


# 第8章内容概要

- 软件维护的定义
- 软件维护的特点
- 软件维护过程
- ★ ■ 软件的可维护性
- 软件再工程过程

# 软件的可维护性

- 软件的可维护性：维护人员理解、改正、改动或改进这个软件的难易程度。
- IEEE Standard Glossary of Software Engineering Terminology, IEEE Std 610.12-1990, Published by the Institute of Electrical and Electronics Engineers, Inc., 345 East 47th Street, New York, NY 10017, USA.   
December 10, 1990.
  - Maintainability: (1) The ease with which a software system or component can be modified to correct faults, improve performance or other attributes, or adapt to a changed environment. (2) The ease with which a hardware system or component can be retained in, or restored to, a state in which it can perform its required functions.
- 决定软件可维护性的因素
- 文档
- 可维护性复审

# 决定软件可维护性的因素

- **可理解性:** 软件可理解性表现为外来读者理解软件的结构、功能、接口和内部处理过程的难易程度。
- **可测试性:** 诊断和测试的容易程度取决于软件容易理解的程度。良好的文档对诊断和测试是至关重要的，此外，软件结构、可用的测试工具和调试工具，以及以前设计的测试过程也都是非常重要的。
- **可修改性:** 软件容易修改的程度和设计原理和启发规则直接有关。
- **可移植性:** 软件可移植性指的是，把程序从一种计算环境（硬件配置和操作系统）转移到另一种计算环境的难易程度。
- **可重用性:** 所谓重用（reuse）是指同一事物不做修改或稍加改动就在不同环境中多次重复使用。

ISO 9126. Software Product Quality Characteristics, <http://www.cse.dcu.ie/essiscope/>

	<b>Analyzability</b>	<b>Attributes of software that bear on the effort needed for diagnosis of deficiencies or causes of failures, or for identification of parts to be modified.</b>
<b>Main-tainability</b>	<b>Changeability</b>	<b>Attributes of software that bear on the effort needed for modification, fault removal or for environmental change.</b>
	<b>Stability</b>	<b>Attributes of software that bear on the risk of unexpected effect of modifications.</b>
	<b>Testability</b>	<b>Attributes of software that bear on the effort needed for validating the modified software.</b>

# 文档

## ■ 总的来说，软件文档应该满足下述要求

- (1) 必须描述如何使用这个系统，没有这种描述时即使是最简单的系统也无法使用；
- (2) 必须描述怎样安装和管理这个系统；
- (3) 必须描述系统需求和设计；
- (4) 必须描述系统的实现和测试，以便使系统成为可维护的。



- 用户文档至少应该包括下述5方面的内容：
  - (1) 功能描述，说明系统能做什么；
  - (2) 安装文档，说明怎样安装这个系统以及怎样使系统适应特定的硬件配置；
  - (3) 使用手册，简要说明如何着手使用这个系统(应该通过丰富例子说明怎样使用常用的系统功能，还应该说明用户操作错误时怎样恢复和重新启动)；
  - (4) 参考手册，详尽描述用户可以使用的所有系统设施以及它们的使用方法，还应该解释系统可能产生的各种出错信息的含义(对参考手册最主要的要求是完整，因此通常使用形式化的描述技术)；
  - (5) 操作员指南(如果有系统操作员的话)，说明操作员应该如何处理使用中出现的各种情况。





- 所谓系统文档指从问题定义、需求说明到验收测试计划这样一系列和系统实现有关的文档。
- 描述系统设计、实现和测试的文档对于理解程序和维护程序极端重要
- 从概貌到每个方面每个特点，从抽象到具体，有逻辑地介绍系统

# 可维护性复审



- 在需求分析阶段的复审过程中，应该对将来要改进的部分和可能会修改的部分加以注意并指明；应该讨论软件的可移植性问题，并且考虑可能影响软件维护的系统界面。
- 在正式的和非正式的设计复审期间，应该从容易修改、模块化和功能独立的目标出发，评价软件的结构和过程；设计中应该对将来可能修改的部分预作准备。



# 可维护性复审

- 代码复审应该强调编码风格和内部说明文档这两个影响可维护性的因素。
- 设计和编码过程中尽量使用可重用的软件构件。
- 在测试结束时进行最正式的可维护性复审，这个复审称为配置复审。配置复审的目的是保证软件配置的所有成分是完整的、一致的和可理解的，而且为了便于修改和管理已经编目归档了。
- 在完成了每项维护工作之后，都应该对软件维护本身进行仔细认真的复审。