

北京信息科技大学
BEIJING INFORMATION SCIENCE & TECHNOLOGY UNIVERSITY



北京信息科技大学 计算机学院
COMPUTER SCHOOL, BEIJING INFORMATION SCIENCE & TECHNOLOGY UNIVERSITY



软件工程

第一章 软件工程学概述

计算机学院 信息安全系 向昶

2025年9月



目录

01

软件危机

02

软件工程

03

软件生命周期

04

软件过程

05

内容梳理及实践建议



北京信息科技大学 计算机学院
COMPUTER SCHOOL, BEIJING INFORMATION SCIENCE & TECHNOLOGY UNIVERSITY

01

软件危机



主要内容



1.1 软件危机

1.2 软件工程

1.3 软件生命周期

1.4 软件过程

要解决的问题

解决方向

解决方案



1.1 软件危机

1.1.1 软件危机的介绍

在计算机软件的**开发**和**维护**过程中**所遇到**
的一系列严重问题。



讨论:

软件开发有那些困难和问题?



1.1 软件危机

软件危机的典型表现

- 1、对软件开发成本和进度的估计常常很不准确
- 2、用户对“已完成的”软件系统不满意的现象经常发生
- 3、软件产品的质量存在问题
- 4、软件维护困难（固有错误修改、加新功能、新硬件）



