

软件协同设计课程之

3.2 软件质量



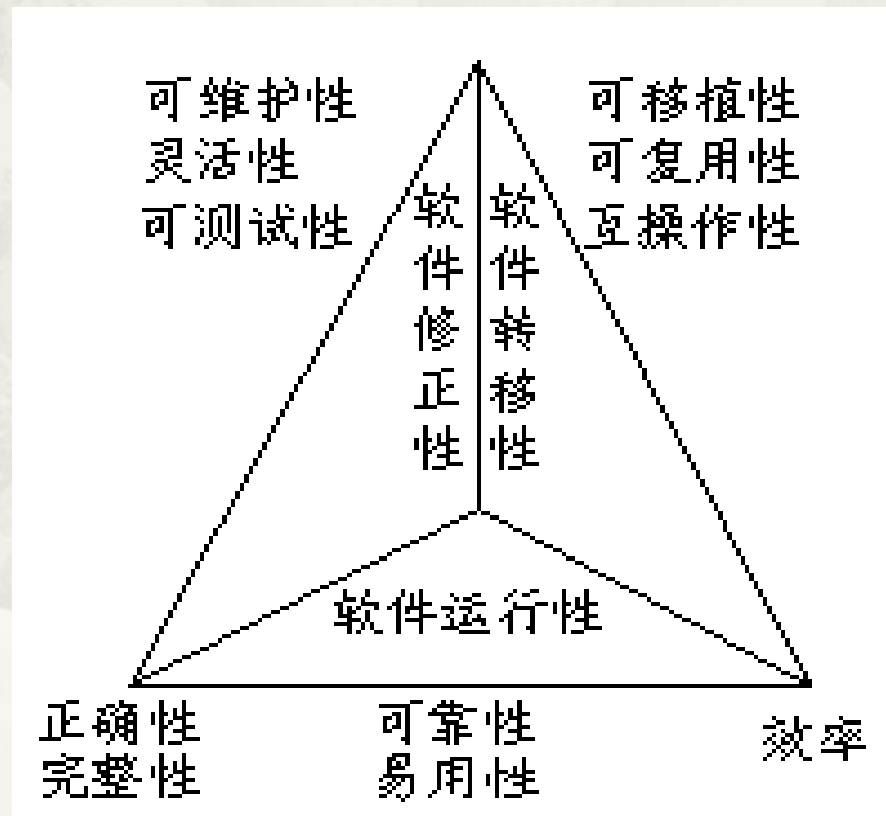
2025年1月

1.软件质量模型

- 课程目标之一：开发出高质量的产品
- 软件质量定义
 - GB/T25000.1-2010 《软件产品质量要求与评价指南》中将软件质量定义为：在规定条件下使用时，软件产品满足明确或隐含要求的能力。
- 典型的软件质量模型
 - McCall模型
 - ISO9126 软件质量模型是评价软件质量的国际标准，由6个特性和27个子特性组成。

