**软件生存周期**

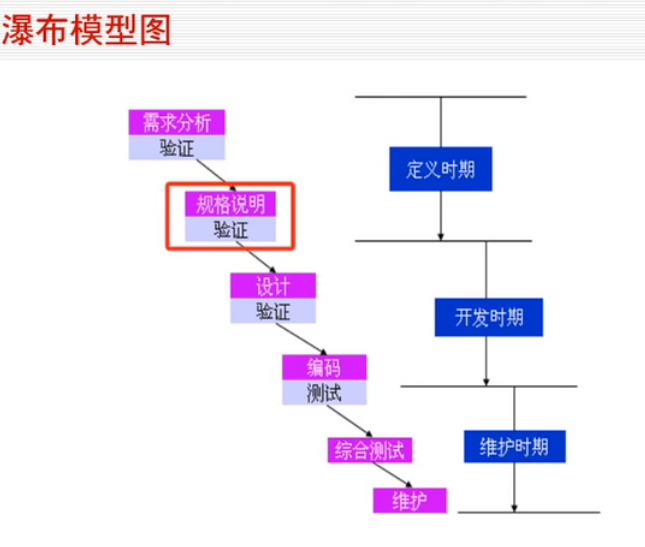
可行性研究与计划

需求分析

总体设计

**常用软件过程模型**

**瀑布模型**



1.阶段具有依赖性和顺序性

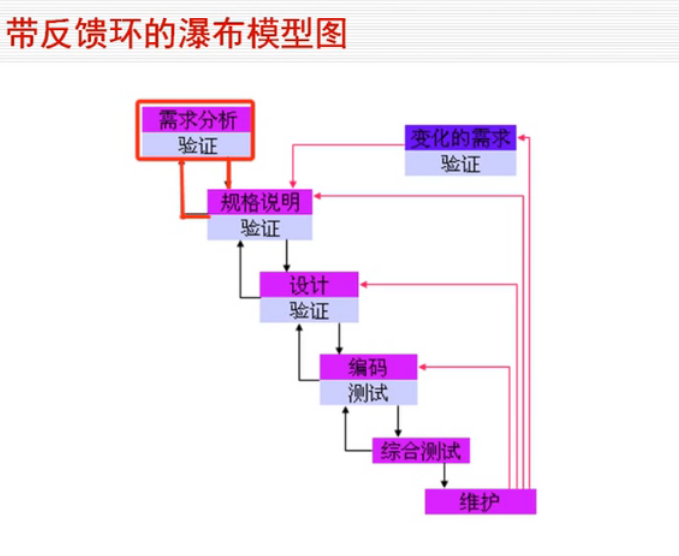
2.推迟实现观点

瀑布模型在编码前设置系统分析、系统设计，推迟程序物理实现，保证工期扎实

3.质量保证观点

坚持每个阶段必须完成完整，准确的文档

坚持每个阶段结束前对文档评审

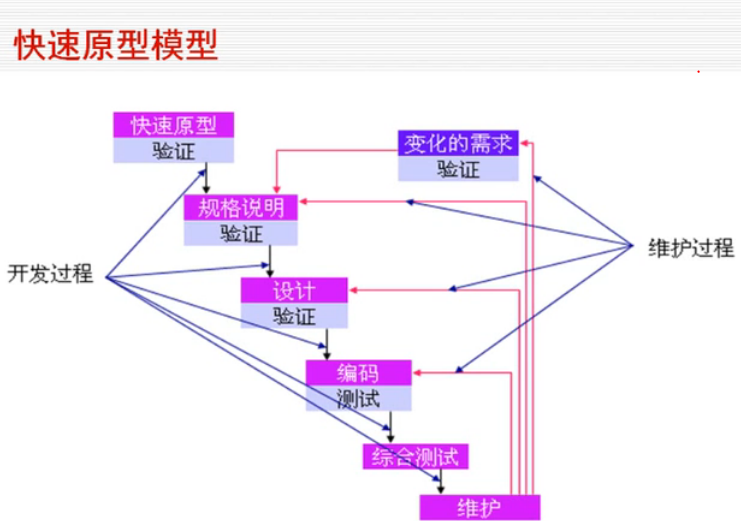


优点：提高软件质量、降低维护成本，缓解软件危机

缺点：缺乏灵活性、无法解决需求不明确的问题。用户不经过实践提出的需求不切实际。

**快速原型模型**

快速建立反映用户主要需求的原型系统，反复由用户评价更改需求

