

一. 敏捷项目管理概述

1. 了解项目的属性，传统项目管理的主要概念

属性：

- 1) 拥有特定的目标
- 2) 临时性：具有明确的开始和结束时间
- 3) 使用逐步细化的方式进行开发
- 4) 通常需要不同领域的资源
- 5) 应该要有一个主要的客户或者赞助商
- 6) 不确定性

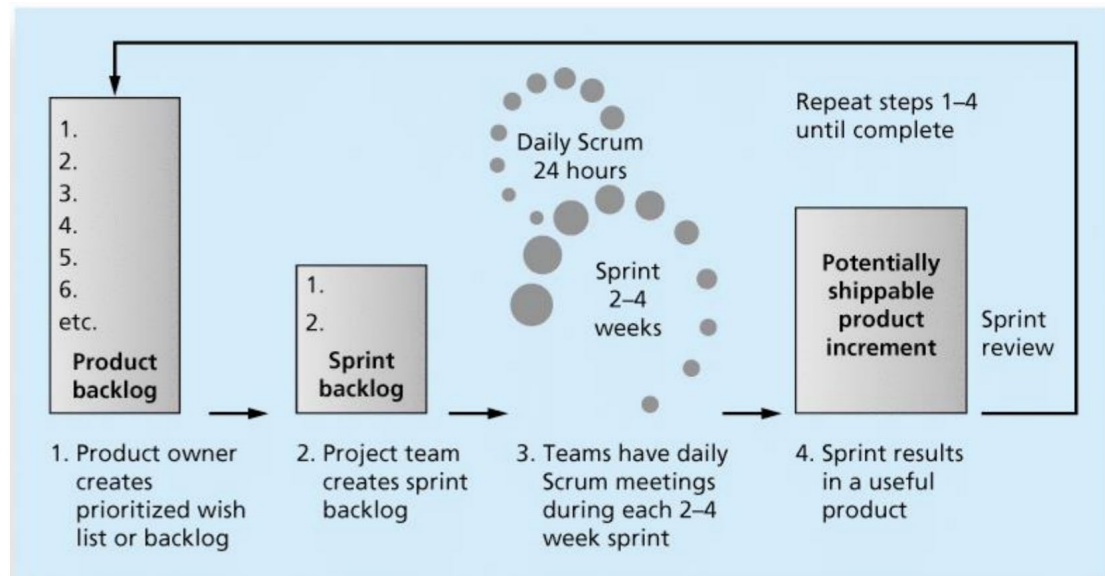
2. 敏捷开发的背景

- 1) 软件危机及软件工程的出现。
- 2) 速度是企业竞争致胜的关键因素，软件项目的最大挑战在于：
 - ① 一方面要应付变动中的需求。
 - ② 一方面要在紧缩的时程内完成项目。
- 3) 传统的软件工程难以满足这些要求。
- 4) 所以软件团队除了在技术上必须日益精进，更需要运用有效的开发流程，以确保团队能够发挥综效。这正是 Agile Process（敏捷的软件开发流程）于近年来兴起的主要原因。

3. 敏捷开发的宣言

- 1) 个体和交互胜过过程和工具。
- 2) 可以工作的软件胜过面面俱到的文档。
- 3) 客户合作胜过合同谈判。
- 4) 响应变化胜过循环计划。

4. Scrum 敏捷开发的框架



5. Scrum 角色，团队及特质

角色：

- 1) **Product owner(产品负责人)**: 是利益相关方的代表，他的工作重点是产品的业务方面，他负责向团队介绍产品愿景那个。负责给出一份明确的，可度量的，合理的产品 backlog，并从业务角度出发对 backlog 中各项问题按优先级排序。
- 2) **Scrum Master**: 是整个团队的导师和组织者，他负责提高团队的开发效率，他常提出培训团队的计划，列出障碍 Backlog。Scrum Master 控制着检查和改进 Scrum 的周期，他维护这一团队的正常运行，并与产品负责人一起让利益相关方获得最大化投资回报。他关心的是这些敏捷开发思想是否能够得到利益相关方的理解和支持。
- 3) **技术负责人**
- 4) **开发团队**: 一个由 5-9 个人组成的跨职能团队，负责组织自己的工作，并为每次冲刺产生期望的结果，通常持续 2-4 周。
- 5) **Stakeholder**: 是参与或受项目活动影响的人
- 6) **敏捷导师** 利益相关方

