



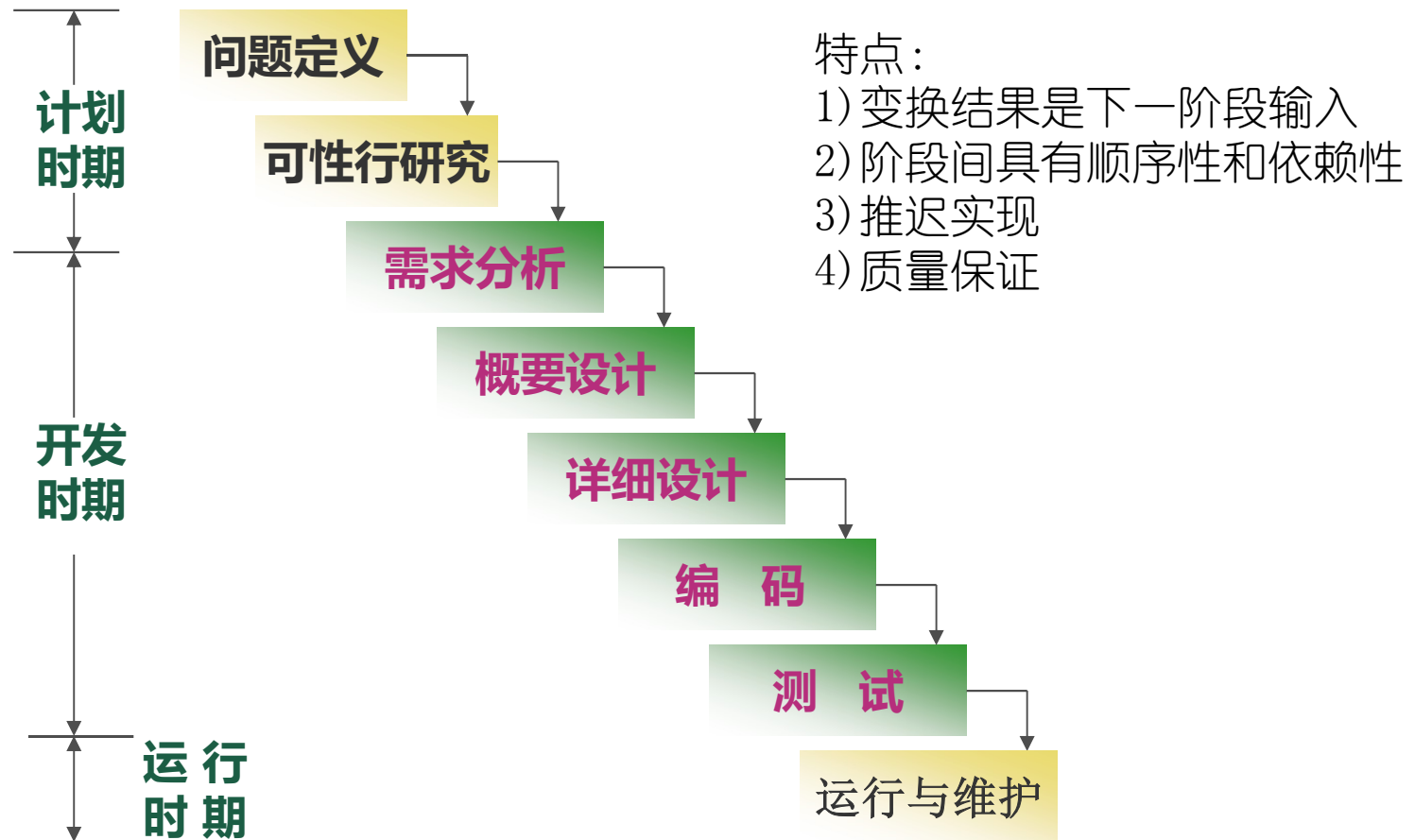
软件过程与项目管理

2021年9月25日星期六

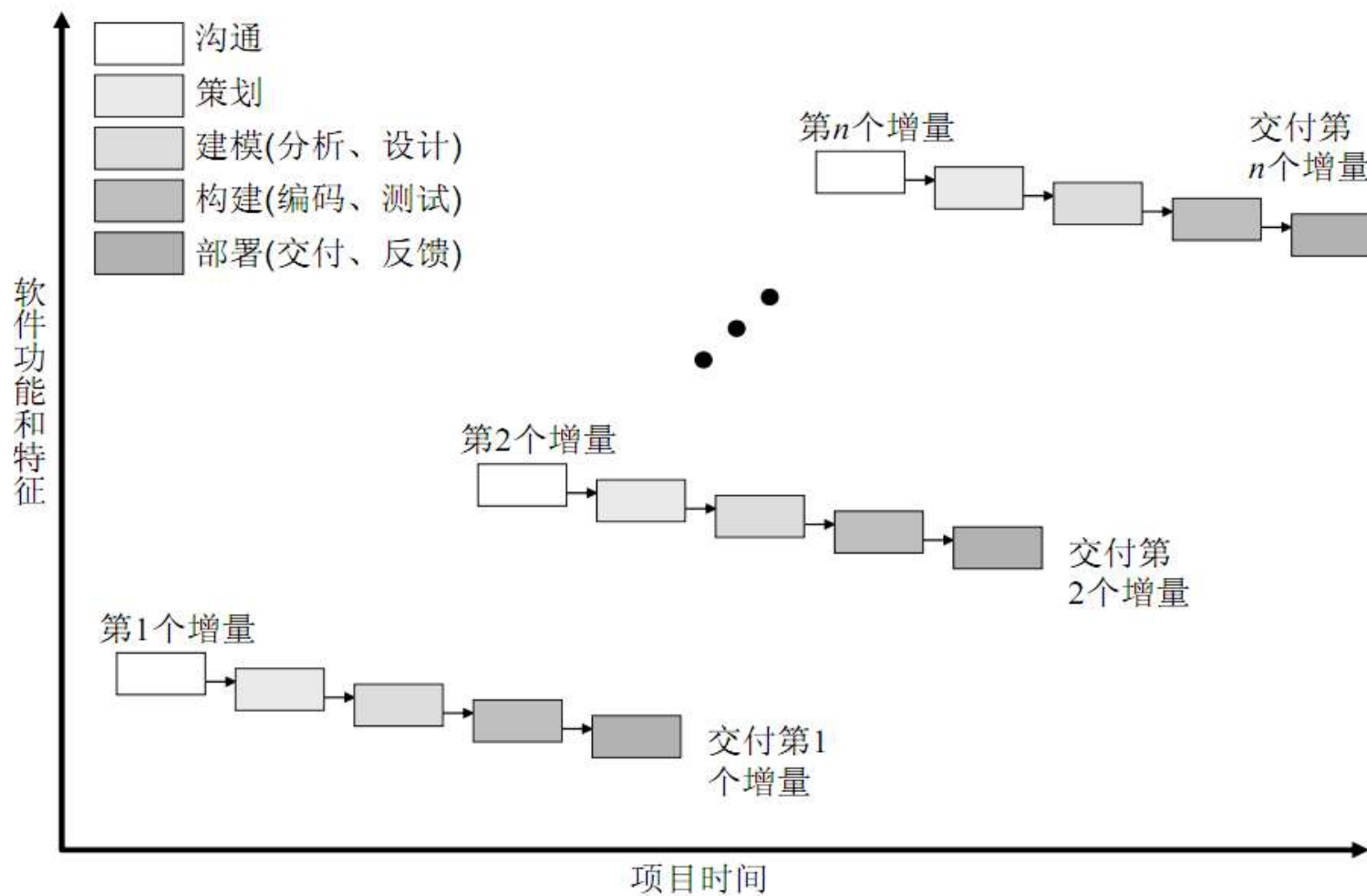
软件过程模型

- ◆根据模型提出的时间不同，可以分为传统工程过程模型和现代软件工程过程模型
- ◆传统软件过程过程模型的主要代表是：编码修正模型、瀑布模型、演化模型和螺旋模型；IBM公司的统一过程模型（RUP）、敏捷过程（AP）和微软解决方案（MSF）等则是现代软件工程过程模型的主要代表