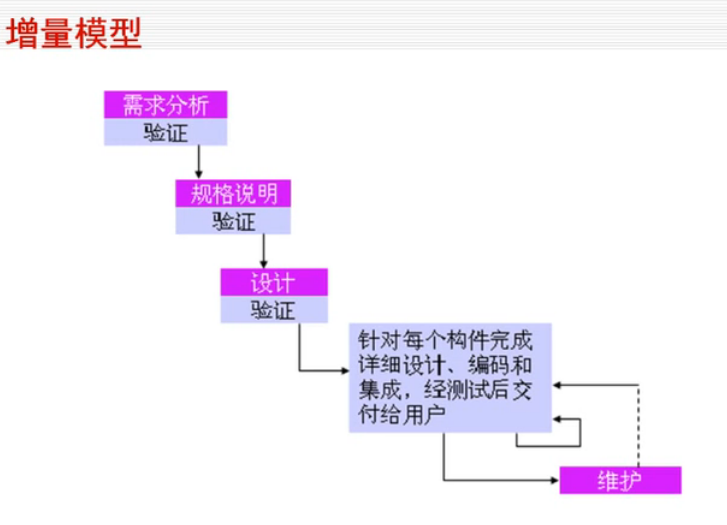


优点：确定需求优于瀑布模型；提供学习手段，通过开发原型和演示开发模型对开发者和使用者了解系统都有积极作用；有的软件原型可以作为最终产品的一部分

缺点：产品质量可能偏低，原型系统内部结构可能不好

**增量模型**

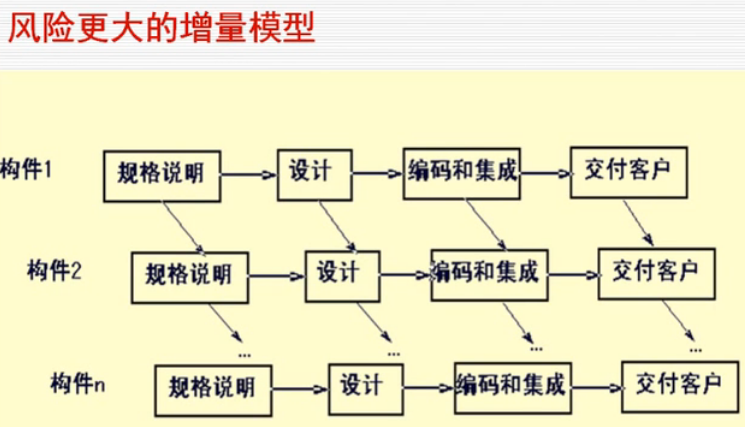
开发软件时将软件产品作一系列增量构件设计、编码、集成和测试；分批向用户提交产品



优点：较短时间向用户提交有用的工作产品；用户有时间学习产品使用；软件结构必须开放，方便向现有产品加入新构件

缺点：做到第三个优点比较困难；

风险： 各构件构建



**螺旋模型**

加入风险分析，常指导大型软件项目开发

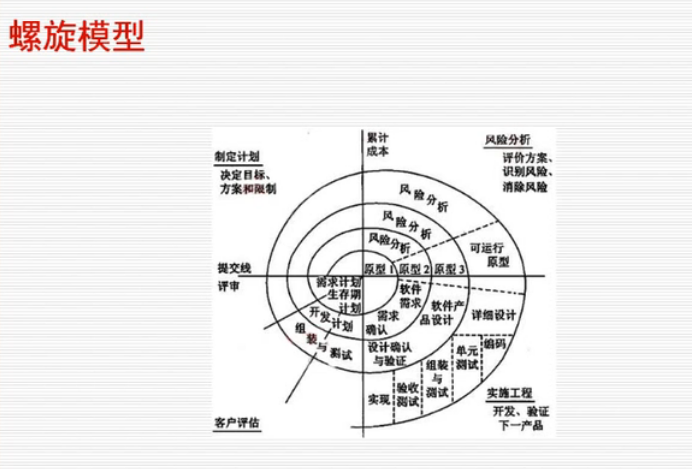
笛卡尔坐标四象限表达四方面活动：

制定计划：确定目标、选定方案、设定约束条件

