

常见的软件生存周期模型

- 瀑布模型
- 增量模型
- 演化模型
- 喷泉模型

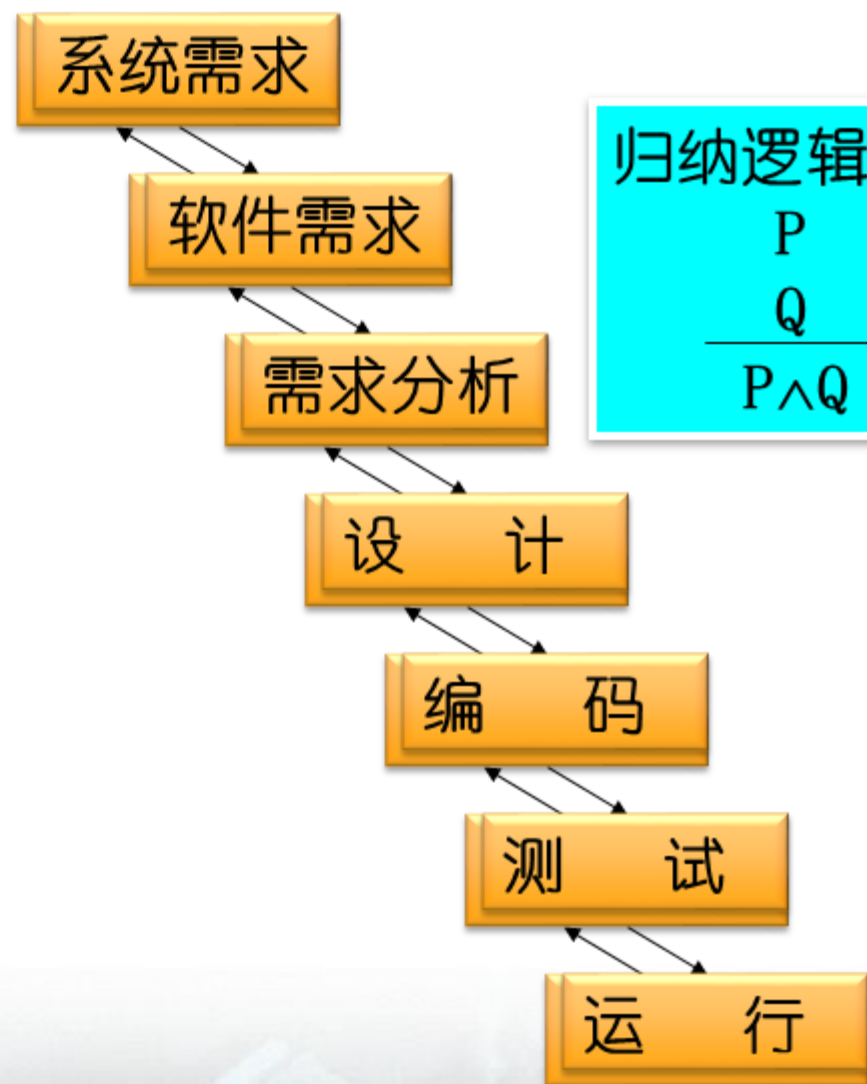
2) 瀑布模型

1970年, W.Royce

(1) 瀑布模型将软件生存周期的各项活动规定为依固定顺序而连接的若干阶段工作;

(2) 瀑布模型规定了每一阶段的输入, 以及本阶段的工作成果, 作为输出传入下一阶段。

----软件生存周期模型



归纳逻辑:

$$\begin{array}{r} P \\ Q \\ \hline P \wedge Q \end{array}$$



北京大学

常见的软件生存周期模型

- 瀑布模型
- 增量模型
- 演化模型
- 喷泉模型

- (1) 项目的开发依次经过：需求、设计、编码和单元测试、集成以及维护 - 这一基本路径。
- (2) 通过每一阶段, 提交以下产品：软件需求规约、设计文档、实际代码、测试用例、最终产品等。工作产品（又称可提交的产品, Deliverables）流经“正向”开发的基本步骤路径。
- (3) “反向”步骤流表示对前一个可提交产品的重复变更（又称为“返工”（Rework））。
 - 由于所有开发活动的非确定性，因此是否需要重复变更，这仅在下一个阶段或更后的阶段才能认识到。
 - 返工不仅在以前阶段的某一地方需要，而且对当前正在进行的工作也是需要的。



