• • •

# 软件工程概述

清华大学软件学院 刘强



# 教学提纲

1	软件工程的产生与发展
2	软件工程基本概念
3	软件质量属性
4	软件过程模型

# 教学提纲

- 软件的发展历程
- 软件开发面临的挑战
- 软件的本质特性
- 软件开发问题的解决途径

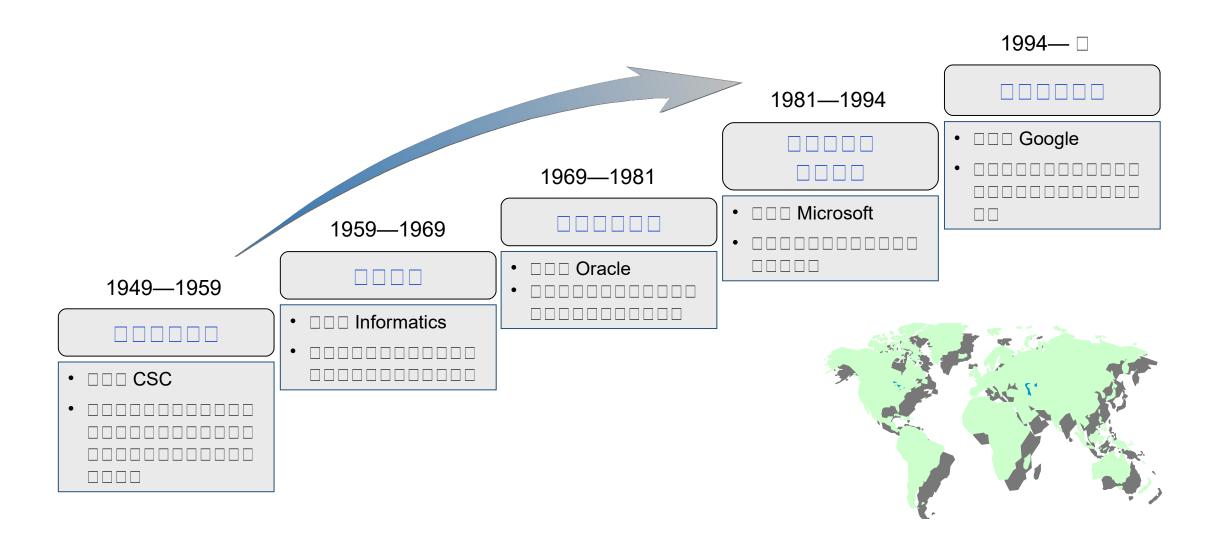
## 软件无处不在



较件无处不在,为我们的生活创造了无限精彩。在当今的信息时代,世界正在变得更加"智慧",万事万物间感知化、互联化和智能化的程度不断加深。

较件工程为这一切做出了巨大贡献, 随着 该学科的成熟发展, 其未来的贡献将不可限量

## 软件发展历程



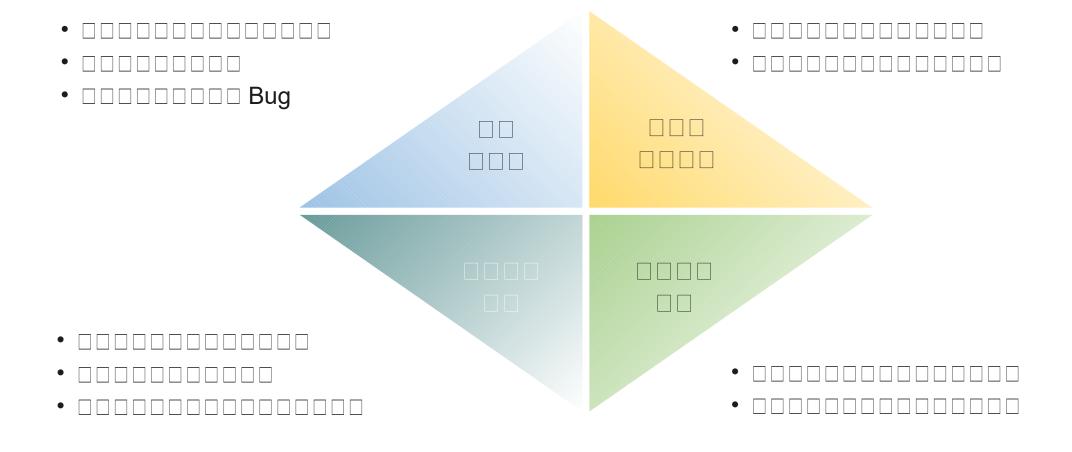
#### 软件开发面临的挑战

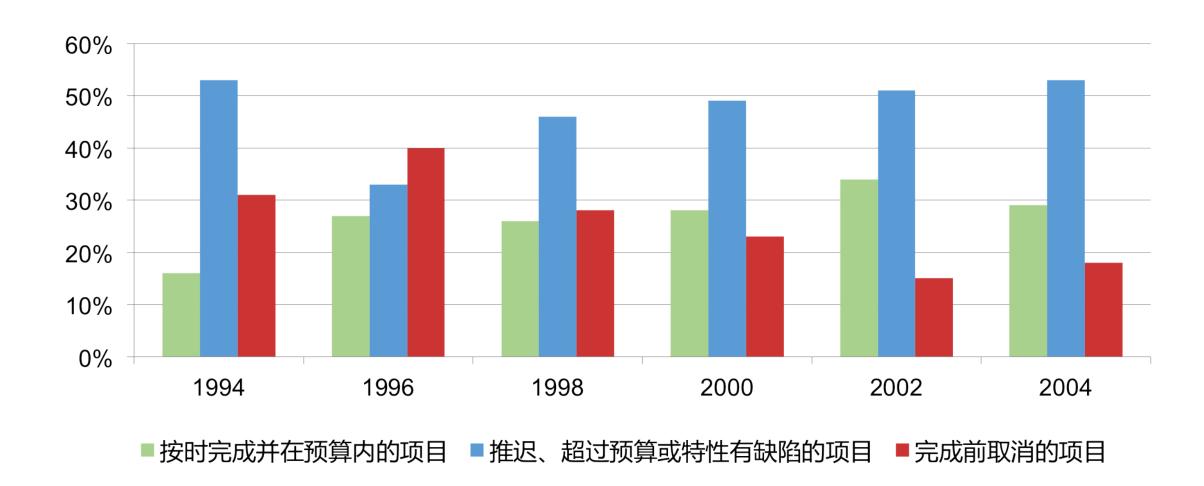
### 软件创新的最大障碍, 复杂性

- 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



# 软件开发面临的挑战





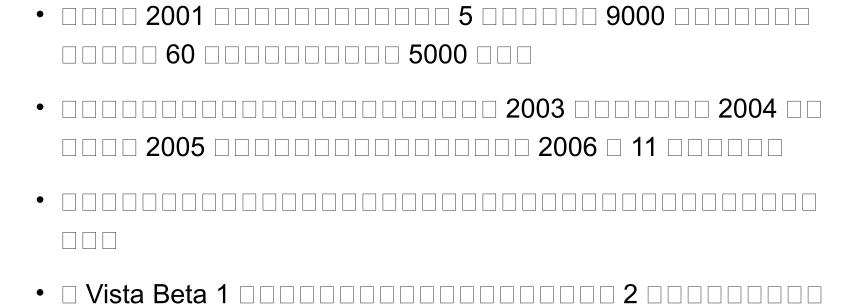
□ □ 1 □ ARIANE 5 □ □

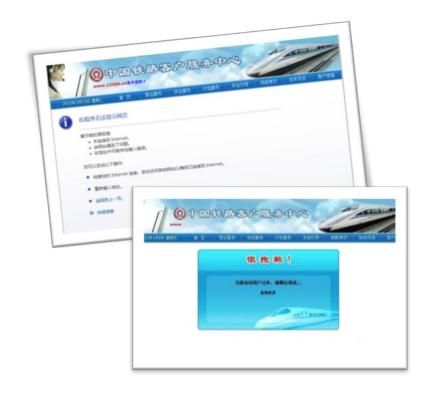


- •
- •

#### □ □ 2 □ Windows Vista □ □







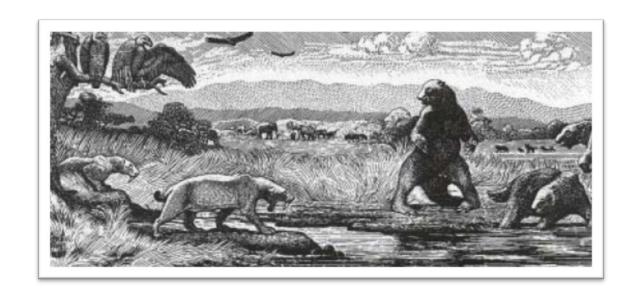






## 一个值得思考的问题

为什么一个看似简单的东 一个看似简单的东 一个有可能变成一个 慈后进度、超出预算、存 在大量缺陷的怪物?