团队成员特质：

- 1) 创建产品
- 2) 自组织、自管理
- 3) 跨职能工作
- 4) 专注且集中办公

6. Scrum 工件，仪式及特点

工件：

- 1) **Product backlog (产品代办列表)**：按照业务价值进行优先级排序的功能列表
- 2) **Sprint backlog (冲刺待办列表)**：product backlog 中具有最高优先级的事项，将在一个 sprint（冲刺）中完成
- 3) **Product Increment (产品增量)**：交付可用的产品
- 4) **Burndown chart (燃尽图)**：展示了一个冲刺中每天的剩余工作量

仪式：

- 1) **sprint planning session (冲刺计划会议)**：团队举行的会议，从 product backlog 中选择一组工作在下一轮的冲刺中完成。
- 2) **Daily Scrum (每日例会)**：开发团队举行的一个简短会议，主要分享进度以及遇到的挑战，以及计划一天的工作。
- 3) **Sprint reviews (冲刺评审会议)**：团队在会议中向利益相关者和 product owner 演示在冲刺过程中已完成的工作。
- 4) **Sprint retrospective (冲刺回顾)**：根据开发团队的实际表现进行的会议过程，寻找改进产品的方法。

特点：

- 1) 是一个敏捷开发流程
- 2) 以团队为基础
- 3) 要求迅速变化情况下迭代和增量开发
- 4) 控制由利益和需求冲突导致的混乱
- 5) 改善交流并最优化合作
- 6) 检测开发过程障碍并将其去除
- 7) 最大化生产率
- 8) 适用于单一的项目到整个组织
- 9) 能让参与者对工作贡献感到满意

二. Scrum 敏捷工作实践

(一) 定义产品愿景和产品路线图

(二) 计划发布与冲刺 *冲刺计划会议*

- 1) 细化需求和估算
- 2) 创建用户故事
- 3) 创建 Product Backlog
- 4) 发布计划：确认团队能够行动的并推出最迫切产品的日期。PO 负责确定发布目标和发布日期。发布计划的两项关键活动（团队成员共同完成）：从产品待办列表（Product backlog）中选择优先级高的用户故事；制定发布计划：目标、日期。团队所有成员承诺发布计划。

5) 冲刺：

(三) 全天的工作

1) 计划每天的工作：

- ① 每日例会（站会）：
 - a. 标注已完成的任务
 - b. 发现和交流：问题和难点
 - c. 提出需要 Scrum 主管介入的工作
 - d. 计划当天工作
- ② 时长：15min
- ③ 发言人：开发人员，PO，Scrum 主管
- ④ 会议目的：沟通交流，不解决问题
- ⑤ 由 Scrum 主管主持，成员轮流回答 3 个问题：
 - a. 昨天我完成了什么工作
 - b. 今天我打算做什么
 - c. 我在工作中遇到了什么困难

2) 跟踪每天的进展，工具：燃尽图、任务板

3) 开发并测试每天的工作

4) 结束一天的工作：

- ① 更新 Sprint Backlog
- ② 根据燃尽图调整任务、人员安排、时间安排
- ③ 更新燃尽图
- ④ PO 更新任务板
- ⑤ SM 检查燃尽图和 Sprint Backlog，及时发现可能的风险