常见的软件生存周期模型

- 瀑布模型
- 增量模型
- 演化模型
- 喷泉模型

关于瀑布模型的几点说明

(1) 瀑布模型的优点

虽然瀑布模型是一个比较“老”的、甚至过时的开发模型，但其优点为：

- ① 在决定系统怎样做之前，存在一个需求阶段，鼓励对系统“做什么”进行规约（即设计之前的规约）。
- ② 在建造构件之前，存在一个设计阶段，鼓励规划系统结构（即编码之前的设计）。
- ③ 在每一阶段结束时进行复审，允许获取方和用户的参与。
- ④ 前一步工作产品可作为下一步被认可的、文档化的基线。允许基线和配置早期接受控制。



常见的软件生存周期模型

- 瀑布模型

- 增量模型

- 演化模型

- 喷泉模型

----软件生存周期模型

(2) 瀑布模型存在的不足

❶ 客户必须能够完整、正确和清晰地表达他们的需求；开发人员一开始就必须理解需求。



❷ 缺乏灵活性。一旦软件需求存在偏差，就会导致开发出的软件产品不能满足用户的实际要求。

❸ 在一个项目的早期阶段，过分地强调了基线和里程碑处的文档，可能要花费更多的时间，用于建立一些用处不大的文档。

❹ 直到项目结束之前，都不能演示系统的能力，增加了项目的风险。



常见的软件生存周期模型

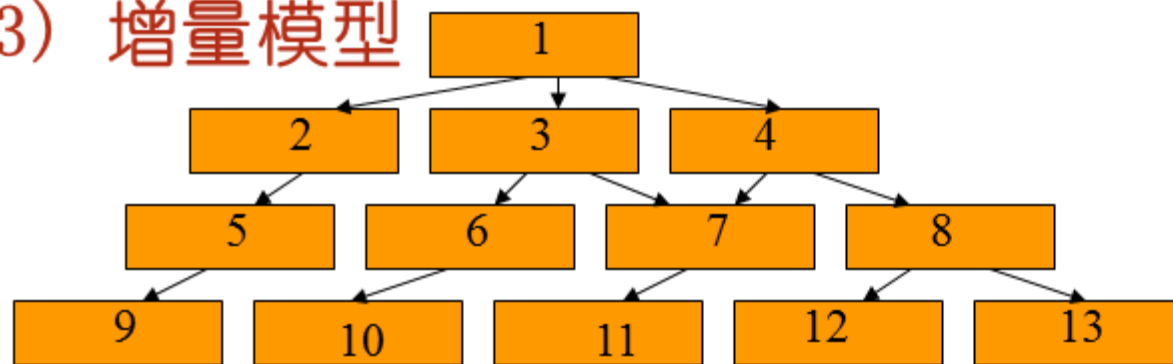
- 瀑布模型
- 增量模型
- 演化模型
- 喷泉模型



----软件生存周期模型



3) 增量模型



增量1

{ 1, 2, 5, 9 }

增量2

{ 3, 6, 7, 4, 10, 11 }

增量3

{ 8, 12, 13 }

增量1

增量2

增量3

管理

增量规约

增量设计

纠错性分析

增量实现

该模型有一个假设，即需求可以分段，成为一系列增量产品，每一增量可以分别地开发。



北京大学

常见的软件生存周期模型

- 瀑布模型
- 增量模型
- 演化模型
- 喷泉模型



----软件生存周期模型



关于增量模型的几点说明：

(1) 增量模型的优点

作为瀑布模型的第一个变体，具有瀑布模型的所有优点。此外，它还有以下优点：

- ① 第一个可交付版本所需要的成本和时间是很少的；
- ② 开发由增量表示的小系统所承担的风险是不大的；
- ③ 由于很快发布了第一个版本，因此可以减少用户需求的变更；
- ④ 允许增量投资，即在项目开始时，可以仅对一个或两个增量投资。

