

AgileDo敏捷实践集

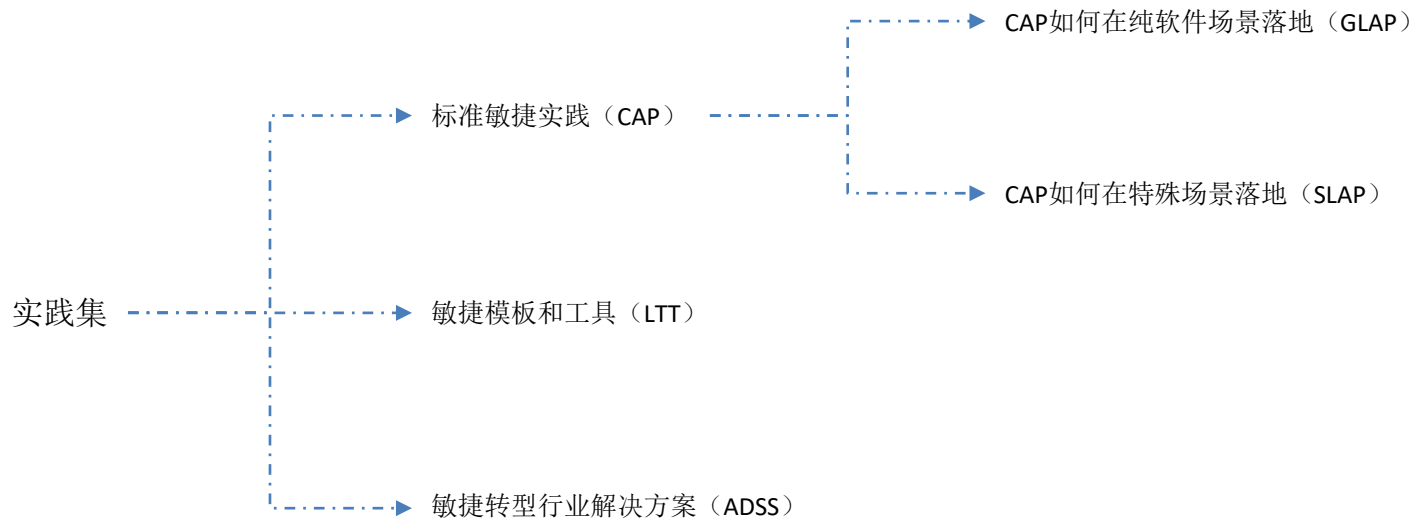
北京迅思威尔科技有限公司 (AGILEDO)

V 1.1

为什么做敏捷实践集

Scrum Guide2020版强调：Scrum 是一个框架，可以很好的作为其他技术、方法论和实践的容器，但是**不完整**（未涵盖全生命周期），**也没有详细说明实施细节**。调查显示，如果仅以Scrum框架来实施Scrum，超过50%的团队**没有达到期望的效果**。AgileDo总结了398个敏捷实践，为团队在实施Scrum时提供详细、完整的技术和方法。

实践集分为内部版本和分享版本。前者在培训和教练等服务中提供，后者以电子刊的方式持续分享。您可以在[公司官网](#)上获得电子刊的最新版本
电子刊的第一个版本是 V1.0，目前发布的版本是 V1.1



CAP: 标准敏捷实践

敏捷主流方法论官方推荐的敏捷开发标准做法

LTT: 敏捷模板和工具

敏捷模板是预定义的框架，可帮助敏捷团队实施和管理敏捷方法。这些模板提供了规划、跟踪和执行敏捷流程的标准化方法。优秀的模板和工具可以帮助敏捷开发实践更好的落地

GLAP: 纯软件场景的落地实践

由于团队、文化差异等原因，标准敏捷实践（CAP）在落地时会遇到各种问题。GLAP的价值是保证标准敏捷实践（CAP）在最常见的场景（本地团队的纯软件开发）中可以成功落地

ADSS: 敏捷转型行业解决方案

针对不同客户行业，提供一揽子的敏捷转型整体解决方案，包括流程、实践、模板和工具的结合。目前提供的解决方案包括证券、银行、智能制造、游戏开发、互联网等20+行业

SLAP: 特殊场景的落地实践

SLAP的价值是保证标准敏捷实践（CAP）在一些特殊的场景中可以成功落地，这些场景包括外包管理、异地协作、嵌入式开发、硬件开发、IPD和CMMI环境等



目 录

CAP

Top30标准敏捷实践 [5](#)

GLAP

一份详尽的DoD清单 [6](#)

50+团队规模化Scrum的组织架构 [8](#)

消除技术债务的敏捷实践 [9](#)

做好Scrum Master的七种武器 [11](#)

敏捷发布规划的第一个交付物 [12](#)

SLAP

异地协作的敏捷开发实践 [14](#)

嵌入式开发的敏捷实践 [16](#)

LTT

精益画布 - 产品开发的第一个模板 [18](#)

ADSS

电商行业200+团队敏捷转型整体流程 [20](#)

CAP: Top30标准敏捷实践

迭代管理

- 增量迭代开发
- 每日站会
- 燃尽图
- 任务板
- Kanban: 看板
- 估算: 计划扑克
- 定期回顾

产品管理

- Backlog: 待办列表
- 用户故事
- 3C:“卡片、对话、确认”
- Story Mapping: 故事映射
- Story Splitting: 故事拆分
- Give-When-Then
- MVP: Minimum Viable Product
- 敏捷发布计划

