

第三章 软件质量工程体系

◆ 1 概述

- 系统工程学的思想
- 软件质量工程体系的建立
- 软件质量工程体系的构成

◆ 2 软件质量因素和指标

◆ 3 软件质量模型分析

◆ 4 软件质量工作层次

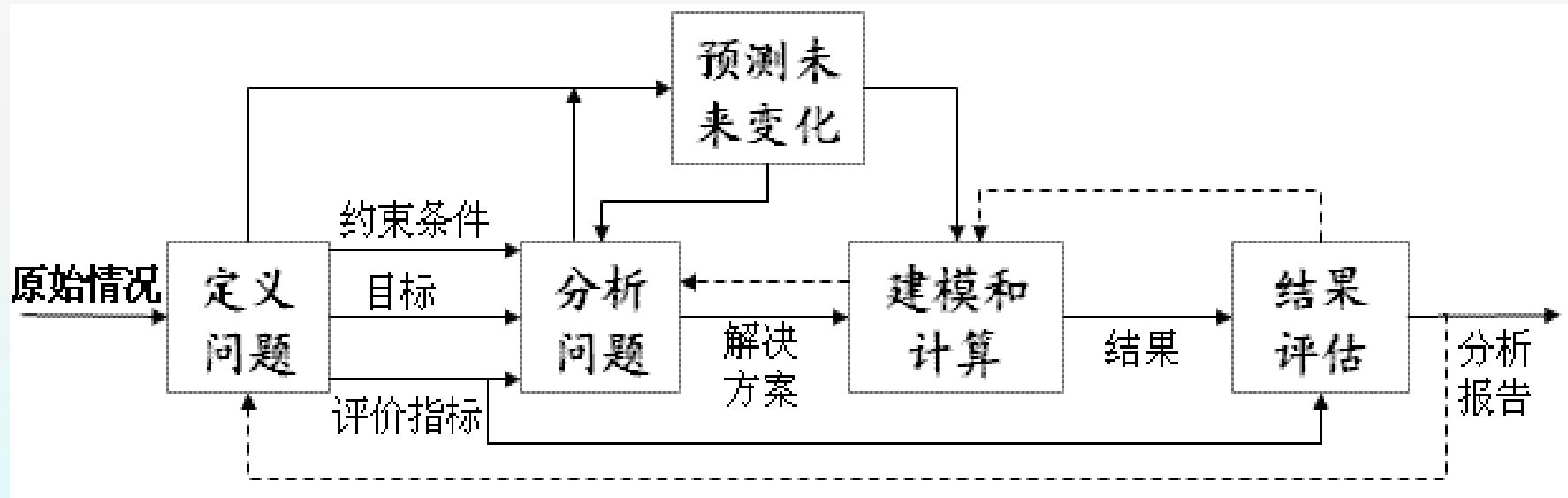
◆ 5 软件质量成本

◆ 6 软件质量标准 and 度量



系统工程学的思想

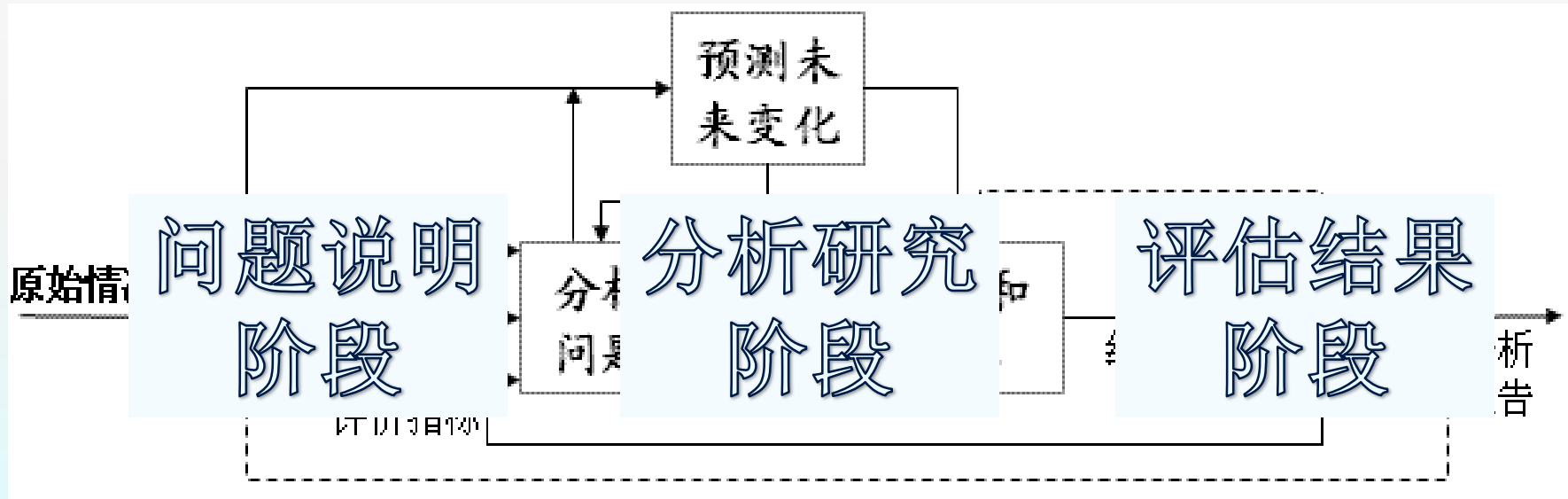
系统工程学是为了研究多个子系统构成的整体系统所具有的多种不同目标的相互协调，以期系统功能的最优化、最大限度地发挥系统组成部分的能力而发展起来的一门科学。



系统分析过程的逻辑结构

系统工程学的思想

系统工程学是为了研究多个子系统构成的整体系统所具有的多种不同目标的相互协调，以期系统功能的最优化、最大限度地发挥系统组成部分的能力而发展起来的一门科学。



系统分析过程的逻辑结构

软件质量工程体系的建立

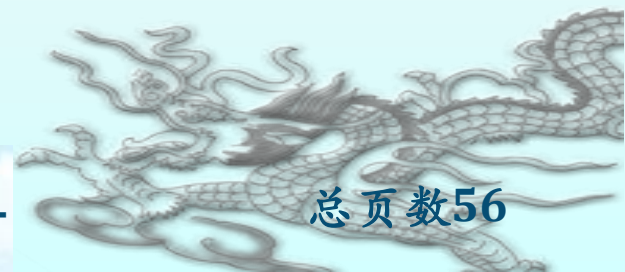
建立和实施质量管理体系的方法

- 确定顾客和其他相关方的需求和期望；
- 建立组织的质量方针和质量目标；
- 确定实现质量目标必需的过程和职责；
- 确定和提供实现质量目标必需的资源；
- 规定测量每个过程的有效性和效率的方法；
- 应用这些测量方法确定每个过程的有效性和效率；
- 确定防止不合格并消除产生原因的措施；
- 建立和应用持续改进质量管理体系的过程。



软件质量工程体系的建立

要将软件质量管理作为一个系统来管理，运用系统科学的方法，通过有关质量的各种信息反馈与调控，对软件质量进行全面、综合的系统性管理，包括软件质量计划、组织、协调、控制，以实现软件质量的目标。