Brooks, F. P., *The Mythical Man Month*, Addison-Wesley, 1975

#### 软件的本质特性



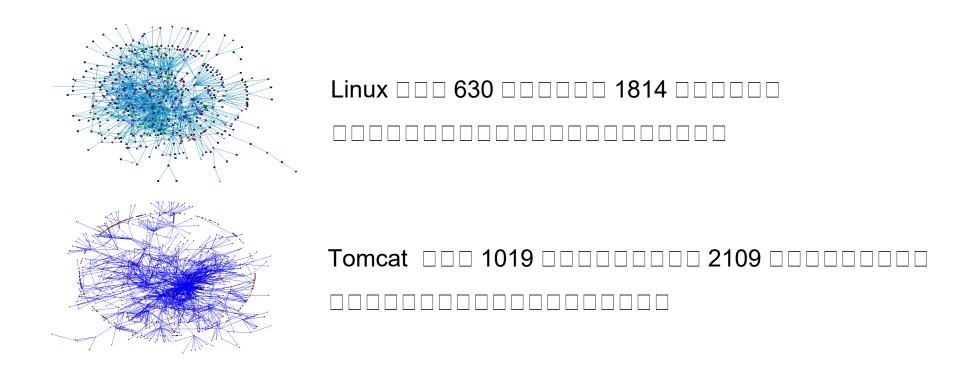
软件具有复杂度、一致性、可变性和不可见性等 固有的为在特性, 这是造成软件开发困难的根本原因

0

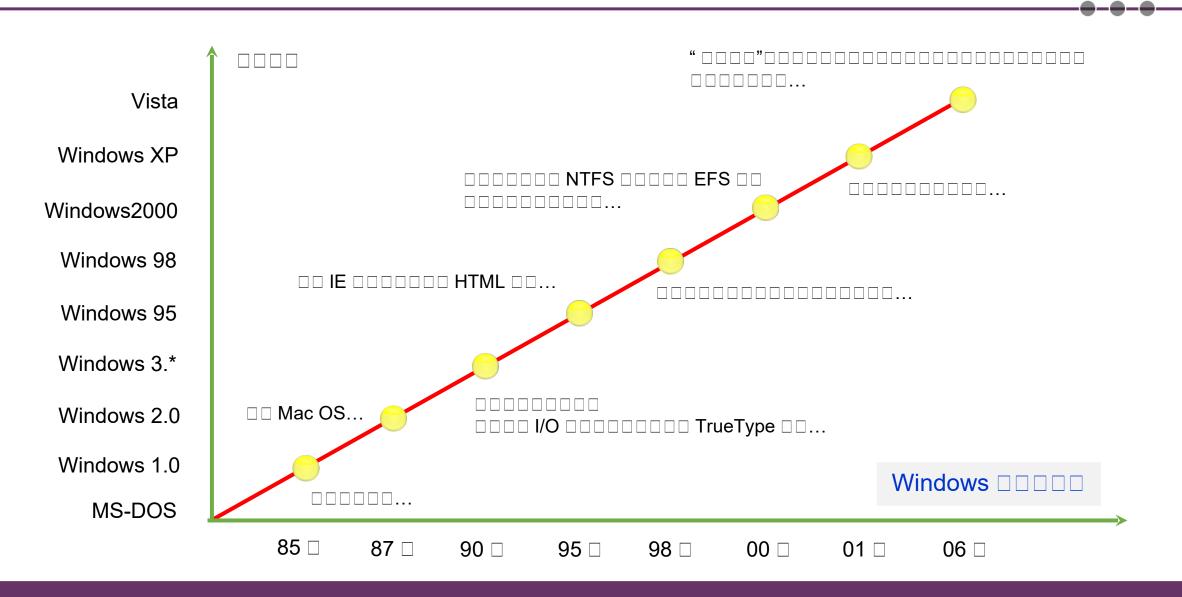
Brooks, F. P., "No silver bullet: essence and accidents of software engineering", *IEEE Computer*, Vol. 20, No. 4, pp.10-19, 1987

# 软件的本质特性:复杂性

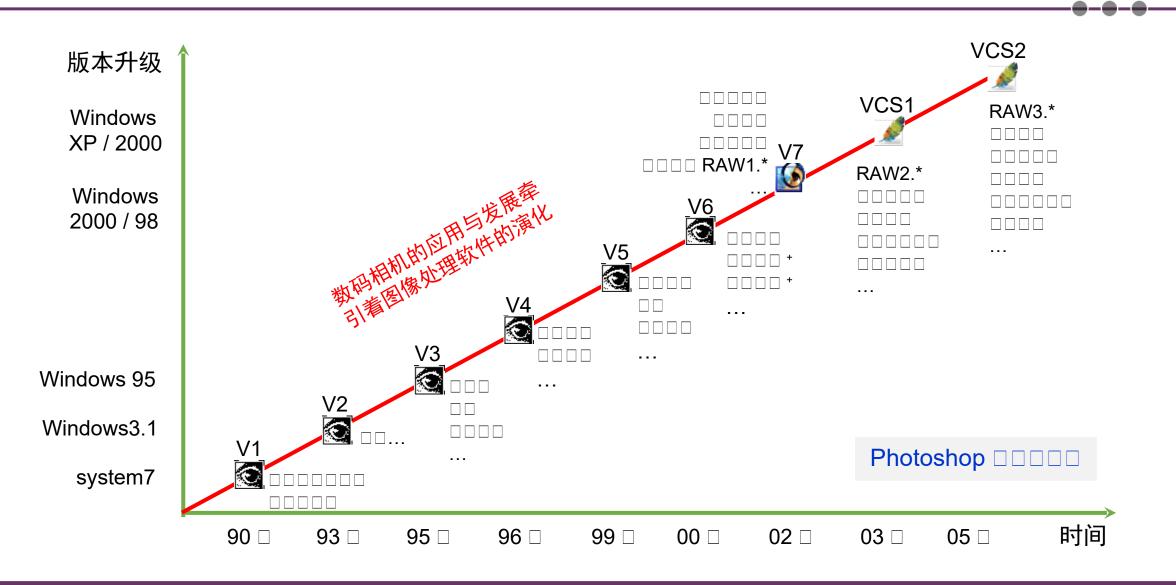
软件是人类思想的外延,人们将自己的思想传送给计算机,当产生的可执行文件被激活运行时,软件便重现人类的意图。



# 软件的本质特性:演化性



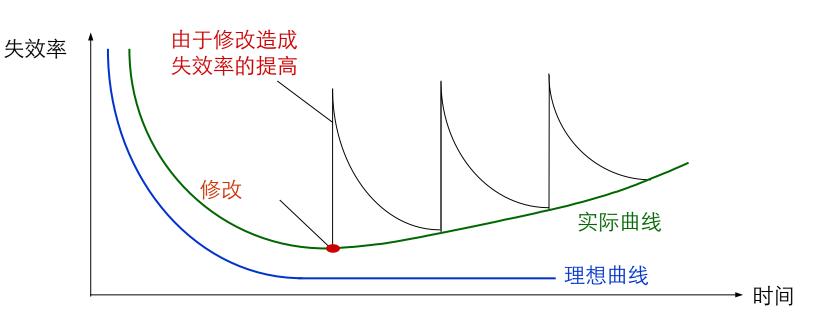
# 软件的本质特性:演化性



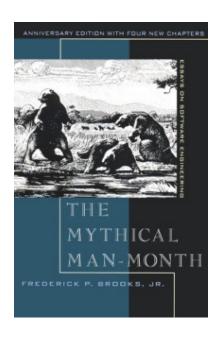
## 软件的本质特性:演化性

- 人们总是认为软件是容易修改的,但忽视了修改所带来的副作用
- 不断的修改最终导致软件的退化,从而结束其生命周期

软件的失效率曲线

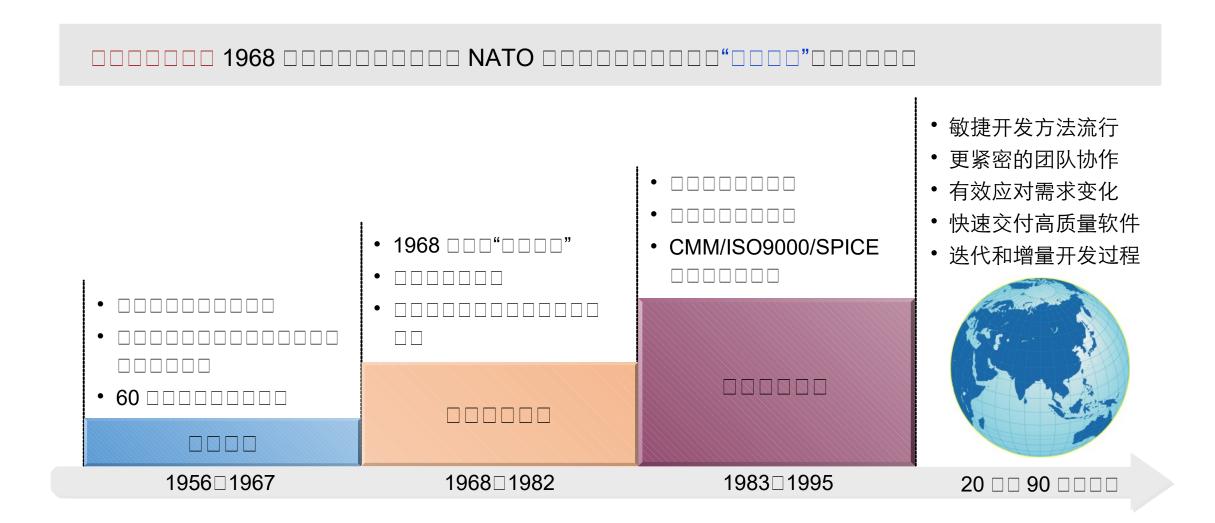


#### 软件的本质特性:不可见性



软件人员太像"皇帝的新衣"故事中的裁缝了。当我来检查软件 开发工作时,所得到的回答好象对我说:我们正忙于编织这件带 有魔法的织物。只要等一会儿,你就会看到这件织物是及其美丽 的。但是我什么也看不到,什么也摸不到,也说不出任何一个有 关的数字,没有任何办法得到一些信息说明事情确实进行得非常 顺利,而且我已经知道许多人最终已经编织了一大堆昂贵的废物 而离去,还有不少人最终什么也没有做出来。