敏捷团队

- 自组织
- 跨职能
- Team Room: 团队室
- 团队拆分
- Scrum of Scrums
- T型技能

内置质量

- 结对编程
- 单元测试
- 持续集成
- TDD: 测试驱动开发
- 代码重构
- BDD: 行为驱动开发
- DoD: 完成的定义
- DoR: 就绪的定义
- 集体所有权
- 学习左移

[点击这里](#), 查看每一个实践的解释

GLAP: 一份详尽的DoD清单

完成的定义 (DoD) 是 Scrum 团队内部对如何使软件做好发布准备的共识。



1. 一份详尽的完成清单包含哪些内容

米奇·曼特尔 (Mickey Mantle) 和罗恩·利奇蒂 (Ron Lichty)

在他们的《管理难以管理的事情》一书中提出了一份详尽的清单，列出了“完成的定义”应包括的内容。在实践中，敏捷团队会在清单中选择最适合自己的内容

项目启动通常是就“功能、任务或冲刺何时完成？”这一问题达成一致的最佳时机。定义“完成”是与业务、开发团队、项目管理、QA (有时还有其他人) 的联合行动，以定义何时可以宣布每个功能完成。

除了编码之外，功能被完成的定义可能还需要

- 设计回顾
- 同行评审 (或结对编程)
- 单元测试
- 静态代码检查
- 签入源代码管理
- 在源代码管理中评论
- 必要的重构
- 在多个平台、浏览器、配置等上进行了测试
- 性能测试
- 必须修复的错误已修复
- 安装脚本编写
- 必须的文档已更新

对于团队来说，同意“完成”的要素并同意交付它们非常重要。两个团队就“完成”的相同定义达成一致的情况很少见。定义甚至可能会在迭代之间、任务之间、冲刺之间发生变化 (经团队同意)。

GLAP: 一份详尽的DoD清单

2. 常见陷阱

- 过分关注标准清单可能会适得其反，该列表需要定义使产品增量达到“完成”状态通常所需的最小工作量
- 除了适用于一般工作的标准之外，个别功能或用户故事可能还有特定的“完成”标准
- 如果“完成”的定义仅仅是一种共同的理解，而不是详细说明并展示在墙上，那么它可能会失去很多有效性；它的很大一部分价值在于成为团队所有成员都知道的明确合同

参考：“Managing the Unmanageable: Rules, Tools, and Insights for Managing Software People and Teams”, Mickey Mantle, Ron Lichy, Addison-Wesley

GLAP: 50+团队规模化Scrum的组织架构

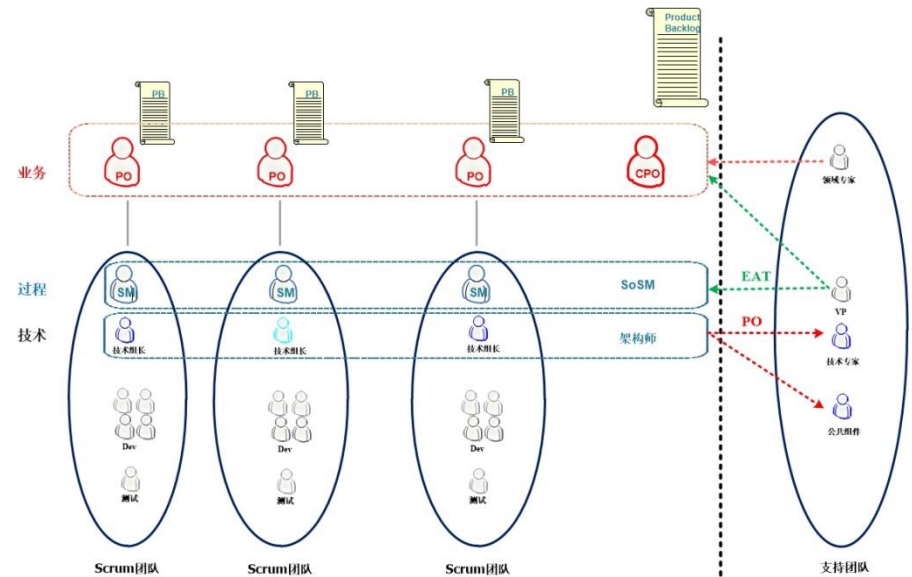
1. 客户背景和需求

客户属于电子商务行业，研发团队规模200+。试点单个团队的Scrum已经取得了很好的效果，希望在50+规模的团队中开展敏捷开发

2. AgileDo解决方案

AgileDo的敏捷教练基于Scrum @ Scale,为客户制定了详细的规模化转型方案，敏捷组织架构是其中的一个重要环节，见右侧的示意图，以三个团队为例介绍：

- 每个Scrum小组内部以Scrum的方式工作
- 产品部增加一名总监（CPO），负责公司层面的产品思路，整合三个团队的交付
- 三个项目组的SM每天（需要的时候随时）以会议的方式沟通10~20分钟，主要是项目组间资源的协调、遇到的障碍如何解决等
- VP作为EAT
- 各个Scrum小组的技术组长成立虚拟架构师团队，架构师团队根据产品部的整体产品思路，提出并实现产品的整体技术架构
- 各个Scrum小组的测试人员成立虚拟测试团队，主要的目的是为了整合测试资源，推出更加高效的测试方法、测试工具