(四) 展示工作和集成反馈

1) 冲刺评审

2) 冲刺回顾

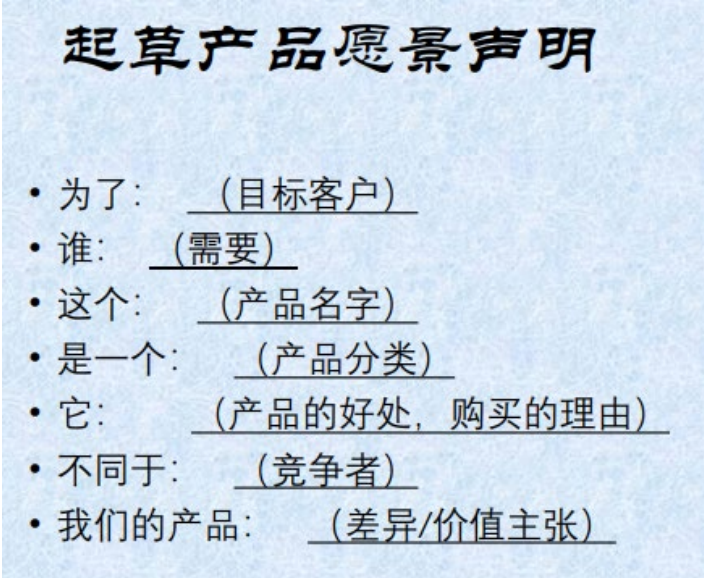
(五) 为发布做准备

- 1) 准备部署产品——发布迭代
- 2) 让组织为产品发布做好准备 *内部准备*
- 3) 让市场为产品发布做好准备 *外部准备*

1. 愿景声明

创建产品愿景声明的步骤：

- 1) 设定产品目标：需要考虑产品的关键目标、客户分析、需求、竞争者、主要差异
- 2) 创建愿景声明的草案：
- 3) 与项目干系人共同确认愿景草案，并修改
- 4) 确定愿景草案



起草产品愿景声明

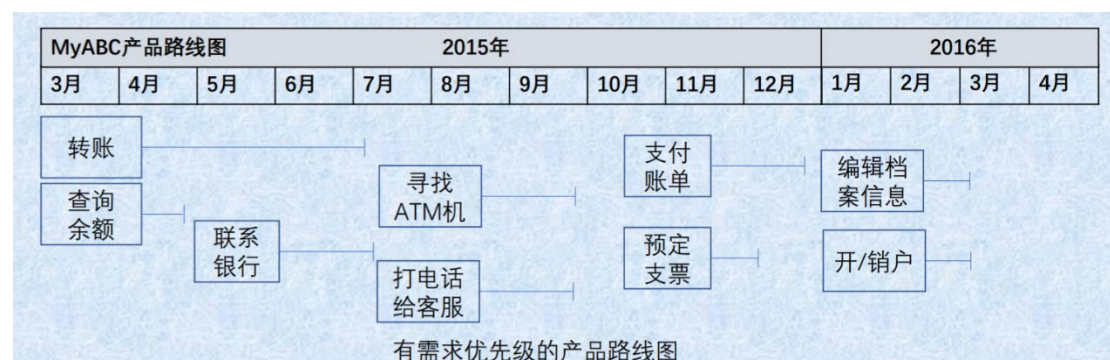
- 为了： (目标客户)
- 谁： (需要)
- 这个： (产品名字)
- 是一个： (产品分类)
- 它： (产品的好处，购买的理由)
- 不同于： (竞争者)
- 我们的产品： (差异/价值主张)

2. 产品路线图，创建的步骤，优先级估算方法

产品路线图是指产品需求的综合提示图，是产品需求的概览，也是组织开发过程的工具。
与创建愿景相比，开发团队参与程度更大。

创建步骤：

- 1) 识别产品需求：
 - ① 通过愿景声明，确定需求的主题 ---- 是最高层次的需求
 - ② 分析需求主题的特性，即拆分为具有若干特性的需求
- 2) 整理产品特性（功能 features）
- 3) 产品特性的估算和排序
 - ① 目的：确定核心需求，识别需求差异
 - ② 给需求的价值和工作量打分
 - ③ 用 Fibonacci 数列作为分值：用相对分数
 - a. 产品路线图的需求：分数在 55—144 （主题、特性）
 - b. 发布计划的需求：分数在 13—34 （史诗故事）
 - c. 冲刺计划的需求：分数在 1—8 （用户故事）
- 4) 决定大致的时间框架
 - ① 根据产品特性的相对优先级排序，创建产品路线图
 - ② 优先级高的排在完成时间的前面
 - ③ 当确定了产品路线图，可以确定产品发布时间
 - ④ 根据产品发布的优先级，确定大致的产品迭代时间增量



工作量估算方法：

- 1) 绝对估算法：以人时或人日为估算单位，计算用户故事的工作量。但是，在计算优先级时，数据不能统一，即用户故事的价值值如果随意取，则算出来的优先级难于理解。
- 2) 相对估算法，也称为故事点(Story Point)估算法：选择一个较小的用户故事，给出故事点数，以此为基准，估算其他用户故事的故事点数。