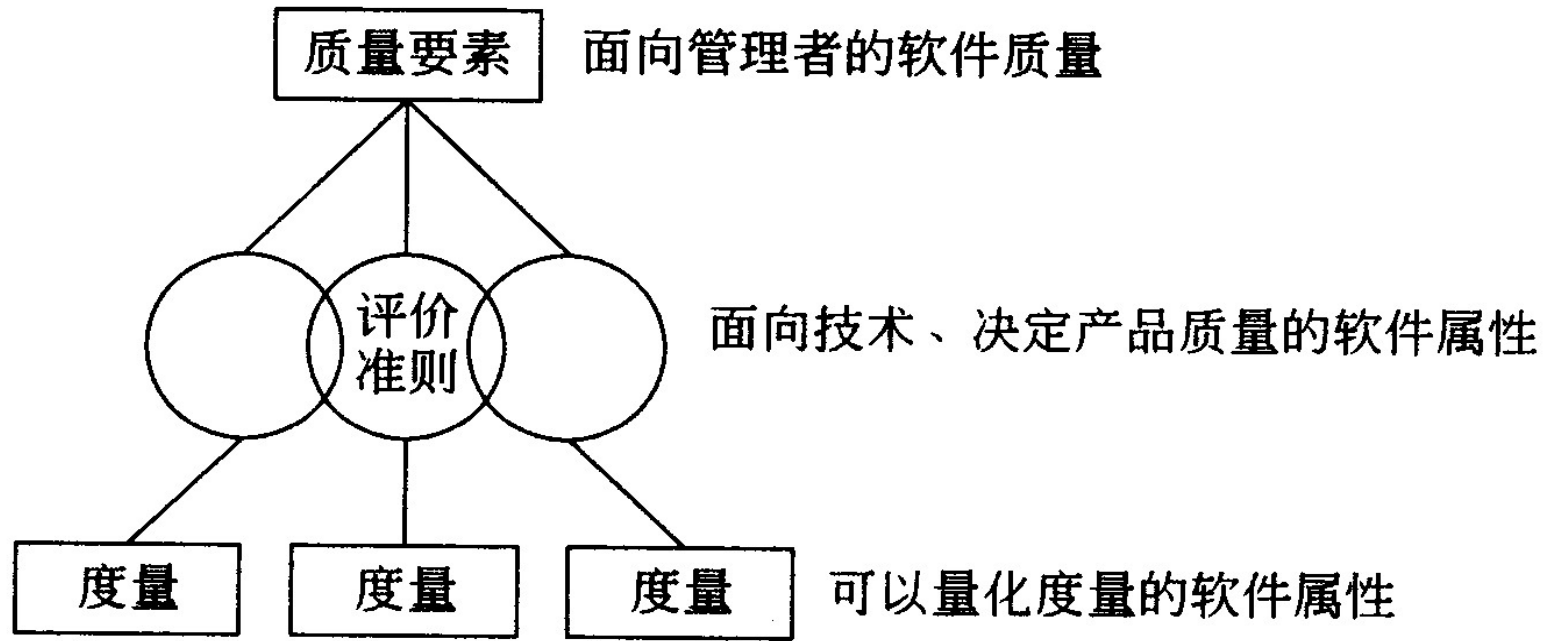
2. McCall模型

- 质量要素:又称为质量特征, 反映软件的质量, 如正确性、可靠性、效率等。
- 从软件产品的运行、修改和迁移三个方面分为11个软件质量要素



3. 软件质量要素评价准则

- 软件质量要素难以直接测量，因此需要为每个质量要素定义一组评价准则
- McCall定义了21种评价准则



4. 质量要素与评价准则的关系

[illegible]

5. 软件质量要素的评价准则-1

(1) 可审计性(auditability)

和标准的符合性、可被检查的容易程度。

(2) 准确性(accuracy)

计算和控制的准确度。

(3) 通信共性(communication commonality)

标准接口、协议和带宽的使用程度。

(4) 完备性(completeness)

所需功能完全实现的程度。

(5) 简洁性(conciseness)

以代码行数来评价的程序的简洁程度。

(6) 一致性(consistency)

在软件开发项目中一致的设计和文档技术的使用。

(7) 数据共性(data commonality)

在整个程序中对标准数据结构和类型的使用。

5. 软件质量要素的评价准则-2

(8) 容错性(error tolerance)

当程序遇到错误时所造成的损失。

(9) 执行效率(execution efficiency)

一个程序的运行性能。

(10) 可扩展性(expandability)

结构、数据或过程设计可被扩展的程度。

(11) 通用性(generality)

程序构件潜在的应用宽度。

(12) 硬件独立性(hardware independence)

软件独立于其运行于之上的硬件的程度。

(13) 自检测性(instrumentation)

程序监视它自身操作并且标识产生的错误的程度。

(14) 模块性(modularity)

程序部件的功能独立性。

5. 软件质量要素的评价准则-3

(15) 可操作性(operability)

程序操作的容易度。

(16) 安全性(security)

控制和保护程序和数据机制的可用度。

(17) 自文档性(self-documentation)

源代码提供有意义的文档程度。

(18) 简单性(simplicity)

一个程序可以没有困难地被理解的程度。

(19) 软件系统独立性(software system independence)

程序独立于非标准编程特性、操作系统特性和其他环境限制的程度。

(20) 可追踪性(traceability)