- 各个Scrum小组的美工成立虚拟美工组，负责公司所有产品的界面（页面）设计，最大的好处是页面风格统一，页面层的技术可以共享
- 成立支持团队，分为技术专家团队、公共组件团队、领域专家团队。每个团队人数很少，但是可以使所有Scrum团队的工作富有效率。架构师团队的Leader是组件团队和技术专家团队的PO，他们的Product Backlog仅技术需求

3. 效果

在这个组织架构的基础上，AgileDo的教练帮助团队完成了50+团队的规模化敏捷的转型。后续开始200+团队的规模化敏捷转型

GLAP: 消除技术债务的敏捷实践

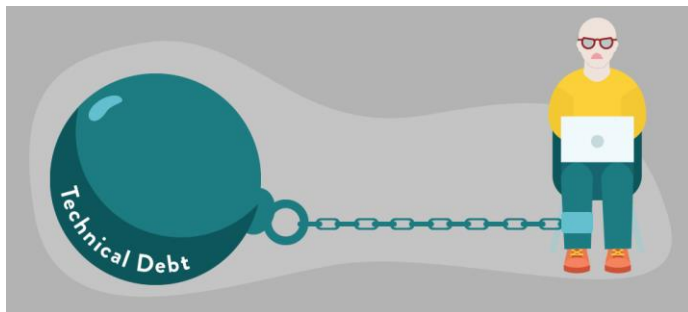
观点

技术债务不是一个“贬义词”，消除的策略是在“业务交付速度”和“代码质量”之间做平衡

1. 什么是技术债务

技术债务是指属于“稍后修复”类别的可扩展性和性能问题，包含但不限于：

- 不遵守编码和工程最佳实践
- 导致重复、不必要的复杂性或未使用组件的代码解决方法
- 推迟技术升级
- 绕过安全相关功能



<https://www.mightybytes.com/blog/technical-debt-agile-and-sustainability/>

2. 消除技术债务的敏捷实践

敏捷方法提供了一组工程最佳实践，可以帮助团队组织起来应对技术债务：

- 仅在仔细考虑其影响后才接受技术债务。如果战略性地接受一定数量的技术债务是可行的
- 对完成有一个明确的定义，其中必须包括可接受的技术债务限制和特定的测试覆盖率值。在发布新功能之前完成代码审查会议可以帮助团队确定他们是否达到了他们定义的可接受质量水平
- 使用定义的代码审查流程，其中包含团队商定的清单
- 利用自动化和持续集成
- 通过集成静态代码分析工具来自动衡量技术债务
- 让技术债务成为团队日常工作的一部分，授权在每个迭代期间采取重点减少技术债务的行动
- 定义技术待办事项列表，这允许团队设置专门用于代码稳定和重构的迭代
- 保持技术债务水平透明，所有团队成员和产品经理都了解当前的技术债务，有利于决策
- 通过讨论他们的愿景和业务战略以及他们愿意接受的技术债务水平，鼓励产品经理参与技术债务管理

GLAP: 消除技术债务的敏捷实践

3. 角色间如何协作

让一个人负责管理技术债务是不可能的，保持技术债务的控制是开发团队、Scrum Master、PO共同承担的责任，尤其是在敏捷环境中。每个人都有自己的角色：

- 开发团队

开发人员负责代码质量，他们需要遵循编码标准和软件工程最佳实践，同时系统地审查和测试他们的代码

- Scrum Master

Scrum Master衡量技术债务并确保团队尊重“完成的定义”。他们还营造了一个有利于团队和PO协作的环境，从而确保了双方的透明度

- PO

PO通过降低业务的耦合度以及明确的用户信息（Who、Why）来帮助开发团队建立消除技术债务的策略。同时还通过向客户解释技术债务对软件质量的影响来为开发团队赢得支持

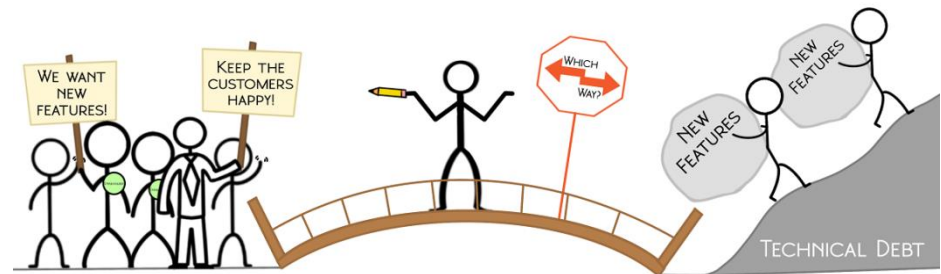
4. 消除技术债务的最佳时机

理论上讲，“重构代码和偿还技术债务的最佳时机是昨天。第二个最好的时间就是现在。” Mario Fusco – Red Hat 的 Drools 核心开发人员

但是，当面临需要编码响应的业务挑战时，开发团队有两种选择。他们可以花费额外的时间来编写可扩展且健壮的代码，或者可以通过快速甚至混乱的修复来解决当前问题，但这将需要未来的技术干预来清理代码。第二个选择产生了技术债务，这是一种临时妥协，开发团队和客户都愿意将业务目标置于软件质量之上。