3. 用户故事，产品待办任务 Product Backlog

用户故事：是指一种对某个产品需求的简单描述（结合需求分析的用例图）

用户故事卡片：

- 标题<名称>
- 作为<用户 或 角色>*
- 我想<采取的行动> *
- 以便<能获得的益处>

注意：

- 发布级别的用户故事个数要求不超过 34 个故事点
- 冲刺级别的用户故事个数要求不超过 8 个故事点，每个故事点要分解为多个任务

Product Backlog：是指确定了用户故事优先级的用户故事列表

PB_ID	Role	User story	价值	工作量	优先级	所属特性	状态
1	普通用户	账户登录验证	34	13	2.62	账户信息	planned
2	VIP用户	账户VIP管理	21	13	1.62	账户信息	planned
...							

4. 敏捷冲刺的概念、要素和角色，每日例会

冲刺：指一次迭代，并提交能够正常工作的产品

一次冲刺的工作：

- 1) 开始时的冲刺计划
- 2) 每日例会
- 3) 开发时间 - 冲刺的主体
- 4) 结束时的冲刺评审和冲刺回顾

只为当前的冲刺选择用户故事

角色：

- 1) **开发团队**
 - ① 完成可交付的产品
- 2) **产品负责人**
 - ① 审核产品待办列表和冲刺计划
 - ② 解决开发中的技术问题
 - ③ 验证每天的测试结果
 - ④ 监督指导持续集成的工作
- 3) **Scrum 主管**
 - ① 每日例会
 - ② 解决遇到的非技术困难，保护开发团队不受外部干扰
 - ③ 保持与 stakeholders 的良好沟通，建立好的人际关系

冲刺 0：敏捷开发启动前的准备阶段：

- 1) 团队组建：组建特性团队
- 2) 前期调研：市场&用户分析
 - ① 高层业务流程图
 - ② 用户故事
 - ③ 产品 Backlog
 - ④ 迭代周期及工作
- 3) 技术准备：
 - ① 技术架构选型与基础设施搭建
 - ② 代码规范制定

5. 冲刺计划，冲刺待办任务 Sprint Backlog

在本次冲刺的第一天召开冲刺计划会议

参加者：product owner、开发团队、scrum 主管、客户

主要工作：

- 制定冲刺目标并选择优先级高的用户故事
- 创建 Sprint backlog，即分解开发任务

会议时长：

- 冲刺为 1 周：开会 2 小时
- 冲刺为 2 周：开会 4 小时
- 冲刺为 3 周：开会 6 小时
- 冲刺为 4 周：开会 8 小时

冲刺待办任务 Spring Backlog: 是冲刺阶段的重要文档，用于描述任务、时间计划和进度的。

6. 燃尽图的概念

（展示了一个冲刺中每天的剩余工作量）

填写：

- 1) 燃尽图须体现 Sprint Backlog 完成进展，即围绕 Sprint Backlog 的任务
- 2) 燃尽图与开发人员数量、周期、任务数量和时数等有关
- 3) 燃尽图的纵坐标也可以是
 - ① 工作时数：时间粒度太小，估算较费时，且冲刺后期可能需较大调整
 - ② 工作人天数：按照几人几天总数估算，粒度比时数估算要大，且可以了解每人完成情况，冲刺后期调整不大（与 Sprint Backlog 的计划对应）
 - ③ 任务点数：任务越多，则折线越多。不能反映任务的用时情况，即不能了解时间进度
- 4) 燃尽图的时数（纵坐标）不包含冲刺的计划、评审和回顾时间

7. 任务板（看板）的表示

- 1) 每个任务做在一个便利贴上，做成一个任务板
- 2) 将任务分为 4 组：待办；正进行；待验收；已完工
- 3) 用移动便利贴展示冲刺进度，实物变化的视觉更真实
- 4) 将任务板放在醒目的位置上，以便每个人都能容易地看到

8. 冲刺开发的三个主要活动，任务完成的定义

三个主要活动：

1) 细化：

- ① 开发前，可以对用户故事和任务进一步的分析或分解
- ② 进一步明确需求，如发现问题，要及时与产品负责人沟通
- ③ 当开发团队开始新任务开发时，任何时候都可以做细化
- ④ 细化活动应集体完成，如有个人独立于团队之外，Scrum 主管应宜于干涉

