

# 第9章内容概要

- 软件项目管理概述
- 软件项目的成本与工作量估算
- 软件项目进度计划
- 软件项目中的人员组织
- 软件项目配置管理
- 软件项目风险管理
- ★ ■ 软件项目质量管理
- 能力成熟度模型

# 软件质量

- 软件质量----软件与明确地和隐含地定义的需求相一致的程度
- 软件质量是软件与明确地叙述的功能和性能需求、文档中明确描述的开发标准以及任何专业开发的软件产品都应该具有的隐含特性相一致的程度

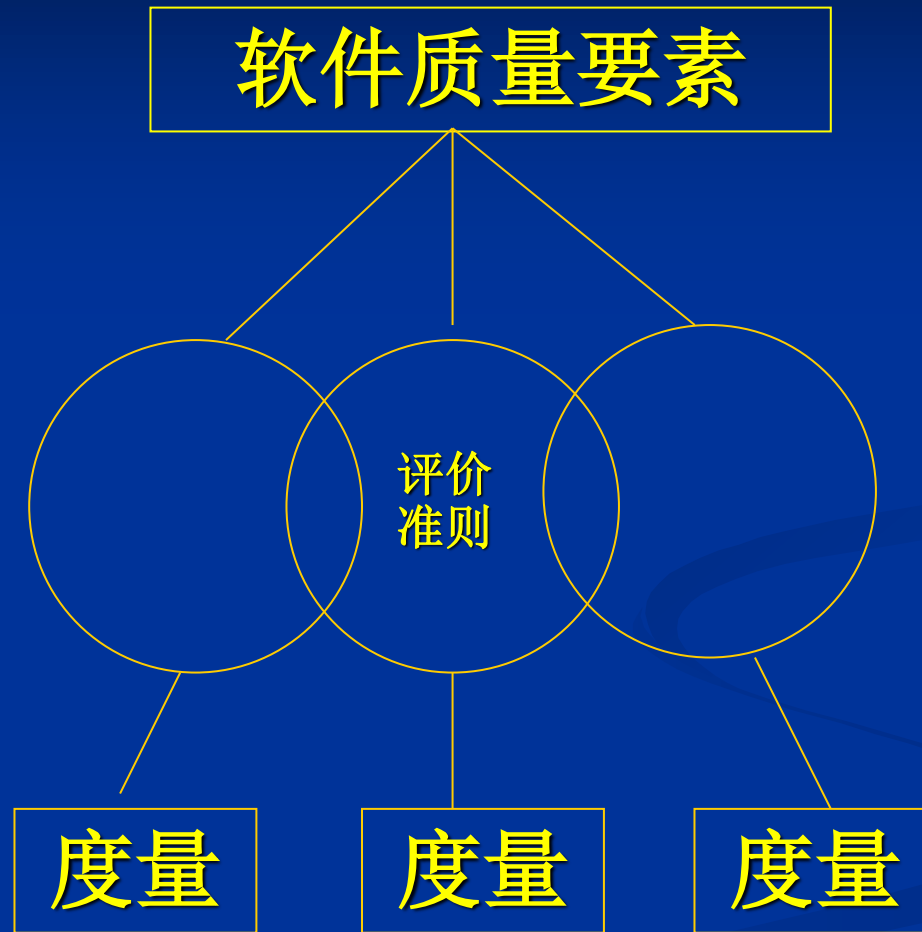
# 软件质量

- 上述定义强调的三个要点：
  - （1）软件需求是度量软件质量的基础
  - （2）指定的开发标准定义了一组指导软件开发的准则
  - （3）通常有一组没有显式描述的隐含需求