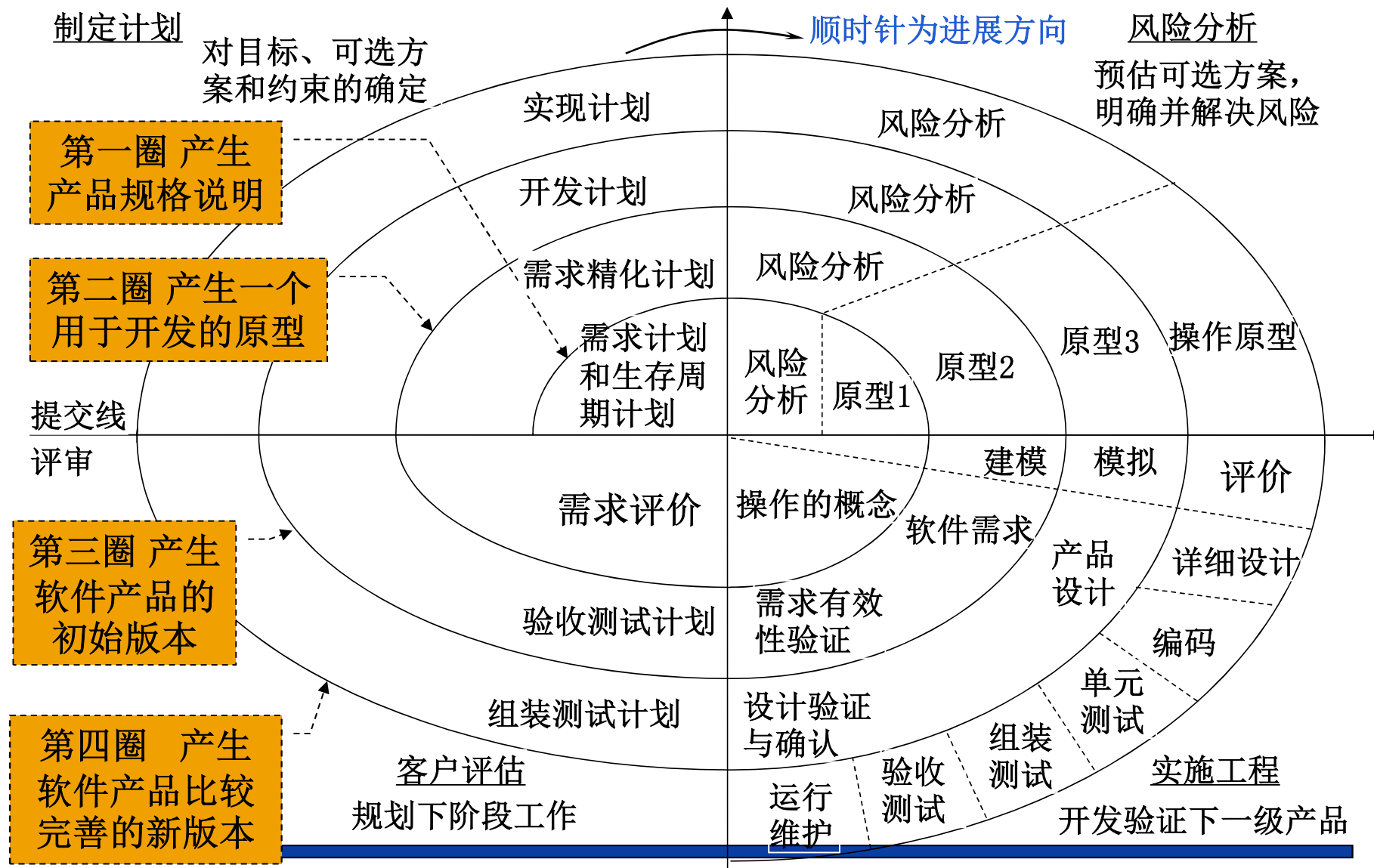
瀑布模型 Waterfall Model



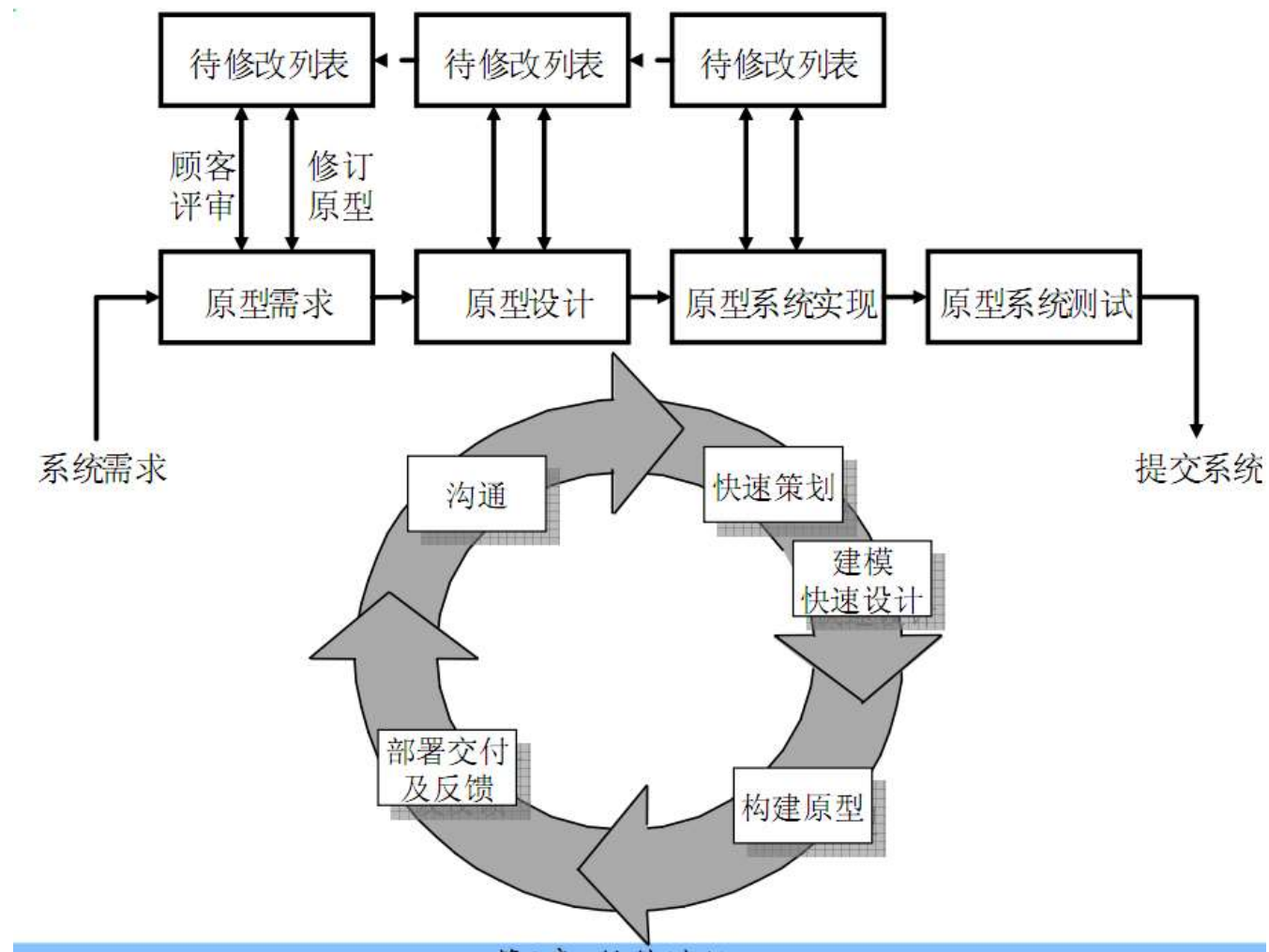
增量模型



螺旋模型



原型模型



传统模型缺陷

◆传统的过程开发模型都是从**管理者的角度**来看待软件开发，存在着**重大缺陷**：

- **忽视变化的存在**。过程阶段划分过细、过于死板，难于适应实际情况的变化。
- **忽视了软件开发是一个智力密集型**的工作，过分强调纪律和文档，导致人的创造性降低。
- **忽视了人与人之间的直接交流**。过多的书面交流既增加整个项目的成本，又导致了误解和沟通障碍的增加。