2) 开发：

- ① 开发团队自主地完成开发工作
- ② 产品负责人参与其中，主要是明确开发的内容，验收开发功能
- ③ Scrum 主管解决开发之外的管理问题

3) 验证：

完成的定义：

- 1) 编码完成；
 - 2) 代码评审完成；
 - 3) 单元测试 Bug 数小于三个；
 - 4) 集成完毕；
 - 5) 文档工作完毕
- 1) 当迭代任务清单上的任务完成时，变为“已完成”状态
 - 2) 定义“已完成”的含义是非常重要的，例如：
 - ① 如何记录软件的变化
 - ② 使用什么样的代码分析工具，发现的问题应当如何处理
 - ③ 进行了什么样的测试，结果是如何记录的，通过标准（如覆盖率、修正的错误）是什么
 - 3) 定义“已完成”意味着定义质量上的需求
 - 4) “已完成”是 0/1 变量：完成或者未完成，所有的任务(task)都完成了，迭代任务才算完成
 - 5) 在第一个迭代开始之前应该定义好，因为它会影响工作量，而且必须文档化，这样团队和产品所有者的理解是一致的

9. 敏捷工程实践：结对编程、持续集成、TDD

结对编程：两位程序员在一台电脑前工作，一个负责敲入代码，而另外一个实时检查每一行敲入的代码。

结对编程的好处：

- 1) 有助于提升代码设计质量
- 2) 研究表明结对生产率比两个单人总和低 15%，但缺陷数少 15%，考虑修改缺陷工作量和时间都比初始编程大几倍，所以结对编程总体效率更高
- 3) 结对编程能够大幅促进团队能力提升和知识传播

持续集成 (CI)：是一项软件开发实践，其中团队的成员经常集成他们的工作，通常每人每天至少集成一次，每次集成通过自动化构建完成。持续集成提供产品质量的快速反馈，保证随时拥有可工作的软件。

持续集成的好处：

- 1) 大幅缩短反馈周期，实时反映产品真实质量状态
- 2) 缺陷在引入的当天就被发现并解决，降低缺陷修改成本
- 3) 将集成工作分散在平时，通过每天生成可部署的软件，避免产品最终集成时爆发大量问题

测试驱动开发 (TDD)：TDD 以测试作为编程的中心，要求在编写任何代码之前，首先编写定义代码功能的测试用例，编写的代码要通过用例，并不断进行重构优化。TDD 要求测试可以完全自动化运行。

测试驱动开发的好处：

- 1) 和代码同步增长的自动化测试用例，能为代码构筑安全网，保证代码重构的质量
- 2) TDD 有助于开发人员优化代码设计，提高代码可测试性

10. 冲刺中 Scrum 主管的任务：识别障碍，解决问题

障碍：基本上，任何阻止团队正常工作的，都可称之为障碍。

- 每日例会可以汇报障碍
- 可能的障碍，如：
 - 1) 无法访问信息系统
 - 2) 所需要的信息不能及时提供或者提供的不正确，如界面规格或者其它软件模块不到位或不正确
 - 3) 开发环境或者原型系统出现问题
 - 4) 其他的任务分配：培训，售前支持
 - 5) 缺乏必要的信息或者相应的知识
- 对于团队提出的各项障碍，Scrum 主管要以列表形式进行记录
- **谁来清除：**所有人。
- scrum 主管负责确定是否已经清除，不一定亲自清除

11. 冲刺评审会议

准备工作：

- 1) 邀请参加的人员
- 2) 计划好会议和时间的地点
- 3) 打印好给参会人员的准备资料
- 4) 制定会议的日程：产品演示，讨论
- 5) 会议的进程控制，讨论中的时间控制等
- 6) 记录、总结与反馈

参加者：开发团队，客户，关注产品的人

内容：

- 1) 展示工作：
 - 产品负责人要知道已完成的用户故事
 - 产品负责人要验证过所演示的可交付功能
 - 演示是非正式的产品展示，但要求展示环境与生产环境一致
 - 产品负责人介绍发布目标、冲刺目标和新增功能
 - 演示过程中，时间要留给核心价值的工作，不要去包装
 - 开发团队展示冲刺完成的功能，即运行程序的主要步骤，不需要展示文档
 - 典型的做法是使用演示形式展示新功能或者底层架构的实现，而不是 PPT
- 2) 收集反馈：
 - 非正式地记录反馈意见，由专人记录，同时产品负责人和 Scrum 主管也要记录要点
 - 产品负责人要分析反馈意见，并且
 - 将新的用户故事放到产品 backlog
 - 将已有用户故事的更改放到冲刺 backlog