1.1 软件危机

软件危机的典型表现

5、软件文档问题（时效性问题、表达准确性问题）。

6、软件成本在计算机系统总成本中所占的比例逐年上升。

7、软件开发生产率提高的速度，远远跟不上计算机应用迅速普及深入的趋势。



1.1 软件危机

1.1.2 产生软件危机的原因

与软件本身特点有关

软件开发与维护的方法不正确有关



1.1 软件危机

1.1.2 产生软件危机的原因

与软件本身特点有关

1 管理和控制
软件开发过
程相当困难。

2 软件缺陷具有隐
蔽性和复杂性，
软件的缺陷有可
能在测试阶段未
能检测出来。

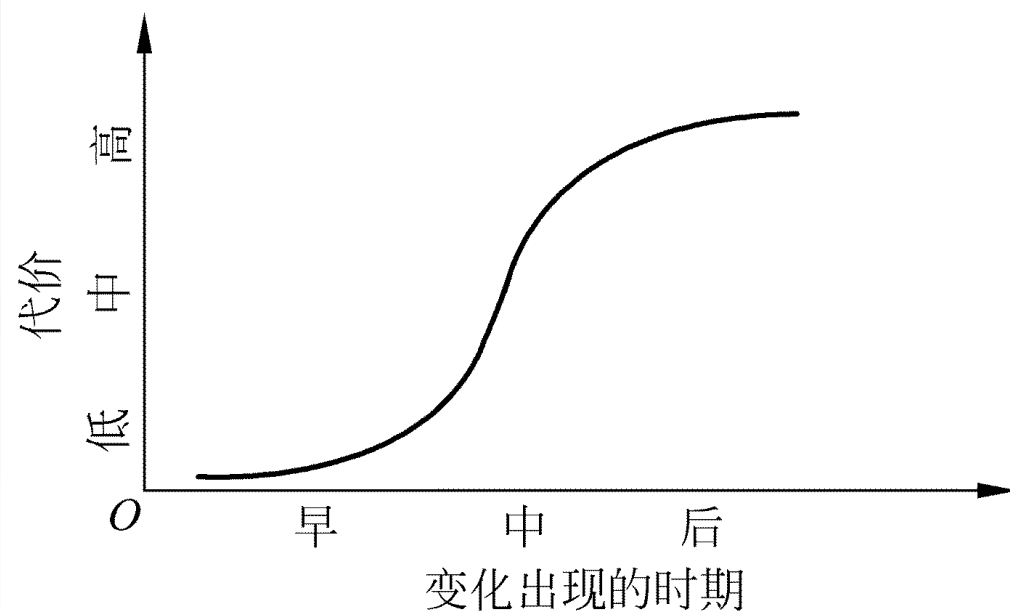
3 软件往往规模庞
大，而且程序
复杂性将随着程
序规模的增加而
呈指数上升。

4 忽视软件需求分
析的重要性，对
用户要求没有完
整准确的认识就
匆忙着手编写程
序。



1.1 软件危机

在软件开发的不同阶段进行修改需要付出的代价





1.1 软件危机



北京信息科技大学 计算机学院
COMPUTER SCHOOL, BEIJING INFORMATION SCIENCE & TECHNOLOGY UNIVERSITY

轻视软件“维护”，导致后续困难重重。



1.1 软件危机

1.1.3 消除软件危机的途径

需要**专门的学科**和**持续的研究**来指导软件开发与维护

1 首先应该对**计算机软件**有一个正确的认识。

2 充分认识到软件是**各类人员**协同配合，共同完成的工程项目。

3 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 持续**探索、开发和使用**更好的软件工具。



主要内容



北京信息科技大学 计算机学院
COMPUTER SCHOOL, BEIJING INFORMATION SCIENCE & TECHNOLOGY UNIVERSITY

02

软件工程



主要内容

1.1 软件危机



1.2 软件工程

1.3 软件生命周期

1.4 软件过程



1.2 软件工程

1.2.1 软件工程的介绍

软件工程概述

实践性非常强的学科

软件工程是指导计算机软件开发和维护的一门工程学科。

采用工程的概念、原理、技术和方法来开发与维护软件，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。



1.2 软件工程

软件具有的本质特性

软件工程关注于大型程序的构造

软件工程的中心课题是控制复杂性

软件经常变化

开发软件的效率非常重要

和谐地合作是开发软件的关键

必须有效地支持它的用户

两种背景的人创造产品



1.2 软件工程

1.2.2 软件工程的基本原理

1、用分阶段的生命周期计划严格管理

2、坚持进行阶段评审

3、实行严格的产品控制

4、采用现代程序设计技术

5、明确各个子目标及完成期限，跟踪进度

6、开发小组的人员应该少而精

7、承认不断改进软件工程实践的必要性



1.2 软件工程

软件工程方法学的三个要素

方法

- 完成软件开发的各项任务的技术方法，回答“怎样做”的问题

工具

- 为运用方法而提供的自动的或半自动的软件工程支撑环境

过程

- 为了获得高质量的软件所需要完成的一系列任务的框架，它规定了完成各项任务的工作步骤



1.2 软件工程

1.2.3 软件工程方法学

1、传统方法学

- 传统方法学也称为生命周期方法学或结构化范型。它采用结构化技术(结构化分析、结构化设计和结构化实现)来完成软件开发的各项任务，并使用适当的软件工具或软件工程环境来支持结构化技术的运用。

