

主要内容

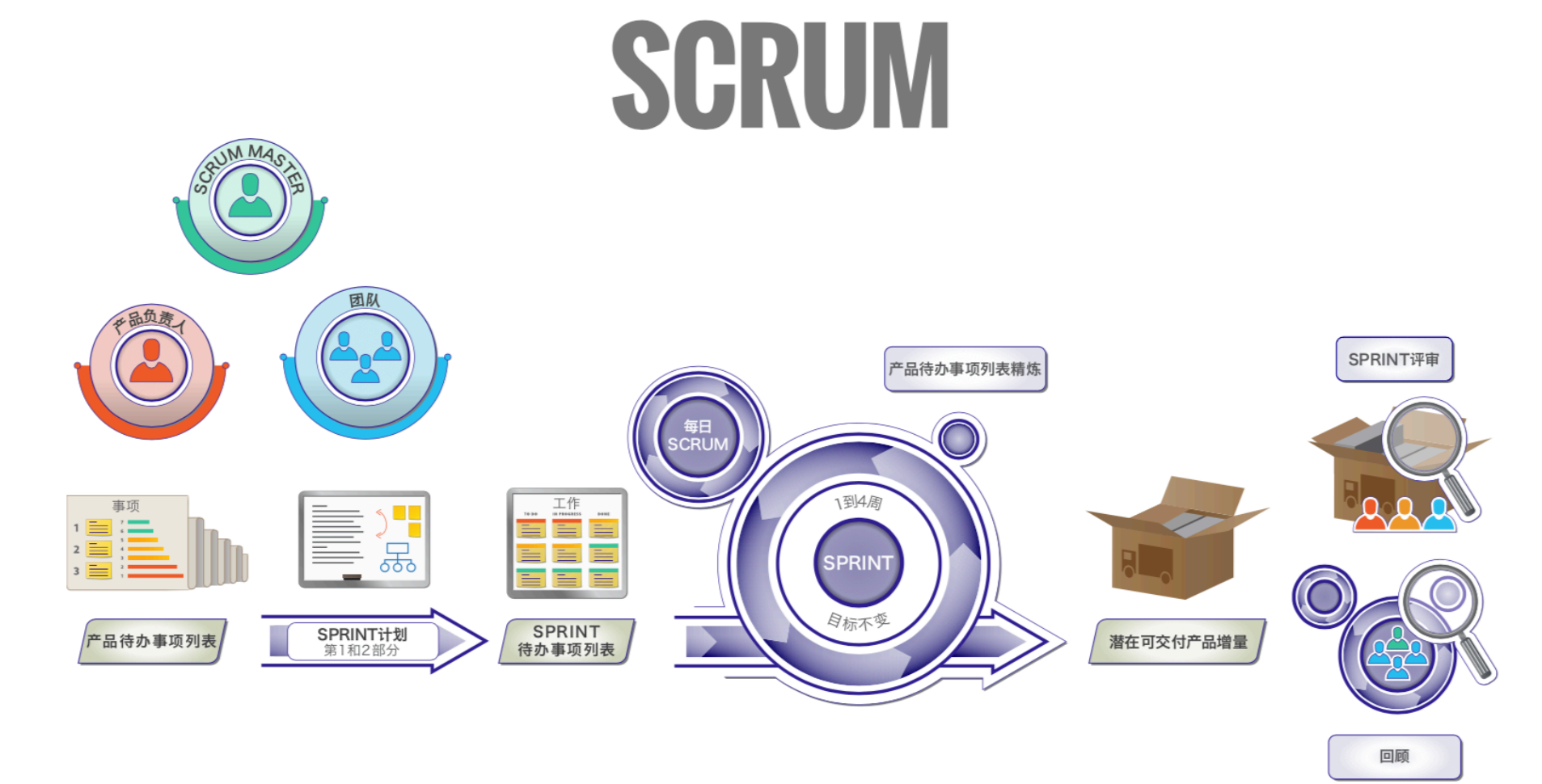
第 I 部分“核心概念”(第 2~8 章), 涉及的主题有: Scrum 框架, 敏捷原则, 冲刺, 需求与用户故事, 产品列表, 估算与速率, 技术债。