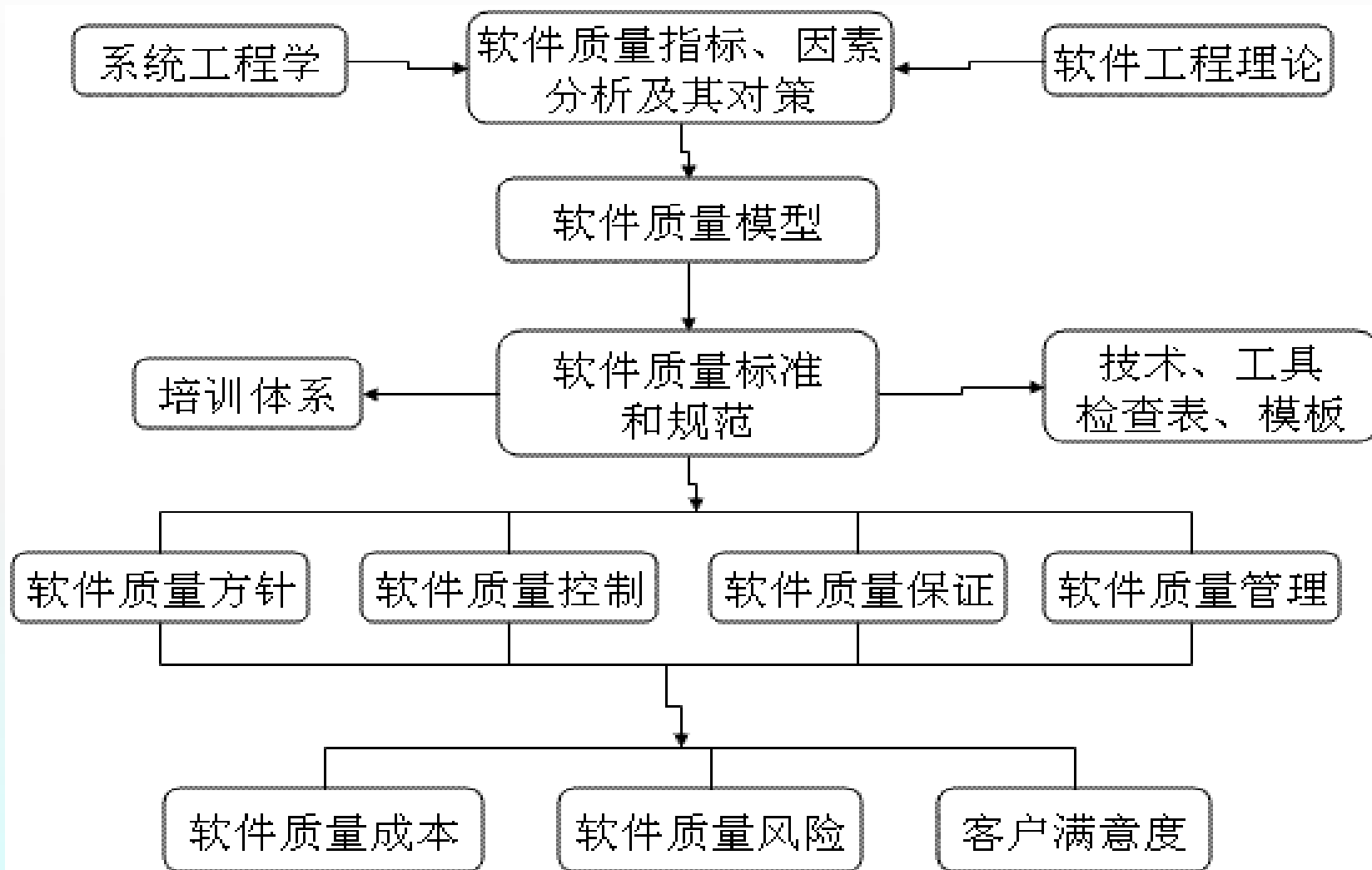
软件质量工程体系的构成

系统工程学

软件工程理论

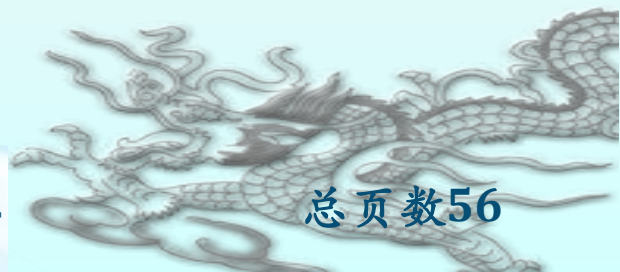


软件质量工程体系的构成



第三章 软件质量工程体系

- ◆ 1 概述
- ◆ 2 软件质量因素和指标
 - 软件质量指标
 - 软件质量因素
- ◆ 3 软件质量模型分析
- ◆ 4 软件质量工作层次
- ◆ 5 软件质量成本
- ◆ 6 软件质量标准 and 度量



软件质量指标

软件的6个品质要素

➤ 正确性：

实现的功能达到设计规范，并满足用户需求的程度

➤ 可靠性：

规定的时间和条件下，仍能维持其性能水准的程度

➤ 易用性：

用户掌握软件操作所要付出的时间及努力程度



软件质量指标

➤ 效率:

软件执行某项功能所需电脑资源（含时间）的有效程度

➤ 可维护性:

当环境改变或软件发生错误时，执行修改或恢复所做努力的程度

➤ 可移植性:

从一个系统/环境移到另一系统/环境的容易程度



软件质量指标

质量指标：

- 衡量那些可识别的软件质量特性的**项目**，
- 有助于软件质量进行度量，
- 利用软件工程方法来达到特定的质量目标。



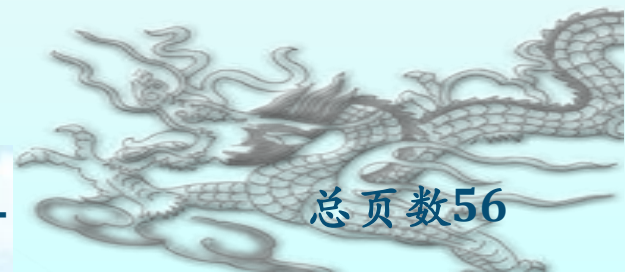
软件质量指标

功能性的质量指标:

- 功能的正确性
- 功能的准确性
- 软件功能的完整性

可用性的质量指标:

- 可操作性
- 通用性
- 一致性



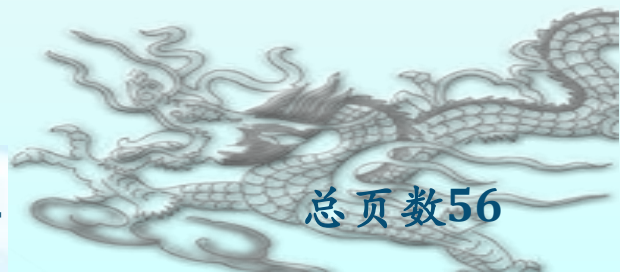
软件质量指标

可靠性的质量指标:

- 系统自我恢复能力
- 健壮性
- 系统的分布性

性能的质量指标:

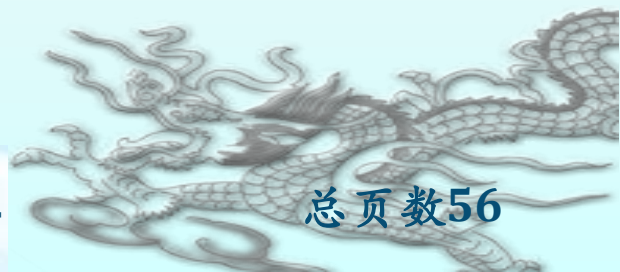
- 有效性
- 安全管理/完整性
- 易存取性



软件质量指标

可维护性的质量指标

- 简单性
- 自我描述性
- 系统兼容性
- 文档质量
- 增强能力/灵活性
- 模块化
- 可测试性
- 可追溯性



软件质量指标

可移植性的质量指标

- 独立性
- 可重用性
- 互操作性
- 虚拟性
- 一般性



软件质量因素

软件质量因素：

影响软件质量特性或影响软件指标的参数。

分类：

- 软件产品的质量因素
- 软件过程的质量因素



软件质量因素

软件产品的质量因素

产品修改

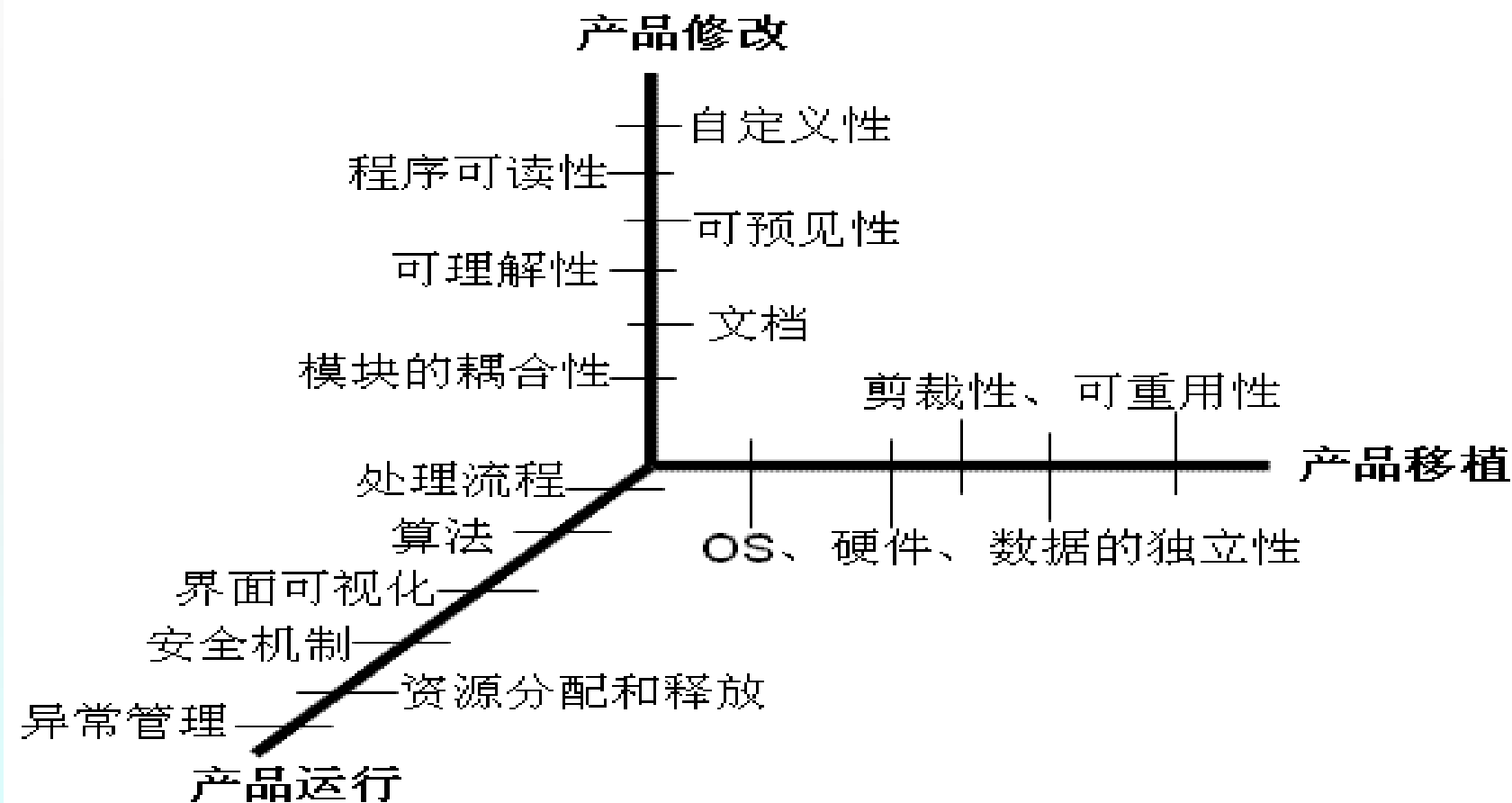
产品移植

产品运行



软件质量因素

软件产品的质量因素



软件质量因素

软件过程的质量因素

➤ 项目计划过程:

和客户的沟通能力、软件产品特性定义的方法、项目计划策略、评审的流程、范围、方式和程度、协同工作流程、合同和用户管理流程和方法、文档编写、管理等的规范和流程

➤ 项目设计过程:

软件产品指标的定义和解释、设计流程、设计标准、协同工作流程、文档编写、管理等的规范和流程



软件质量因素

➤ 项目实施过程:

变更控制流程、执行过程跟踪方法、
流程和相适应的系统、缺陷处理流程、
文档编写、管理等的规范和流程

➤ 软件维护过程:

变更控制流程、用户反馈、相应处理机制、
回归测试流程

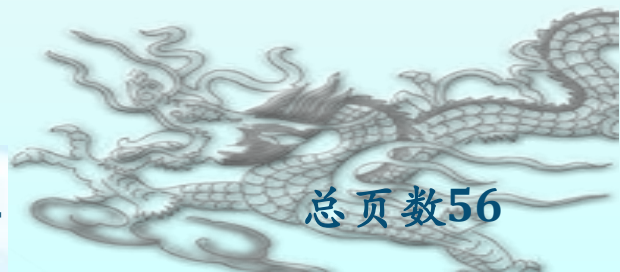
➤ 软件商业环境过程:

软件改进的策略、产品开发模式、市场定位、
产品标准等



第三章 软件质量工程体系

- ◆ 1 概述
- ◆ 2 软件质量因素和指标
- ◆ 3 软件质量模型分析
 - 软件质量模型
 - 模型评价和分析
- ◆ 4 软件质量工作层次
- ◆ 5 软件质量成本
- ◆ 6 软件质量标准和度量