2、面向对象方法学

- 与传统方法相反，面向对象方法把数据和行为看成是同等重要的，它是一种以数据为主线，把数据和对数据的操作紧密地结合起来的方法。



1.2 软件工程

面向对象方法学：

概念：与传统方法相反，面向对象方法把数据和行为看成是同等重要的，它是一种以数据为主线，把数据和对数据的操作紧密地结合起来的方法。



1.2 软件工程

面向对象方法学的优点：

- 降低了软件产品的复杂性，提高了软件的可理解性，简化了软件的开发和维护工作。
- 面向对象方法特有的继承性和多态性，进一步提高了面向对象软件的可重用性。



软件
危机

典型表现

消除的途径

软件
工程

基本原理

传统方法学
面向对象方
法学

软件生命周期



北京信息科技大学 计算机学院
COMPUTER SCHOOL, BEIJING INFORMATION SCIENCE & TECHNOLOGY UNIVERSITY

03

软件生命周期



主要内容

1.1 软件危机

1.2 软件工程



1.3 软件生命周期

1.4 软件过程



1.3 软件生命周期

软件生命周期由软件定义(又称系统分析)、软件开发和运行维护(也称为软件维护)3个时期组成, 每个时期又进一步划分成若干个阶段。

软件定义



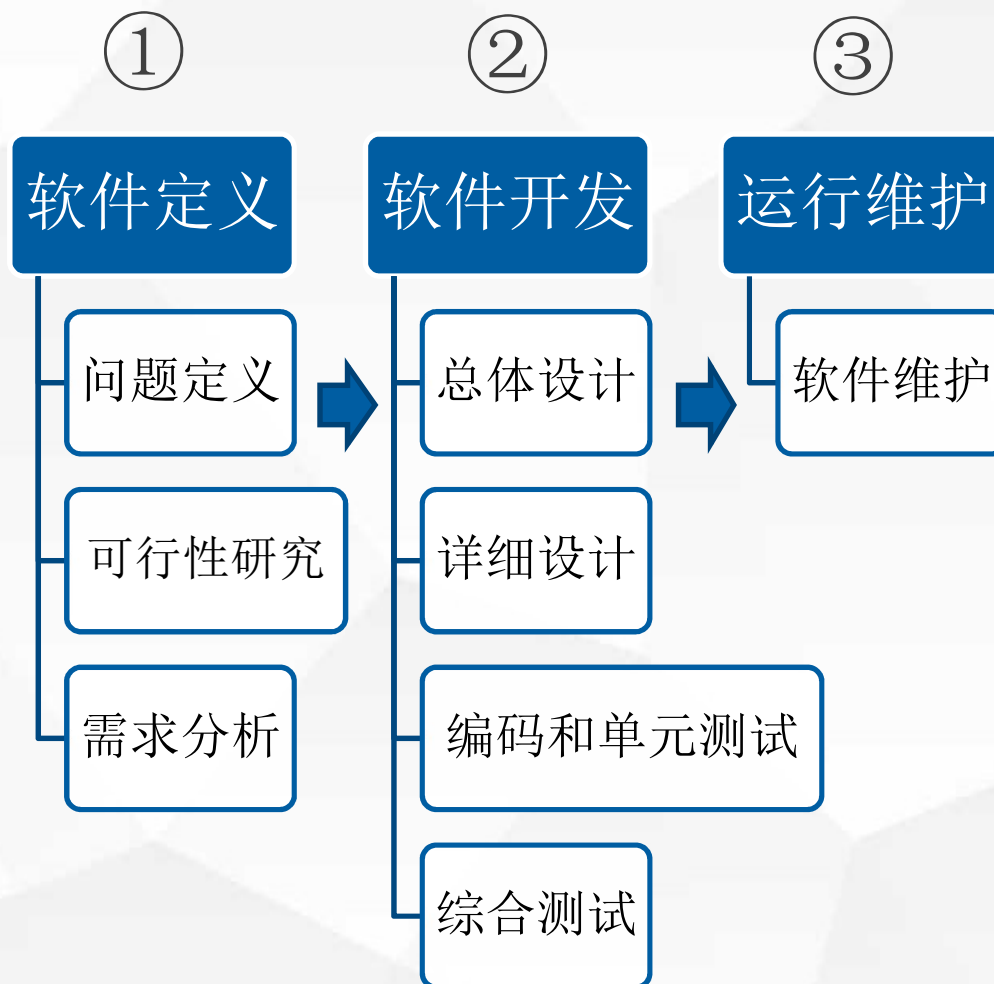
软件开发



运行维护



1.3 软件生命周期





04

软件过程



主要内容

1.1 软件危机

1.2 软件工程

1.3 软件生命周期



1.4 软件过程



1.4 软件过程

软件过程是为了获得高质量软件所需要完成的一系列任务的**框架**，它规定了完成各项任务的工作**步骤**。

软件过程描述为了开发出客户需要的软件，什么人（**who**）、在什么时候（**when**）、做什么事（**what**）以及怎样（**how**）做这些事以实现某一个特定的具体目标。



1.4 软件过程

1.4.1 瀑布模型

瀑布模型一直是唯一被广泛采用的生命周期模型，现在它仍然是软件工程中应用得最广泛的过程模型。

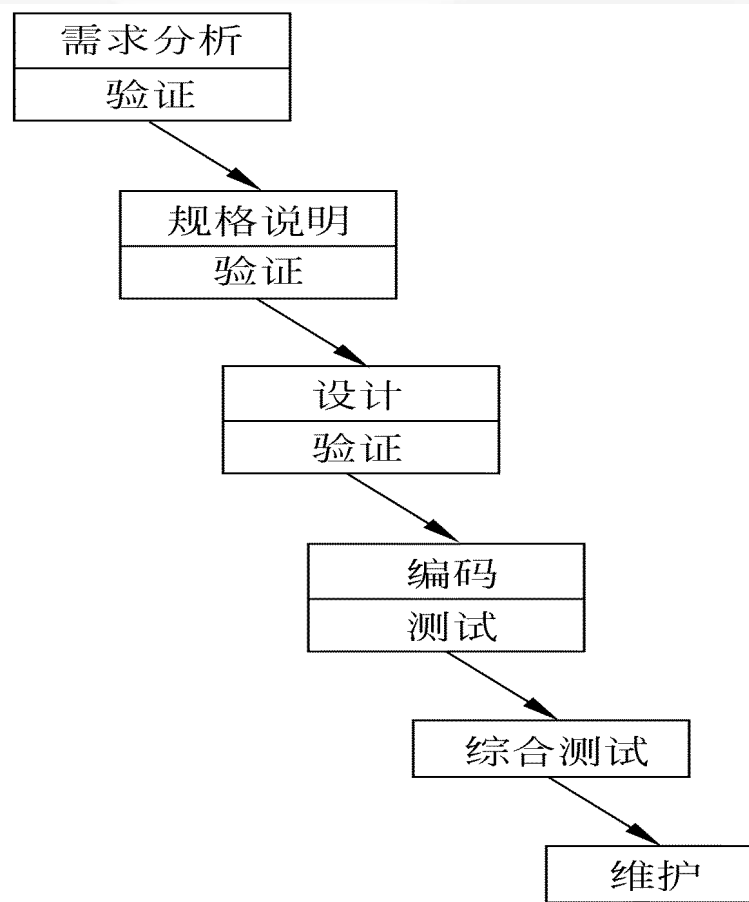




顺序完成



北京信息科技大学 计算机学院
COMPUTER SCHOOL, BEIJING INFORMATION SCIENCE & TECHNOLOGY UNIVERSITY



传统的瀑布模型



1.4 软件过程

1.4.1 瀑布模型

按照传统的瀑布模型开发软件，有下述的几个特点。

a) 阶段间具有顺序性和依赖性：

两重含义：

- ①必须等前一阶段的工作完成之后，才能开始下一阶段的工作；
- ②前一阶段的输出文档就是后一阶段的输入文档，因此，只有前一阶段的输出文档正确，下一阶段的工作才能获得正确的结果。