第 II 部分“角色”(第 9~13 章), 涉及的主题包括: 产品负责人、ScrumMaster、开发团队、Scrum 团队构成和经理。

第 III 部分“规划”(第 14~18 章), 主题包括: Scrum 规划原则、多层次规划、产品组合规划、构想/产品规划和版本规划。

第 IV 部分“冲刺”(第 19~22 章), 主题包括: 冲刺规划、冲刺执行、冲刺评审和冲刺回顾。

Scrum的基本框架：



基本概念：Scrum框架、敏捷原则

角色：产品负责人（PO）、ScrumMaster、团队，主要职责、技能要求

关键环节：产品待办事项列表、迭代计划会议、迭代待办事项列表、每日站会、迭代评审会议、迭代回顾会议

冲刺细节思考和分享

- 背景介绍：
- 12月底参加吕毅老师的ScrumMaster（CSM）认证课程
 - 1月上旬准备尝试Scrum敏捷开发，确定1周一次迭代，已进行3次迭代
 - 团队结构：2个前端、3个后端、1个iOS、1个安卓、2个设计、1个产品
 - 需求端：PC Web、小程序、iOS、安卓、其他（技术重构等）

- 工作中迭代计划会议遇到的问题：
- 产品待办列表不够详细，只有大的需求没有小的功能点；需求不明确，如XXX优化
 - 出现无法评估的需求，或者没法评估工时
 - 不是按优先级排列的单个需求列表，多个业务端分别排列需求优先级

这3个问题基本都是“产品待办列表”的问题，翻看书籍P128页有介绍“产品待办列表”就绪的检查表：

就绪的定义	
<input type="checkbox"/>	清楚表达业务价值
<input type="checkbox"/>	有开发团队能够理解的足够多的细节，这样就能针对是否能够完成做出明智的决策
<input type="checkbox"/>	已经识别出依赖关系，不存在阻碍 PBI 完成的外部依赖关系
<input type="checkbox"/>	为了完成 PBI，团队人手配备齐全
<input type="checkbox"/>	PBI 做过估算、足够小、很容易在一个冲刺中完成
<input type="checkbox"/>	接收标准清晰并且是可测试的
<input type="checkbox"/>	如果有性能标准的话，性能标准是已经定义并且可测试的
<input type="checkbox"/>	Scrum 团队很清楚在冲刺评审中如何演示 PBI