所以，消除技术债务的策略是在“业务交付速度”和“代码质量”之间做平衡。如果以下的场景越来越多，则说明消除技术债务的权重也越来越大

- 团队一直处于压力之下，经常处于救火的状态
- 即使产品资源和资金增加，仍看到产出减少
- 客户因为产品质量或服务的下降，逐渐失去对产品的信任



参考：

1. <https://www.pedallog.com/blog/methodology/how-to-manage-technical-debt/>
2. 技术债：从隐喻到理论与实践，作者：Philippe Kruchten、Robert L. Nord 和 Ipek Ozkaya

GLAP: 做好Scrum Master的七种武器

做一个好的Scrum Master，是要帮助到团队更好的交付。可以重点关注：

1. 理解Scrum Master的职责

职责做好了，自然是一个好的Scrum Master

ScrumGuide2020标明了Scrum Master的职责：负责按照 Scrum 指南中的定义建立 Scrum，并通过帮助 Scrum 团队和组织内的每个人了解 Scrum 理论和实践来做到这一点

2. 团队的组织架构

根据产品/项目的团队规模，确定好团队协作的组织架构（单团队、Scrum@Scale还是其他），保证每个团队的跨职能

3. 工作室 (Team Room)

安排工作室 (Team Room)，增强项目的透明和团队的沟通，该空间配备了团队可能需要的各种便利设施：工作站、白板和活动挂图、用于显示任务板、项目计划或其他图表的墙壁空间等

4. Scrum框架

保证Scrum的框架可以工作，也就是经典的3-5-3

5. 度量和检查

可以鼓励团队有勇气去尝试，但同时更需要做过程的度量和检查，以发现问题并及时调整，例如缺陷的数量、种类，计划和实际的偏差等

6. 内部教练

最好可以以内部教练的方式帮助到团队。Scrum 是一个框架，可以很好的作为其他技术、方法论和实践的容器，但是不完整（未涵盖全生命周期），也没有详细说明实施细节。如果在这两个方面帮助到团队，会有更好的效果。例如一个想法如何拆分到Backlog，Backlog中的用户故事是否覆盖足够的需求领域，如何估算更加靠谱等等



GLAP: 敏捷发布规划的第一个交付物

发布规划是一种将产品的实施组织到发布中并预测交付日期或里程碑的技术。

1. 如何做发布规划

1.1 发布规划决定了发布的内容和日期

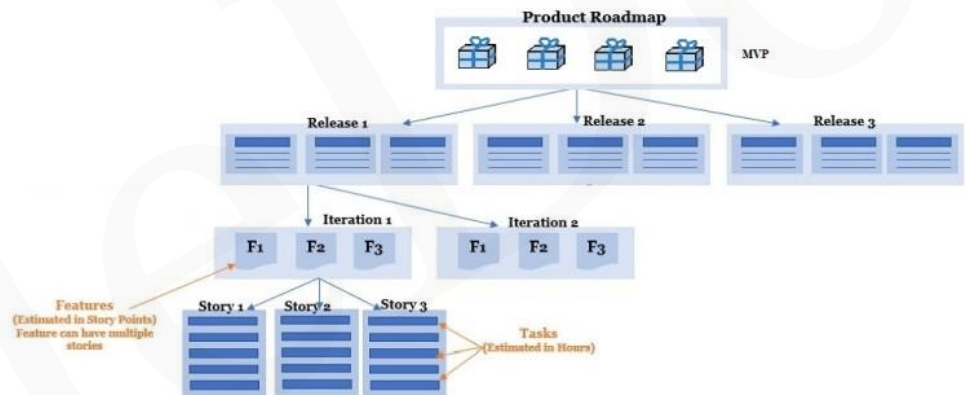
发布内容取决于MVP的制定，发布日期和发布范围取决于Scrum团队的速度和产品待办事项的估计

1.2 如何平衡速度、范围和日期

在规划发布时，要平衡速度、范围和日期，以便制定切合实际的计划。基本上有两种选择：确定发布内容并确定发布日期，或者确定发布日期并相应协调内容

1.3 定期调整

发布计划是产品待办事项细化的一部分，并在此框架内定期更新。例如，如果产品待办事项条目或速度发生变化
敏捷发布计划包括初始发布计划和持续计划。初始发布计划是第一个交付，是产品交付规划的整体框架。在整个过程中，人员可能会被替换，功能可能会发生变化，发布计划也会持续被修改