## 软件开发问题的解决途径



# 教学提纲

- 系统的本质与工程的方法
- 什么是软件工程
- 软件工程的基本要素
- 软件工程学科发展

#### 系统的本质

ENTITIES:

Particulate matter

Oxygen

Carbon dioxide

Water

Nitrogen

Nose

Mouth

Trachea
Bronchial tubes

Lungs

Alveoli

#### **ACTIVITIES**:

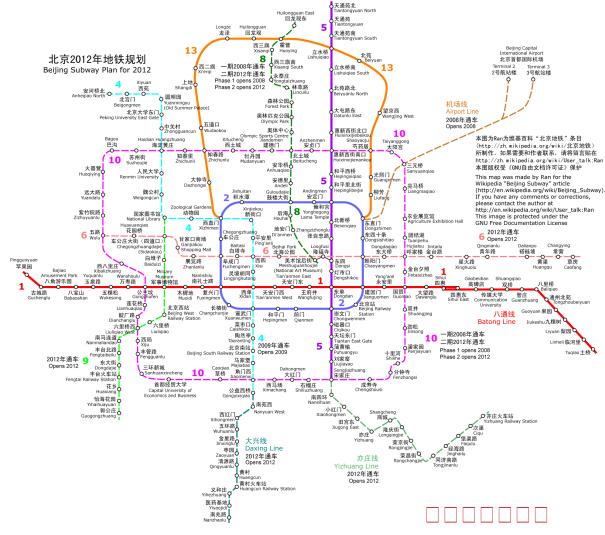
Inhale gases

Filter gases

Transfer molecules to/from blood



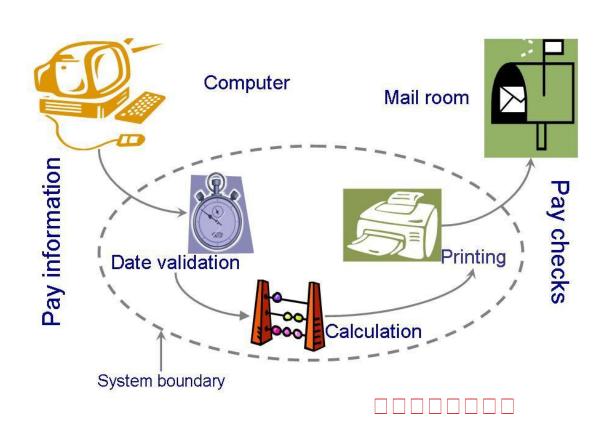
Boundary

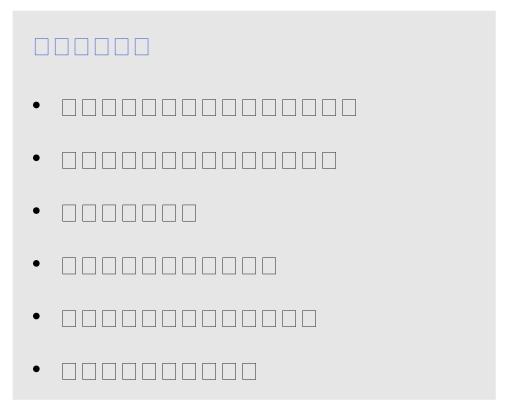






# 系统的本质





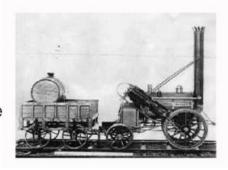
#### 工程的方法

#### 工程是将理论和所学的知识应用于实践的科学,以便经济有效地解决实际问题。



- craft
  - personal skill
  - experience
  - flexible materials

- engineering
  - tables and tools
  - recorded knowledge
  - controlled materials



#### ■ 规模上的差异

- 花园小道 vs. 汽车高速公路
- 树上小屋 vs. 摩天大楼
- 加法程序 vs. 医院档案系统
- 手工(Craft):小规模的设计与建造
  - 简单问题与单一目标
  - 个人控制与个人技能
- 工程(Engineering):大规模的设计与建造
  - 复杂问题与目标分解
  - 团队协作、需要考虑运营、管理、成本、质量 控制、安全等

# 工程的方法

# 需求分析和定义 软件设计 编码实现 软件测试 软件交付与维护

注重系统的构建过程

#### 工程的特征

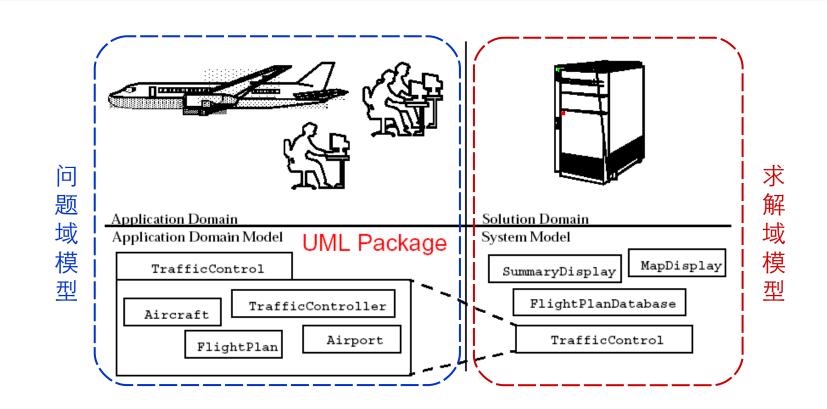
- 平衡与决策
- 度量与验证
- 训练有素的过程
- 团队协作与角色分工
- 系统地运用工具
- 工程原则、标准和实践
- 重用设计和设计制品