- 针对问题1和2，有非常详尽的操作实践指导：
- 提供足够多的细节，清楚表达业务价值
 - 识别依赖关系，提前对需求进行估算

针对问题3，原文内容：产品待办列表应该是大小合适、经过估算、而且有优先顺序。

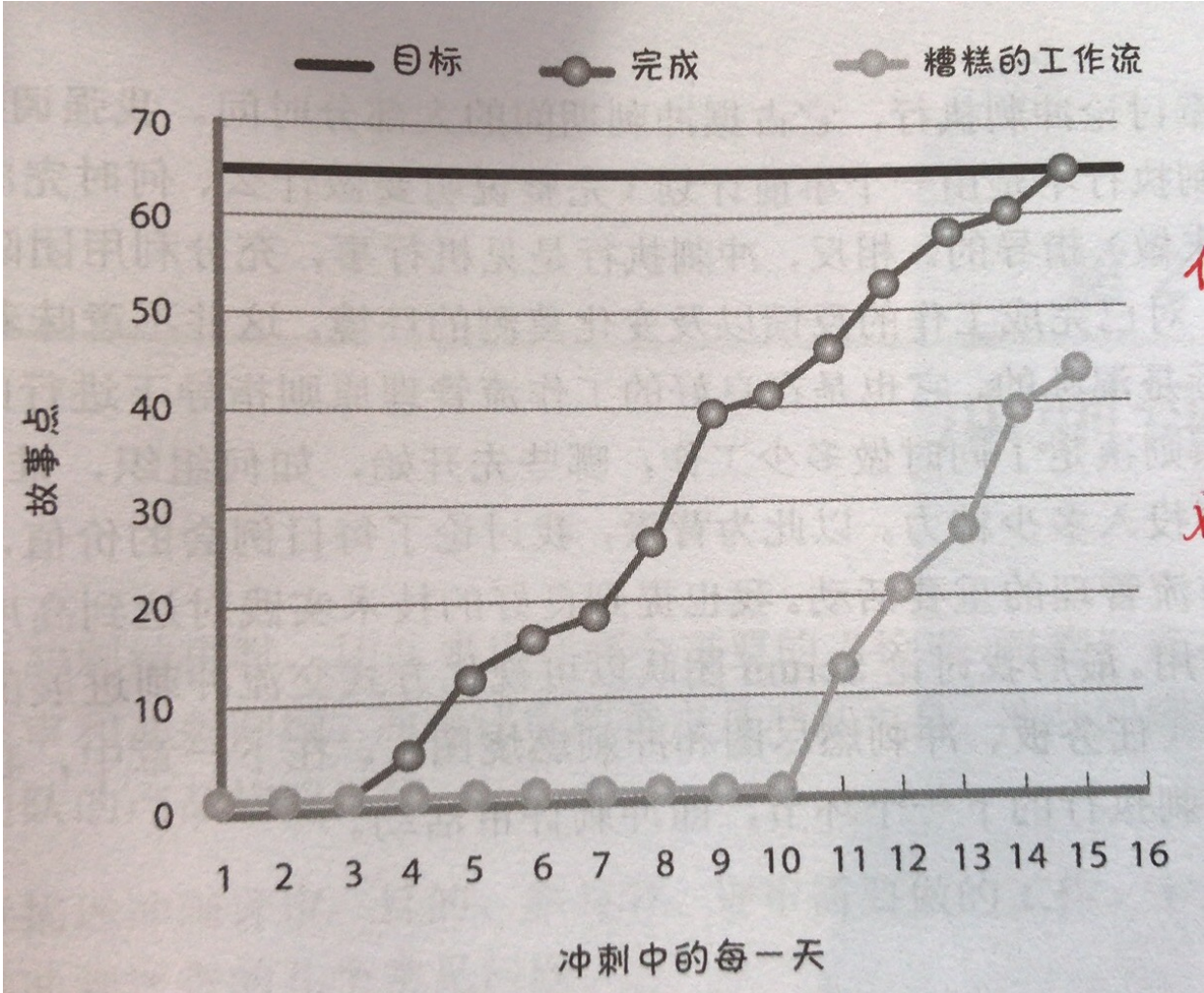
最终和产品讨论做出的改变：除iOS和安卓外，其他需求端合并一个需求列表。

感觉有时候带着问题去阅读，总会获得意外的收获！

- 工作中每日站会遇到的问题：
- 需求Review确认（测试）太慢，基本都在周四进行Review

- 对照书籍中的介绍，虽然形式上是Scrum开发，但本质上还是“瀑布式”开发方式，不是“蜂拥式”的开发方式：
- 瀑布式
 - 每个人都是一个独立的个体，分别去完成每个需求项中特定的内容
 - 一直到最后才进行Review测试
 - 会有很大的风险
 - 蜂拥式，
 - 几个人组成一个小团体，共同为某一个功能项而努力，直到需求被Review测试完成

- 从“瀑布式”到“蜂拥式”的转变：
- 在看板中添加迭代需求的燃尽图，更直观的展现趋势问题，明确目标
 - 鼓励团队成员发展多技能，团队成员之间可以更灵活搭配



- 看书时想到的问题：
- 如何保证让团队主动承诺需求项时，与团队的实际生产能力相匹配（不会少领）？
- 从实际操作环节来看，团队领取任务时出于个人荣誉感和羞耻感，基本不会出现少领的情况
 - 每次迭代都会有历史生产能力留存，可以对照约束

最后

通过阅读书籍，发现要想把Scrum玩得好，发现还有很多很多细节需要优化迭代，除了正确实施Scrum流程中的各个环节还包括对技术团队的技术要求（自动化测试、重构、持续继承等）。

Scrum除了对需求进行迭代之外，对本身的迭代流程也需要进行优化迭代，是一个自我进化的过程。

虽已上路，路漫漫其修远兮，吾将上下而求索！