

软件工程技术基础
——流程与方法

第一章 软件工程概述

北京邮电大学 网络空间安全学院
芦效峰

1

1

《软件工程技术基础》学习目的

掌握软件开发方法论

体验软件工程各阶段的主要工作，特别注意吸取教训；

- 学会与他人合作，培养团队精神。
- 成为一名项目经理/产品经理的必修课

2

2

授课内容

- 软件工程概述
- 可行性研究
- 需求分析 (UML)
- 概要设计 (UML)
- 详细设计
- *人机交互
- *编码规范
- 软件测试
- 软件维护

3

3

主要学习内容：

- 软件
- 软件工程
- 软件生存周期
- 常用软件开发过程模型
- 软件开发方法简述
- 软件文档

4

4

1 软件

1.1 软件的概念及特点

- 从1946年第一台计算机诞生，计算机已经进入各个领域、各个行业，进入每个家庭，成为现代社会必不可少的工具。
- 计算机强大功能主要依赖于**计算机软件**



5

5

计算机程序

- 计算机的工作是用程序、软件来控制的
- IEEE定义：软件是计算机程序、处理和与计算机系统运行相关的文档和数据。
- **程序是按事先设计的功能和性能要求执行的指令序列**
- **数据是程序运行的基础和操作的对象**
- **文档是与程序开发，维护和使用有关的图文材料**

6

6

2 软件的特点

- **软件是一种逻辑实体，而不是具体的物理实体。具有抽象性，不可描述性**
- **在软件的运行和使用期间，没有硬件那样的机械磨损，老化问题。但存在退化问题。（用户体验）**
- **软件的生产与硬件不同，在它的开发过程中没有明显的制造过程，而是研制。是程序员脑力劳动的结果。**
是否和其他行业一样需要工匠精神？如何体现？

7

7

- 可变性，软件在生产过程中甚至在投入运行之后，也还可以再改变。
- 软件必须和运行软件的硬件保持一致（当前这个特点变的突出了！）
- 软件的运行受计算机系统的影响
- 软件一旦研制开发成功，其生产过程就变成复制过程。

8

8

软件开发成本昂贵

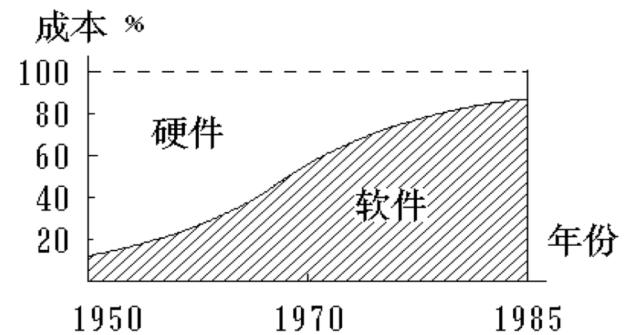
- 软件的开发至今尚未完全摆脱手工艺的开发方式。软件成本相当昂贵，软件开发需要投入大量的、高强度的脑力劳动，人员成本很高，开发人员的工资大于运维等部门。

两面性

- 原因1：软件本身是复杂的
 - 实际问题的复杂性
 - 程序逻辑结构的复杂性
- 原因2：人际关系等非技术原因更复杂
- 软件成本超过硬件

9

9



10

软件的分类型

1. 按软件的功能进行划分：

- 系统软件 IAAS
- 支持软件 PAAS
- 应用软件 SAAS

11

11

软件的分类型

• 系统软件

控制和协调计算机及外部设备，支持应用软件开发和运行的系统

- 操作系统
- 数据库管理系统
- 设备驱动程序
- 通信处理程序等

12

12

· 支持软件

支撑各种软件的开发与维护的软件，
又称为软件开发环境

- 文本编辑程序
- 文件格式化程序
- 程序库系统
- 支持需求分析、设计、实现、测试和支持管理的软件

13

13

· 应用软件

满足用户不同领域、不同问题的应用

- 商业数据处理软件
- 工程与科学计算软件
- 计算机辅助设计 / 制造软件
- 系统仿真软件
- 智能产品嵌入软件
- 医疗、制药软件
- 事务管理、办公自动化软件
- 计算机辅助教学软件