从一个设计表示或实际程序部件跟踪到需求的能力。

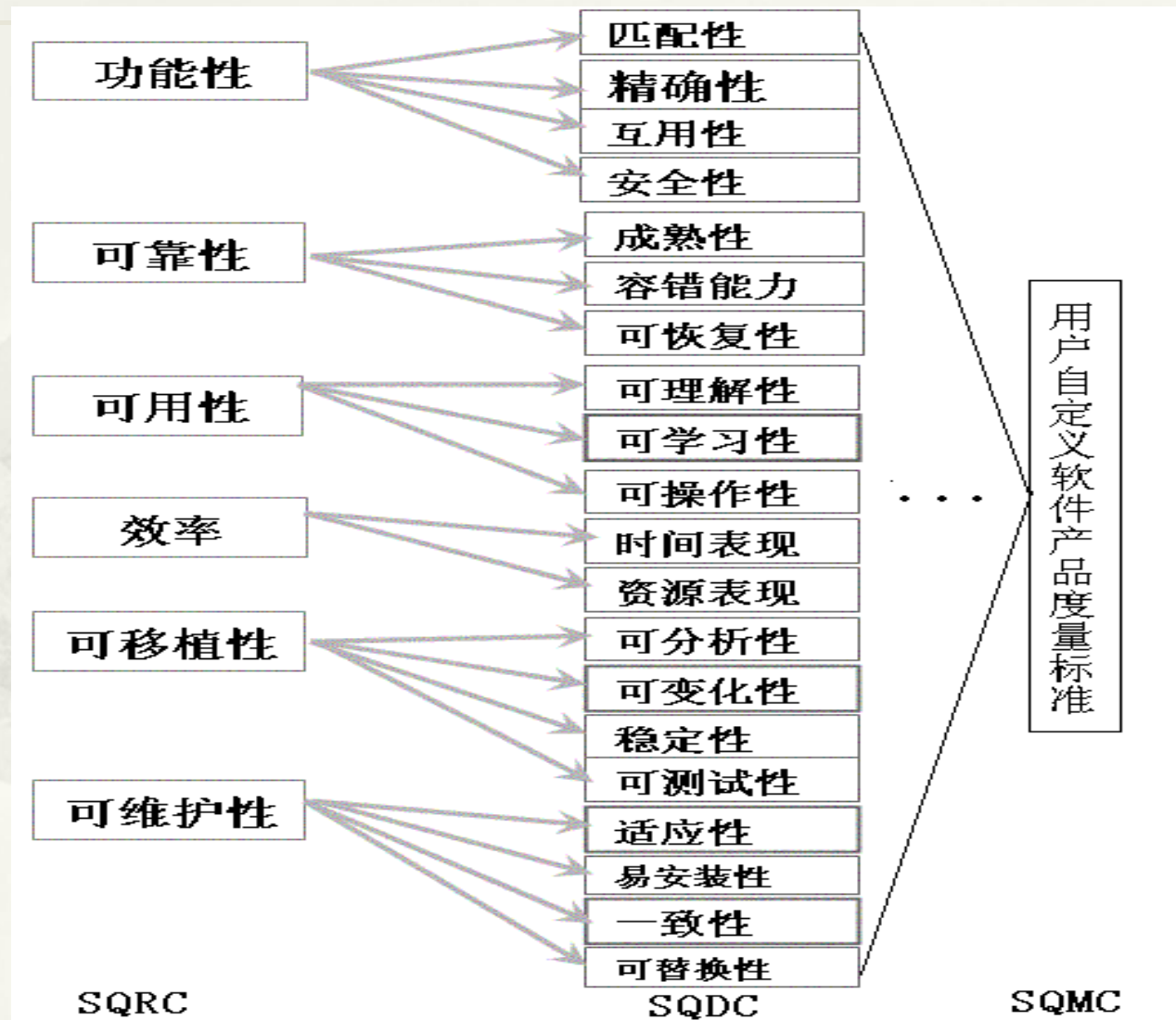
(21) 易培训性(training)

软件支持使得新用户使用系统的能力。

6. 软件质量特征 (ISO 9126)

- ❑ 功能：是满足明确或隐含的需求的那些功能。
- ❑ 可靠：在规定的一段时间和条件下，软件维持其性能水平的能力。
- ❑ 易用：由用户为使用软件所需作的努力和评价。
- ❑ 效率：在规定条件下，软件的性能水平与所使用资源量之间关系。
- ❑ 可维护：进行指定的修改所需的努力的大小。
- ❑ 可移植：软件从一个环境转移到另一个环境的能力。

ISO 9126软件质量三层模型



小结

本讲介绍了软件质量的概念、两种质量模型、以及质量模型的质量要素与评价准则