软件质量模型

软件质量指标和软件质量因素的分析是为了建立软件模型。

软件模型是描述他们之间的关系，分析软件质量因素究竟如何影响质量指标的，从而寻求最优的质量保证解决方案，最终达到软件质量目标。



软件质量模型

- **McCall 软件质量模型 (GE模型, 1977)**

11个指标构成,分为产品操作,产品修订和产品转移

- **Boehm 模型 (1978)**

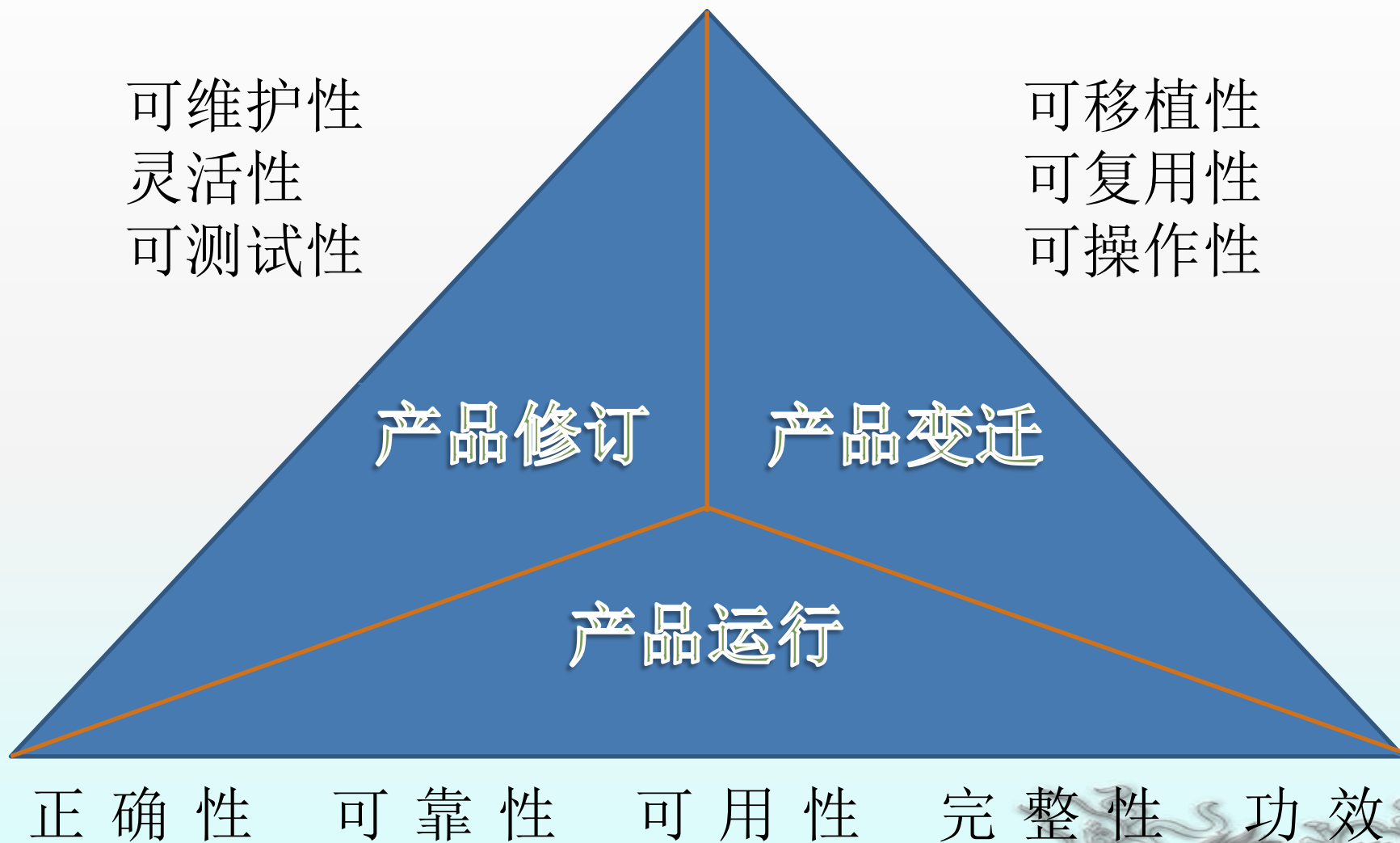
基于很多特性和标准

- **ISO 简化的软件质量模型**

包括6个质量目标,每个质量目标有很多属性



McCall 模型



Boehm 模型



ISO简化的软件质量模型

- 按照ISO/TC97/SC7/WG3/1985-1-30/N382, 软件质量度量模型由 3 层组成:

高层是软件质量需求评价准则 (SQRC)

中层是软件质量设计评价准则 (SQDC)

低层是软件质量度量评价准则 (SQMC)

- ISO认为, 应对高层和中层建立国际标准, 在国际范围内推广软件质量管理 (SQM) 技术, 而低层可由各使用单位视实际情况制定。
- ISO的 3 层次模型来自McCall等人的模型。高层、中层和低层分别对应于McCall模型中的特性、度量准则和度量。



ISO简化的软件质量模型

ISO/TC97/SC7/WG3/1985-1-30/N382

SQRC (8个元素)

功能性
可靠性
可使用性
效率
可维护性
可移植性
互连性
安全性



ISO/IEC9126

SQRC (6个元素)

功能性
可靠性
可使用性
效率
可维护性
可移植性
~~互连性~~
~~安全性~~

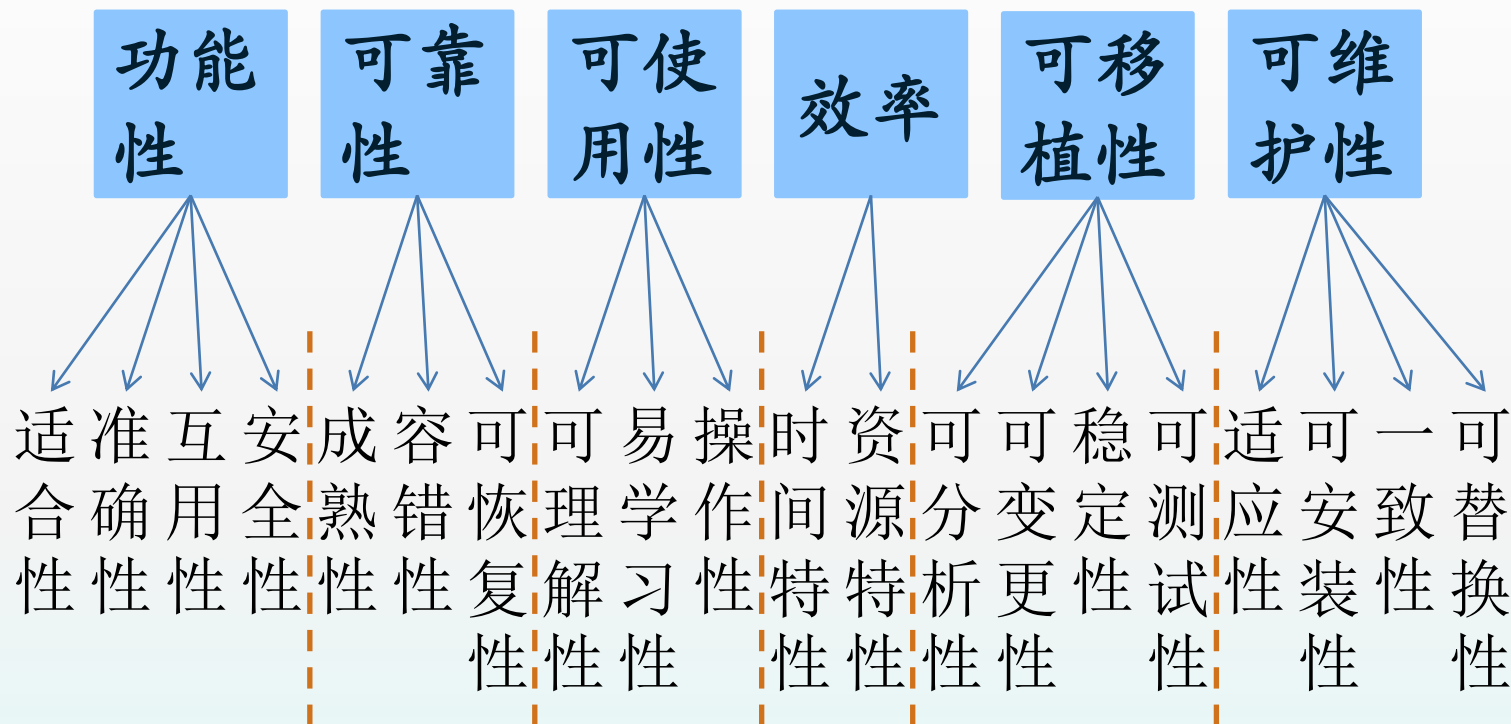


ISO简化的软件质量模型

SQ
RC

SQ
DC

SQ
MC



用户自定义软件产品度量标准



模型评价和分析

➤ 存在差别:

软件质量特性、影响因素或质量指标的定义不完全一致;

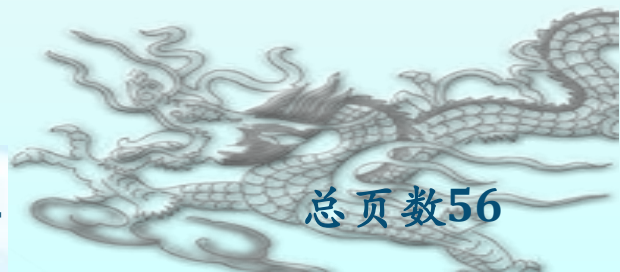
➤ 思想接近:

总体上要表达的思想非常接近;

➤ 目的相同:

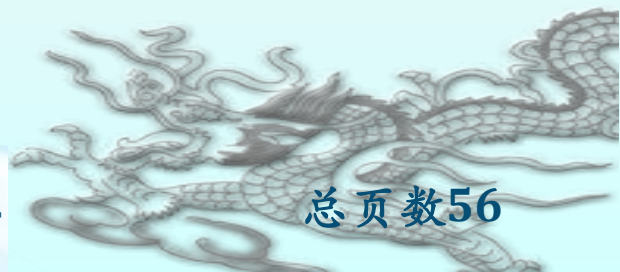
软件质量因素、准则和度量3者综合的软件质量结构模型;