- •
- 0000000

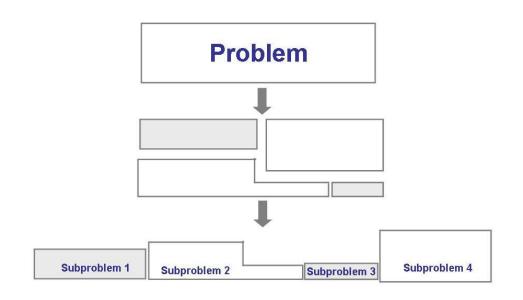


软件工程是一项建模活动,通过抽象找到事物的重要特征而忽略非本质的细节,从不同侧面建立系统模型,有效地简化和处理复杂性。

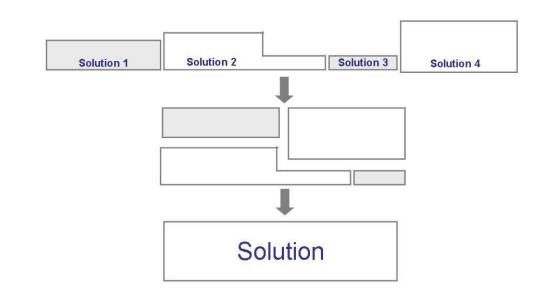


软件工程是一项解决问题的工程活动,它不仅限于算法设计,还要通过试验、设计复用

、系统评估等手段找到一个客户可接受的方案。



分析:将复杂的问题分解成可理解并能够 处理的若干小问题进行研究和分析。



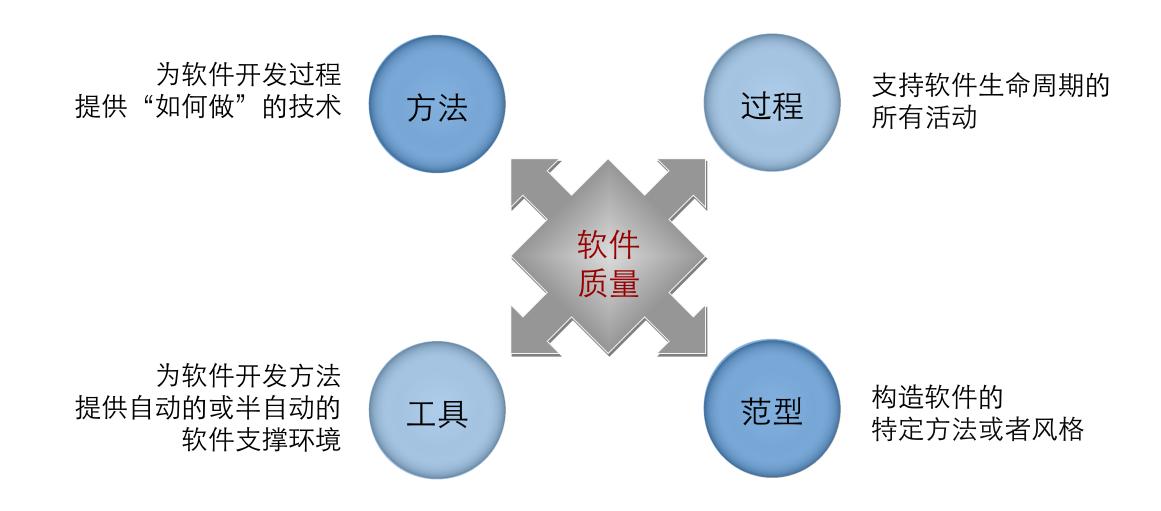
合成: 在分析的基础上,针对各个不同部分给出解决方案,并将这些小构造块组合成一个大系统

软件工程是一项受软件工程原理指导的活动,软件工程师需要捕捉和理解一个系统的基本原理模型,并根据评价标准提出合理的开发决策。

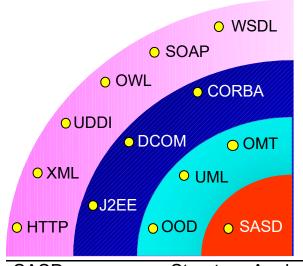


- 00000000000000000

### 软件工程的基本要素



### 软件工程方法(范型)



面向服务:在应用表现层次上将软件构件化,即应用业务过程由服务组成,而服务由构件组装而成。

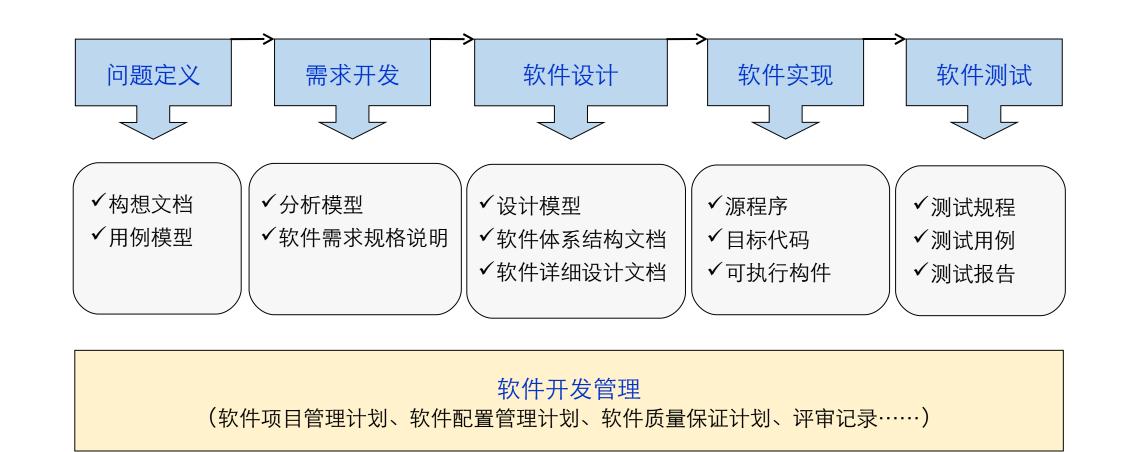
面向构件: 寻求比类的粒度更大的且易于复用的构件, 期望实现软件的再工程。

面向对象: 以类为基本程序单元,对象是类的实例化,对象之间以消息传递为基本手段。

面向过程: 以算法作为基本构造单元,强调自顶向下的功能分解,将功能和数据进行一定程度的分离。

SASD	Structure Analysis, Structure design	
OOD	Object-Oriented Programming	
UML	Unified Modeling Language	
OMT	Object Modeling Technology	
CORBA	Common Object Request Broker Architecture	
DCOM	Distribute Component Object Model	
J2EE	Java2 Platform Enterprise Edition	JAVA 🗆 🗆 🗆 🗆
HTTP	Hypertext Transfer Protocol	
XML	Extensible Markup Language	
UDDI	Universal Description, Discovery and Integration	
OWL	Ontology Web Language	□ □ Web □ □
SOAP	Simple Object Access Protocol	
WSDL	Web Service Describe Language	

### 软件工程过程



### 举例:麦当劳的过程管理







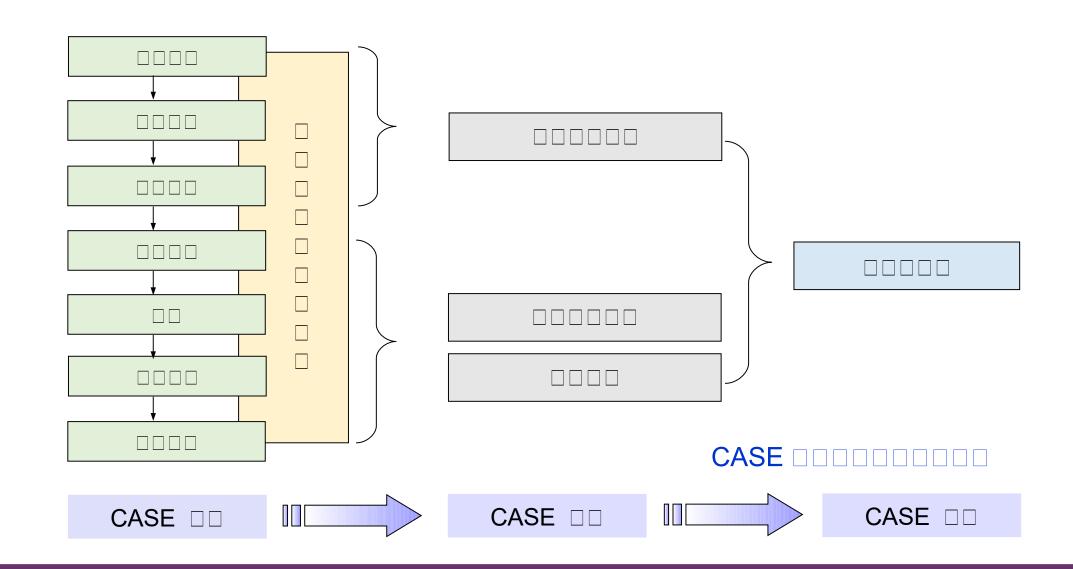
- 流水线作业: 机械化、批量化、标准化
- 四无主义:无论何人,无论何时,无论何地,无品质差异
- 质量控制: 在过程细节上精益求精

- •□□□□□□ 17cm □□□□□□□□□□

- •□□□□□□□ 83% □□□ 17% □□□□□□□
- •

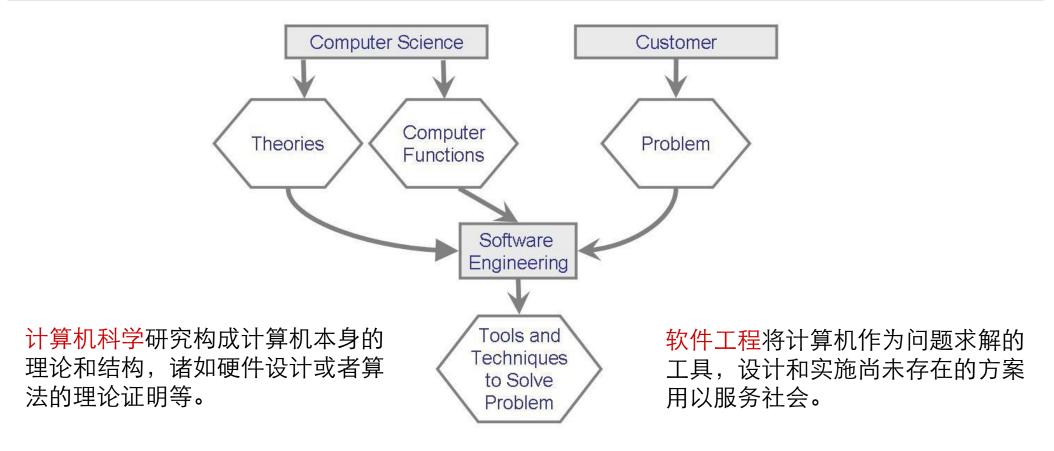
- •□□□□□□□□□□□□ 4 °C □□□□□□□□

# 软件工程工具



#### 软件工程与计算机科学的区别

科学是发现世界上已经存在的事物,回答"是什么"和"为什么"的问题。工程是创造世界上从未存在的事物,回答"做什么"和"怎么做"的问题。



## 举例:电子商务网站商品推荐



#### 学术界:

- •基于数据本身评价推荐效果,如准确率等;
- •如果一个用户本来打算购买《软件工程》这本书,那 么推荐系统准确预测该结果是否有价值?