14

14

2. 基于软件规模进行划分

类别	参加人员数	研制期限	源程序行数
微型	1	1 ~ 4周	0.5k
小型	1	1 ~ 6月	1k ~ 2k
中型	2 ~ 5	1 ~ 2年	5k ~ 50k
大型	5 ~ 20	2 ~ 3年	50k ~ 100k
甚大型	100 ~ 1000	4 ~ 5年	1M(=1000k)
极大型	2000 ~ 5000	5 ~ 10年	1M ~ 10M

15

15

3. 基于软件工作方式划分：

- 实时处理软件
- 分时软件
- 交互式软件
- 批处理软件

16

16

· 按软件服务对象的范围划分：

- **项目软件**
- **产品软件**

· 按使用的频度进行划分：

- **一次使用**
- **频繁使用**

· 按软件失效的影响进行划分：

- **高可靠性软件**
- **一般可靠性软件**

17

17

软件的种类如此众多，软件的功能这么强大，软件是如何发展起来的呢？从最初的软件到现在的巨型系统软件，软件的发展是不是一帆风顺呢？

18

18

3 软件的发展

- 第一阶段，程序设计阶段，从1946年到20世纪60年代初。
- 第二阶段，程序系统阶段，从20世纪60年代初到20世纪70年代初。程序的规模已经发展的很大，需要多人分工协作，小组模式。
- 第三阶段，软件工程阶段，从20世纪70年代中期至80年代中期。在这个阶段，软件工程师把工程化的思想加入到软件的开发过程中。
- 第四阶段，面向对象阶段，80年代中期至今

19

19

软件神化

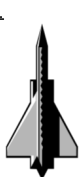
- 20 世纪60年代以前，计算机刚刚投入实际使用，软件设计往往只是为了一个特定的应用而在指定的计算机上设计，基本上个人设计、个人使用。
- 60年代中期，大容量、高速度计算机的出现，使计算机的应用范围迅速扩大，软件开发急剧增长。高级语言开始出现；操作系统的发展引起了计算机应用方式的变化。软件系统的规模越来越大，复杂度越来越高，软件可靠性问题也越来越突出。
- 原来的个人设计、个人使用的方式不再能满足要求，在计算机软件的开发和维护过程中遇到的一系列严重的问题。

20

20

1963年美国飞往火星的火箭爆炸，造成**1000万**美元的损失。原因是FORTRAN程序：

● `DO 5 I=1, 3`
误写为：`DO 5 I=1, 3.`



● 1967年苏联“联盟一号”载人宇宙飞船在返航时，由于软件忽略一个小数点，在进入大气层时因打不开降落伞而烧毁。



21

21

• IBM公司的 OS/360，共约100万条指令，花费了5000个人年；经费达数亿美元，而结果却令人沮丧，错误多达2000个以上，系统根本无法正常运行。OS/360系统的负责人Brooks这样描述开发过程的困难和混乱：“...像巨兽在泥潭中作垂死挣扎，挣扎得越猛，泥浆就沾得越多，最后没有一个野兽能够逃脱淹没在泥潭中的命运。”

22

22

60年代后期软件开发出现的问题

- ① 软件开发费用和进度失控。费用超支、进度拖延的情况屡屡发生。
- ② 软件的可靠性差。尽管耗费了大量的人力物力，而系统的正确性却越来越难以保证，出错率大大增加，造成的损失惊人。
- ③ 生产出来的软件难以维护。很多程序缺乏相应的文档资料，程序中的错误难以定位，难以改正，有时改正了已有的错误又引入新的错误
- ④ 用户对“已完成”的系统不满意现象经常发生。

软件危机！！

23

23

产生软件危机的原因分析：软件的特点

在软件开发和维护的过程中存在这么多严重问题，一方面与软件本身的特点有关，另一方面也和软件开发与维护的方法不正确有关。

1、软件不同于硬件：它是计算机系统逻辑部件，在写出程序代码并在计算机上运行之前，软件开发过程的进展情况较难衡量，软件开发的质量也较难评价，因此管理和控制软件开发过程相当困难。