- **过分注重过程**。认为符合过程就能导致正确的结果。

敏捷开发吸收了历史上各种软件开发中的最佳实践，如迭代、原型、用户驱动

敏捷模型(1)

◆敏捷开发试图将开发者的视角也加入到软件过程管理中来。

◆**2001年Kent Beck等16位知名专家共同发起了敏捷联盟，并发表了“敏捷开发宣言”：**

- 个人与交流 胜于 开发过程和工具
- 可运行的软件 胜于 面面俱到的文档
- 客户协作 胜于 合同谈判
- 响应变化 胜于 按部就班遵循计划

注意：宣言中右边的各项并非没有价值，只是左边的各项价值更大

◆最核心理念

- 基于适应而非预测
- 以人为导向而非过程导向

目标：快速、增量地发布软件

敏捷模型(2)

◆敏捷宣言背后遵循的原则：小步快跑，及时反馈

敏捷鼓励：

使沟通更便利的团队结构和协作态度

快速交付可运行产品而非中间文档

客户以开发团队中的一员的身份参与项目

根据实际情况灵活调整项目计划

不强调文档，转向强调可运行的软件片段；
开发者与顾客之间的频繁沟通；
快速开发，快速反馈，快速修改；
连续不断的短周期迭代；

◆敏捷模型很容易适应变化并迅速做出自我调整，在保证质量的前提下，实现企业效益的最大化。敏捷过程在保证软件开发有成功产出的前提下，尽量减少开发过程中的活动和制品——Just enough

敏捷模型(3)