注：如果采用增量投资方式，那么客户就可以对一些增量进行招标。然后，开发人员按提出的截止期限进行增量开发，这样客户就可以用多个契约来管理组织的资源和成本。



北京大学

常见的软件生存周期模型

- 瀑布模型
- 增量模型
- 演化模型
- 喷泉模型

----软件生存周期模型

(2) 缺点:



如果增量模型不适于某些项目，或使用有误，则有以下缺点：

- ❶ 如果没有对用户的变更要求进行规划，那么产生的初始增量可能会造成后来增量的不稳定；
- ❷ 如果需求不像早期思考的那样稳定和完整，那么一些增量就可能需要重新开发，重新发布；
- ❸ 管理发生的成本、进度和配置的复杂性，可能会超出一些组织的能力。



北京大学

常见的软件生存周期模型

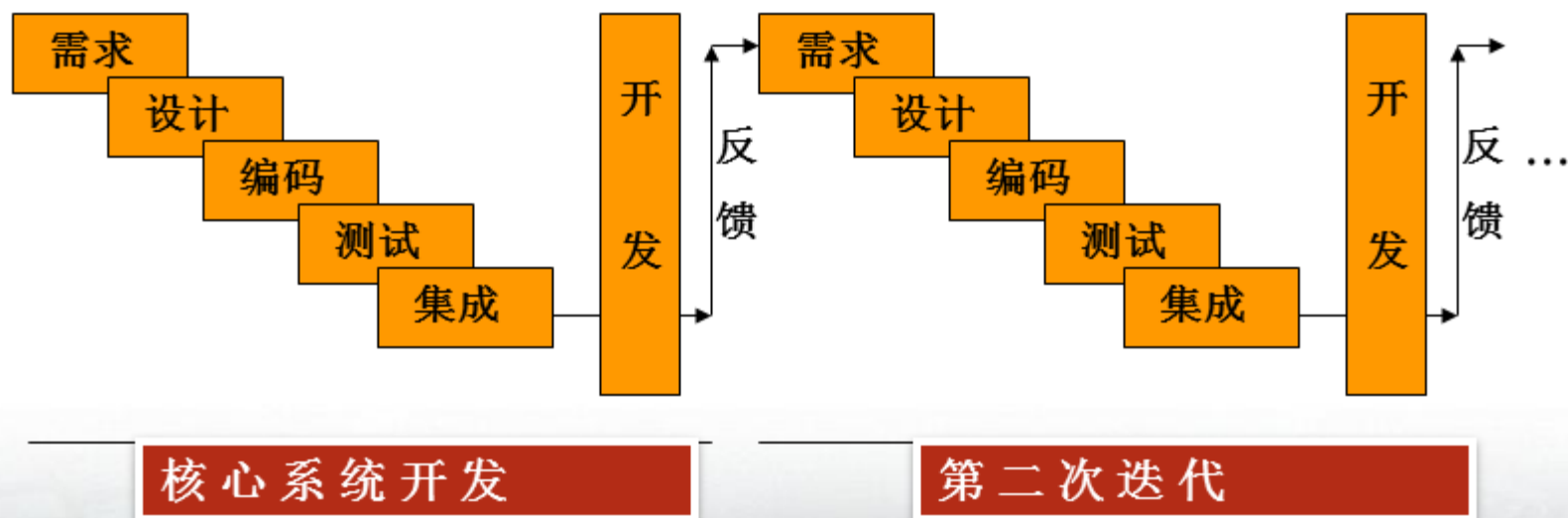
- 瀑布模型
- 增量模型
- 演化模型
- 喷泉模型

----软件生存周期模型

4) 演化模型 (Evolutionary model)

是一种有弹性的过程模式，由一些小的开发步组成，每一步历经需求分析、设计、实现和验证，产生软件产品的一个增量。通过这些迭代，完成最终软件产品的开发。

- 针对事先不能完整地定义需求的软件开发
- 针对用户的核心需求,开发核心系统
- 根据用户的反馈,实施活动的迭代



北京大学

常见的软件生存周期模型

- 瀑布模型
- 增量模型
- 演化模型
- 喷泉模型



----软件生存周期模型

5) 喷泉模型

- 特征：迭代
无缝
- 与面向对象技术的
关系