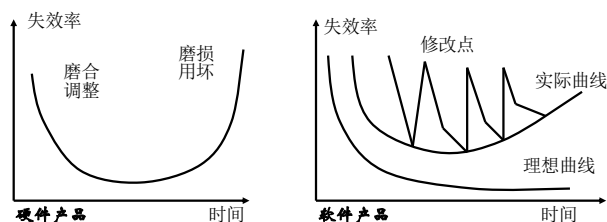
2、软件在运行过程中不会因使用时间过长而被用坏，如果运行中发现错误，绝不会是使用造成的，一定是遇到了一个在开发时期引入并且在测试阶段没能检测出来的故障，因此软件维护通常意味着改正或修改原来的设计。

24

24

3、软件不同于一般程序：规模庞大，必须有严格而科学的管理。孙悟空如果缺乏唐僧的管理，什么结果？

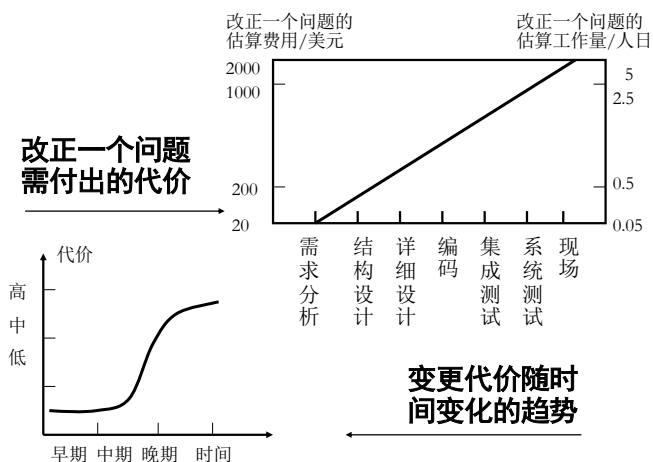
4、软件专业人员对软件开发和维护有不少的糊涂观念，在实践过程中或多或少地采用了错误的方法和技术，忽视软件需求分析的重要性，认为开发软件就是写程序并使之运行，轻视软件维护，这可能是软件危机的主要原因。



硬件/软件产品失效率曲线

25

25



26

26

软件危机的原因

- (1) 没有统一的、规范的方法论的指导。
文件资料不全，忽视交流，缺乏方法论的指导
- (2) 忽视软件开发前期的需求分析。
编程越早，需求分析不充分，完成时间越长
- (3) 忽视软件文档也是造成开发效率低下的原因。
对软件开发、维护、用户都很重要
- (4) 忽视测试阶段的工作，或不负责任的测试员，提交用户的软件质量差。
“错误是不可避免的”，该如何理解这句话？
- (5) 轻视软件的维护。
- (6) 开发计划不合理，缺乏经验和数据积累。

27

27

2.解决软件危机的途径达成共识

- 1、软件开发应该是一种组织良好、管理严密、各类人员协同配合、共同完成的工程项目。
- 2、必须充分吸收和借鉴人类长期以来从事各种工程项目所积累的行之有效的原理、概念、技术和方法，特别是要吸取几十年来从事计算机硬件研究和开发的经验教训。
- 3、应该推广使用在实践中总结出来的开发软件的成功的技术和方法。
- 4、应该开发和使用更好的软件工具。

28

28

总之，为了解决软件危机，既要有技术措施（方法和工具），又要有必要的组织管理措施。

- 技术措施
- 管理措施

按工程化的原则和方法组织软件开发工作是有用的，是摆脱软件危机的一个主要出路。软件工程正是从管理和技术两方面研究如何更好地开发和维护计算机软件的学科。

29

29

2 软件工程

1968 年北大西洋公约组织 的计算机科学家在联邦德国召开国际会议，第一次讨论软件危机问题，并正式提出“软件工程”一词，从此一门新兴的工程学科——软件工程学——为研究和克服软件危机应运而生。