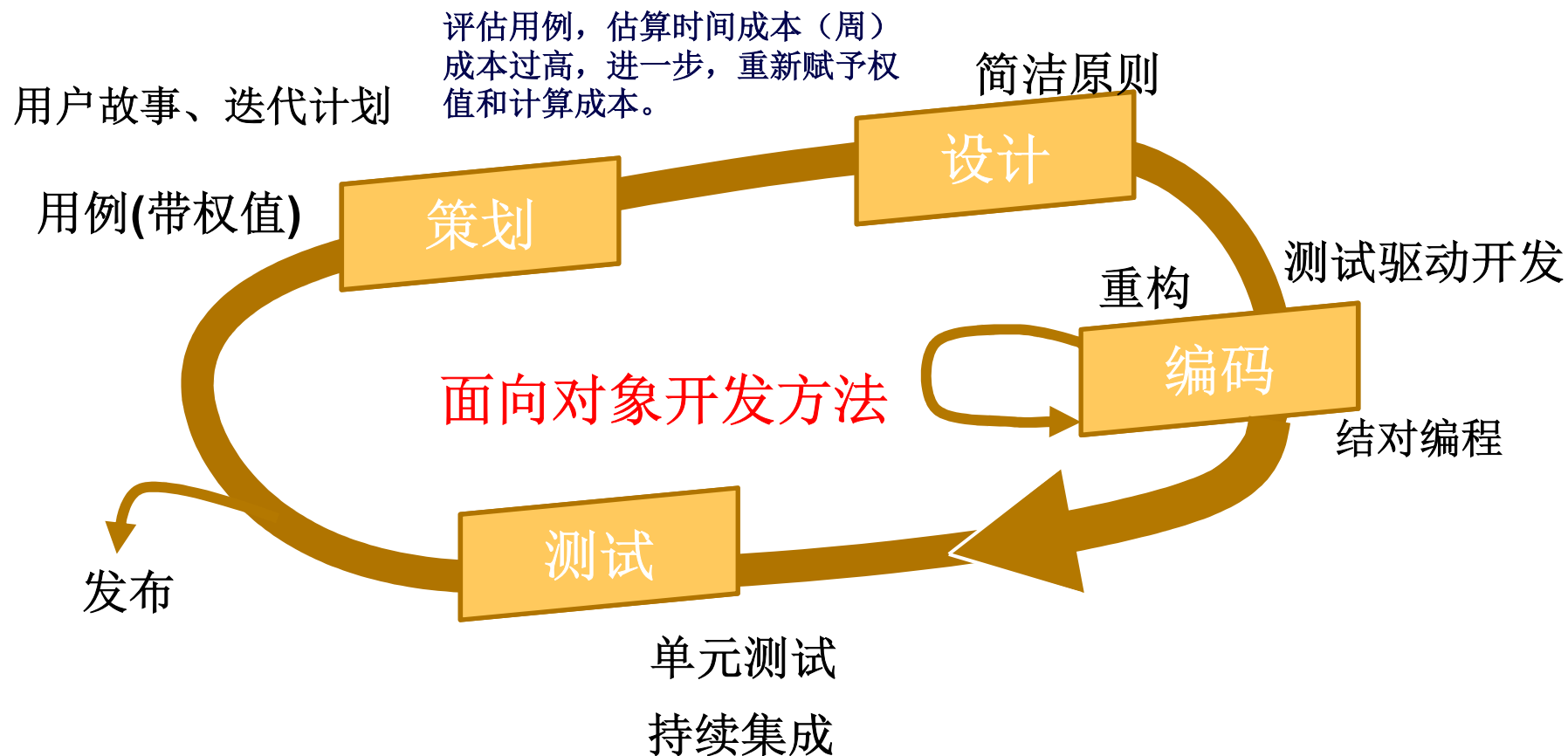
- ◆敏捷开发强调团队及其成员应该具备下列要素（续）：
 - **决策能力**。项目团队在项目问题上必须有自主决策权。
 - **相互尊重和信任**。具体体现就是良好、高效的沟通。
 - **不断学习**。团队应从多种来源（包括过去的失败）中学习经验。
 - **自我组织**。团队自己组织自身、安排进度来完成项目，并对此负责。

极限编程（XP）

- ◆由Kent Beck、Ward Cunningham、Ron Jeffries等人提出反响最大、最为完善的敏捷过程方法。
- ◆价值观：沟通、反馈、简化、勇气
- ◆特点：
 - 测试成为开发的核心
 - 纪律性与灵活性巧妙结合。

敏捷过程中最负盛名地一个，“极限”二字地含义是指把好好地开发实践运用到极致，广泛运用于需求模糊且经常改变地场合。

极限编程过程模型



测试驱动开发

统一过程(Unified Process, UP)

◆统一过程（Unified Process，简称UP）是一种流行的、构造面向对象系统的、迭代式软件开发过程，源于UML作者的软件开发过程。

◆RUP一般从三个视角来描述过程：

- 动态视角；
- 静态视角；
- 实践视角

统一过程模型

◆RUP核心工作流程

➤ 6个过程工作流

- 业务建模
- 需求
- 分析设计
- 实现
- 测试
- 部署

➤ 3个支持工作流

- 配置和变更管理
- 项目管理
- 环境

工作
内容

