目录

8.1 维护的基本概念

目的

- 交付后品质的早期安定化
- 本管理的结果的可以回馈到下期的开发过程，提高下期开发的品质
- 提高故障对应能力以提高顾客满意度
  迅速地，正确地对应

8.1 维护的基本概念

- 维护是重要的

影响顾客满意度

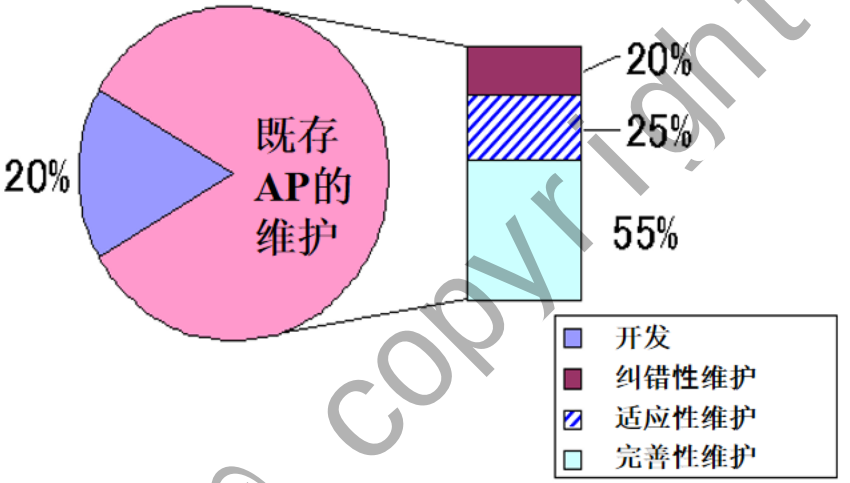
→

必须迅速地正确地对应用户的需要

维护的种类

- （1）纠错性维护。
  由于前期的测试不可能揭露软件系统中所有潜在的错误，用户在使用软件时仍将会遇到错误，诊断和改正这些错误的过程称为纠错性维护。
- （2）适应性维护。
  由于新的硬件设备不断推出，操作系统和编译系统也不断地升级，为了使软件能适应新的环境而引起的程序修改和扩充活动称为适应性维护。
- （3）完善性维护。
  在软件的正常使用过程中，用户还会不断提出新的需求。为了满足用户新的需求而增加软件功能的活动称为完善性维护。
- （4）预防性维护

维护的种类



完善性维护的比例最高，次之是适应性维护
预防性维护的量最少

8.2 软件维护的特点

- 维护的难点

  软件维护是人们对既丰富多彩又会令人心酸的活动的统称。

  其中丰富多彩的活动是指那些反映客观世界变化、能使软件系统更加完善的修改和扩充工作。

  令人心酸的活动是指那些永无修止、并且改了旧错却引起新错让人欲哭无泪的工作。

维护的难点

- （1）软件人员经常流动，当需要对某些程序进行维护时，可能已找不到原来的开发人员。只好让新手去“攻读”那些程序。
- （2）人们一般难以读懂他人的程序。在勉强接受这类任务时，心里不免嘀咕：“我又不是他肚子里的虫子，怎么知道他如何编程。”
- （3）当没有文档或者文档很差时，你简直不知道如何下手。
- （4）很多程序在设计时没有考虑到将来要改动，程序之间相互交织，触一而牵百。即使有很好的文档，你也不敢轻举妄动，否则你有可能陷进错误堆里。
- （5）如果软件发行了多个版本，要追踪软件的演化非常困难。
- （6）维护将会产生不良的副作用，不论是修改代码、数据或文档，都有可能产生新的错误。
- （7）维护工作毫无吸引力。高水平的程序员自然不愿主动去做，而公司也舍不得让高水平的程序员去做。带着低沉情绪的低水平的程序员只会把维护工作搞得一塌糊涂。