ISO模型第一层（质量特性）和第二层（准则）的关系非常清楚，不像McCall模型和Boehm模型那样存在交叉关系。



第三章 软件质量工程体系

- ◆ 1 概述
- ◆ 2 软件质量因素和指标
- ◆ 3 软件质量模型分析
- ◆ 4 软件质量工作层次
 - 软件质量方针
 - 软件质量控制
 - 软件质量保证
 - 软件质量管理
- ◆ 5 软件质量成本
- ◆ 6 软件质量标准 and 度量



软件质量方针

- 在质量控制、质量保证和质量管理之上是质量方针。
- 即在质量方针指导下，质量管理指挥和控制组织的质量活动，协调质量的各项工作，包括质量控制、质量保证和质量改进。



软件质量方针

质量方针

- 由组织的最高管理者正式发布的、
- 该组织的质量宗旨、目标和质量方向，并形成文件。
- 质量方针是企业经营总方针的组成部分，
- 是企业管理者对质量的指导思想和承诺。
- 建立质量方针和质量目标为组织提供了关注的焦点，在一个质量管理体系中起着关键作用。
- 质量方针应体现软件公司的组织目标、顾客的期望/需求和组织内部质量行为的准则。



软件质量控制

质量控制

- 是一个**设定标准**(根据质量要求)、测量结果,判定**是否达到了预期要求**,对质量问题采取措施进行**补救并防止再发生**的过程
- 质量控制已不再仅仅是检验,而更多地倾向于确保生产出来的产品满足用户要求的**过程控制**。



软件质量保证

质量保证

- 为保护产品和服务充分满足消费者要求的质量而进行的有计划有组织的活动，致力于提供对满足质量要求的信任。

对象

- 内部质量保证

组织向自己的管理者提供信任；

- 外部质量保证

组织向外部客户或其它方提供信任。



软件质量保证

质量保证

活动分类

➤ 复审

在软件生命周期的每个阶段结束之前，都用结束标准对该阶段生产出的软件配置成分进行严格的技术审查等活动；

➤ 内审

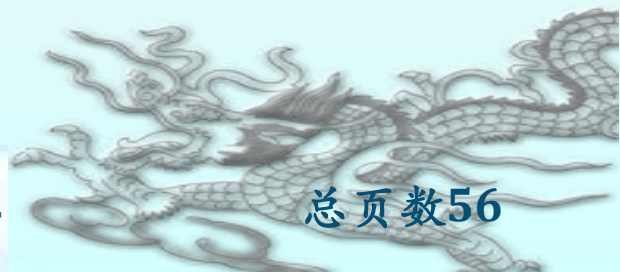
检查组织内部是否遵守已有的模板、规则、流程等。



软件质量改进

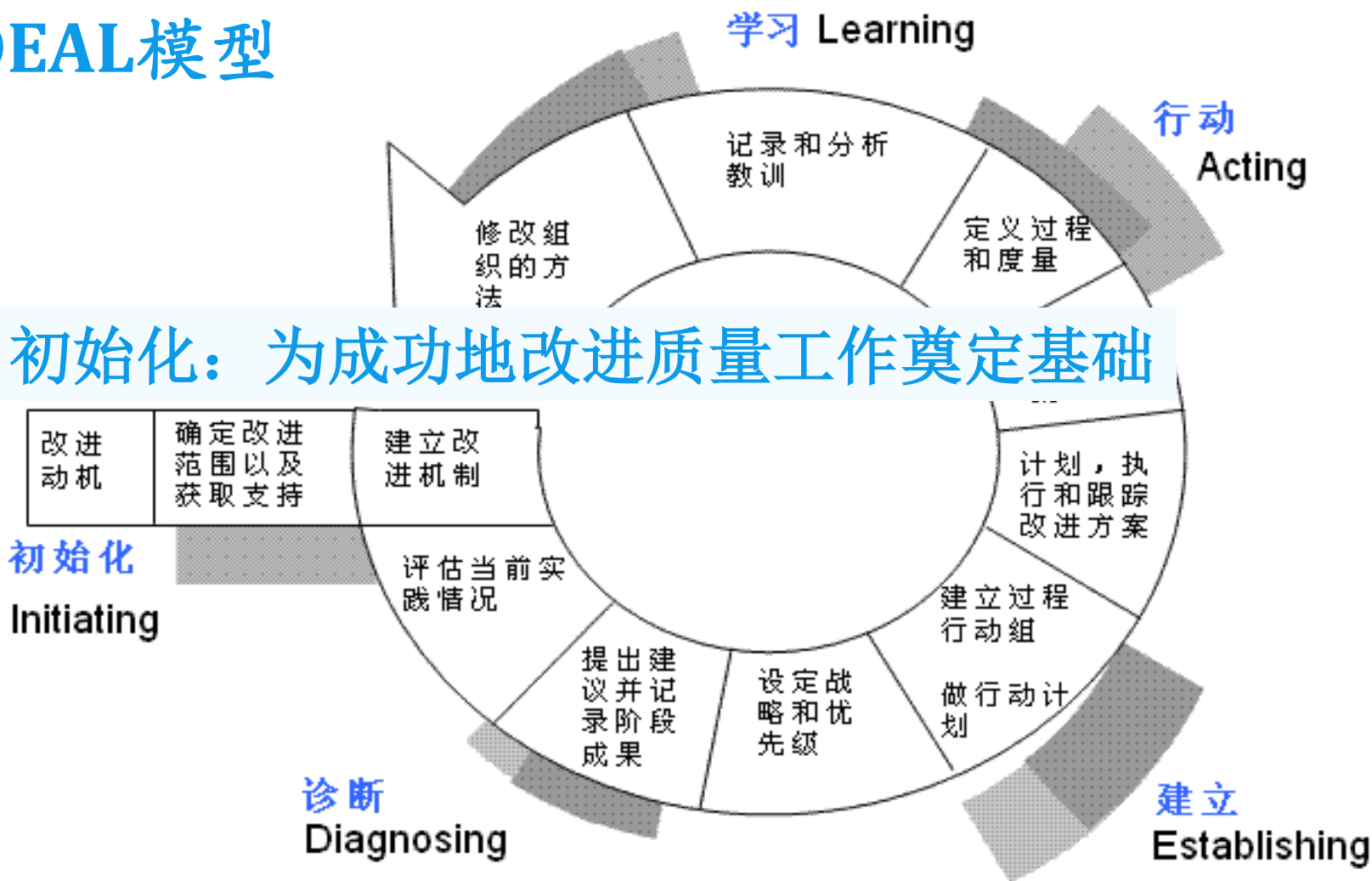
质量改进

- 致力于增强满足质量要求的能力，是一个持续过程。
- 在质量改进工作中，有许多模型，包括**IDEAL模型**、**六西格玛模型**、PDCA模型、PEIS模型、CMM模型和SPICE模型等。



