

1 软件工程导论

- 软件工程的产生与发展

- 软件的发展过程

- 软件与程序的区别

- 程序

- 程序是为实现特定目标或解决特定问题而用计算机语言编写的命令序列的集合

- 软件

- 软件是**程序**、**数据**以及开发、使用和维护所需要的所有**文档**的总称

- 文档：描述程序的研制过程、方法和使用的图文资料

- 软件的身份转变

- 1960：程序

- 1980：工具

- now：服务

- 信息时代软件的作用

- 知识的管理者

- 知识的传递者

- 知识的生产者

- 软件的本质特性

- 复杂性

- 一致性

- 软件不能独立存在，需要依附于一定的环境（如硬件、网络以及其他软件）

- 软件必须遵从人为的惯例并适应已有的技术和系统

- 软件需要随接口不同而改变，随时间推移而变化，而这些变化是不同人设计的结果

- 可变性（演化性）

- 不可见性

- 软件开发面临的挑战

- 软件开发面临的挑战

- 客户不满意

- 交付的功能用户不需要

- 交付的日期没有保障

- 用户使用时发现bug

- 风险与成本问题

- 开发团队专注技术，忽视风险

- 无能力预测成本，导致预算超支

- 项目过程失控

- 客户需求变化频繁
 - 无法预见软件交付质量
 - 盲目遵从流程，忽略客户业务价值
- 无力管理团队
 - 无法评估开发人员能力与工作进度
 - 难以提升团队能力与效率
- 软件生产过程中面临的问题
 - 软件开发周期长
 - 开发成本高
 - 交付前无法找到错误
 - 维护软件成本高
 - 无法估计软件开发与维护过程
- 为什么出现这些问题
 - 客观上
 - 软件产品开发的复杂度和难度随软件规模呈指数级别增长
 - 主观上
 - 软件开发人员缺乏工程性的、系统性的方法论。没有区分“产品研发”与“程序编码”，忽视需求分析、轻视软件维护
- 软件创新的最大障碍——复杂性
 - 复杂的基础架构
 - 缺乏适用的开发流程
 - 存在众多的应用系统
 - 每年扩展现存系统并开发新系统
 - 软件生产环境本身的复杂度
- 软件开发问题的解决途径
 - 软件工程
 - 是什么
 - 软件工程是：1. 将系统性的/规范化的/可量化的方法应用于软件的开发/运行和维护，即工程化应用到软件上；
 - 2. 对 1 中所述方法的研究
 - 基本目标
 - 较低的开发成本
 - 按时完成开发任务并及时交付
 - 实现客户要求的功能
 - 所开发软件具有良好的性能
 - 较高的可靠性、可扩展性、可移植性
 - 软件维护费用低

- 内容
 - 工具
 - | 为过程和方法提供自动或半自动化支持
 - 方法
 - | 技术上的解决办法
 - 过程
 - | 确定若干框架活动
 - 质量关注点

- 工程的特征

工程的特征

- 平衡与决策
- 度量与验证
- 训练有素的过程
- 团队协作与角色分工
- 系统地运用工具
- 工程原则、标准和实践
- 重用设计和设计制品

- 工程化思想

- 分而治之
- 折中
- 复用

- 软件工程的基本概念

- 软件工程的基本要素

- 方法（范型）
 - | 如何做的技术：面向服务/面向构建/面向对象/面向过程
 - 过程
 - 工具

- 软件工程所关注的目标

- 功能性需求

软件所实现的功能达到它的设计规范和满足用户需求的程度

- 完备性
 - 正确性
 - 健壮性
 - 可靠性

- 非功能性需求

效率：软件实现其功能所需要的计算机资源的大小，“时间-空间”；

可用性：用户使用软件的容易程度，用户容易使用和学习；

可维护性：软件适应“变化”的能力，系统很容易被修改从而适应新的需求或采用新的算法、数据结构的能力；

可移植性：软件不经修改或稍加修改就可以运行于不同软硬件环境(CPU、OS和编译器)的能力；

清晰性：易读、易理解，可以提高团队开发效率，降低维护代价；

安全性：在对合法用户提供服务的同时，阻止未授权用户的使用；

兼容性：不同产品相互交换信息的能力；

经济性：开发成本、开发时间和对市场的适应能力。

商业质量：上市时间、成本/受益、目标市场、与老系统的集成、生命周期长短等。

系统能够完成所期望的工作的性能与质量

- 软甲工程学科发展
- 什么是好的软件
 - McCall 质量模型
 - ISO9126 质量模型
 - 商业环境下的软件质量