8.3 纠错性维护的步骤和方法（1）

在进行维护时的难点和问题点

理解是困难的、变更是危险的

- 维护要求的正确的文档化技术不足
  →什么样的文档是易于维护的
- 不能充分获得分析故障原因情报
- 把握变更的影响范围是很复杂的
- 确定变更后的确认范围是很困难的

修正**bug**的时机随着各个阶段的后移、工数，费用的花费会越来越大。
→
所以**bug**在前面的阶段就被摘出是十分重要的。

8.3 纠错性维护的步骤和方法（2）

故障发生后

- 确认故障问题点
  - 发生时日
  - 该当制品名、版本
  - OS版本
  - 现象的内容
    发生时的现象、事前的操作内容、发生后的操作（确认是否有log的写入）、再现性的有无
  - 确认其他运行的SW
  - ＨＷ的器械名称（打印机名、网络）
  - ＬＯＧ文件（event log、其它log）

8.3 纠错性维护的步骤和方法（3）

对应方法

■作业内容

- 过去履历的确认･･･ 检索同样的现象、确认是否已经回答过了
- 再现测试 ･･･ 是否发生同样现象
- **dump**解析、**log**解析
- 用户的失误、其它程序的问题的排除
- 代码解析(找不到原因、无法再现时的手段)
- 回避策略的研究

■必要的资料

- **dump**、**log**、设计书、**list**
- 用户的应用程序、源代码

8.3 纠错性维护的步骤和方法（4）

修改时的注意点

(1) 设计思想的一致性
(2) 修改的处所尽可能的少，不要比需要修改的地方多
(3) 尽量不使用共享系统中的已有变量，而使用局部量
(4) 不要建立公用子程序，而建立各自独立的子程序
(5) 坚持修改后的复审
(6) 建立修改文档
(7) 注意修改相应的文档

8.4 软件的可维护性

- 可理解性
- 可测试性
- 可修改性
- 可移植性
- 可重用性

可读性，文档，设计的时候考虑将来功能的扩展

8.5 维护的管理

- 由用户或专门人员提供问题报告：

  登记号，软件名称，编号，版本号码，报告人姓名，单位，报告时间，问题描述等。

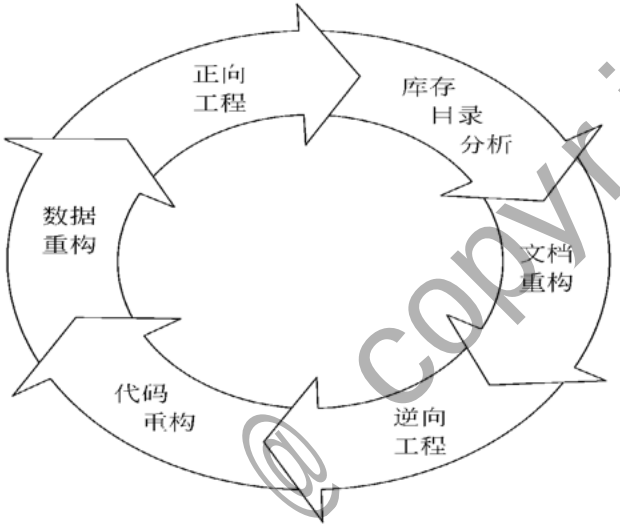
- 维护人员提供修改后报告：

  登记号，修改日期，修改人名，修改内容，测试情况，修改工数等。

8.6 软件再工程过程
(Software Reengineering)

- 再工程是一个重构活动（类比 重建一所房子）
  - 开始重建前，首先检查一下房子。确定它是否确实需要重建？
  - 在拆掉并重建房子前，确定其结构是否牢固。若结构良好，则可能是“改造”。
  - 在开始重建前，确保你已经了解房子最初是如何建造的。（墙内部，了解布线、管道以及内部结构。）
  - 如果开始重建，应该使用最现代的，耐久的材料。
  - 如果决定重建，一定要采用严格的方式，使用现在及将来都将获得高质量的做法。