#### 工业界:

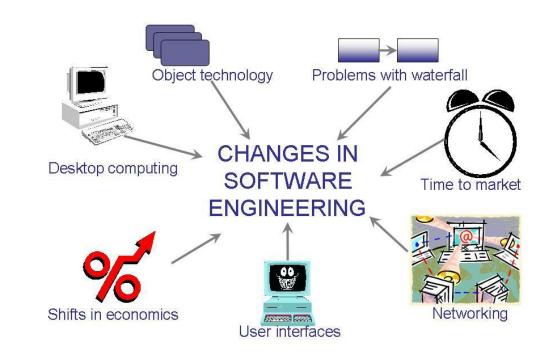
- •基于商业应用评价推荐效果,如受推荐影响的转化率和购买率等;
- •对于用户而言,希望推荐系统使其购买更多并非预想的书;对于出版社而言,希望推荐系统增加图书的潜 在购买人群。

#### 软件工程学科发展

Wasserman, Anthony I., "Toward a discipline of software engineering", *IEEE Computer*, Vol. 13, No. 6, pp.23-31, 1996

#### 改变软件工程的关键因素:

- •交付时间的重要性
- •计算技术在经济中的转变
- •功能强大的桌面计算
- •互联网络
- •面向对象技术
- •图形用户界面
- •瀑布模型的不可预知性



### 软件工程学科发展

抽象

软件建模方法

用户界面原型化

软件体系结构



Wasserman's Discipline of Software Engineering

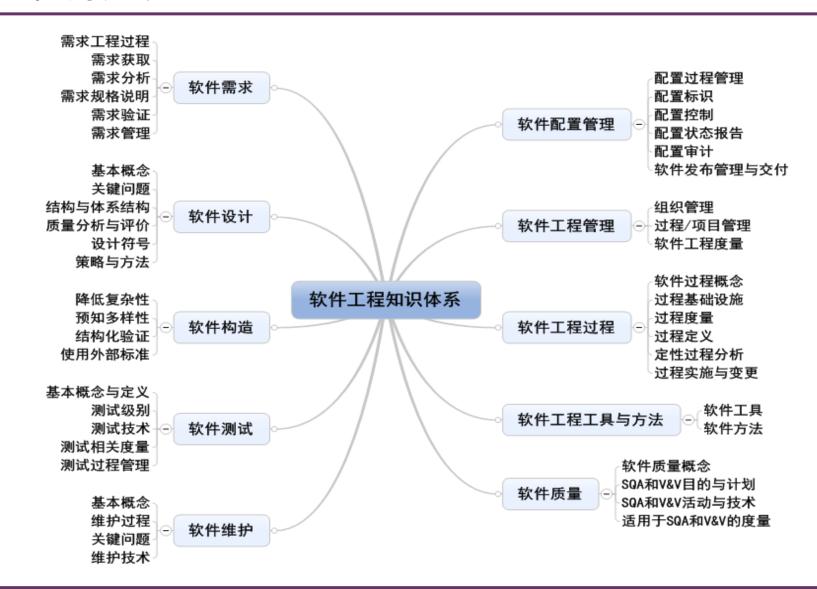
软件过程

软件复用

度量

工具与集成环境

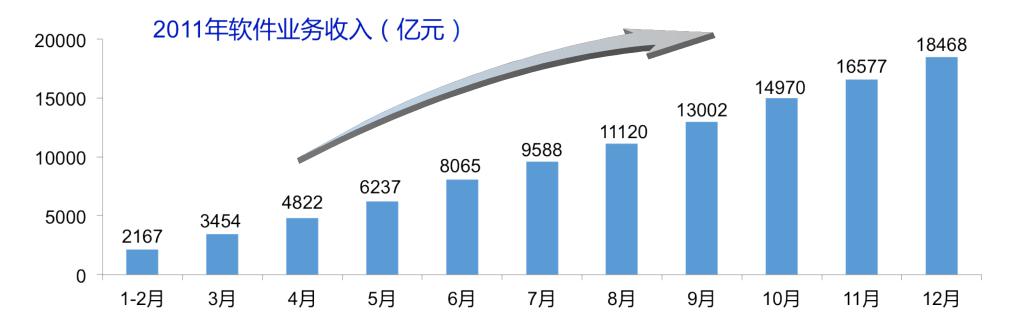
#### 软件工程学科发展



#### 中国软件产业的需要



- 软件产业步入新的快速发展阶段,收入保持快速增长势头
- 软件行业结构调整步伐加快,骨干企业运行态势良好
- 基础软件产业化能力差,产业创新体系不健全,核心技术缺乏
- 人才结构矛盾突出,高层次、复合型、领军型人才依然缺乏



#### 中国软件产业的需要

#### 软件人才是软件企业最宝贵的财富,其工作特点是创新



#### 软件人才职业要求

- ✓ 良好的思想政治素质
- ✓ 恪守伦理和工程职业道德
- ✓ 掌握数学、自然科学和工程科学知识
- ✓ 具备解决复杂工程问题的能力
- ✓ 解决与工程有关经济、环境与社会问题
- ✓ 良好的人际交流与合作能力
- ✓ 具备国际视角和跨文化竞争与合作能力



#### 软件人才素质要求

既要:细致、严谨、认真、甘于寂寞、乐于奉献、

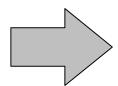
责任感强

又要: 兴趣广泛、开阔思路、善于观察自然和社会

、有创新创业精神

喻为: 动静结合

还要: 过硬的技术、良好的文化底蕴和艺术素养



面对软件技术的迅猛发展,软件人员要不断学习

才能保持持续创新,适应工作需要和软件技术发展的势态

### 教学提纲

3

- McCall □□□□
- ISO9126 □□□□

### 什么是好的软件

Quality is never an accident; it is always the result of intelligent effort.