关键点：

- 1) 展示“真实”的产品，不允许有作弊：Team 应在真实环境中展示可运行的软件，判断是否达到“完成”标准；
- 2) 收集反馈：产品负责人根据验收情况及客户反馈意见，及时调整产品 Backlog 和冲刺 backlog
- 3) 产品负责人要避免产品“镀金”，即不要过度开发，只完成用户故事描述的功能

好处：

- 1) 通过演示可交付的功能可以确认项目的进度，具有真实性
- 2) 能尽早的获得用户对产品的反馈，使产品更加贴近客户需求

12. 冲刺回顾会议

准备工作：

- 1) 计划好会议和时间的地点
- 2) 回顾内容：工具、结对、分工、解决的问题等，做成道具
- 3) 开会的方式：讨论式、头脑风暴、引领式等
- 4) 记录、总结与反馈

目的：分享好的经验和发现改进点，促进团队不断进步。以行动为导向，主要是总结和改进方案。

参与者：主要是开发团队参加：

- 1) Scrum 主管—主持会议
- 2) 产品负责人
- 3) 开发团队
- 4) 可能还包括客户或其他干系人

内容：开发平台/测试工具/管理平台等的选择，当前使用的情况，是否需要沟通学习，结对编程组合是否合理...

围绕三个问题：

- 1) 本次冲刺有哪些做得好
- 2) 本次冲刺在哪些方面还能做得更好
- 3) 在下次冲刺中哪些方面可以改进

关键点：

- 1) 会议气氛：Team 全员参加，气氛宽松自由，畅所欲言，头脑风暴发现问题，共同分析根原。
- 2) 关注重点：Team 共同讨论要解决的问题的优先级，将精力放在最需要的地方，讨论不要分散，关注几个改进即可。
- 3) 会议结论要跟踪闭环：可以放入冲刺 backlog 中。

好处：

- 1) 激励团队成员
- 2) 帮助团队挖掘优秀经验并继承
- 3) 避免团队犯重复的错误
- 4) 营造团队自主改进的氛围

13. 发布冲刺的工作：内部准备和外部准备

内部准备：

- 1) 除了产品生产部门，企业的其他组织部门要为产品发布做好准备
- 2) 产品负责人和 Scrum 主管要准备一个发布冲刺待办列表
- 3) 识别和确定完成发布活动的干系人
- 4) 为发布产品的评审会准备正式的 PPT，解释产品的背景、目标、价值，确保干系人充分理解产品和产品客户
- 5) 评审会应该由市场部或运营部主持，产品负责人只是负责冲刺发布任务的汇报

外部准备：

- 1) 营销支持——确定产品推广、品牌营销的时机
- 2) 客户测试——运营和市场要把客户反馈转化为产品推广的依据
- 3) 营销物料——产品推广文案、新闻发布会、产品包装等
- 4) 支持渠道——所有类型客户到技术支持部的渠道要畅通

三. 在企业中实施敏捷

1. 企业组织中实施敏捷的步骤

- 1) 制定实施策略
- 2) 构建意识，培养热情
- 3) 实施团队转型
- 4) 确定实验的项目
- 5) 确定成功的标准：时间、成本、质量、风险、变更、交付的成果
- 6) 培训计划与实施
- 7) 总结和改进
- 8) 推广

2. 实现敏捷转型的系统工程

敏捷转型覆盖 7 个方面：实践、组织、过程、绩效考核、管控、文化、技术和业务调整。敏捷在敏捷转型不同阶段，敏捷转型框架的 7 个方面引入的优先级不一样，初期以实践为主。

- 1) **实践 (Practices)**: 团队根据自身情况选择合适的实践应用。
- 2) **组织(Organization)**: 组织适应产品业务架构、产品管理办公室(PMO)、测试、产品管理等。
- 3) **过程(Process)**: 流程、敏捷项目管理策略等，由企业管理者和敏捷导师共同制定。
- 4) **绩效度量 (Performance)**: 业务价值、质量、过程改进。
- 5) **管理监控(Governance)**: 包括管控点设置，如投资评审和风险评估等。
- 6) **文化理念(Culture)**: 从命令控制式的管理文化转向领导协作的文化，加强透明性和实际性。
- 7) **技术和业务调整(Alignment)**: 业务策略、IT 策略、企业/产品架构、产品组合管理、合同、能力规划等调整。

