

极限编程评注



撰文 / 莫映

题记

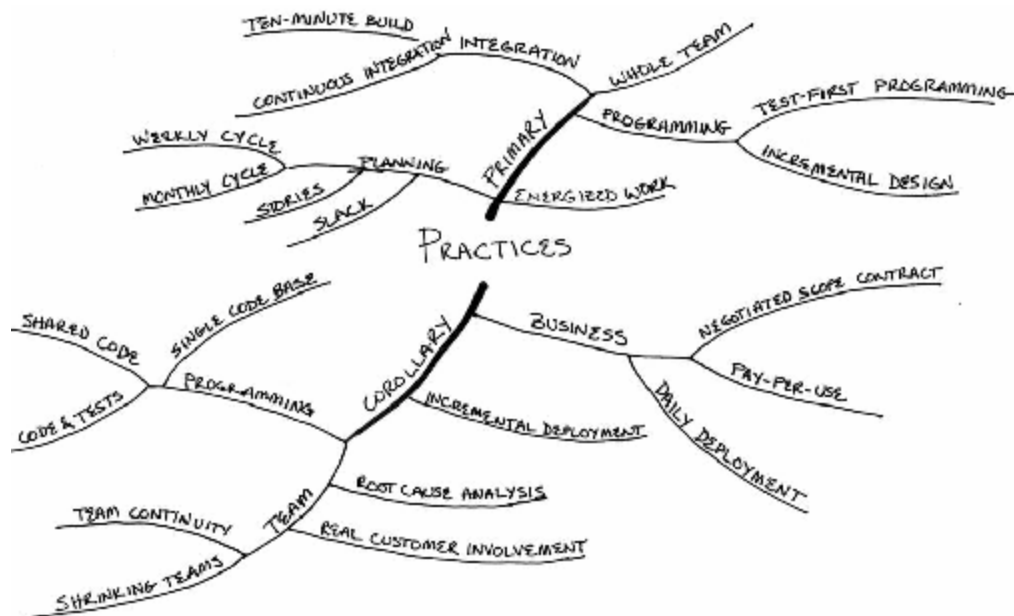
前不久，笔者有幸受编辑之邀，为 Kent Beck 的《解析极限编程（第二版）中文版》做校审工作。该书第一版发行于 1999 年，时隔数年，作为最具影响的敏捷软件开发方法之一，极限编程（以下简称 XP）已经日渐成熟。如果说该书第一版更像是程序员的宣言，内容多着墨于技术，那么作为 XP 缔造者的 Kent Beck，在第二版中对 XP 的内涵与外延又重新做了诠释，让该书有了更为广泛的受众。校审工作除了给我带来阅读的愉悦，也赋予了我重新回顾自己以往敏捷实践的极好机会。本文作为这一回顾的产物，在解释 XP 的同时，也融入了笔者的一点个人体会，希望能对读者更好的理解 XP 有所助益。

如何理解 XP

实践、价值观、原则是构成 XP 的三大基石。本节将对这三者以及彼此间的关系做一简单评注，它将有助于读者更深刻的理解 XP 的本质。关于 XP 实践的详细讨论，读者可以在书本中找到。

实践

实践是看得见摸得着的，很明确，也很有针对性。而且，即使对实践还没有深刻的理解，也不会妨碍我们直接拿来就用。XP 为我们提供的部分实践颇具可操作性，例如：持续集成、迭代计划，这些实践往往很有希望获得立竿见影的效果。不过，也有部分实践带着点“extreme”的味道，从而造成了一些人认为 XP 过于极端的误会。不过正如 Kent 所说，一些在几年前看似极端的 XP 实践，如今对于一个应用 XP 的团队，或是受 XP 思想熏陶的普通团队，早已经习以为常。当团队习惯了 XP 实践，“极端”也就消弭于无形了。



图：XP实践汇总（摘自《解析极限编程（第二版）》）

价值观

价值观是我们在某一情景下喜欢或不喜欢某事物的根源。价值观存在的必要性可以通过一个简单的例子来说明：作为XP实践之一的迭代计划，虽然我们可以不假思索的实施，但是负责估算故事的程序员们却可以隐瞒实际的估算值。这和实践本身实施与否无关，而对价值观认同与否有关——隐瞒估算意味着回避有效的沟通。透过这个例子我们也可以看到，确有必要将价值观置于一个显见的地位加以强调，因为这样可以避免实践的生搬硬套。事实上，其他软件开发方法也有各自的价值观，即使是一个没有明确开发方法指导的团队，价值观也依然体现在成员们的日常行事之中。

