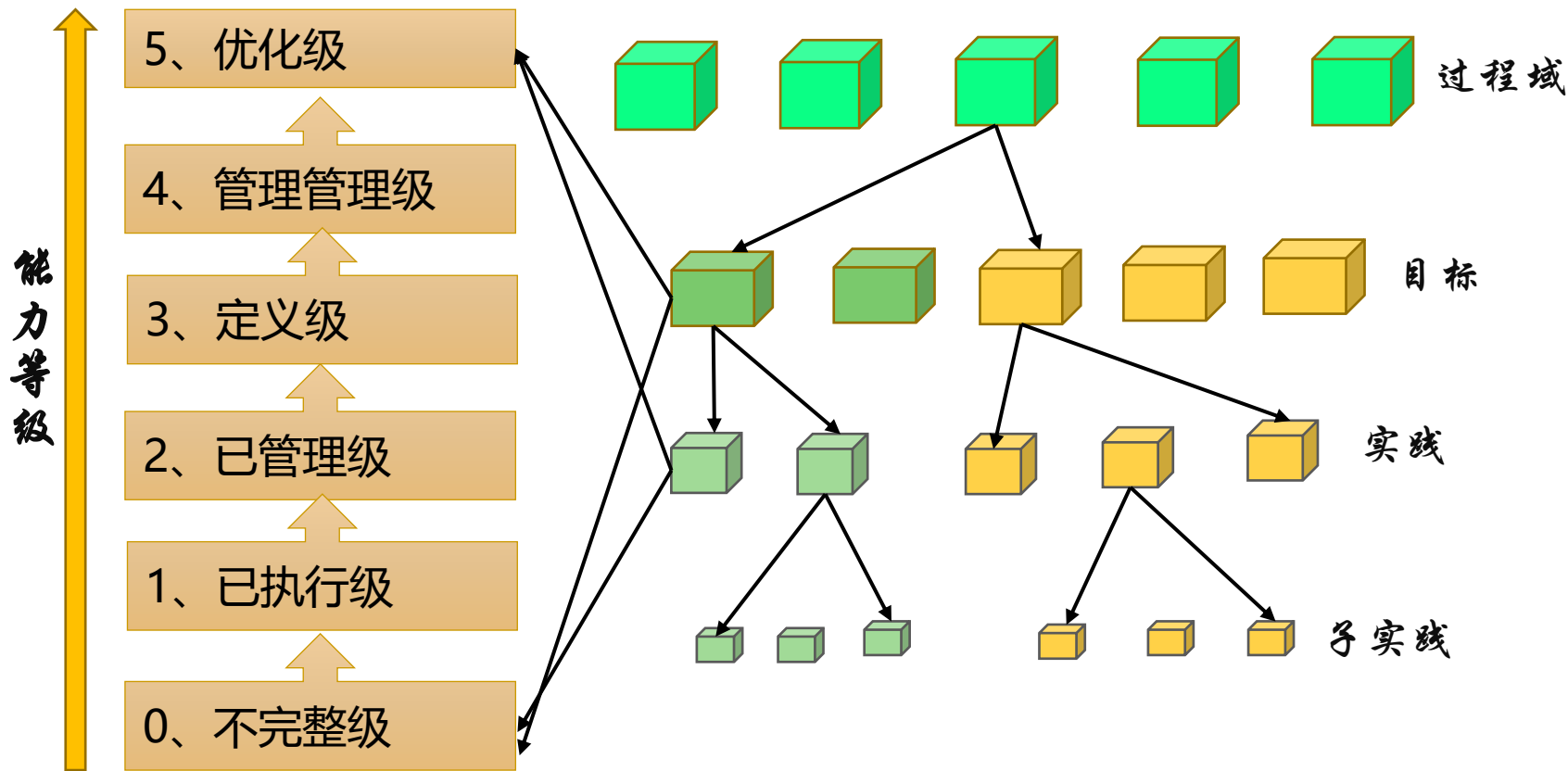
1.6 CMM/CMMI模型

— CMMI阶段表示关键过程域

可重复级	已定义级	已管理级	可优化级
需求管理 项目策划 项目监督与控制 供应商合同管理 过程和产品质量保证 配置管理 度量和分析	组织级过程焦点 组织级过程定义 组织级培训 集成化团队 集成化项目管理 组织级集成环境 需求开发 技术解决方案 产品集成 验证 确认 风险管理	量化项目管理 组织级过程性能	原因分析与解决方案 决策分析与解决方案 组织级改革和实施