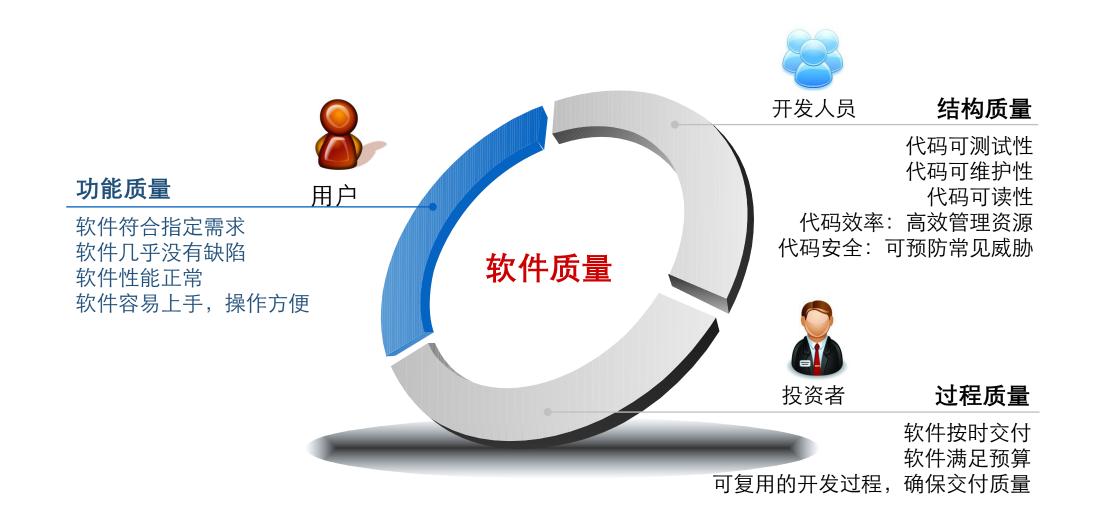
—— John Ruskin

#### 



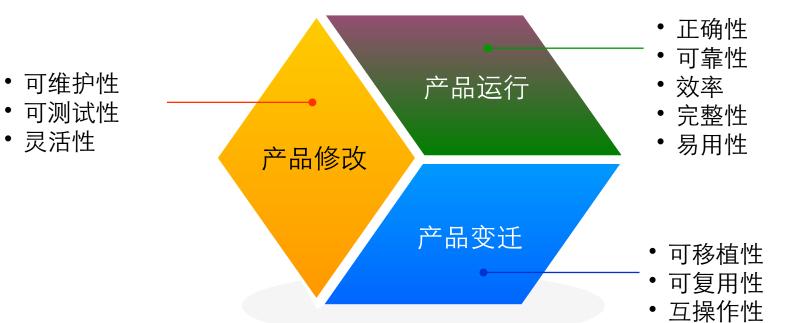


### 什么是好的软件



#### McCall

软件质量是许多质量属性的综合体现,各种质量属性反映了软件质量的不同方面。人们通过改善软件的各种质量属性,从而提高软件的整体质量。



#### McCall

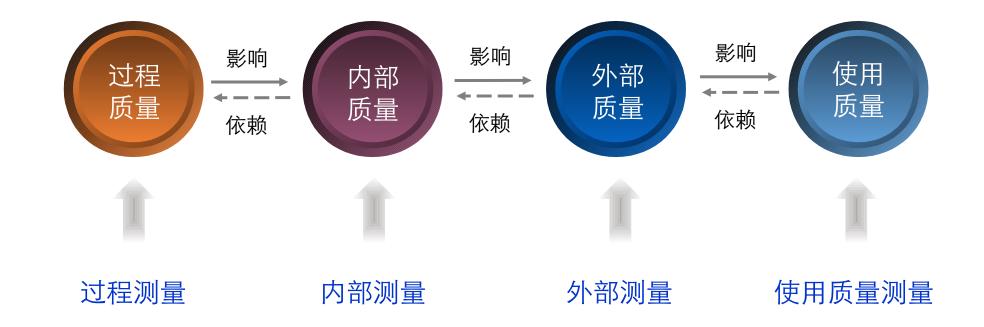
#### 质量属性

- 正确性: 软件满足需求规格说明和完成用户任务目标的程度。
- 可靠性: 软件在规定时间和条件下无故障持续运行的程度。
- 效率: 软件系统完成预定功能所需的计算机资源量。
- · 完整性: 对未授权人员访问软件或数据的可控程度。
- 易用性: 用户学习、操作、准备输入和解释输出所需要的工作量。
- 可维护性: 定位和修复软件中的一个错误所需的工作量。
- 可测试性: 测试程序以确保它能完成预期功能所需的工作量。
- 灵活性: 修改或改进一个已经投入运行的软件所需的工作量。
- 可移植性: 将程序从一个硬件或软件环境移植到另一环境所需的工作量。
- 可复用性: 程序可以再次用于另一个应用中的程度。
- 互操作性: 将一个系统连接到另一个系统所需的工作量。

软件过程

软件产品

产品效用

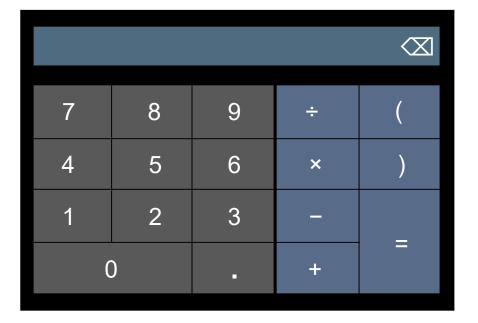


#### ISO9126 | | | |









#### 商业环境下的软件质量

讨论思考:

软件质量的重要性是无容置疑的,那么是不是质量越高就越好? 软件产品是否应该追求"零缺陷"?



举例:在航天器发射之前,只要发现任何异常,就会立即取消发射指令,直到异常被消除为止。前苏联甚至做得更过分,许多重大武器系统的负责人都签了生死状,系统研制成功则获得英雄勋章,失败则被枪毙。



许多互联网软件(例如新浪微博、百度导航等)在产品仍然存在一定缺陷的情况下就发布上线,之后再不断更新版本修复已有的缺陷。这种系统为什么不像航天系统一样,在发布前应修改所发现的任何缺陷?

#### 商业环境下的软件质量

讨论思考:

软件质量的重要性是无容置疑的,那么是不是质量越高就越好? 软件产品是否应该追求"零缺陷"?

#### 商业目标决定质量目标:

- •为了提高用户对产品的满意度,企业必须提高产品的质量,但是不可能为了追求完美的质量而不惜一切代价。当企业为提高质量所付出的代价超过收益时,这个产品就没有商业价值了。
- 企业必须权衡质量、效率和成本三个因素,产品质量太低了或者太高了,都不好。理想的质量目标不是"零缺陷",而是恰好让广大用户满意,并且将提高质量所付出的代价控制在规定的预算之内。