2. 发布规划的第一个交付物：初始发布计划

2.1 计划包含的层次

初始发布计划会按照层次制定，具体的层次包括：MVP、发布版本、迭代、功能和用户故事（上图所示）

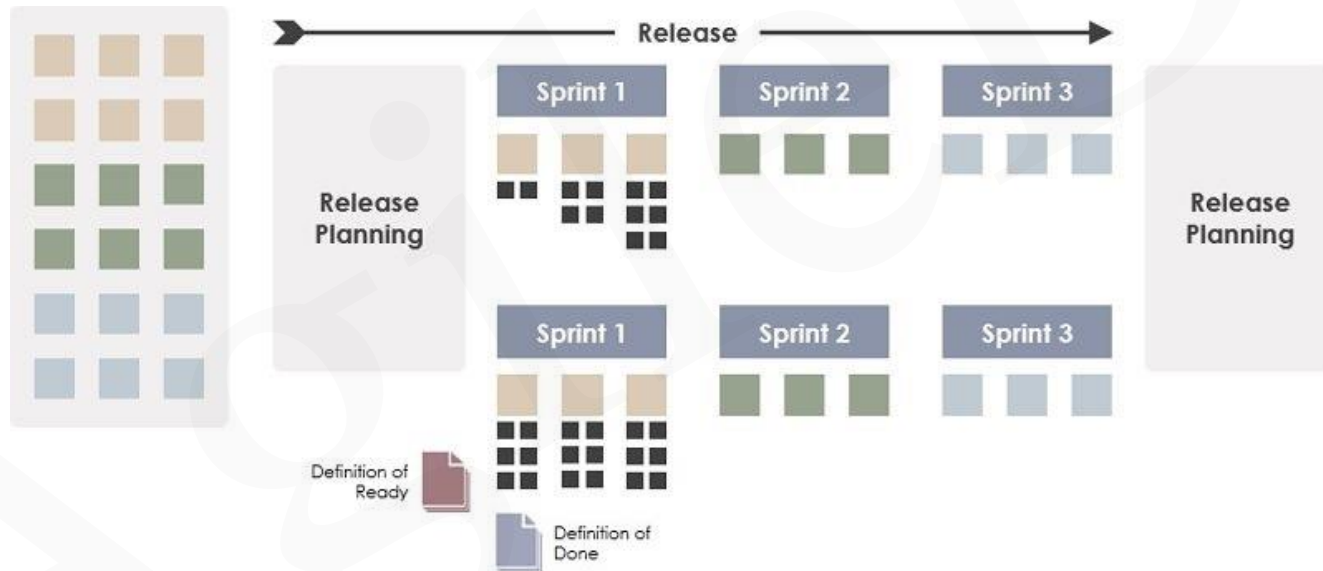
GLAP: 敏捷发布规划的第一个交付物

2.2 计划制定的详细程度

初始发布计划，要适应需求的变化，按照“越近的计划越关注细节，越远的计划越关注目标”的原则制定

2.3 初始计划的示例

下图展示的是一个初始发布计划的具体例子，由两个团队共同完成一个产品



SLAP: 异地协作的敏捷开发实践

1. 客户背景

客户属于传统软件产品研发行业，在北京、武汉、济南三地分别有研发团队，三地协作共同研发一款软件产品

2. 客户需求

采用敏捷开发，提高交付的速度，快速响应市场



3. 遇到的困难和AgileDo的解决方案

3.1 三地的团队各自有独立的研发流程，不容易协作

项目建立伊始，就要决定如何协作：

- AgileDo的敏捷教练和所有团队成员一起组织了一个“工作协议”活动。我们定下来用 Scrum @ Scale 作为异地协作的模式
- 确定一些实践方式，如怎样做结对、用哪些工具、质量目标、工作的完成定义等等。然后在Wiki上记录下来，这样整个团队就有了共识。一旦这些共识需要修改，比如在回顾会议上提出改进，这些实践就要在wiki上更新，有新人加入的时候，他们看到的总是最新内容
- 为了让这些实践可以更好的落地，项目开始的最初，北京、武汉两地的团队各自派出三名核心成员，和济南的团队一起工作了三周，在工作中进行实践的优化和达成共识



SLAP: 异地协作的敏捷开发实践

3.2 异地协作如何处理业务依赖

我们的模式是让每个本地团队尽可能自给自足，以减少对对方的业务依赖。具体而言，是在Sprint 0的阶段业务切分时就做到三地的业务是松耦合的，每个团队的业务相对独立，这个是CPO和三地的PO工作的重点之一

3.3 异地协作如何处理集成的风险

三地的团队有公共的持续集成服务器，核心代码要求必须有Unit Test，核心业务必须由本地的核心开发人员结对完成，每个版本的代码提交都会进行代码的自动集成

3.4 异地协作如何处理平时的沟通

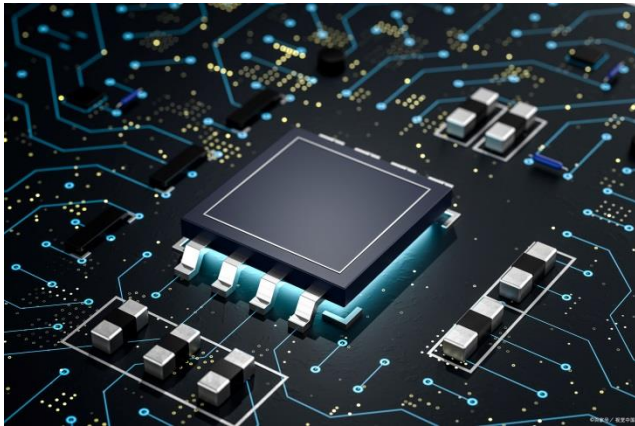
- 腾讯会议等软件是我们的首选，微信群的讨论也是我们异地多人讨论的常用方式
- 在Wiki上会分享三地对应事项的紧急联系人，以保证任何事情都可以无需等待的处理
- 在公共服务器上有项目管理软件，记录三地的项目进展状态。因为我们是分布式团队，虽然每个团队都有物理白板进行每日站会，但我们会要求所有的团队都在工具上继续项目进展的细节，以帮助其他地方的团队更好的了解整个项目的进展和风险

4. 效果

以上列出的困难和解决方案仅仅是工作中的一部分。AgileDo的教练和客户团队现场持续共同工作，以解决更多实际场景中遇到的障碍。目前，客户三地的团队协作已经可以稳定的以两周的频率交付产品的增量版本