1.6 CMM/CMMI模型

— CMMI的连续表示方法



1.6 CMM/CMMI模型

— CMMI连续表示关键过程域

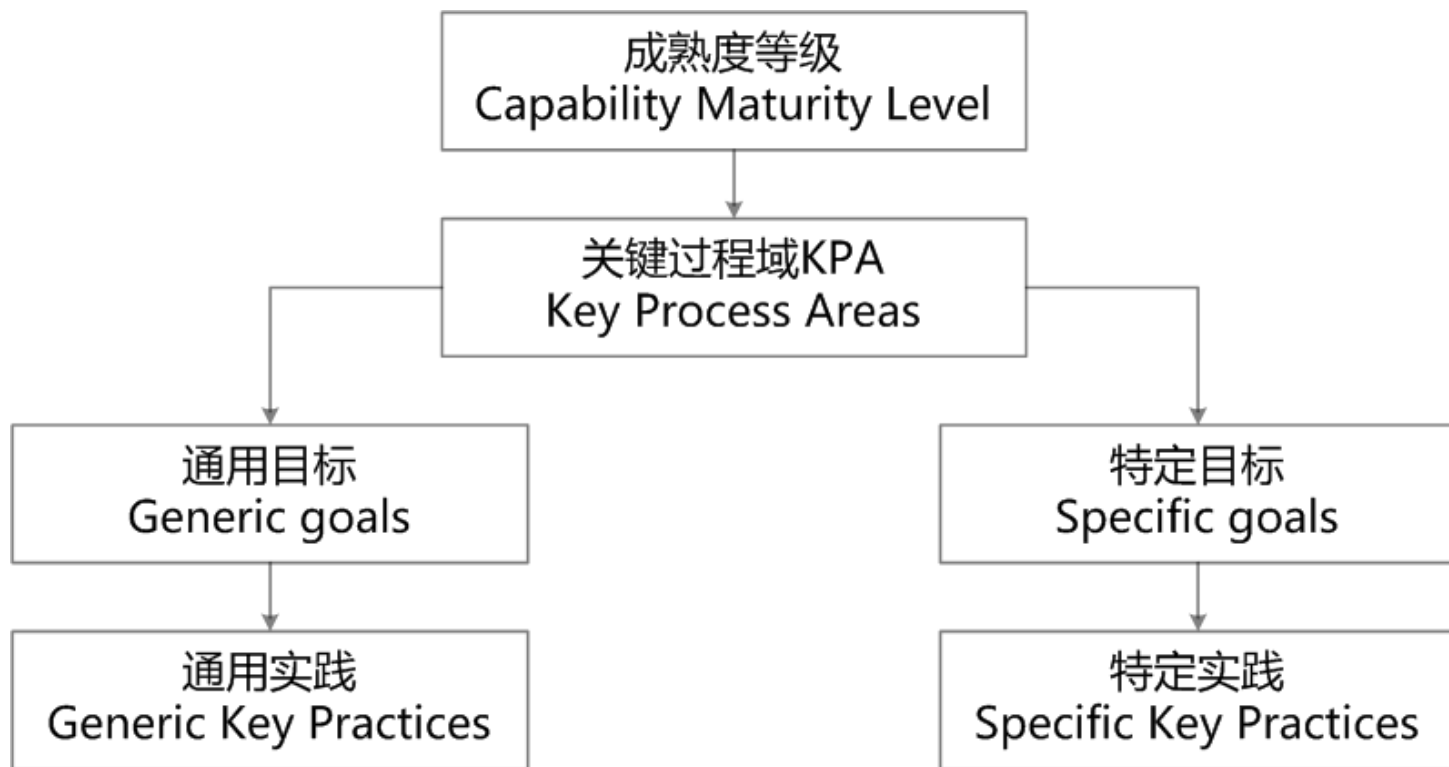
关键过程域是最重要的软件过程活动，是企业需要集中力量改进的部分，用于达到增强过程能力的效果。包括过程、项目、工程、支持四类关键过程域。