尊重

XP的价值观包括：**简单、沟通、反馈、勇气、尊重**。其中，尊重是《解析极限编程（第二版）》才引入的。**Kent**将它看作其他价值观的基础，足见其重要性：在一个团队中，如果成员对彼此的工作互不尊重、漠不关心，再好的方法也不会奏效，再先进的技术也无济于事。知道尊重的重要性并不等于不会犯错。稍加留意就会发现，人们很容易在日常工作中不时的在无意间破坏着尊重，而具体表现则是各异的：一味的强调个性和自我，一意孤行而不考虑别人的观点；上班时间没有效率的工作，从而影响了他人的工作；对自己代码中的“**Bad Smell**”置若罔闻，却让后继者叫苦不迭。没有人天生优于他人，每一位团队的贡献者都应该受到他人的尊重，这是XP得以成功实施的最基本保障。

XP的价值观构成了一个有机的整体，任何一条都不应被孤立的看待，否则就会误入歧途：没有其他价值观的保证，勇气就是鲁莽的；当涌现大量反馈时，应该缩短周期，让问题尽量简单；大家为了沟通而坐在一起讨论，但过后却无人付诸行动——将沟通作为起点并没错，但还要有其他价值观的保证才行。

原则

原则是介乎价值观与实践之间的，作为连接两者的桥梁，原则将有助于我们更好的理解实践。还是以例子加以说明：XP 倡导沟通，并将其作为价值观，而洋洋洒洒数百页的文档也算是一种沟通手段。因而，单从价值观出发，我们无法就这一“实践”的好坏做出明确的判断。于是，原则就有了用武之地。

另一个有趣的例子出自 Kent 的妙笔：如果想在写代码的时候获得代码质量的反馈，我们可以做持续构建和单元测试。但假如是在给婴儿换尿布，要想获得反馈，做持续构建和单元测试显然是让人啼笑皆非的。正确的做法应该是将婴儿抱起来，看看尿布会不会掉下来。由此可见，同样是作为价值观的反馈，在不同情境下所采用的实践是不一样的。为了让我们能够在众多实践中很好的做出选择，以体现价值观，必然存在一个中介，一种与具体领域相关的指导原则。

掌握了价值观和原则，实践就变得不那么重要了，我们甚至可以创造出自己的实践，而不是拘泥于书上的参考实践，只要这没有违背价值观。关于实践、价值观和原则三者的关系，Kent 举了一个园艺师的例子，非常生动：我们可以从书本上获得园艺知识（实践），但这并不能使我们脱胎成园艺师（XPer）。首先，园艺师要比我们懂得更多；其次，即使是我们所掌握的，他也比我们做得更好。园艺技能很重要，因为没有它压根就不可能成为园艺师。但是，即便学会所有技能，也依然成不了园艺师；也许我们正得意于自己的剪枝技巧，可园艺师却认为我们正在摆弄的那棵树根本就不应该种在这里。这是他根据花园的整体感做出的直观判断（价值观）。此外，在原则方面，园艺师也比普通人更有水准。我们也许知道如何将万寿菊和草莓种在一起，但是园艺师还知道“半裁法”（原则）对于两种毗邻作物的生长会有互补作用，万寿菊可以驱赶掉一些专门啃食草莓的害虫。这就是一位真正的园艺师（XPer）和普通人（普通程序员）的区别。

如何实施 XP

要成为真正的 XPer，真正领会 XP 的价值观和原则，唯有通过实践才行。对于如何应用 XP，我们可以通过入水的例子加以解释：有很多方法进入水池。你可以先把一个脚趾头伸到水里；也可以一步一步的往水里走；还可以做一个漂亮的高台跳水动作；或者像炮弹一样砸向水面，然后溅起一大片水花。没有哪种方法是绝对正确的。

从哪里开始

实施 XP 也和入水的道理一样。一次一个实践也许会看到稳步的改善，多个实践同时实施或许会有“加合”效应。下面给出的几条建议将有助于读者考虑如何开始实施 XP：

- 首先采用哪一项实践，最好根据实际环境，从你认为最有机会改进的地方入手。比如：我所了解的一个团队最早尝试的一项 XP 实践是持续集成，这在很大程度上是因为它听起来很“酷”，并且很快就能看到效果。是的，这个理由听起来并不理智，但是随着实践的深入，人们马上就会回归理智。不管怎样，从这样的实践开始，并没有什么大的不可逆转的风险。
- 在 XP 实践中，有些被称为基本实践，另一些被称为扩展实践，应该首先从基本实践入手。基本实践从任何一项开始都是可以的，因为它们彼此独立，互不影响。而扩展实践则是建

