

# Chapter 1

---

## ■ **Software & Software Engineering**

*Slide Set to accompany*

*Software Engineering: A Practitioner's Approach, 7/e*

**by Roger S. Pressman**

# What is Software

---

Software is:

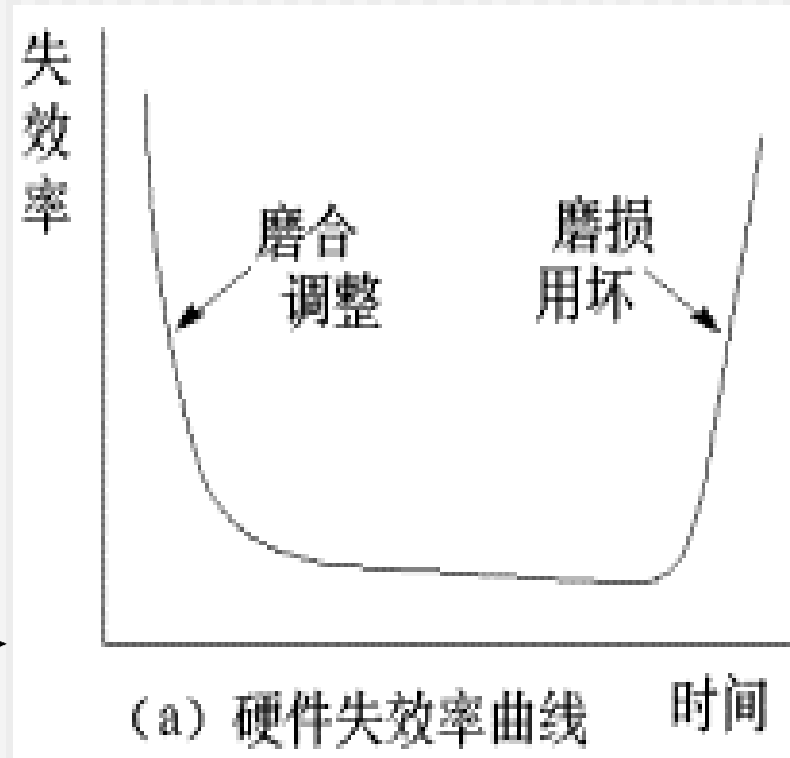
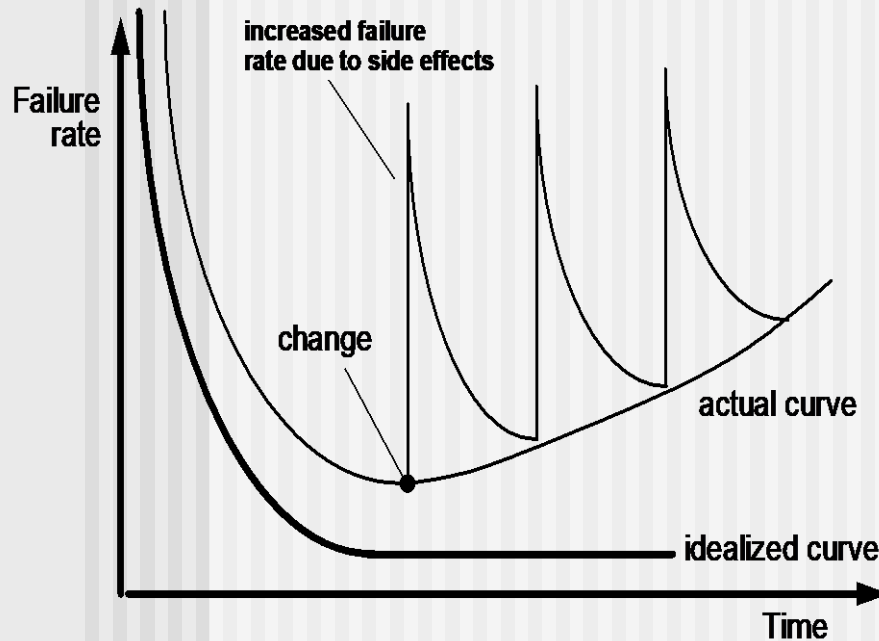
- (1) **instructions** (computer programs) that when executed provide desired features, function, and performance;
- (2) **data structures** that enable the programs to adequately manipulate information and
- (3) **documentation** that describes the operation and use of the programs.