风险分析：评估方案，识别和消除风险

实施工程：软件开发

客户评估：评估开发工作，计划下一阶段工作



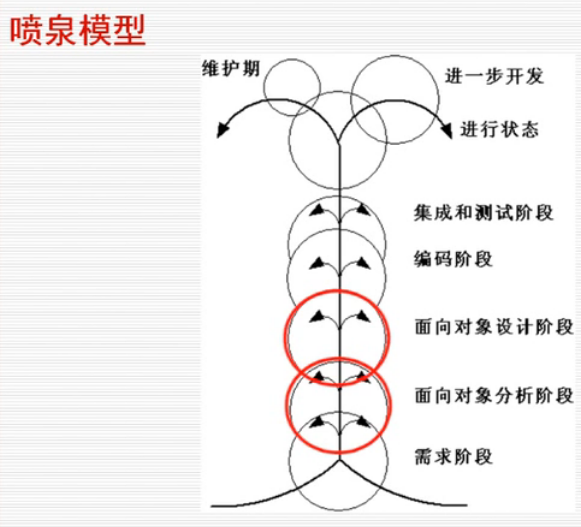
优点：大型软件开发项目有较好的风险控制

缺点：需要良好的风险评估经验；契约开发通常需要事先指定过程和产品

**喷泉模型**

面向对象方法的生命周期模型

迭代：求精，系统某部分被重复工作多次，相关功能在每次迭代中逐渐加入

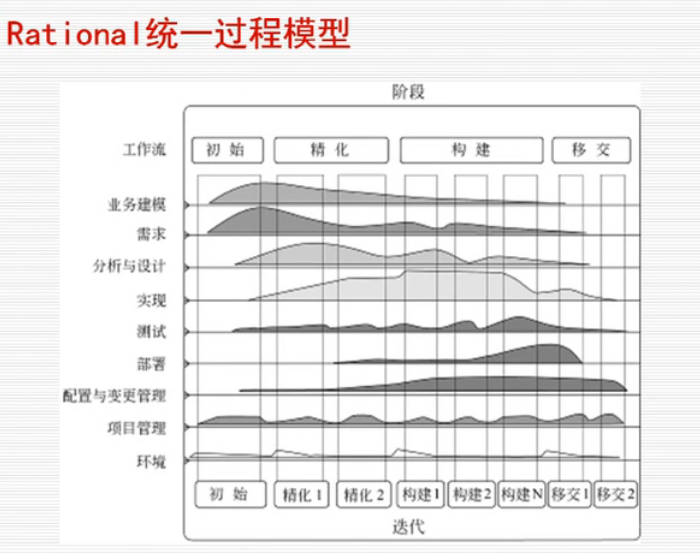


优点：无缝，同步开发，提升效率节省时间

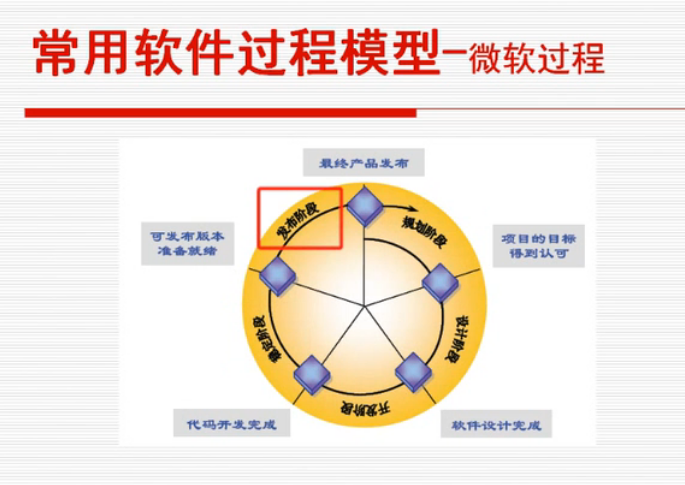
缺点：可能随时附加各种信息、需求、资料，文档管理困难

**Rational 统一过程**

强调以迭代和建增方式开发软件



**微软公司软件开发过程**



递进式的开发策略