表 2-2 CMMI v1.1 过程域分类

过程管理类	项目管理类	工程类	支持类
机构过程焦点 (OPF) 机构过程定义 (OPD) 机构培训 (OT) 机构过程性能 (OPP) 机构改进和部署 (OLD)	集成项目管理 (IPM) 风险管理 (RSKM) 集成供应商管理 (ISM) 集成化团队 (IT) 项目策划 (PP) 项目监督与控制 (PMC) 供应协议管理 (SAM) 项目定量管理 (QPM)	需求制订 (RM) 需求开发 (RD) 技术解决方案 (TS) 产品集成 (PI) 验证 (VER) 确认 (VAL)	原因分析与解决方案 (CAR) 决策分析与解决方案 (DAR) 组织集成环境 (OEM) 过程与产品质量保证 (PPQA) 配置管理 (CM) 度量与分析 (MA)

1.6 CMM/CMMI模型

— CMMI的过程域



1.6 CMM/CMMI模型

— CMMI的过程域

过程域是一系列实践和目标的集合，通过执行这些实践就能实现一组相应的目标。

过程域描述

目的
简要说明
有关的过程方面
特定目标
通用目标
特定实践
典型工作产品
子实践
通用实践
通用实践的详细说明

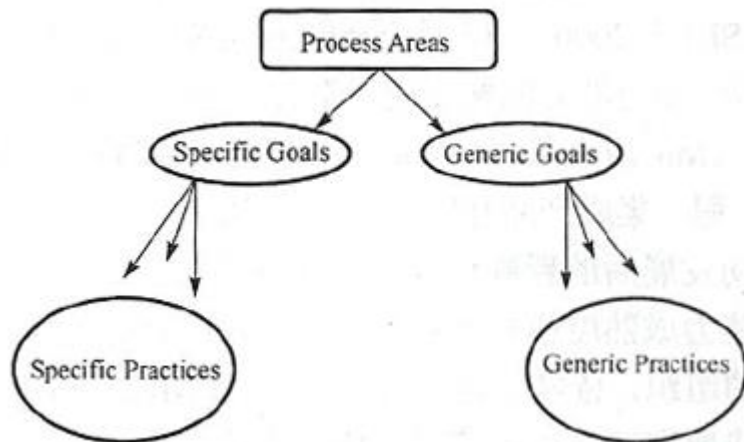
目标表明了每个关键过程域的范围、界限和意图。每一组目标都概括了该关键过程域的一组关键实践，并且可作为一个组织或一个项目是否有效实施了该关键过程域的判定标准。

关键过程域	目 标
需求管理 requirements management	<ul style="list-style-type: none">• 应对软件需求加以控制,以建立软件工程和管理活动的基线• 软件计划、软件产品和活动均与需求保持一致
软件项目策划 software project planning	<ul style="list-style-type: none">• 将对项目的估算写成文件,以供项目策划和跟踪使用• 项目的活动和承诺都应制定计划并形成文件• 项目相关的小组和人员都要对项目有关的承诺取得一致意见
软件项目跟踪 和监督 software project tracking and oversight	<ul style="list-style-type: none">• 将计划实施取得的成果和计划实施情况与计划对照跟踪• 在计划实施中所得到的实际结果和执行情况与软件计划有较大偏离时,要采取纠正措施加以控制• 项目相关的小组和人员对承诺的变更取得一致意见

1.6 CMM/CMMI模型

— CMMI的目标

包括通用目标GG、特定目标SG。通用目标作用于全部的关键过程域、特定目标作用于具体的一个关键过程域。



通用目标（GG）：某个过程方面达到了通用目标，即意味着该过程方面的实施是有效的、可重复的和持久的，并且实现了某种程度的制度化。

以需求管理为例

GG2 把需求管理过程制度化为受管理过程
把该过程作为受管理过程加以制度化。

通用实践（GP）：通用实践适用于所有过程方面。通用实践提供的是制度化的特性，这些特性将确保相应的过程实施有效、可重复和持久。

以需求管理为例

GP2.8 监督和控制该过程

对照计划监督和控制“需求管理”过程，并且采取适当的纠正措施。

1.6 CMM/CMMI模型

— CMMI通用目标和实践

GG1	满足特定目标	GP1	执行特定实践
GG2	制度化已管理过程	GP2.1	建立组织方针政策
		GP2.2	策划该过程
		GP2.3	提供资源
		GP2.4	分配职责
		GP2.5	培训人员
		GP2.6	管理配置
		GP2.7	确定干系人并使其介入参与
		GP2.8	监督和控制过程
		GP2.9	客观评价符合度
		GP2.10	与高层管理者一起审查过程状态
GG3	制度化已定义过程	GP3.1	建立已定义过程
		GP3.2	收集过程信息和改进经验
GG4	制度化定量管理过程	GP4.1	建立过程性能和质量的指标
		GP4.2	稳定过程性能
GG5	制度化可优化过程	GP5.1	保证过程的持续改进
		GP5.2	分析问题的根本原因

