

敏捷在 PMBOK 中的应用

1. Why 敏捷

1.1 工作的不确定性

项目工作包括可确定的工作与高度不确定的工作。可确定的工作项目具有明确的流程，它们在以往类似的项目中被证明是有效的。团队可以利用较少的工作增量验证自身的工作，对后续工作做出相应调整。当团队交付小的增量时，他们能够更快更准确地理解真正的客户需求。新的设计、解决问题和之前未做过的工作都是探索性的。它要求主题专家携手合作，解决问题并创建解决方案。

高度不确定的项目变化速度快，复杂性和风险也高。这些特点可能会给传统预测法带来问题，传统预测法旨在预先确定大部分需求，并通过变更请求过程控制变更。而敏捷方法的出现是为了在短时间内探讨可行性，根据评估和反馈快速调整，形成闭环的改进。

1.2 市场的不确定性

随着互联网时代的到来，外界的变化更加迅速，给项目带来更多变更的可能。传统的瀑布项目管理变更的方式过于严格，面对多变的环境，需要一个更轻量级更灵活能够快速响应外界变化的方法，才能快速响应外界机会。敏捷通过小版本，快交付这样小步快跑的方式推出原型，获得客户的验证和反馈，根据反馈调整，最终达到客户要求，既能减少返工的缺陷，也能保证做出来的功能是客户需要的。

2. What 敏捷

2001 年 2 月，17 位软件开发领域的领军人物聚集美国犹他州雪鸟（snowbird）雪场讨论两天，概况出一些可以让软件开发团队快速工作、响应变化能力的价值观和原则，采用“敏捷”一词代表他们所倡导的开发思想。敏捷不是一种软件开发方法，而是一族，共同点是：以人为核心、迭代、循序渐进。成立敏捷联盟，宣言如下：

我们一直在实践中探寻更好的软件开发方法，身体力行的同时也帮助他人。由此我们建立了如下价值观：

- 个体和互动 胜过 流程和工具
- 可工作的软件 胜过 详尽的文档
- 客户合作 胜过 合同谈判
- 响应变化 胜过 遵循计划

虽然右项有其价值，我们更重视左项价值。

🌈 个体和交互高于过程和工具

流程和工具是项目中需要的，将团队的目的聚焦于个体参与和互动。项目是通过人来完成，而不是工具。困难也是由人来解决的，而不是流程。个体的参与和交互会有利于项目成功。但是，并不是说流程和工具对于项目的成功没有帮助，这些反而是重要的组织资产。

🌈 可以工作的软件高于面面俱到的文档

软件项目以创造有价值、高质量的软件为首要目标。然而很多人经常更多关注一些临时的可交付成果，例如：泛泛的文档。却不太关注于支持项目最终目标的可工作的软件。没有文档描述的软件在技术支持和维护上一定会出现问题和阻碍。但是只有详尽的文档而没有完成软件对于任何一个组织而言都是没有价值的。所以，对于文档是需要有一个度的把握，文档是需要的。

客户合作高于合同谈判

知识型项目是动态的，特别是软件系统；软件是无形的和难以参考的。每一个软件系统都是独特的。外界需求变化很快，技术的革新非常迅速。我们应该识别出将要发生变更的事件，与客户在共同定义“完成”。这将取决于更加互信的关系和更灵活的合同模式，同时也需要将我们的工作重点从没有附加价值的活动（例如：对范围的争论）转移至更富有价值的工作上。

响应变化高于遵循计划

遵循计划是指按计划行事，中间可能需要采取纠正措施，目的是为了使预期的未来绩效与项目计划一致而做的一切事情。响应变化则是适应的过程，通过卓越构想和不断反馈来采取适应措施，适应的目的是对实践而非预订计划的回应。

最初的计划有时候是不完全完善的。如果我们努力将项目按照最初的计划执行，这样就会使我们投入更多的精力去响应了必然的变更，从而导致投入的精力也是一种浪费。

响应变化高于遵循计划对于有着很高变更比率的软件项目尤为重要。同样，我们通过相应变化来代替抑制变化和使用大量的时间和精力来遵循一个大型计划。敏捷项目有着很高的工作列队的可视性，通过未完项和任务看板来形成计划。敏捷价值观的主旨就提倡适应性计划，积极要求全员参与。