软件质量改进

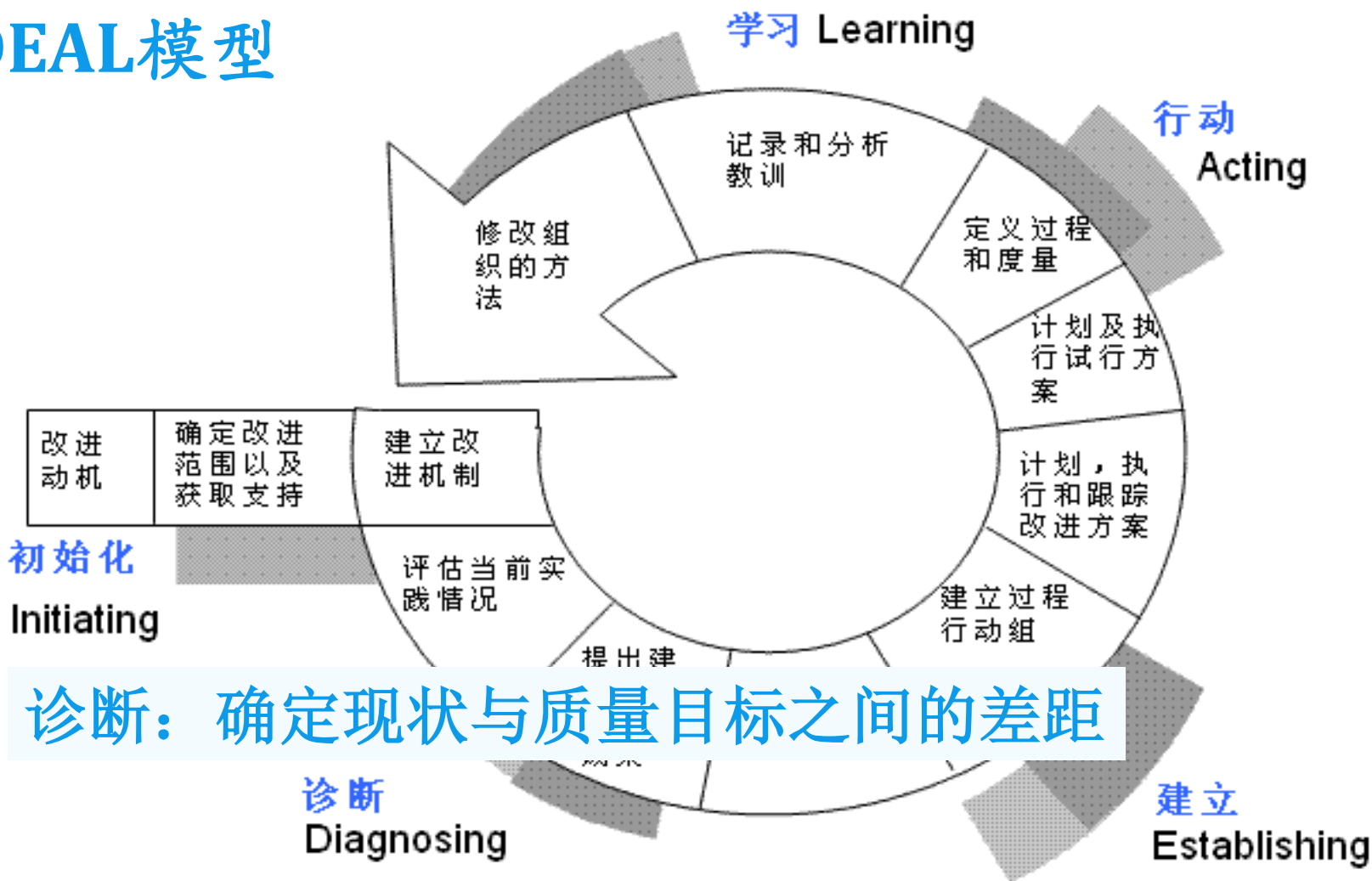
IDEAL模型



初始化：为成功地改进质量工作奠定基础

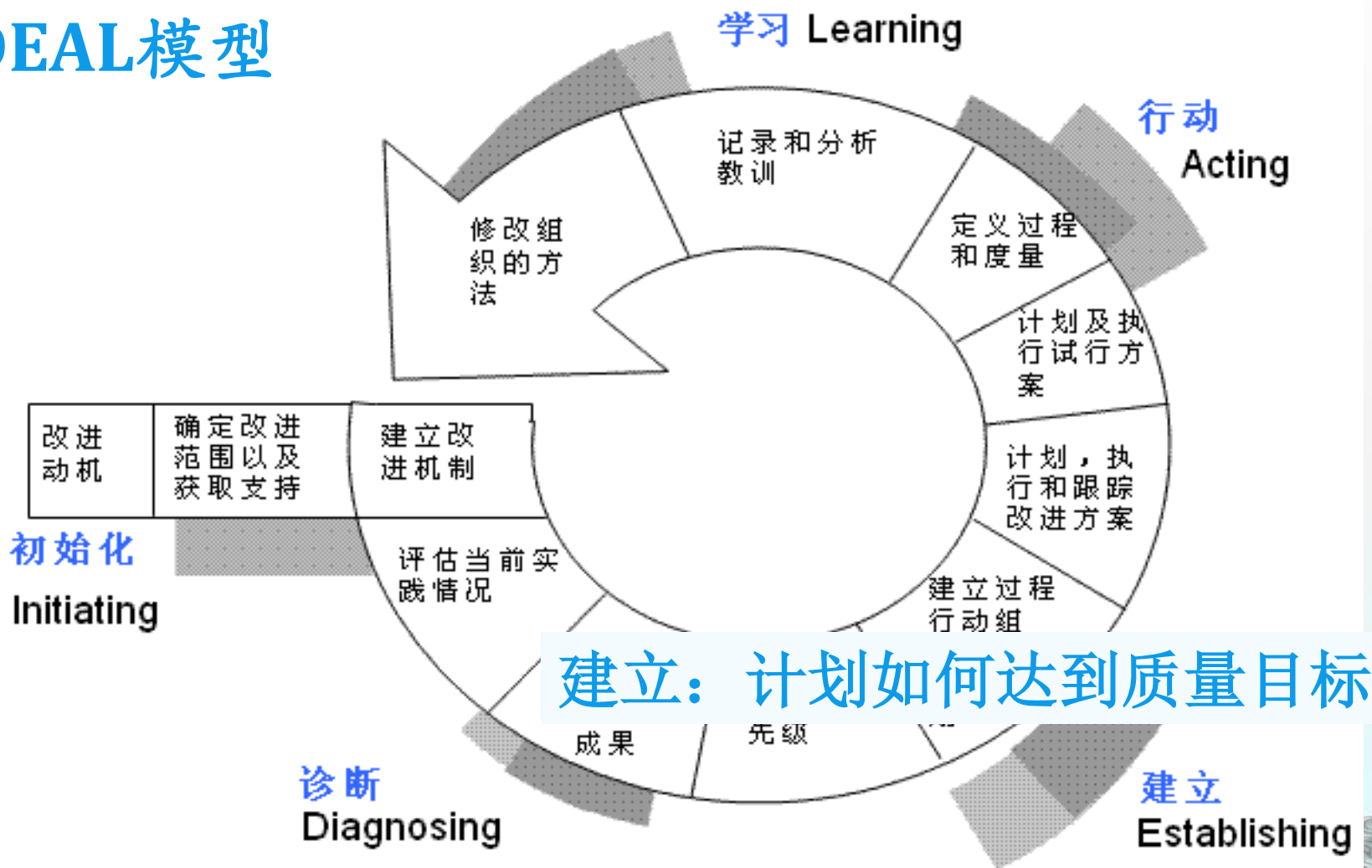
软件质量改进

IDEAL模型



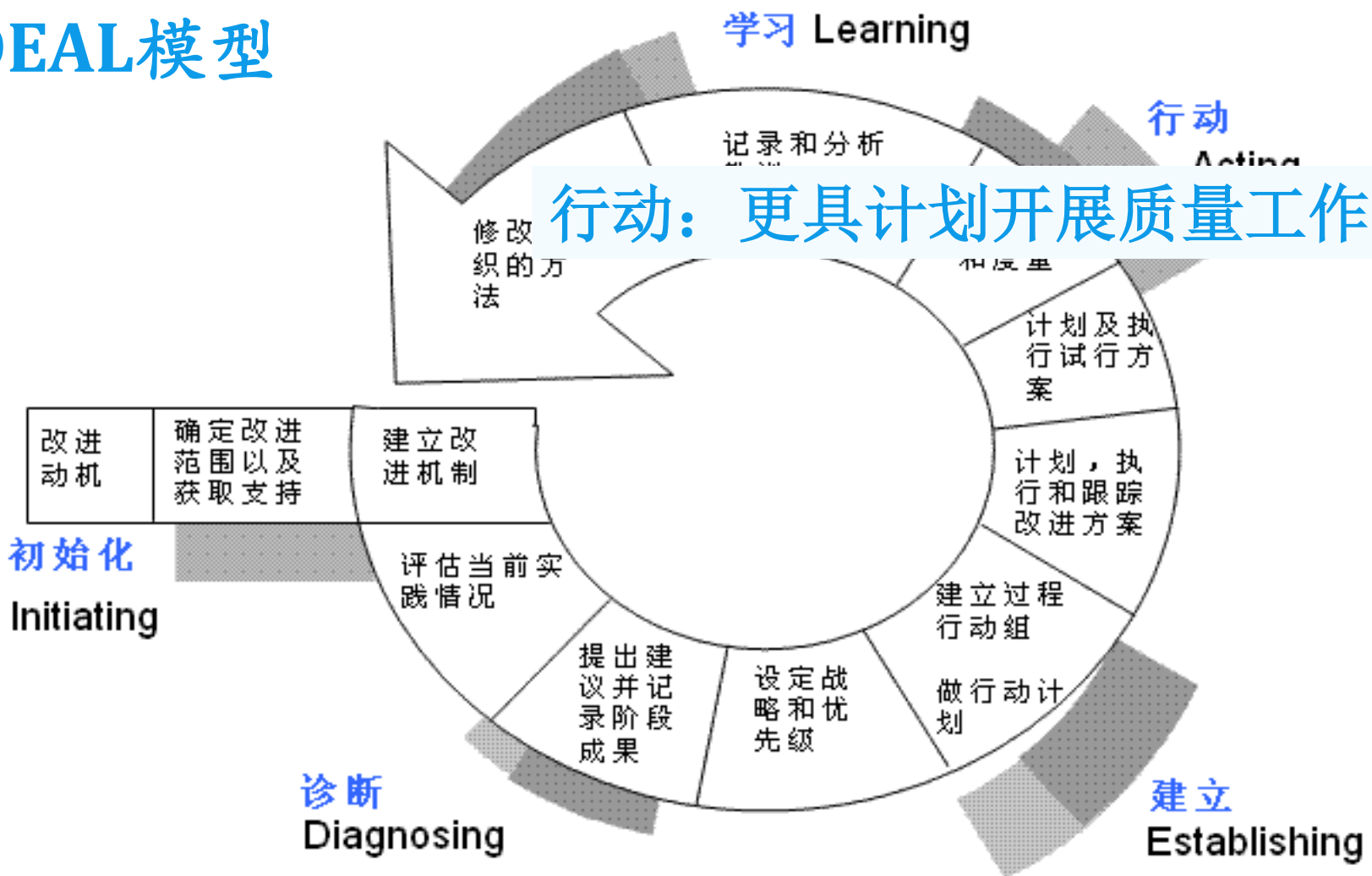
软件质量改进

IDEAL模型



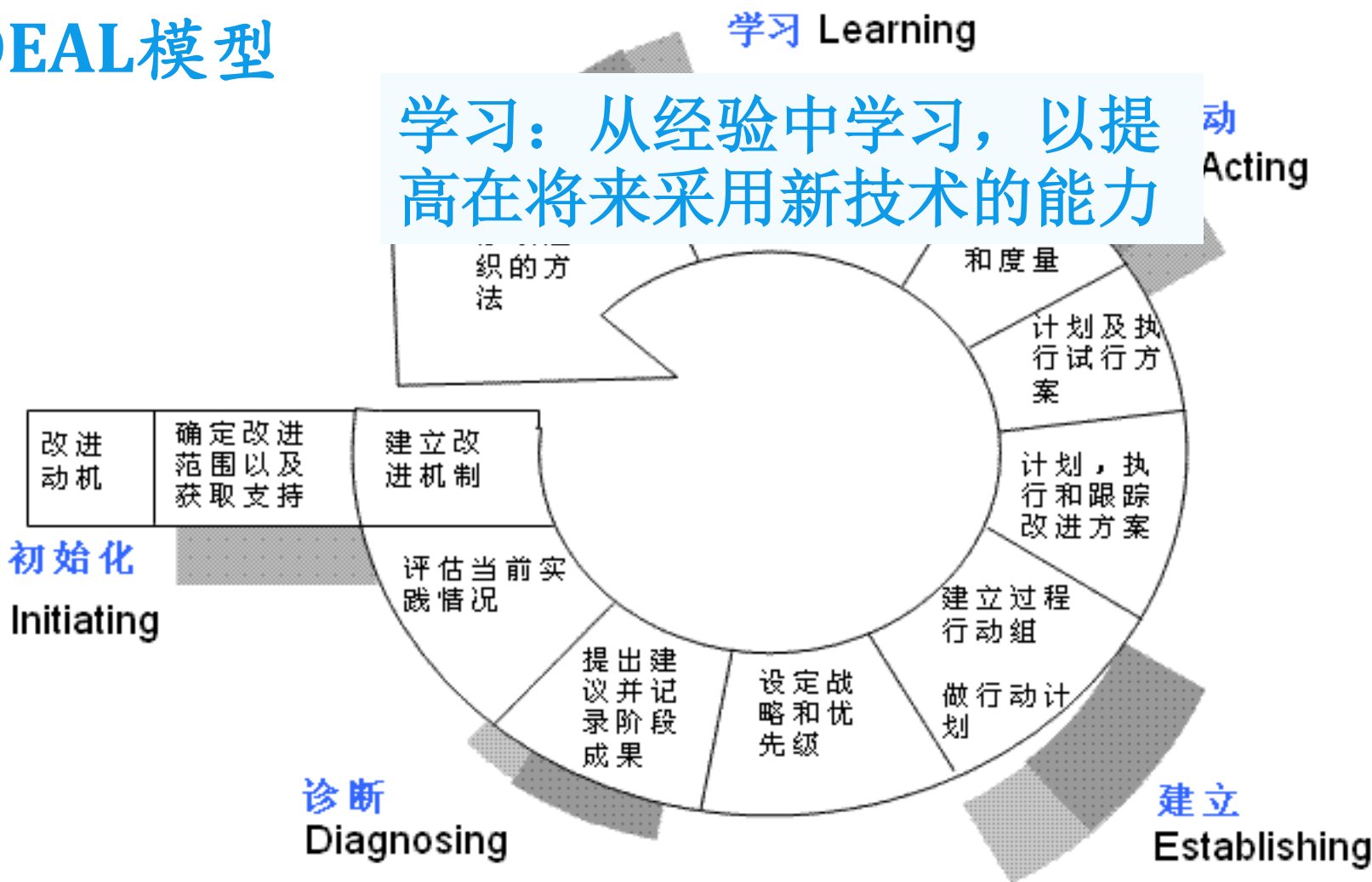
软件质量改进

IDEAL模型



软件质量改进

IDEAL模型



软件质量改进

六西格玛模型

- 在统计学中称为标准差，用来表示数据的分散程度。达到6西格玛的程度，就几近于完美地达成顾客要求，在一百万个机会里，只找得出3、4个瑕疵。
- 为了达到6西格玛，首先要制定标准，在管理中随时跟踪考核操作与标准的偏差，不断改进，最终达到6西格玛。
- 现已形成一套使每个环节不断改进的简单的流程模式：界定、测量、分析、改进、控制。



软件质量改进

质量改进步骤

➤ 界定

确定需要改进的目标及其进度，企业高层领导就是确定企业的策略目标，中层营运目标可能是提高制造部门的生产量，项目层的目标可能是减少次品和提高效率。界定前，需要辨析并绘制出流程。