3. 如何组织实施敏捷

- 1) 组织与个人的承诺。
- 2) 选择正确的项目团队成员：
 - ① PO: 业务能力强，做事果断。
 - ② Scrum 主管: 有影响力，善于沟通。
 - ③ 开发成员: 愿意合作的。
- 3) 创建合适敏捷的环境: 团队要整齐，要及时调整。
- 4) 持续地支持敏捷: 选择合适的项目，拥有一位敏捷导师。

四. 其他敏捷项目管理开发框架

1. 掌握不同的敏捷开发框架

- 1) eXtreme Programming (XP) : 极限编程
- 2) Lean Software Development (Lean): 精益软件开发
- 3) Adaptive Software Development (ASD): 自适应软件开发
- 4) Feature-Driven Development (FDD): 特征驱动开发
- 5) Crystal Methodologies: 水晶族方法
- 6) Dynamic Systems Development Method (DSDM): 动态系统开发方法
- 7) Agile Unified Process (AUP): 敏捷统一过程

2. 敏捷度量和敏捷原则

度量:

- 1) 40% 终端用户: 功能, 最小训练, 增量交付
- 2) 30% 赞助商: 创收, 花费减少, 风险减小
- 3) 20% 公司: 可比较的优势, 品牌认知, 依赖性
- 4) 10% 团队: 认知, 开发技能, 授权

原则:

- 1) 我们最优先要做的是通过尽早的、持续的交付有价值的软件来使客户满意
- 2) 即使到了开发的后期, 也欢迎改变需求
- 3) 经常性地交付可以工作的软件, 交付的间隔可以从几周到几个月, 交付的时间间隔越短越好
- 4) 在整个项目开发期间, 业务人员和开发人员必须天天都在一起工作
- 5) 项目由有激情的、值得信任的个体合作完成
- 6) 在团队内部, 最具有效果并且富有效率的传递信息的方法, 就是面对面的交谈
- 7) 工作的软件是首要的进度度量标准
- 8) 敏捷过程提倡平稳的开发节奏; 发起人、开发者和用户应该能够保持一个长期的、恒定的开发速度
- 9) 不断地关注优秀的技术和好的设计可以增强敏捷能力
- 10) 简单化是根本 (不做过度设计和预测)
- 11) 最好的构架、需求和设计是来源于自组织的团队
- 12) 每隔一定时间, 团队会在如何才能更有效地工作方面进行反思, 并对自己的行为进行相应调整

3. Scrum Of Scrums

- 把 Scrum 扩展到大型项目团队的一个实践
- Scrum of Scrums 可以是多层次的
- 是跨团队的沟通与交流
- 由各个 Scrum 团队的代表参加 Scrum of Scrums 会议
- 会议也采用固定的频率，如每周 2 次
- 会议时长也控制在 15-30 分钟
- 会议中提出的问题要及时处理

项目中的多个 Scrum 团队：企业级，产品级，项目级

4. 敏捷项目管理需要的传统项目的知识领域

- 1) 成本管理
- 2) 沟通管理
- 3) 风险管理
- 4) 采购管理
- 5) 干系人管理

5. 敏捷项目管理的关键测量指标

- 1) 冲刺目标成功率
- 2) 缺陷数量
- 3) 项目总工期
- 4) 产品上市时间
- 5) 项目总成本
- 6) 投资回报率
- 7) ROI 预算中的新请求（新的产品特性可能转化成更高的产品收入）
- 8) 资金调配
- 9) 满意度调查
- 10) 团队成员流动率

6. 敏捷项目管理的十大好处

- 1) 更好的产品质量
- 2) 更高的客户满意度
- 3) 更高的团队士气
- 4) 增强合作和责任感
- 5) 可定制的团队结构（允许团队成员按自己特定的工作风格和个性组建团队，根据成员特定的技能进行分工）
- 6) 更多的测量标准（燃尽图，故事点、速率、ROI、时间表和预算）
- 7) 提高绩效可视性（看板、站会、冲刺评估）
- 8) 项目更易监控（有多个环节可以控制和调整）
- 9) 提高项目可预测性
- 10) 降低风险