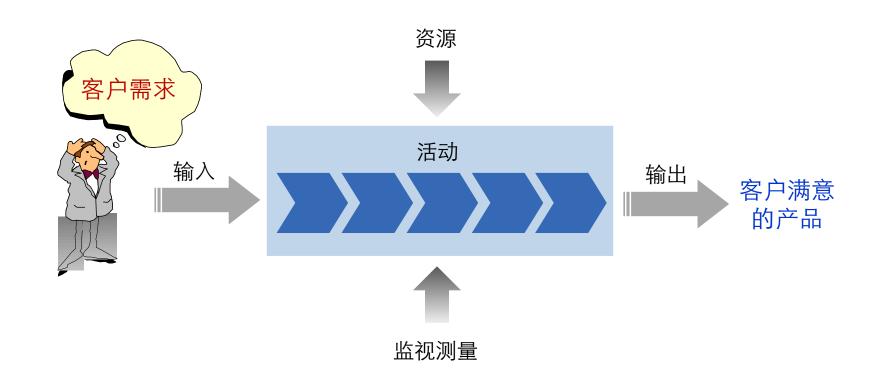
### 教学提纲

4

- 软件开发活动
- 瀑布模型
- 原型化模型
- 阶段化开发
- 可转换模型

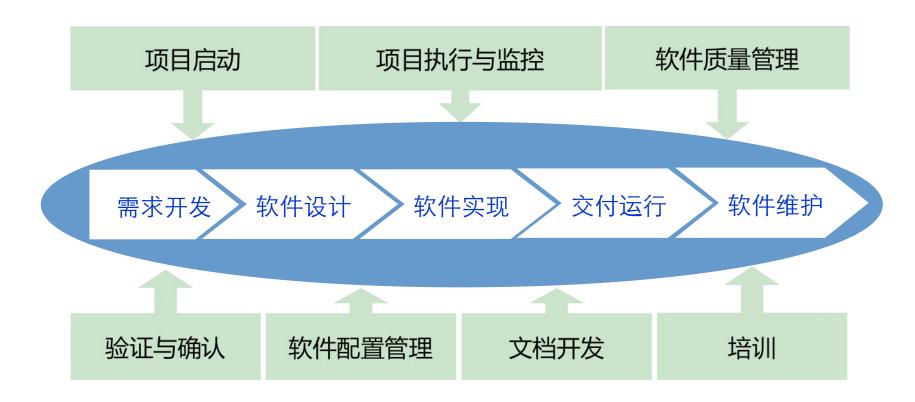
### 过程方法

过程是一组将输入转化为输出的相互关联或相互作用的活动; 过程方法是系统地识别和管理组织内所使用的过程, 保证更有效地获得期望的结果。



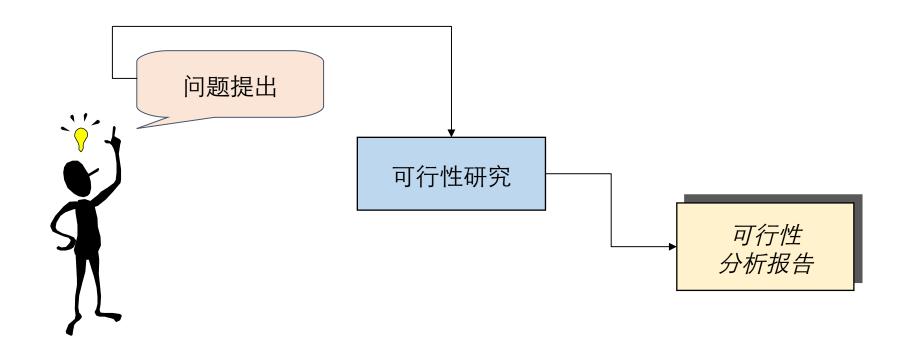
### 软件过程

软件过程是为了获得高质量软件而实施的一系列活动,它定义了工作任务、项目里程碑、交付物和质量保证点。



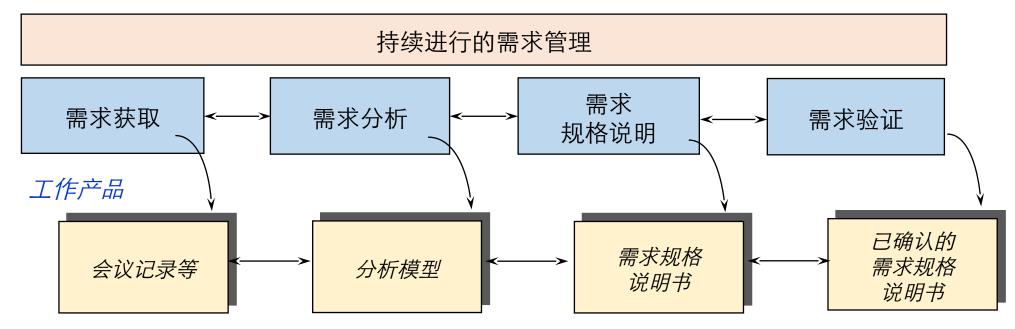
问题定义:人们通过开展技术探索和市场调查等活动,研究系统的可行性和可能的

解决方案,确定待开发系统的总体目标和范围。

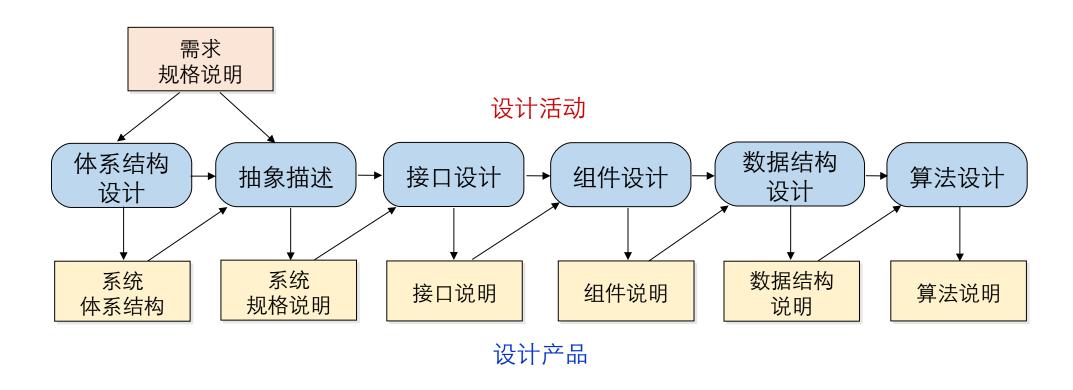


需求开发: 在可行性研究之后,分析、整理和提炼所收集到的客户需求,建立完整的需求分析模型,编写软件需求规格说明。

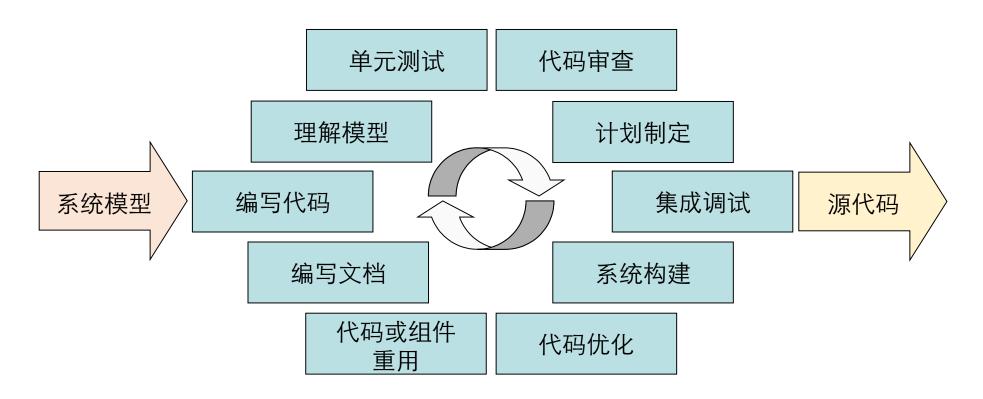
#### 活动



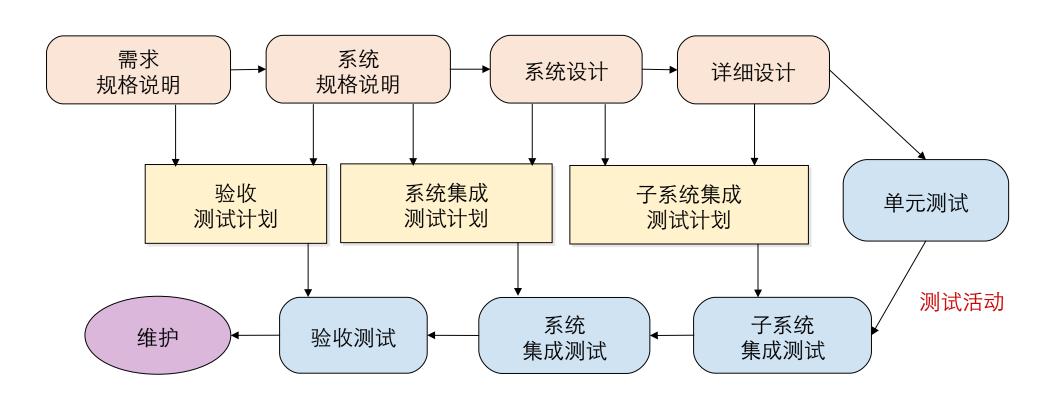
软件设计:根据需求规格说明,确定软件体系结构,进一步设计每个系统部件的实现算法、数据结构及其接口等。



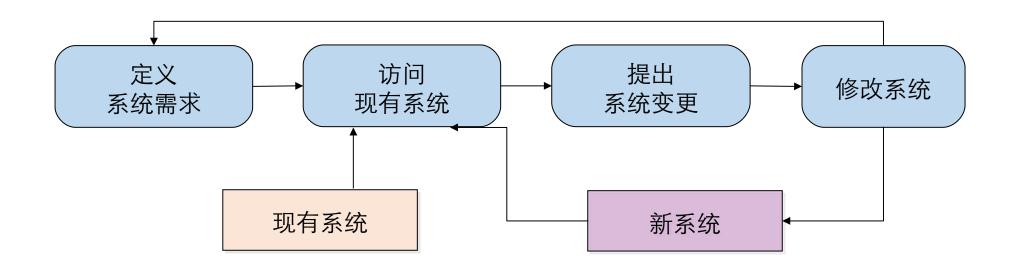
软件实现: 概括地说是将软件设计转换成程序代码,这是一个复杂而迭代的过程,要求 根据设计模型进行程序设计以及正确而高效地编写和测试代码。



软件测试:检查和验证所开发的系统是否符合客户期望,包括单元测试、子系统测试、 集成测试和验收测试等。

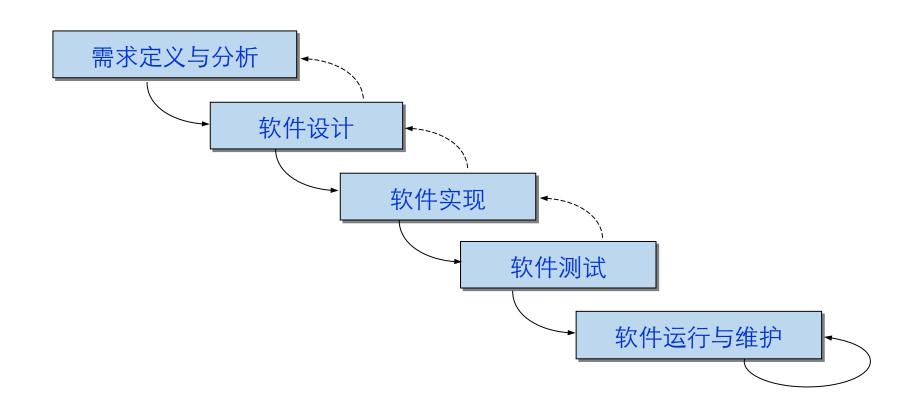


软件演化:系统投入使用后对其进行改进,以适应不断变化的需求。完全从头开发的系统很少,将软件系统的开发和维护看成是一个连续过程更有意义。



### 瀑布模型

瀑布模型的开发阶段严格按照线性方式进行,每一个阶段具有相关的里程碑和交付产品,且需要确认和验证。



### 瀑布模型

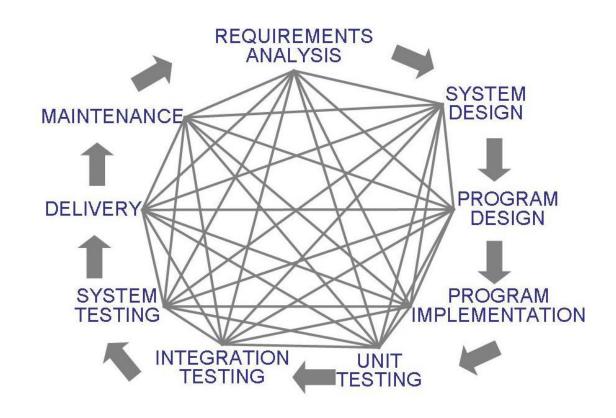
讨论: 瀑布模型是否反映了实际的软件开发过程? 软件开发作为一个问题求解过程, 应当具备什么特点?