SLAP: 嵌入式开发的敏捷实践

嵌入式开发常见的风险包括：**需求变更、代码质量、学习债务、团队间沟通不畅、硬件延期、软件和硬件强依赖、后期软硬件整合风险**等。虽然和纯软件开发相比，嵌入式开发对硬件依赖更大，而硬件开发有耗时长、变更成本高、技能更专业因而不易结对等特点，但是敏捷作为一种应对不确定和动荡的环境并最终取得成功的方法，其价值观、原则、方法论和具体实践还是为嵌入式开发提供了非常大的帮助。



目前AgileDo团队已经有 **60+** 具体的**技术实践和模式**，以实现嵌入式开发的更高质量、快速交付、聚焦客户价值和应对需求的变更。这里分享其中的 **8** 个：

1. 模块化

以用户故事的方式对业务进行“切片”，每个模块可以更早的独立进行硬件、固件和应用软件等的整合

2. 学习左移

在转向硬件实现之前，开发和运行仿真测试，以便客户评估。通过模型更新快速响应不断变化的需求。

3. 模拟测试

为了减少软件测试对硬件的依赖，模拟器、Mock、Stub都是常用的方式。如果项目允许，推荐实现模拟器

4. TDD

测试驱动开发，可以有效保证代码的质量。同时在任何可行的场合，和硬件一起测试，自动化是最佳选择

SLAP: 嵌入式开发的敏捷实践

5. 增量开发

增量（在 Scrum 方法中称为“冲刺”），通过对功能进行优先级排序，可以更早期地降低风险并强调核心功能，不仅团队获得成就感，利益相关者也获得有形的价值，并且可以及时提供反馈

6. 任务板

当冲刺开始时，工程师需要一个工具来在团队内沟通他们正在执行的任务、任务之间的依赖以及任务的状态。任务板是跟踪任务和传达状态的有效方法

7. 每日站会

每天15分钟的团队沟通，可以快速发现问题，以便尽早解决问题

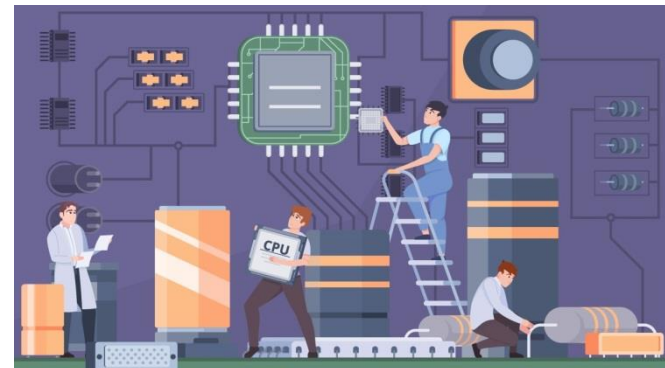
8. DoR

鉴于嵌入式开发中软件和硬件的依赖，任务开始之前的“就绪标准”可以帮助明确依赖，减少可能的等待

总结:

敏捷应用在嵌入式开发中的优点

- 更快的产品开发
- 更高质量的交付成果
- 持续降低风险
- 加强开发团队内部以及与利益相关者的协作
- 项目状态更加透明、准确



LTT: 精益画布 - 产品开发的第一个模板

从想法到产品的第一步，是“精益画布”。实现没有客户价值的需求，在精益开发中被视为最严重的研发浪费。敏捷开发为了提高研发的效率，充分吸收了精益开发中消除浪费的优秀实践。“精益画布”就是其中的实践之一

1. 什么是精益画布

精益画布，它是可视化精益创业的方法，涉及一个表格，其中包含有关精益创业原则的所有关键考虑因素。

2. 精益画布的关键元素

• 问题

如果市场无法通过某些服务或优惠满足人们的需求，那就是一个问题，问题意味着机会。（如果市场已经有竞品，在此可以列出竞品的名单）

• 解决方案

产品准备如何做来解决市场出现的问题

• 关键指标

它涉及帮助了解产品是否成功的具体措施，通常是具体的数字结果

• 产品的独特价值

为客户提供的独特卖点，它可以帮助产品在竞争中脱颖而出

Problem	Solution	Unique value proposition	Unfair Advantage	Customer Segment
	Key metrics		Channels	
Cost structure			Revenue streams	