# Nature of Software

---

- 软件的生产 and 硬件不同，没有明显的制造过程。
- 软件不会磨损。
- 软件开发至今尚未完全摆脱手工开发方式。
- 软件是一种逻辑实体，具有抽象性。
- 软件的开发和运行受到计算机系统的限制，对计算机系统有不同程度的依赖性。

# Wear vs. Deterioration (磨损)



# 软件发展与软件危机

	程序设计时代	程序系统时代
时间	1946-1956	1956-1968
生产方式	个体手工劳动	作坊式小集团合作
使用工具	机器、汇编语言	高级语言
开发方法	个人编程技巧	个人编程技巧，开始提出结构化方法
硬件特征	价格贵，存储容量小，运行可靠性差	速度、容量及工作可靠性明显提高，价格降低
软件特征	只有程序设计概念，不重视程序设计方法	程序员数量猛增，但开发人员素质差，开发技术没有新突破，软件危机产生
软件定义	等于程序	等于程序加技术文档

83年，王永民开发五笔输入法

# Software's Chronic Crisis

---

软件危机是指落后的软件生产方式无法满足迅速增长的计算机软件需求，从而导致软件开发与维护过程中出现一系列的严重问题的现象。

软件危机的表现:

- Do not provide the desired functionality
- Take too long to build
- Cost too much to build
- Require too much resources (time, space) to run
- Cannot evolve to meet changing needs
- Quality can not be guaranteed

# Software's Chronic Crisis

---

- From US DoD (国防部), Software Contracts in 1979
  - 60% schedule overruns
  - 50% Cost overruns
  - 45% Unusable
  - 29% Never delivered
  - 19% Need rework
  - 2% Usable exactly as delivered
- 丹佛国际机场行李处理系统，1989年动工，参加的公司有400多家。预计1993年10月启用，最终在1995年1月启用，落后进度13个月，超出预算近20亿美元，总造价52亿美元

# Software's Chronic Crisis

---

- 软件危机产生的原因
  - 一方面是与软件的复杂性和本身的特点有关
  - 另一方面是由软件开发、维护、管理方法不正确有关

如：Windows2000有5000万行代码；

Exchange2000和 Windows2000涉及开发人员

	Exchange2000	Windows2000
项目经理	25人	约250人
开发人员	140人	约1700人
测试人员	350人	约3200人



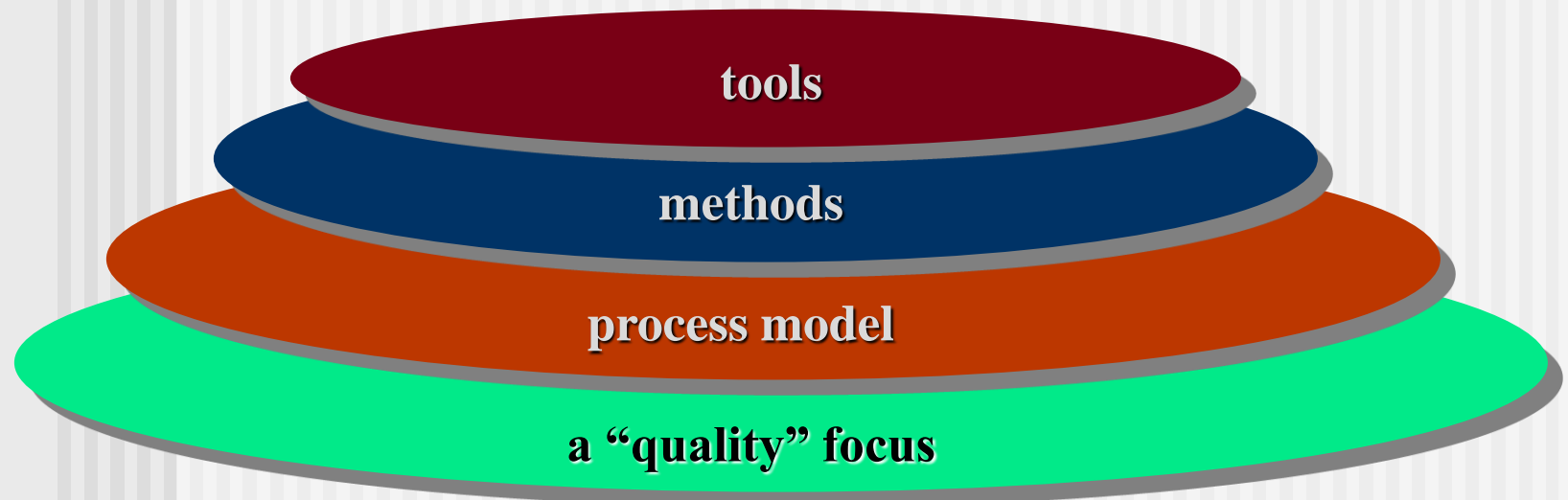
# 软件工程定义

---

- 1968年，北大西洋公约组织首次提出“**软件工程**”概念。
- 软件工程主要研究 软件生产的客观规律，建立与系统化软件生产有关的**概念、原则、方法、技术和工具**，指导和支持软件系统的生产活动，以期达到**降低软件生产成本、改进软件产品质量、提高软件生产率**的目标。

# 软件工程核心要素

---



- 1) Software engineering must rest on an organizational commitment to **quality**.