➤ 测量

以灵活有效的衡量标准测量和权衡现存的系统与数据，了解现有质量水平。



软件质量改进

质量改进步骤

➤ 分析

利用统计学工具对整个系统进行分析，找到影响质量的少数几个关键因素。

➤ 改进

运用项目管理和其他管理工具，针对关键因素确立最佳改进方案。

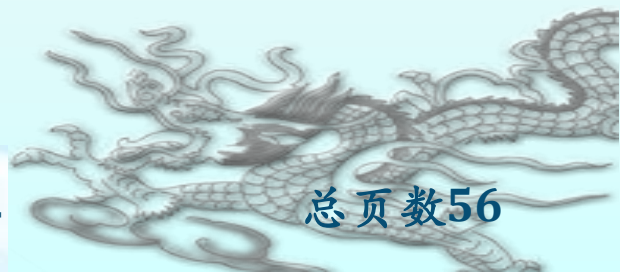
➤ 控制

监控新的系统流程，采取措施以维持改进的结果，以期整个流程充分发挥功效。



第三章 软件质量工程体系

- ◆ 1 概述
- ◆ 2 软件质量因素和指标
- ◆ 3 软件质量模型分析
- ◆ 4 软件质量工作层次
- ◆ 5 软件质量成本
 - 质量成本概念
 - 引入质量成本的意义
 - 劣质成本PONC和COPQ
 - 软件的劣质成本
- ◆ 6 软件质量标准和度量



质量成本概念

质量成本是为确保和保证满意的质量而发生的费用以及没有达到满意的质量所造成损失的总和，即包括保证费用和损失费用。

质量成本 = 保证成本 + 损失成本

保证成本 = 预防成本 + 评价成本



质量成本概念

➤ 预防成本

预防产生质量问题（软件缺陷）的费用，是企业的计划性支出，专门用来确保在软件产品交付和服务的各个环节不出现失误。

➤ 评价成本

交付和服务环节上，为评定软件产品或服务是否符合质量要求而进行的试验、软件测试和质量评估等所必需的支出。

➤ 损失成本

分为内部的和外部的：

- 内部损失成本指在软件发布之前发现质量问题，而要求重做、修改和问题分析所带来的成本；
- 外部损失成本指产品售出后因质量问题而产生的一切损失和费用等。



质量成本概念

引入质量成本的意义：

质量成本将质量与企业经济效益直接联系起来，质量得以货币语言来表达，质量语言和货币语言形成对话，从一个务虚的概念转换成一个务实的概念，使企业管理层对质量及其管理的意义和作用，有了新的认识，更容易树立质量至上的理念，进一步加大质量管理力度，使企业立于不败之地。这是质量成本对社会经济发展的重大贡献。



软件的劣质成本

➤ PONC(Price of Nonconformance)

- “不符合要求的代价”或称“劣质成本”。
- 指由于缺乏质量而造成的人力、财力、物力以及时间成本的浪费。
- PONC是在“零缺陷”质量管理中，为了更有效地衡量质量成本而引入的一个重要概念。

➤ COPQ(Cost of Poor Quality)

- “不良成本”或称“劣质成本”。
- 指所有由过程、产品和服务中的质量缺陷引起的费用。
- COPQ是“6西格玛”质量管理中的一个重要概念，用于有效地衡量质量成本、质量改进过程在经营效益上的表现。



软件的劣质成本

劣质成本的分类

➤ 故障成本

- 质量成本中的外部故障成本、内部故障成本;
- 采取返工、返修、纠正等补救措施所花费的成本。

➤ 过程成本

- 非增值成本(非增值的预防成本和鉴定成本);
- 低效率过程成本(如多余的操作、重复的作业等);
- 机会损失成本(指如果没有缺陷而就不会发生的费用等)。

➤ 损失成本

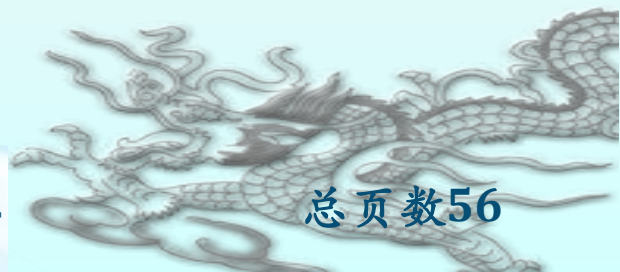
- 顾客损失成本(指给顾客所造成的各种额外的费用及负担);
- 信誉损失成本。

软件劣质成本竟高达46%--80%



第三章 软件质量工程体系

- ◆ 1 概述
- ◆ 2 软件质量因素和指标
- ◆ 3 软件质量模型分析
- ◆ 4 软件质量工作层次
- ◆ 5 软件质量成本
- ◆ 6 软件质量标准 and 度量
 - 软件质量标准体系介绍
 - 软件质量度量的地位



软件质量标准体系介绍

➤ 标准体系空间

- 纵向：分为产品和过程质量管理标准
- 横向：分为通用和各个行业的质量标准
- 范围：分为国际和国内标准

➤ 标准体系层次

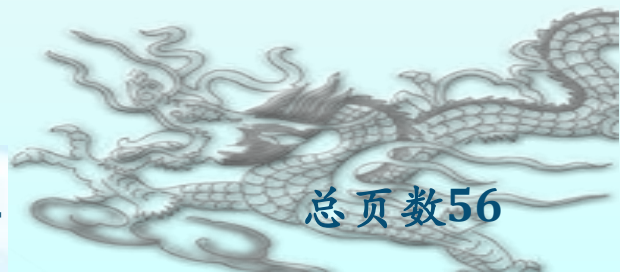
- 原理标准：描述各个原理级的关键组织标准。
- 要素标准：原理标准中的各个要素的详细性能要求的标准，必须执行。
- 指南和补充：为如何把原理或要素标准应用于特定场合而提供指导性的文件。



软件质量标准体系介绍

	原理	要素标准	指南
产品特性	9126-1	TR 9126-2/3/4	
		15026	
软件产品		12119	
产品文档		9127	18019
功能规格		14143-1/2	TR 14143-3/4/5

- 9126 软件产品质量的模型和度量
- 15026 系统和软件完整性级别
- 9127 客户软件包的用户文档和覆盖信息
- 12119 软件包质量需求和测试
- 14143 软件测量功能性规格测量



软件质量标准体系介绍

	原理	要素标准	指南
软件过程	12207/A MD1的过程结果	12207/14764	TR15271
		TR15846	ISO9000-3
		TR16326	TR9294
		15939	
		14598	
		15910	18019
系统过程	15288	15288 标准部分	15288 指南

- 12207 软件生存周期过程
- 14764 软件维护
- TR15846 软件配置管理
- TR16326 软件工程项目管理
- 15939 软件测试过程
- 14598 软件产品评价
- 15910 软件用户文档过程
- 15271 ISO/IEC 12207使用指南
- 15288 系统生存周期过程



软件质量度量的地位

➤ 项目质量度量

度量软件项目特征和项目执行的质量状态，
包括项目的资源使用效率、项目性能、项目风险等。

➤ 产品质量度量

度量软件产品的特性和质量属性，
包括软件产品的功能、复杂性、设计特征、性能和可靠性等。

➤ 过程质量度量

度量软件开发和维护的改进过程，
包括过程中某一时刻的状态（时间切面）、历史数据分析度量和未来变化预测的度量等。

采用定量软件工程，制定软件产品质量的度量准则，
可以提高软件开发过程管理的可视性，降低劣质成本，
提高软件产品的质量。



小结

- 现代软件质量工程体系继承了全面质量管理思想，要求组织中每个人都承担质量的责任，将质量控制、保证和改进的流程融于整个的软件开发生命周期；
- 软件质量工程体系，涵盖质量计划、质量风险管理、质量成本控制和质量计划的实施等内容。质量计划的制定又受质量文化影响、质量方针指导，通过对影响质量各种因素的分析，了解可能存在的质量风险，从而加以回避、控制。通过对软件产品、过程的测量和质量的度量，不断改进软件开发过程，以达到软件质量预先设定的目标。