1.4 软件过程

1.4.1 瀑布模型

b) 推迟实现的观点

瀑布模型在编码之前设置了系统分析与系统设计的各个阶段，分析与设计阶段的基本任务规定，在这两个阶段主要考虑目标系统的逻辑模型，不涉及软件的物理实现。



1.4 软件过程

1.4.1 瀑布模型

c) 质量保证的观点:

软件工程的基本目标是优质、高产。为了保证所开发的软件的质量，在瀑布模型的每个阶段都应坚持两个重要做法。

- ① 每个阶段都必须完成规定的文档
- ② 每个阶段结束前都要对所完成的文档进行评审



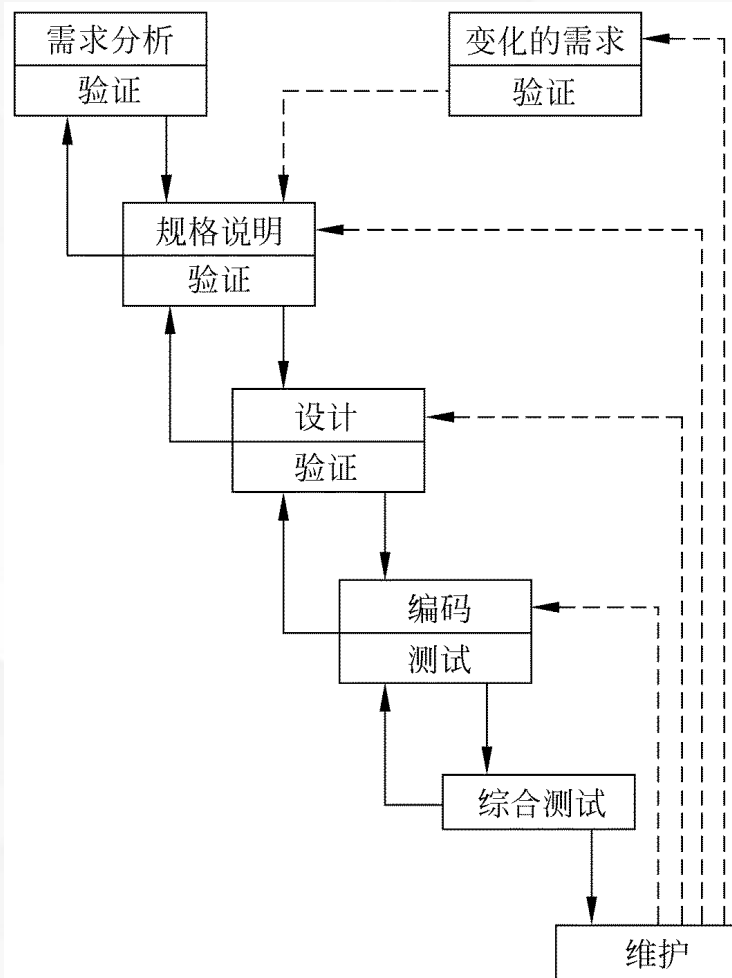
1.4 软件过程

1.4.1 瀑布模型

传统的瀑布模型过于理想化了，事实上，人在工作过程中不可能不犯错误。实际的瀑布模型是带“反馈环”的。



- 1、图中实线箭头表示开发过程，
虚线箭头表示维护过程。
- 2、实际的瀑布模型当在后面阶段发现前面阶段的错误时，需要沿图中左侧的反馈线返回前面的阶段，修正前面阶段的产品之后再回来继续完成后面阶段的任务。