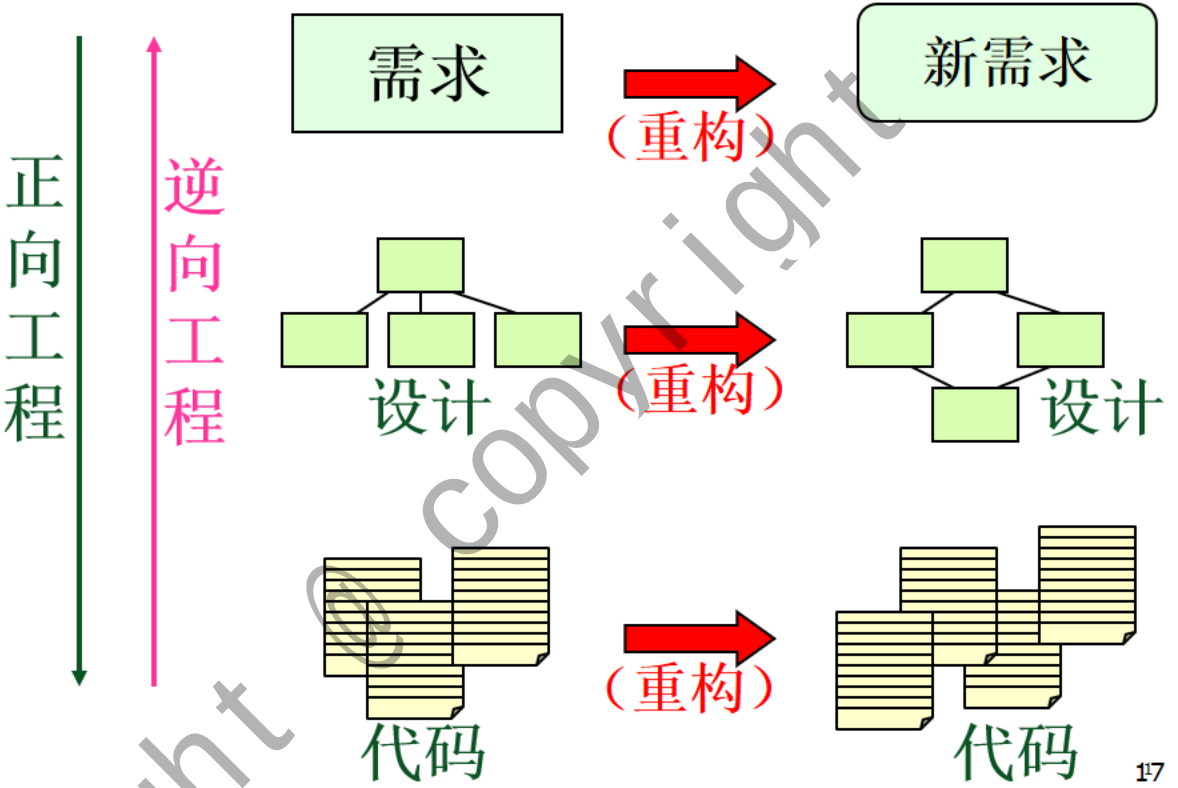
8.6 软件再工程过程（Software Reengineering）



软件再工程过程模型

软件再工程是一类软件工程活动，是一个工程过程，它将逆向工程、重构和正向工程组合起来，将现存系统重新构造为新的形式。

软件再工程过程示意图



软件再工程过程模型所定义的6类活动

- 库存目录分析
- 文档重构
- 逆向工程
- 代码重构
- 数据重构
- 正向工程

本章小结

- 软件维护的4类活动
  （改正性、适应性、完善性、预防性）
- 决定软件可维护性的基本要素
  （可理解、可测试、可修改、可移植和可重用性）
- 文档是影响软件可维护性的决定因素
- 软件再工程（预防性维护）

练习题

- 判断题：

  各类软件维护活动，一般来说，纠错性维护占整个维护工作的比重最大。（ X ）

- 现有纠错性维护作业中，修改代码时的注意点有哪些？

END.

谢谢！