• 优势

竞争对手无法复制或者成本高昂的竞争优势

• 渠道

如何获得客户

• 客户群

产品的目标客户（此处可以列出早期的产品采用者规划。早期的产品采用者指产品的最早版本进行交互并提供反馈的人员）

• 成本结构

包括研发成本、客户的获取成本、销售成本等

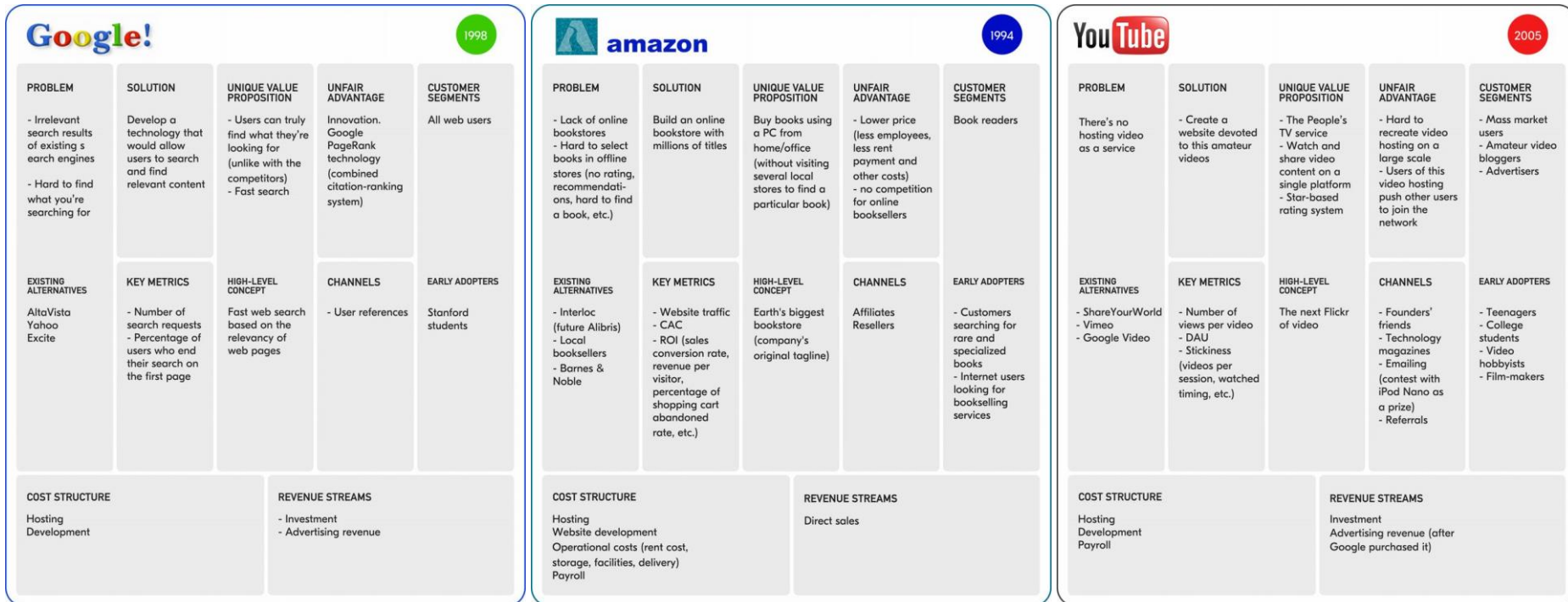
• 收入

产品为企业带来的各种类型的收入

LTT: 精益画布 - 产品开发的第一个模板

3. 精益画布的典型示例

以下展示的是Google（1998年成立）、Amazon（1994年成立）以及YouTube（2005年成立）三家公司在成立之初的精益画布分析



ADSS: 电商行业200+团队敏捷转型整体流程

1. 客户背景

客户属于电子商务行业，研发团队规模200+

2. 客户需求

试点单个团队的Scrum、50+团队的规模化敏捷已经取得了很好的效果，准备在200+规模的团队中开展敏捷转型

3. AgileDo解决方案

AgileDo的敏捷教练基于Scrum @ Scale、DAD（规范敏捷交付）和Lean（精益软件开发），为客户制定了详细的规模化敏捷转型方案，方案需要兼顾产品开发和项目开发。这里简要介绍一下整体的研发流程

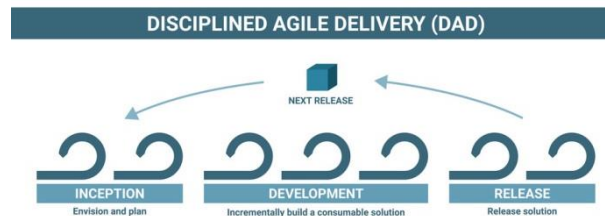
3.1 流程框架

流程分为两个主要阶段：启动阶段和发布阶段，每个阶段包含若干迭代。

3.1.1 启动阶段

主要是为产品构建阶段做好准备，减少返工成本。关键的节点包括但不限于：

- 组建团队
- 最初的想法逐步细化为需求列表
- 进行初步的架构设计
- 建立需求的依赖关系
- 对需求做敏捷估算
- 建立发布计划



Lean Software Development



ADSS: 电商行业200+团队敏捷转型整体流程

3.1.2 发布阶段

主要是为了把产品制作过程中的风险降低到最小，尽快上线并持续改进。

关键的节点包括但不限于：

- 拆分产品为多个子产品，保证每个子产品团队在5~9人以内
- 以MVP为标准建立发布规划，以保证系统测试、子产品团队、技术专家等多个团队高效协作
- 子产品团队保证每个迭代交付可工作的软件，技术专家团队保证为子产品团队提供企业级技术储备和支持，系统测试保证每个子产品的整合测试以及非功能测试，以保证子产品整合为一个产品后是可以工作的
- 子产品团队是一个典型的敏捷团队，PO和Team可以是Scrum + Scrum方式协作，也可以是Kanban + Scrum的方式协作
- 交付阶段中产品上线后存在持续的运营和运维。在这个过程中产生的持续需求改进和线上的重要Bug等，可以考量Kanban或是ScrumBan
- 在整个交付阶段，商业价值驱动、产生较少的缺陷（含技术债务）、消除研发浪费是三个最重要的考量因素