1.4 软件过程

1.4.1 瀑布模型

瀑布模型有许多优点：

- ① 可强迫开发人员采用规范的方法
- ② 严格地规定了每个阶段必须提交的文档；
- ③ 要求每个阶段交出的所有产品都必须经过质量保证小组的仔细验证。



1.4 软件过程

1.4.1 瀑布模型

瀑布模型的缺点：

- ① 用户较晚才能看到软件产品
- ② 变更不易
- ③ 有时最终的软件产品不能满足用户需求



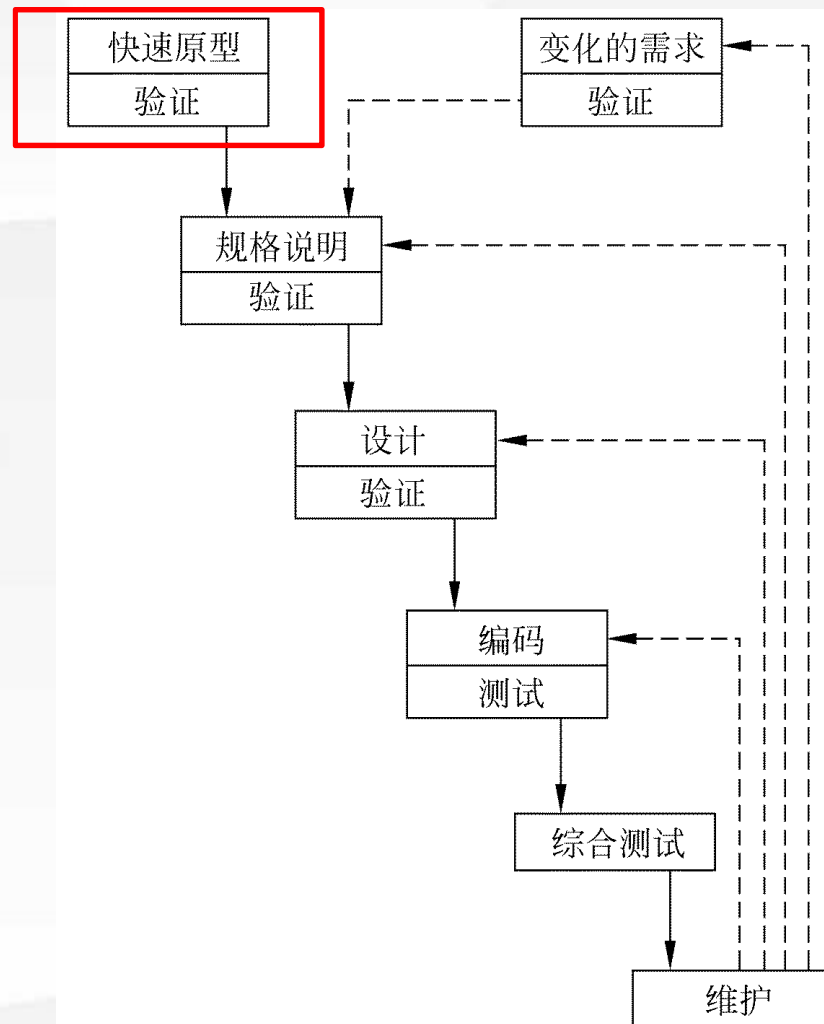
1.4 软件过程

1.4.2. 快速原型模型

概念：

快速原型是快速建立起来的可以在计算机上运行的程序，它所能完成的功能往往是最终产品能完成的功能的一个子集。

1.4.2. 快速原型模型

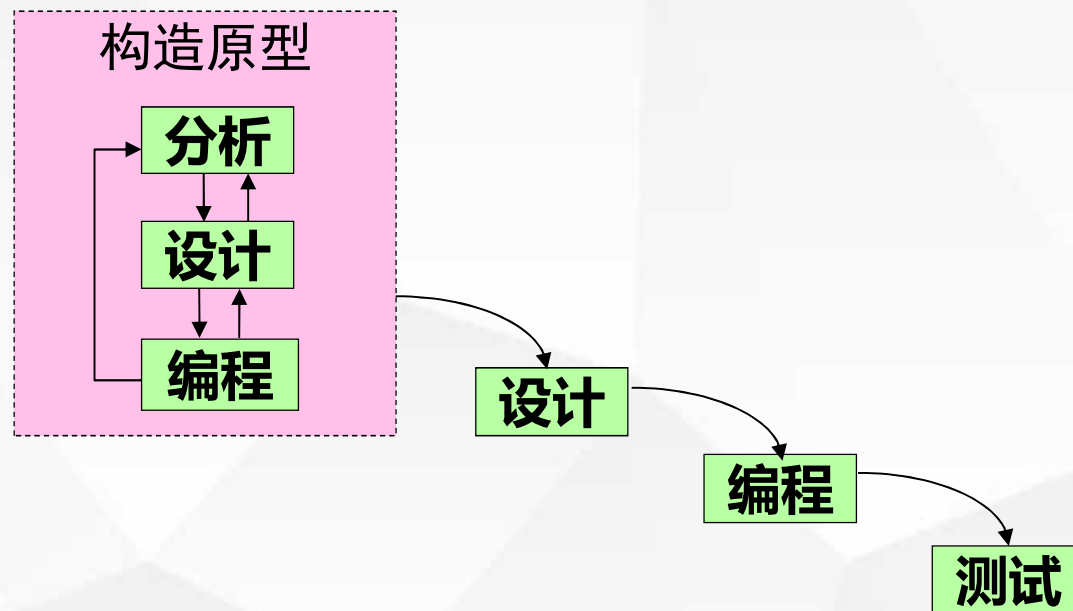


实线箭头--开发过程
虚线箭头--维护过程



1.4 软件过程

1.4.2. 快速原型模型





1.4 软件过程

1.4.2. 快速原型模型

- 快速原型模型可以在早期看到系统原型
- 软件产品的开发基本上是线性顺序进行的



1.4 软件过程

1.4.2. 快速原型模型

优点:

- 原型系统与用户交互过
- 开发人员对系统的理解和认识较深入，避免一些错误和误解快速原型。



1.4 软件过程

1.4.3. 增量模型

使用增量模型开发软件时，把软件产品作为一系列的增量构件来设计、编码、集成和测试。

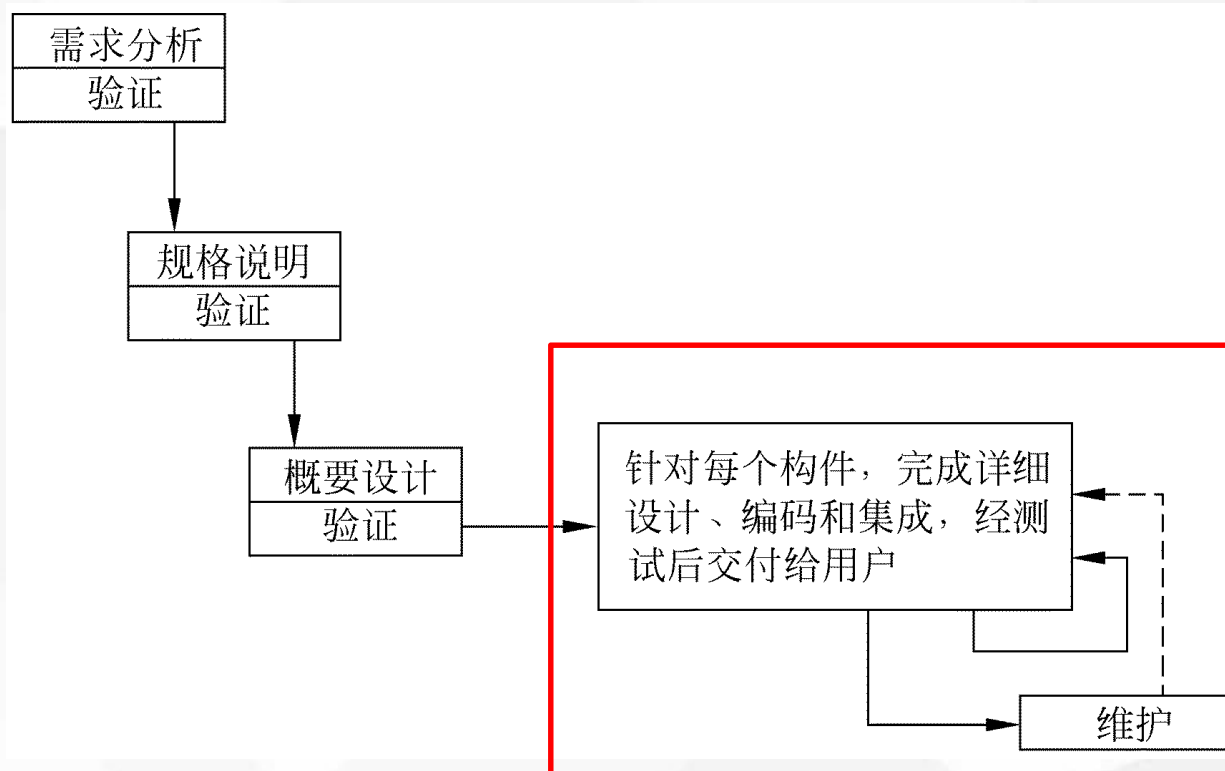


图 增量模型



1.4 软件过程

1.4.3. 增量模型

优点:

能在较短时间内向用户提交可完成部分工作的产品。

逐步增加产品功能可以使用户有较充裕的时间学习和适应新产品，从而减少一个全新的软件可能给客户组织带来的冲击。



1.4 软件过程

1.4.3. 增量模型

使用增量模型的困难：

增量开发时，不能破坏原来已经开发出的产品，不能影响已有用户使用。

软件的体系结构设计要考虑增量开发，便于向现有产品中加入新构件的过程必须简单、方便。

增量迭代



北京信息科技大学 计算机学院
COMPUTER SCHOOL, BEIJING INFORMATION SCIENCE & TECHNOLOGY UNIVERSITY

系统

子系统1

子系统2

子系统3



系统

子系统1

子系统2

子系统3

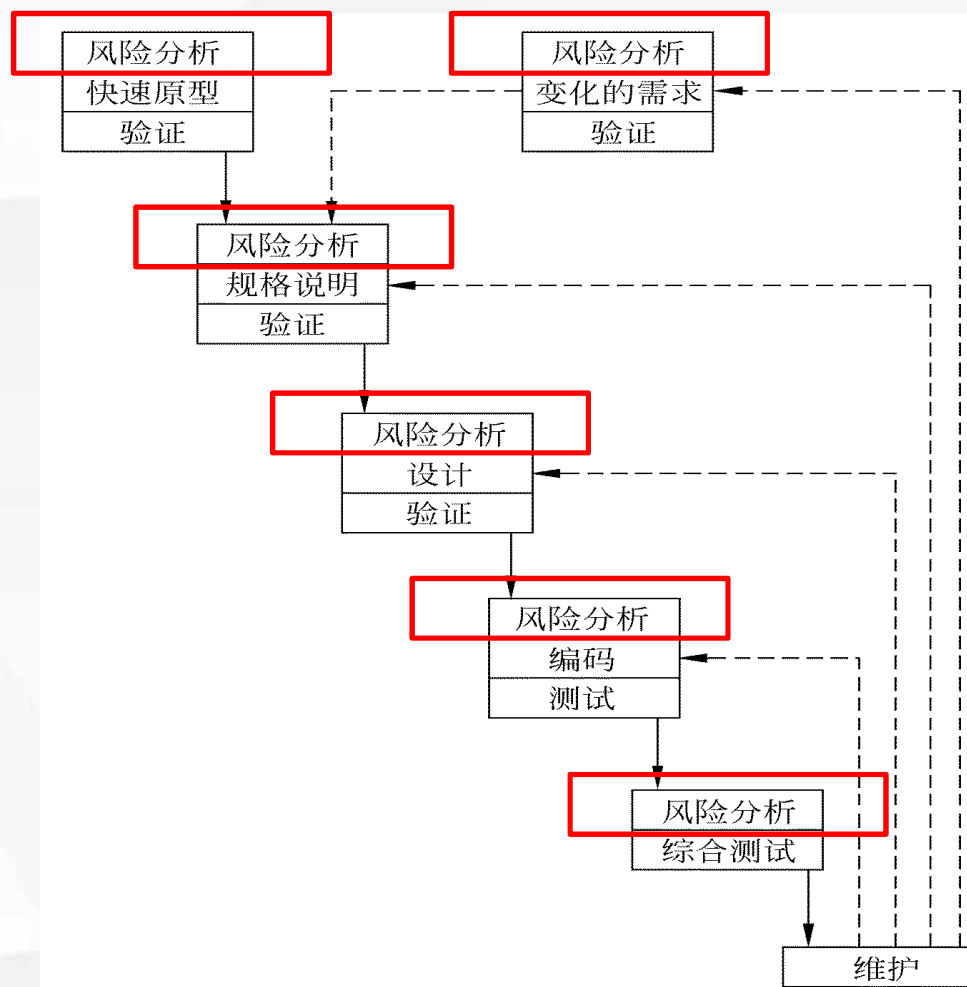


1.4 软件过程

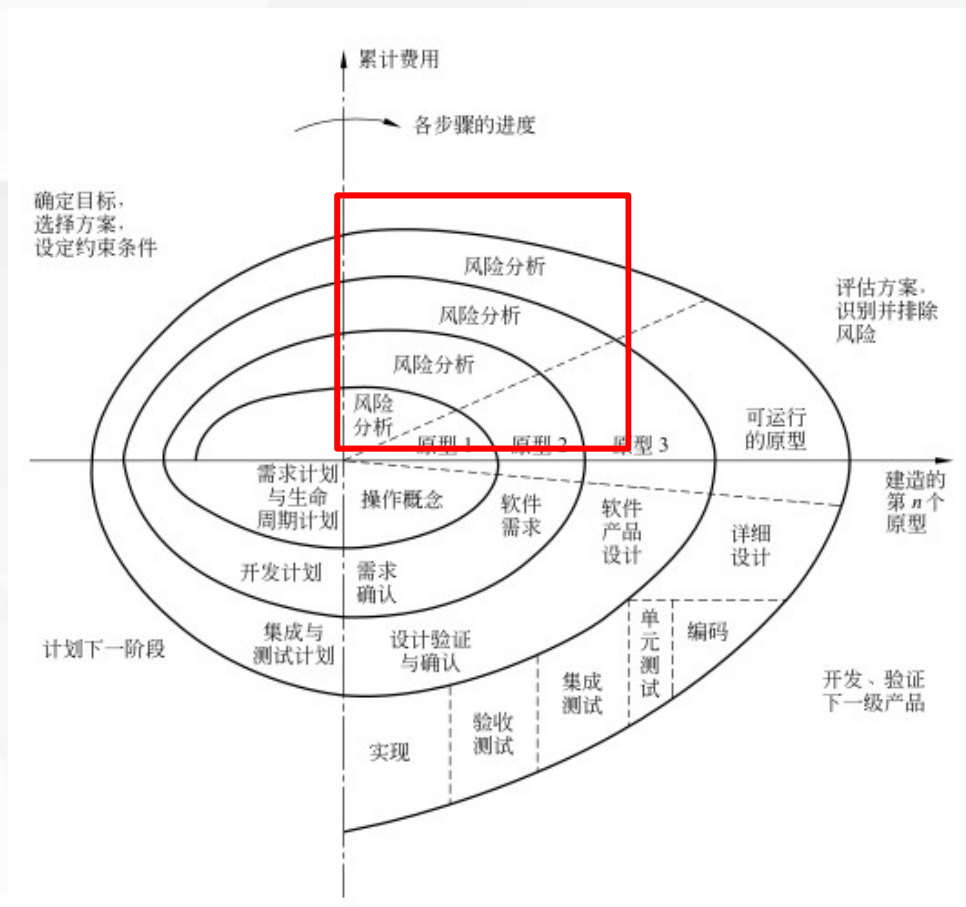
1.4.4 螺旋模型

使用原型及其他方法来尽量降低风险。

是在每个阶段之前都增加了风险分析
过程的快速原型模型。



简单的螺旋模型



- 适合大规模软件开发
- 风险驱动

完整的螺旋模型



1.4 软件过程

1.4.7. 敏捷过程与极限编程

敏捷过程为了使软件开发团队具有高效工作和快速响应变化的能力，17位著名的软件专家于2001年2月联合起草了敏捷软件开发宣言。敏捷软件开发宣言由下述4个简单的价值观声明组成。