30

30

2.1 软件工程的定义和内容

1983年IEEE给软件工程的定义：“软件工程是开发、运行、维护和修复软件的系统方法。” 强调“系统方法”不是“个人技巧”。

专家观点：

Fairly认为：“软件工程学是为了在成本限额以内按时完成开发和修复软件产品所需要的系统生产和维护技术及管理学科”。强调“成本限额以内”及“技术和管理两个方面”。

Fritz Bauer认为：“软件工程是为了经济地获得可靠的且能在实际机器上有效地运行的软件，而建立和使用的完善的工程化原则”。强调“经济地开发出高质量的软件，应建立和使用的完善的工程化原则”。

31

31

1993年IEEE进一步给出了一个更全面的定义：

软件工程是：（1）把系统化的、规范的、可度量的途径应用于软件开发、运行和维护的过程，也就是把工程化应用于软件中；（2）研究（1）中提到的途径。

32

32

2004年由IEEE/ACM联合发布的Software Engineering2004报告强调了对软件工程的新定义，即软件工程是：以系统的、学科的、定量的途径，把工程应用于软件的开发、运营和维护；同时，开展对上述过程中各种方法和途径的研究。

33

33

软件工程是指导计算机软件开发和维护的工程学科.采用工程的概念、原理、技术和方法来开发和维护软件,把经过时间考验而证明正确的管理技术和当前能够得到的最好的技术方法结合起来,以经济地开发出高质量的软件并有效地维护它，这就是软件工程。

34

34

软件工程的内容

软件工程是一种层次化的技术，如图1-1所示。和其他工程方法一样，软件工程是以质量为关注焦点，以相关的现代化管理为理念。

软件工程的基础是过程层。

软件工程的过程是为获得软件产品，在软件工具支持下由软件人员完成的一系列软件工程的的活动，贯穿于软件开发的各个环节。它定义了方法使用的顺序、要求交付的文档资料，是软件开发各个阶段完成的标志。

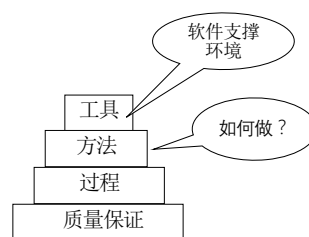


图1-1 软件工程层次图

35

35

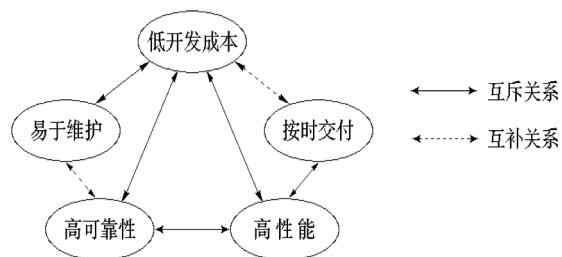
软件工程的目标

- 付出较低的开发成本；
- 达到要求的软件功能；
- 取得较好的软件性能；
- 开发的软件易于移植；
- 需要较低的维护费用；
- 能按时完成开发工作，及时交付使用。

36

36

目标之间的关系



37

37

2.2 工程化思想

- 到底什么是工程化呢？
- 类比我们较熟悉的建筑工程，盖一个大楼
- IEEE定义：工程化是把系统的、受训到、可以计量的方法应用到结构、机械、产品、系统或处理过程。
- 软件开发工程化的目标是通过规范软件开发的步骤和每一步的要求，使软件开发像在流水线上生产机械零件一样

38

38

2.3 软件工程的基本原理

1. 用分阶段的生存周期计划严格管理开发过程。
软件开发周期划分成若干阶段，并相应地制定出切实可行的计划
2. 坚持进行阶段评审。
因此，软件的质量保证工作不能等到编码阶段结束之后再进行。
3. 实行严格的产品控制。
实行基准配置管理（又称变更控制），当需求变更时，其他各个阶段的文档或代码都要随之相应变化，以保证软件的一致性。

39

39

软件工程的基本原理

4. 采用现代程序设计技术。
5. 结果应能清楚地审查
明确地规定开发小组的责任和产品标准。
6. 开发小组的人员应少而精。
减少交流时间
7. 承认不断改进软件工程实践的必要性。

40

40