质量是软件产品的生命线, 软件工程以质量保证为基础

2) The foundation for software engineering is the **process** layer.

软件开发包括若干子过程：**开发过程**、维护过程、管理过程等；



# Cooking Process

(1) 过程与质量密切相关; (2) 不同过程适用于不同项目



备料

- 1、连皮猪肉，去毛洗净，放入冷水锅内煮熟。
- 2、将煮好的猪肉冲洗干净放凉，切成薄而大的肉片。

炒制

- 3、锅内放一茶匙油，放入肉片和姜片，用小火慢慢煎炒至出油，表面色泽变金黄色。
- 4、加入红油豆瓣酱。翻炒至出红油后，再放入一匙生抽，料酒翻炒。
- 5、投入青蒜白及红、青椒，翻炒至断生即可。
- 6、最后放入青蒜叶，翻炒片刻即可出锅。

备料

1. 猪肉切丝，加老抽和淀粉抓匀，放一旁备用；
2. 子姜，红椒切丝；

炒制

3. 锅里下油，油热后下少许子姜丝(因为肉抓了淀粉，容易粘锅，先下少许子姜丝，会有所改善)
4. 把肉丝放下去，翻炒至8分熟，放入剩余的子姜丝，翻炒至肉全熟，下红椒翻炒几下，放盐，翻炒均匀即可出锅。



# Methods

---

3) Software engineering **methods** provide the technical how-to's for building software.

定义“**开发过程**中每项活动，任务的**实现方法**”。

Methods encompass a broad array of tasks that include **requirements analysis, design, program construction, testing, and support.**

# Tools

---

4) Software engineering **tools** provide automated or semi-automated support for the process and the methods.

方法的实现需要**支撑工具**，如分析工具、设计工具、编译工具、测试工具等。

当这些工具被**集成**起来使得一个工具产生的信息可以被另外一个工具使用时，一个支持软件开发的系统就建立了，称为**计算机辅助软件工程**（CASE）。

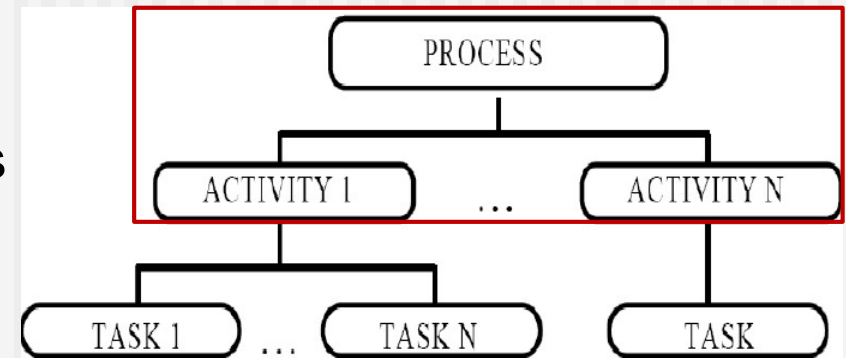
# A Process Framework

---

- 过程框架定义了若干个**框架活动**，为实现完整的软件工程过程建立了基础。
- 这些活动可**普遍应用**于所有软件开发项目，无论项目的规模和复杂性如何。
- 过程框架
  - 框架性活动
  - 普适性活动

# Framework Activities

- Communication
- Planning
- Modeling
  - Analysis of requirements
  - Design
- Construction
  - Code generation
  - Testing
- Deployment



层次结构



# Umbrella Activities

---

普适性活动贯穿项目始终，帮助软件团队管理和控制项目进度、质量、变更和风险

- Software project tracking and control
- Formal technical reviews
- Software quality assurance
- Software configuration management
- Work product preparation and production
- Reusability management
- Measurement
- Risk management

# Adapting a Process Model

---

- A process adopted for one project might be significantly different from which is adopted for another project.
- 某企业过程标准