05

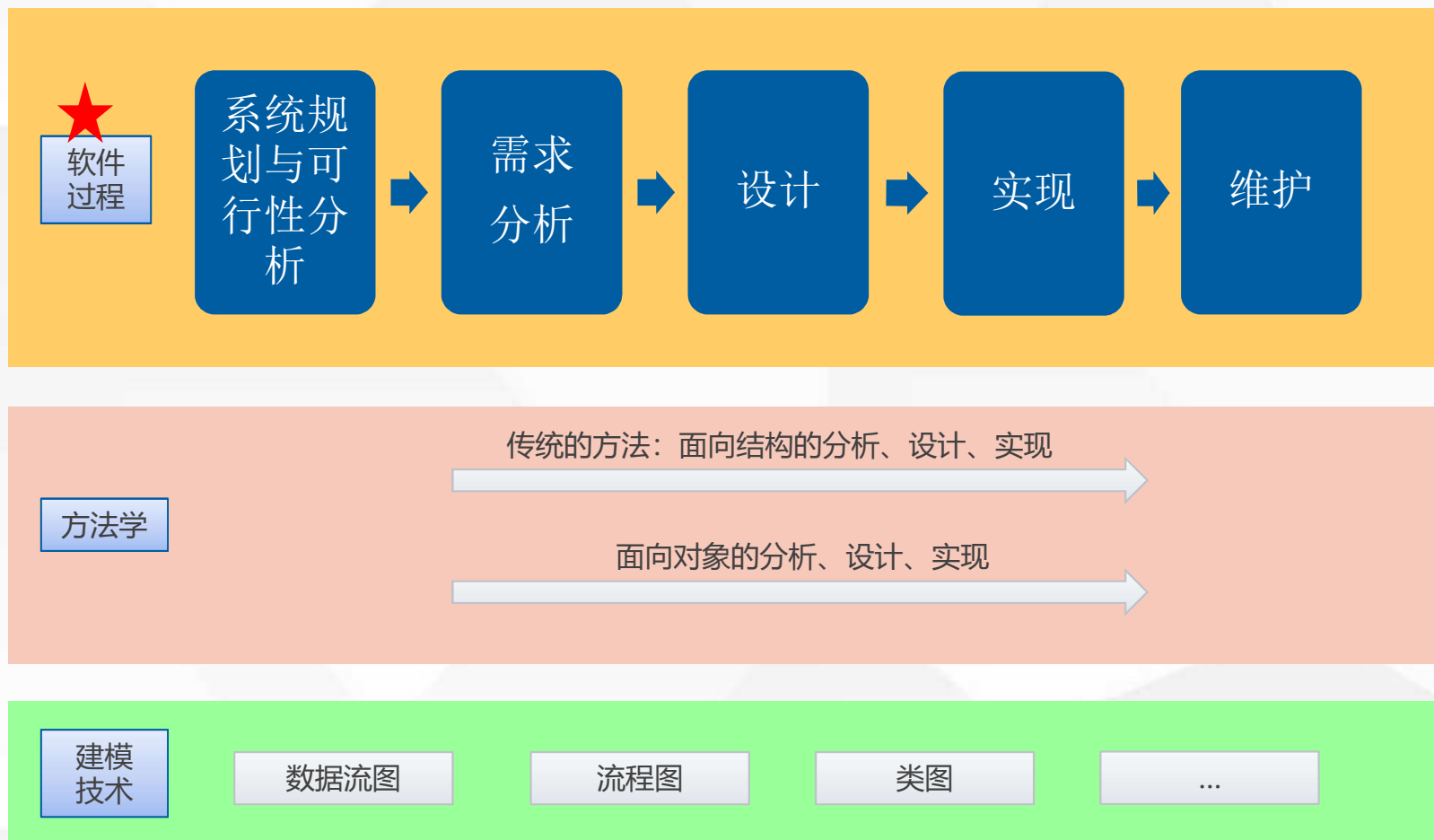
内容梳理及 作业



内容梳理



北京信息科技大学 计算机学院
COMPUTER SCHOOL, BEIJING INFORMATION SCIENCE & TECHNOLOGY UNIVERSITY





建议学习的工具等



北京信息科技大学 计算机学院
COMPUTER SCHOOL, BEIJING INFORMATION SCIENCE & TECHNOLOGY UNIVERSITY

- Eclipse
- Git 软件版本配置管理
- Axure 原型开发工具
- 百度飞浆、PyTorch等
- 前端、后端、MySQL

作业

- 复习第一章
- 确定项目
- 请各组组长，**周日晚10点之前**把项目题目发给课代。



北京信息科技大学 计算机学院
COMPUTER SCHOOL, BEIJING INFORMATION SCIENCE & TECHNOLOGY UNIVERSITY



本章结束