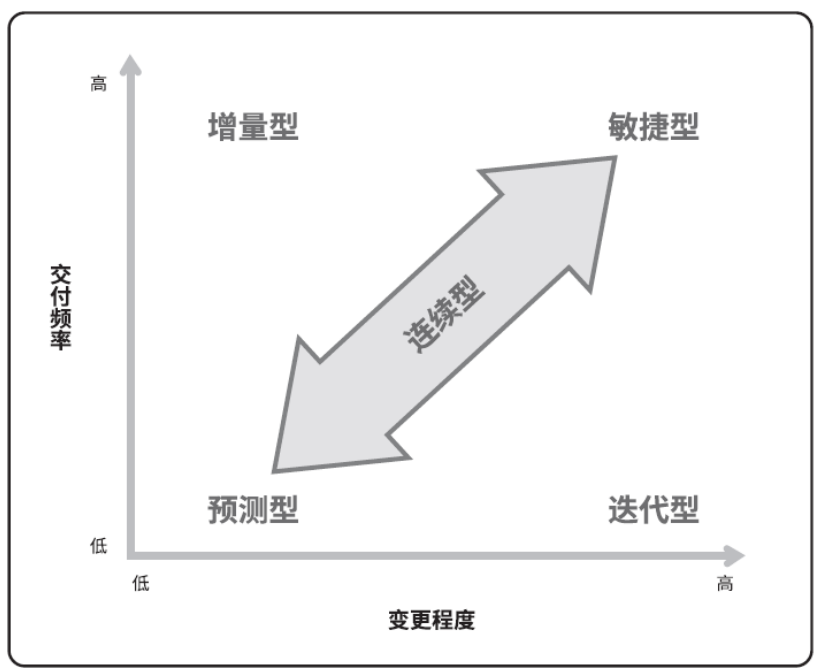
3. 敏捷 vs PMP

传统知识领域	敏捷中的做法
整合管理	1) 迭代和敏捷方法能够促进团队成员以及相关领域专家的身份参与整合管理。团队成员自组织去决定计划以及组件的整合方式。 2) 在敏捷环境下，项目经理的期望不变，但是把具体产品的规划和交付授权给团队，项目经理关注于营造一个合作的氛围，确保团队有能力应对变更，培养通才专家，而非专才。
范围管理	1) 在项目早期缩短定义范围的时间，持续探索和明确范围； 2) 不断涌现的需求会导致真实的业务需求与最初所述的业务需求存在差异，敏捷通过构建原型并发布多个小版本来明确需求。 3) 范围会在整个项目生命周期内被定义和明确，把需求纳入待办事项。
进度管理	1) 采用短周期开展工作，审查结果并及时调整，包括迭代型进度计划（scrum）和拉动式按需进度计划（看板） 2) 大型组织中需要制定长期路线图（roadmap），完整的交付需要系列技术，包括预测法、适应性方法或者混合方法。

成本管理	<ol style="list-style-type: none"> 1) 对范围不明确并经常发生变化的项目，详细的估算没有多大意义，因为有可能估算好的需求后续就不做了。可以采用轻量级估算方法快速进行高层级预测。详细的估算适用于准时制的短期规划。 2) 如果易变的项目也遵循严格预算，会需要频繁更改范围和进度计划，以保持在成本基准内。
质量管理	<ol style="list-style-type: none"> 1) 在整个项目期间频繁进行质量审核，而不是在项目结束时才进行 2) 定期回顾，检查质量过程的效果，寻找问题的根本原因，建议改进举措；后续回顾会议评估举措的可行性并决定后续迭代是否采纳； 3) 为促进频繁的增量交付，敏捷关注于小批量工作，纳入尽可能多的项目可交付成果的要素。小批量的目的是在项目生命周期早期发起缺陷和问题，减少变更代价。
资源管理	<ol style="list-style-type: none"> 1) 易变性高得项目需要最大限度集中和协作团队，打造通才的自组织团队 2) 协作旨在提高生产率和促进创新的问题解决，协作型团队可以促进不同工作的整合，改善沟通，增加知识分享以及提供工作分配的灵活性。协作型团队集中分配任务和决策所需要的时间更少； 3) 变更性高的项目，实物和人力资源规划的可预测性比较低，快速供应资源的协议对控制成本和实现进度至关重要。
沟通管理	<ol style="list-style-type: none"> 1) 在模糊的环境中，要对不断变化的情况进行更频繁和快速的沟通。所以，需要简化获取信息的渠道，频繁进行团队检查并让团队集中办公。 2) 用透明的方式发布项目状态并邀请相关方评审项目组件，可以促进与高级管理层和相关方的沟通。
风险管理	<ol style="list-style-type: none"> 1) 越是变化的环境越是存在不确定性和风险，需要应对变化就要采用敏捷方法管理项目。通过跨职能团队和增量式的工作产品，加快知识分享，在每个迭代期间识别，分析和管理风险。 2) 根据当前风险敞口的理解的加深，定期更新需求文件，随着项目进展重新排列工作优先级。
采购管理	<ol style="list-style-type: none"> 1) 敏捷环境中需要与特定卖方协作来扩充团队，营造风险共担式采购模型，让卖方和买方共担项目风险，共享项目奖励 2) 大型项目可以通过主体协议（如主要服务协议 MSA）来管辖整体协作关系，而把适应性工作写入附录或者补充文件，这样一来，变更只针对适应性工作，而不会影响主体协议。
相关方管理	<ol style="list-style-type: none"> 1) 高度变更的项目需要相关方有效互动和参与。为了开展及时高效的讨论，适应型团队会直接与相关方互动，而不是通过层层的控制 2) 客户，开发人员在动态的共创过程中交换信息，实现更高的相关方参与度和满意度。 3) 采用高透明度的工具把工件发布到公共空间，让相关方达成共识。

4. 敏捷生命周期

4.1 生命周期的适应性



没有任何一个生命周期能够地适用于所有的项目，每个项目都能在上述连续区间中找到一个点，根据其背景特征，实现最佳平衡。四个典型的生命周期如下：

预测型生命周期：充分利用已知和已经证明的事物。不确定性和复杂性的减少，允许项目团队将工作分解为一系列可预测的小组。（关注：成本降低）

迭代型生命周期：允许对部分完成或未完成的工作进行反馈，从而对该工作进行改进和修改。（关注：持续改进）

增量型生命周期：可向客户提供完成的可交付成果，让客户能够立即使用。（关注：快速交付）

敏捷生命周期：同时利用迭代属性和增量特征。会对产品进行迭代，创建完成的可交付成果；获得早期反馈。可以提前发布产品，率先交付价值最高的工作，项目可以更早产生投资回报。（关注：交付价值）

4.2 生命周期的特征比较

特点				
方法	需求	活动	交付	目标
预测	固定	整个项目仅执行一次	一次交付	管理成本
迭代	动态	反复执行直至修正	一次交付	解决方案的正确性
增量	动态	对给定增量执行一次	频繁更小规模交付	速度
敏捷	动态	反复执行直至修正	频繁小规模交付	通过频繁交付和反馈实现客户价值