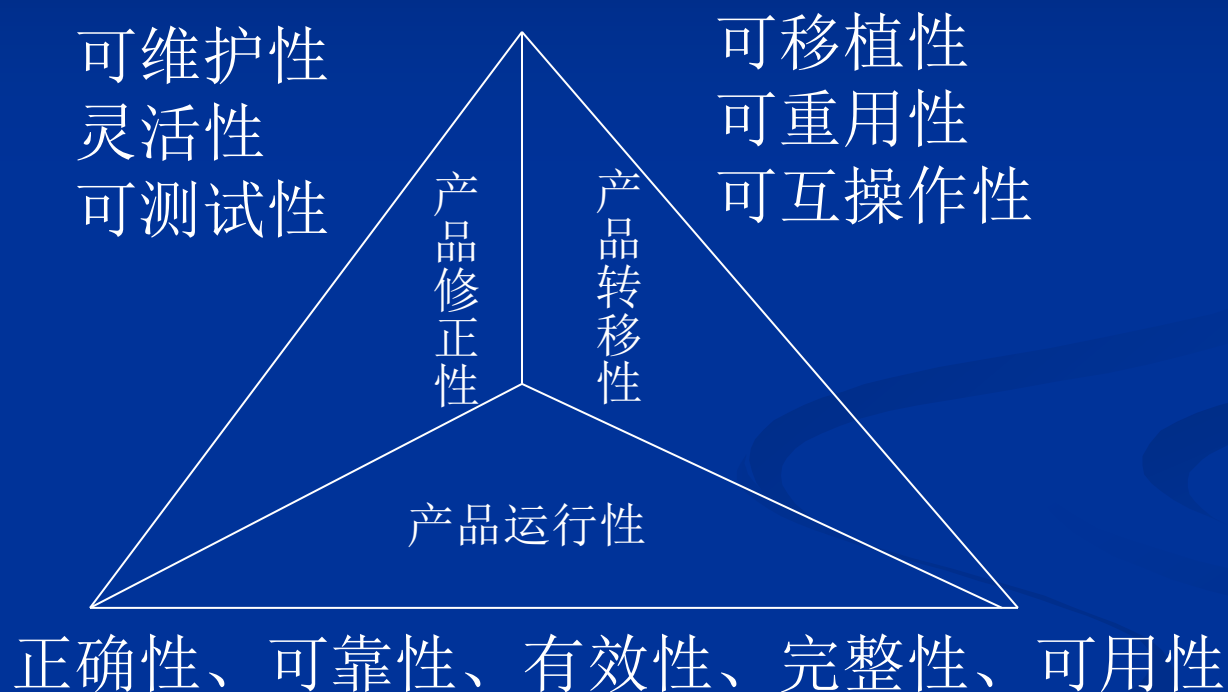
# 软件质量模型

- 现有的多种软件质量模型有一个共同点：把软件质量特性定义成分层模型，其中最基本的叫做基本质量特性，它可以由一些子质量特性定义和度量。
- 1976年 Boehm质量模型
- 1979年 McCall质量模型
- 1991年 ISO质量特性国际标准 (ISO/IEC9126)

# 软件质量模型



# 软件质量要素(McCall)



# 质量要素评价准则(McCall)

- 可审查性 (Auditability)
- 准确性 (Accuracy)
- 通信通用性 (Communication commonality)
- 完全性 (Completeness)
- 简明性 (Conciseness)
- 一致性 (Consistency)
- 数据通用性 (Data Commonality)
- 容错性 (Error-tolerance)

# 质量要素评价准则(McCall)

- 执行效率 (Execution Efficiency)
- 可扩充性 (Expandability)
- 通用性 (Generality)
- 硬件独立性 (Hardware Independence)
- 检测性 (Instrumentation)
- 模块化 (Modularity)
- 可操作性 (Operability)
- 安全性 (Security)



# 质量要素评价准则(McCall)

- 自文档化 (Self-documentation)
- 简单性 (Simplicity)
- 软件系统独立性 (Software System Independence)
- 可追踪性 (Traceability)
- 易培训性 (Training)

# 软件质量质量保证措施

- 基于非执行的测试（复审或评审）
  - 保证编码之前各阶段产生的文档的质量
- 基于执行的测试（软件测试）
  - 需要在程序编写之后进行，是保证程序质量的最后一道防线
- 程序正确性证明
  - 使用数学方法严格验证程序是否与它的说明完全一致

# 软件质量质量保证措施

## ■ 参加软件质量保证工作的人员

- 软件工程师：采用先进的技术方法和度量，进行正式的技术复审以及完成计划周密的软件测试
- SQA小组：辅助软件工程师以获得高质量的软件产品。主要活动是：计划、监督、记录、分析和报告，通过确保软件过程的质量来保证软件产品的质量。

# 软件质量质量保证措施

- 技术复审的必要性
- 走查
- 审查
- 程序正确性证明（参见讲义第7章最后一节）

# 走查 (Walkthrough)

- 4~6名成员，以走查规格说明为例，起草者，负责该规格说明的管理员，客户代表，设计组代表，SQA小组成员（组长）
- 走查材料发放，列出“不理解术语”和“不正确术语”
- 走查组长引导走查，标记错误而不改正错误，不超过2小时
  - 参与者驱动
  - 文档驱动

# 审查 (Inspection)

- 审查组由4人组成：管理员及技术负责人（组长）、当前被审查阶段的代表、下一阶段代表、SQA小组代表
- 比走查范围和步骤多
  - 综述-文档编写者向审查组综述该文档
  - 准备-评审员阅读文档，列出发现的错误类型，并按照发生频率将错误类型分级
  - 审查-评审组仔细走查整个文档
  - 返工
  - 跟踪