瀑布模型产生于硬件领域,它是从制造业的角度看待软件开发的。制造业是重复生产某一特定的产品,而软件开发并不是这样的,随着人们对问题的逐步理解以及对可迄方案的评估,软件在不断演化。因此,软件开发是一个创造的过程,而不是一个制造的过程。

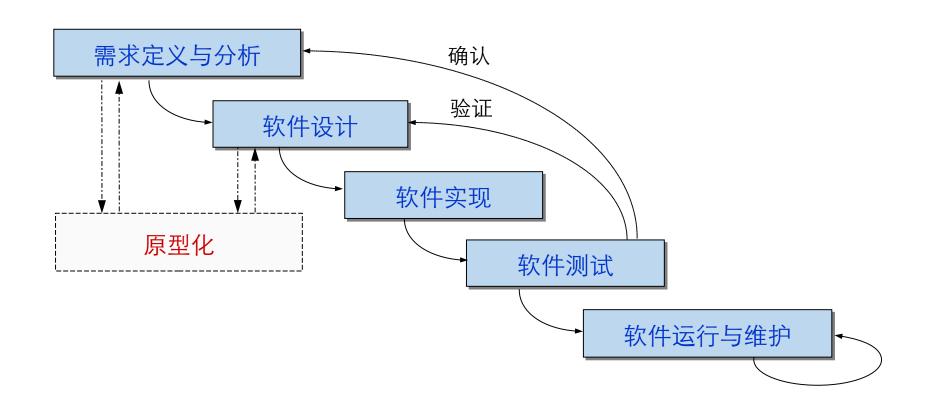
### 软件开发的迭代性

软件开发具有迭代性,需要不断地反复尝试,通过比较和选择不同的设计,最终确定令人满意的问题解决方案。



#### 原型化模型

原型化模型需要迅速建造一个可运行的软件原型,它使用户和开发人员对系统的相关方面进行检查,以决定是否合适和恰当。



#### 需求原型化

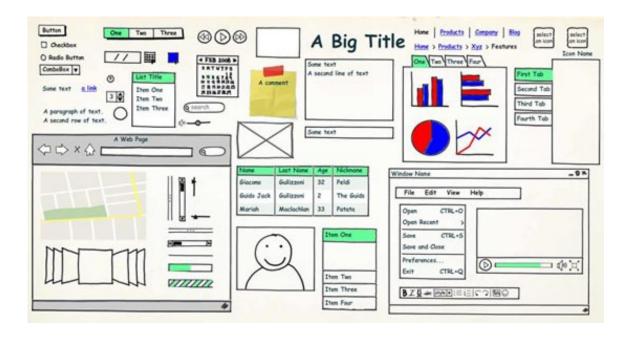
纸上原型是一种原型设计方法,通常在多张纸张和卡片上手绘或打印并裁剪成模块,用以表示不同的界面元素,再组合拼凑并粘贴到背景板上,构造成模拟真实产品界面的原型。这种方法显然构建更快、修改更方便。



http://v.youku.com/v\_show/id\_XMjcyMTM1OTI=.html

### 需求原型化

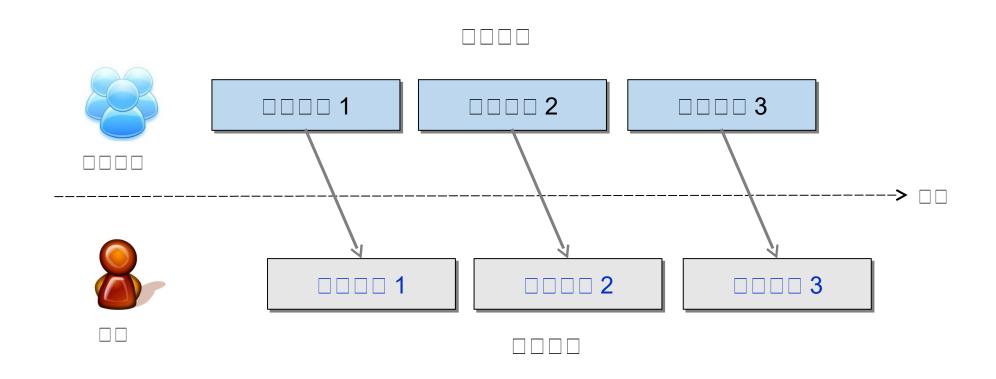




http://www.balsamiq.com

### 阶段化开发

今天的商业环境需要快速地推出新产品,阶段化开发使得软件系统能够一部分一部分 地交付,从而缩短软件开发周期。

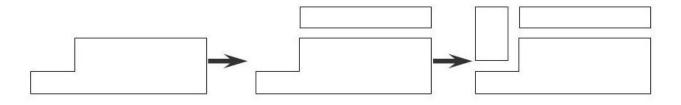


#### 阶段化开发

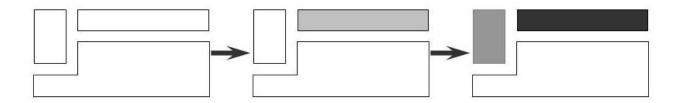
增量模型: 在每一个新的发布中逐步增加功能直到构造全部功能。

迭代模型:一开始提交一个完整系统,在后续发布中补充完善各子系统功能。

#### INCREMENTAL DEVELOPMENT

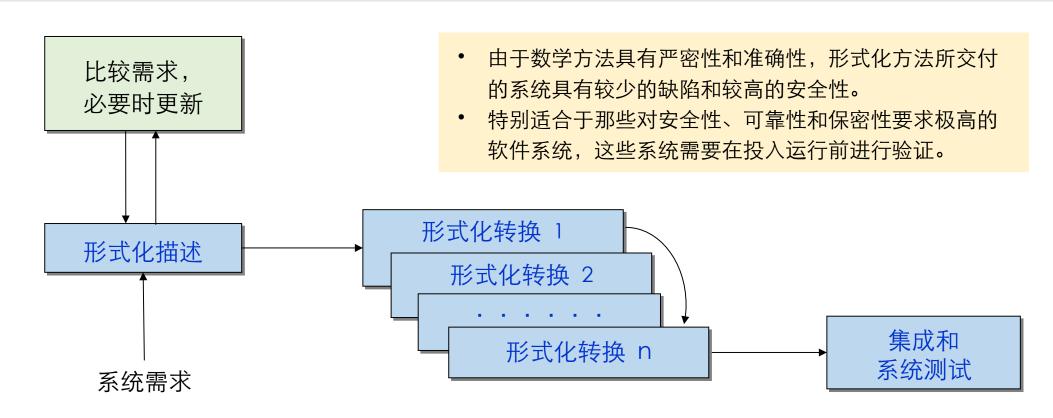


#### ITERATIVE DEVELOPMENT



#### 可转换模型

可转换模型是采用形式化的数学方法描述系统,并利用一系列转换将形式化的需求规格说明变为可交付使用的系统。



#### 

- Booked Braking System ABS Booked Braking System ABS Booked Braking System ABS Booked Braking System Booked Braking Syst



#### 

Dodo Dodo Anti-locked Braking System Dodo ABS Dodo Dodo

 Dodo DO

### 

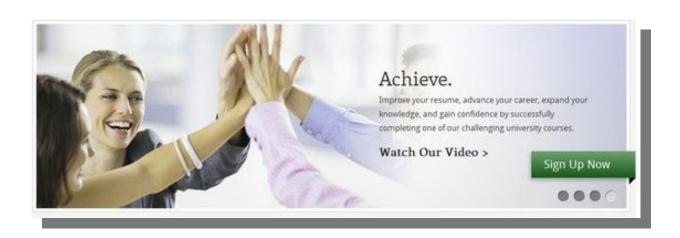
#### 实例分析:

- 嵌入式控制系统
- 该系统对安全性和可靠性要求极高,需要在投入运行前进行验证
- 适合采用形式化方法



#### 实例二: 网络公开课程网站

- 某公司准备开发一个大规模在线公开课程网站,支持学校将自己的课程录像、课件及参考资料等公布在网上,学生可以进行自主学习。
- 该系统将教育、娱乐和社交网络结合在一起,创造了一种新型的网络教育模式,对传统的高等教育模式带了很大的冲击。



#### 实例二: 网络公开课程网站

- 某公司准备开发一个大规模在线公开课程网站,支持学校将自己的课程录像、课件及参考资料等公布在网上,学生可以进行自主学习。
- 该系统将教育、娱乐和社交网络结合在一起,创造了一种新型的网络教育模式,对传统的高等教育模式带了很大的冲击。

#### 实例分析:

- 系统需求会经常发生变化,业务模式存在不确定性
- 系统应该易于维护和修改
- 适合采用增量模型或迭代模型



#### 思考讨论

#### 实验项目二:微信公共账号(自定账号名)

\_\_\_\_\_\_

- ■地点查询:输入搜索地点,可以返回地图位置以及该地点的图片。比如搜索"二校门",可以返回"二校门"的地图位置和相关图片。
- ■<mark>智能问答</mark>:对于初到清华的新生,肯定有些常见的问题,比如怎么选课、寝室怎么申请上网等,助手可以较智能地回答这些问题。
- ■人工服务: 这个人工服务的参与者既是新生,也应该是热心的学长学姐。新生在遇到自动回答处理不了的问题时,可以选择人工求助。新生可以使用文字或语音提问,学长学姐在空闲时可以主动去访问平台,系统返回一个求助任务列表,学长学姐可以选择解答。之后,新生可以查看自己提交的问题并听取解答,对解答满意之后可以关闭该问题。这个类似很多网站上的在线客服,不过本项目利用微信平台来完成,而且客服人员变成了"学长学姐"。
- ■其他功能:可选项,各□□□自由发挥提出□□□□□□。

### 思考讨论



你认为实验项目二适合采用哪一种过程模型? 说出你这样的理由。 对于你这样的过程模型,开发活动应该此何组织? 0 0 0

# 谢谢大家!

## **THANKS**