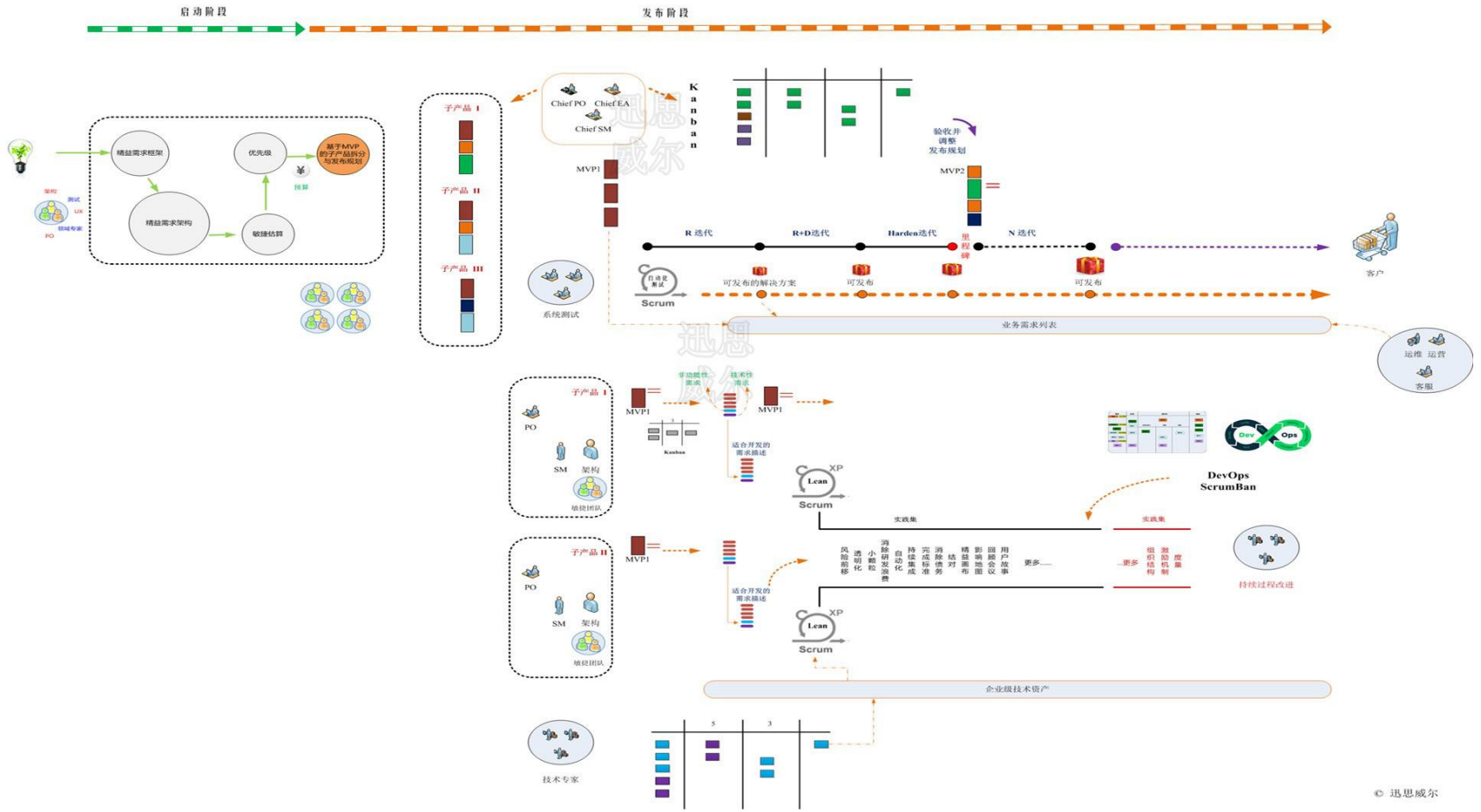
3.3 说明

- 产品和项目的研发稍有不同。产品的发布阶段包含构建阶段和运营、运维阶段，构建阶段比较短，更加专注在前几个稳定版本的发布，产品持续优化、运营和运维阶段占据了产品研发周期的大部分；项目的发布阶段包含构建阶段和项目验收阶段，构建阶段比较长也更加重要
- 对于产品，启动阶段会比较短，需要后续发布阶段，根据各种反馈做持续改进；对于项目则不同，启动阶段会相对较长，需求分析的颗粒度和详细程度也会不一样

4.效果

200+团队的规模化敏捷转型已经初见成效，交付节奏稳定，协作流畅。AgileDo的教练和客户正在紧密合作，以优化流程的各个细节

ADSS: 电商行业200+团队敏捷转型整体流程



14年，迅思威尔专注提升客户的交付质量和效率

我们的目标是发现最适合客户的敏捷开发流程，帮助企业快速交付，控制风险，提高效率

14年专注“敏捷+DevOps”，500+客户积累的解决方案实践集，保证每一个客户的落地

通过与众多管理工具厂商长达10年的合作，我们帮助客户更好的通过工具落地流程



公司介绍

您可以从以下三种服务中得到帮助

培训

通过现场的企业内训，讲解敏捷开发、DevOps的理念，分享业界的实践，解决客户研发中的痛点，并把最终的解决方案和管理工具相结合

教练

定制适合企业自身特点的流程改进方法，使企业切实得到敏捷带来的好处，同时为企业培养成熟的敏捷团队和成熟的内部教练

工作坊

您已经对敏捷开发的理论有了足够的了解，但在团队的敏捷转型中遇到了实际的障碍。如果您的需求是只针对自己团队实际问题进行讨论，并得到解决方案，那么工作坊是最佳选择



提供的服务



总部地址：中国北京 海淀区中关村东路18号财智国际大厦B座16层



联系人：李妍

电话：18701521797

邮箱：liyan@agiledo.cn



网址：www.agiledo.cn



联系我们