特定目标 (SG)：只适用于一个特定的过程方面并且涉及该过程方面的唯一性特性，这些特性描述的是必须实施哪些内容才能达到该过程方面的目的。

以需求管理为例：

SG1 管理需求

对需求进行管理并识别与项目计划和工作产品的不一致之处。

特定实践 (SP)：是为达到相应的特定目标所执行的重要活动。特定实践描述的是那些为实现某过程方面的特定目标所期望的活动。

以需求管理为例

SP1.3 管理需求变更

随着各项需求在项目推进期间发生演变的同时，对需求的变更进行管理。

1.6 CMM/CMMI模型

— CMMI特定目标和实践

需求开发			
SG1	开发客户需求	SP1.1	引导需求
		SP1.2	将相关干系人的业务需要转化为客户需求
SG2	开发产品需求	SP2.1	建立产品或产品组件需求
		SP2.2	分配产品组件需求
		SP2.3	识别接口需求
SG3	分析并确认需求	SP3.1	建立操作概念和场景
		SP3.2	建立必要的功能和质量属性定义
		SP3.3	分析需求
		SP3.4	分析需求以取得平衡
		SP3.5	确认需求
需求管理			
SG1	管理需求	SP1.1	理解需求
		SP1.2	获得对需求的承诺
		SP1.3	管理需求的变更
		SP1.4	维护需求的双向可追溯性
		SP1.5	确保项目和需求之间的一致性

过程改进：企业遵循CMMI的阶梯式演化框架，把握关键过程域，指导软件项目和软件过程的改进，提升企业的软件过程能力和成熟度，提高产品质量和企业效益。

能力评估：评估企业的软件过程能力和成熟度等级，授予相应的资质证书，在招投标、项目申报、企业认定方面大有用处。

CMM建立了一组软件机构过程成熟度的评价准则，提供了一个阶梯式的演化框架，清晰地描述了软件过程的关键元素，包括软件工程和管理方面的优秀实践。

企业可以有选择地引用这些关键实践指导软件过程的开发和维护，以不断地改善组织软件过程，实现成本、进度、功能和产品质量等目标。

- 过程改进应有助于业务而不是为了自身的原因
- 改进意味着不同组织的不同目标:
 - 什么是组织的商业目标?
 - 如何度量进展?
- 改进是长期的、战略性的努力工作:
 - 根据基本出发点, 预期什么效果呢?
 - 度量的效果如何?

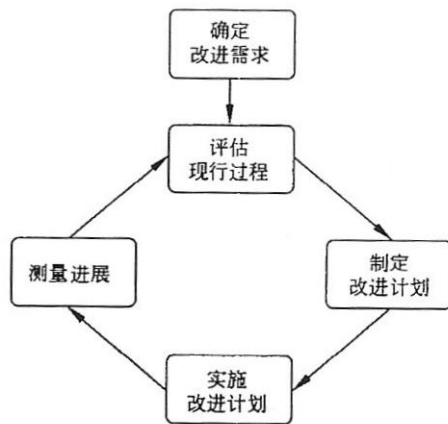
1.6 CMM/CMMI模型

— 过程改进路径

(1) 确定改进需求：明确改进的动机和必要条件，解决好为什么要改进的问题；

(2) 评估现行过程：掌握现行过程状态，解决哪些问题，确定改进目标；

(3) 制定改进计划：发现薄弱环节、制定改进计划，引进必要的管理技术和支持工具，就是回答如何进行改进；



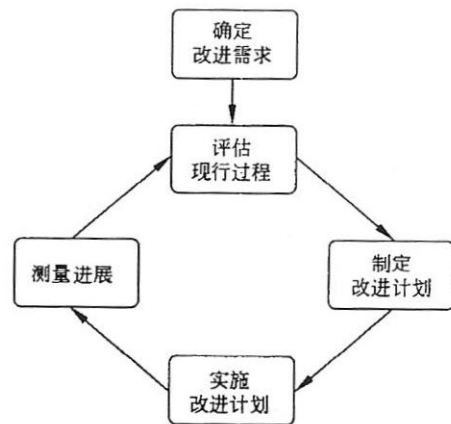
过程改进的基本步骤

1.6 CMM/CMMI模型

— 过程改进路径

(4) 实施改进计划：投入必要的资源，开展改进工作；

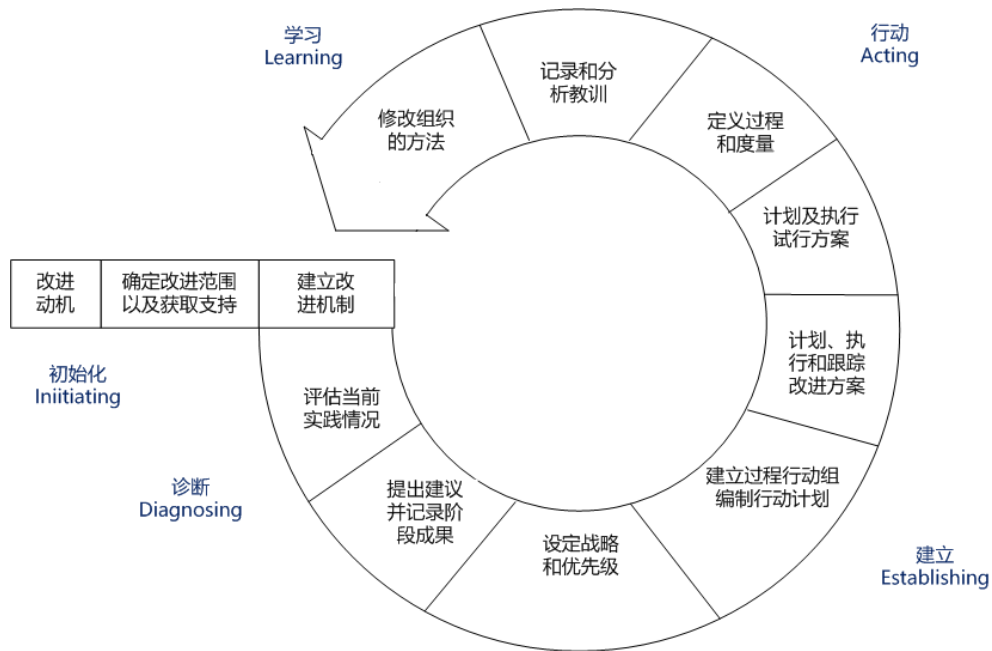
(5) 测量改进的进展：了解改进的成效和存在问题。



过程改进的基本步骤

1.6 CMM/CMMI模型

— 过程改进框架IDEAL



缩写	英文名称	阶段	主要内容
I	Initiating	初始化	商业论证、改进动机、改进目标、获得支持、建立机制
D	Diagnosing	诊断	调查评估、改进意见、改进方案
E	Establishing	建立	行动策略和优先级、行动计划、行动组织
A	Acting	执行	定义过程和度量、改进活动、监督和控制
L	Leveraging	推进	收集数据、分析评估、吸收成果、总结教训、调整过程

CMM是基于政府评估软件承包商的软件能力发展而来的，有两种通用的评估方法用以评估组织软件过程的成熟度，即软件过程评估和软件能力评估。

(1) 软件过程评估

用于确定一个组织当前的软件工程过程状态及组织所面临的软件过程的优先改善问题，为组织领导提供报告以获得组织对软件过程改善的支持。

(2) 软件能力评价

用于识别合格的软件承包商或者监控软件承包商开发软件的过程状态，以保证在预算和进度要求范围内制造高质量软件产品。

过程评估与能力评价差别

分 项	软件过程评估	软件能力评价
发起者	软件组织自身	获取软件的组织或客户
目的	<ul style="list-style-type: none">• 确定软件过程所处状态• 找到当前需要最优先解决的问题• 改进软件过程,提高产品质量	<ul style="list-style-type: none">• 了解能力成熟度,识别过程风险• 选择承包商• 监控已进行项目的过程状态
对象	软件组织自身	投标者,供选择的承包商或已承担项目的承包商
结论及结论使用	<ul style="list-style-type: none">• 过程的强项和弱项• 制定改进行动计划	风险报告、成熟度判定合格否
结论所属	软件组织自身	获取软件的组织或客户