立在某些基本实践基础之上的，因此最好是在相关基本实践初步奏效之后再行实施，否则就有可能退回原地。至于哪些是基础实践，哪些是扩展实践，读者可以在书本上找到答案。

- **改变团队首先要从自身做起。**希望别人做自己不想做的事情，让别人冒自己不想冒的风险，从自身利益出发把新发现的知识和技能强加于人，这都是无礼的，也很难见效，而且还会破坏团队的凝聚力。按照 **XP** 的价值观，我们应该尊重团队里所有的人，甚至包括团队外、组织内的其他人，只要他们有助于我们的 **XP** 实践。即使 **XP** 的实施没有得到组织上层的支持，至少也要让团队内的气氛和生产力能够令人满意。变革之路始于脚下，因为这是你可以控制的变量。通过示范作用，你将用事实说服别人，你的方法是正确的。进而通过团队的共同努力，你将再一次用事实说服组织上层，以期团队改进创造良好的外部环境。
- **在为团队制定改进计划时，我们依然可以因循“XP 风格”。**就像迭代计划一样，我们可以在改进伊始列出所有可能的改进“故事”，然后在其中挑选一两个最有希望成功的，作为本次迭代周期内的改进目标。在迭代周期结束时，评估改进的情况，然后从“故事墙”上选择下一代中需要改进的内容。团队一定要在周期结束时做好反省工作，如果一个团队不能对过去的经历很好的反省，那它就无法健康的成长。

途中跑

在团队改进起步之后，总会遇到这样那样的问题，如何排除困难巩固改进的成果，下面的建议或许会有帮助：

- **要做好“迂回前进”的心理准备。****XP** 只是为我们提供了一个解决问题的契机，但它并不能掩盖或消除团队以往就存在的问题。在改进过程中，我们甚至很容易退回到原有的价值观和实践。任何团队改进都是在引入变化：打破大家以往积累形成的习惯，形成新的习惯。团队改进的过程就是从一个稳态跃迁到另一个稳态。实验证明，人们面对变化的本能反应往往是抵触。人们在行政压力之下也许不会拒绝改变，但是未必情愿改变。也许你很难一时说服他人违背自己的意愿，因为他们并未把自己看作集体的一部分，出于自身利益的考虑，加上外界压力的驱使，最终他们还是会倒退回原来的工作习惯。不要惊讶于这样的现象，也不必因此而得出错误的结论，比如：**XP** 太难实践了，**XP** 根本就是胡扯。此时，不妨冷静回顾一下你的改进之旅，反省过后再决定从哪里重新开始（前面的建议或许对你有帮助）。我曾经了解到一个项目，由于先期控制不力，导致系统上线后问题颇多。位于客户现场的部署人员与后方开发者的沟通又很不充分，导致 **bug** 从提交、修改再到确认的代价很高，而且还时有反复。当团队领导者意识到项目面临危急关头时，当机立断组织了多次现场与后方的“专场”沟通会议。这种极端的沟通形式可以算是团队对以往忽视沟通的一种弥补，同时也说明：错误的价值观和实践往往会把项目带入泥淖，但是经历过痛苦和挫折之后，人们也往往更加愿意尝试剧烈的变化。正所谓塞翁失马，焉知非福，也许距离成功已经不远了。
- **保持权力与责任的统一。**每个人都有畅所欲言的权利，团队上下应该对此有一致的认同。不过包括团队领导者在内的所有人都不应该忘记，在提出自己观点的同时，也要准备好用行动来支持这些观点。架构师从自身角度出发做出一个决定而不必承担后果，别的程序员还不得不服从；程序员根据自己的理解对系统提出质疑，却从来没有仔细求证过；成员们强烈建议编码时要有常用公共类和统一的异常处理机制，却不见行动……团队会因此而很快失去和谐。让所有成员真正明白这个道理，团队改进就会容易得多。

- **平衡个人与团队的关系。**团队需要多样性，多一个声音就多一种选择。但是团队中的每个人也应该清楚地认识到，要想获得个人成功，所有人都必须作为一个整体一起工作，这才是最明智的。一些人根据自己的偏好，在代码中引入新的技术，却没有意识到伴随而来的风险；另一些人不管代码中是否已有现成实现，就开始不假思索的根据自己的习惯实现重复模块……这些都有悖于 **XP** 的价值观，那些不惜代价要求个体自由的成员还需要认真学习价值观。优秀团队能够很好的处理个性与共性的矛盾，当人们建立起彼此信任之后，他们会发现一起工作的结果反而可以让每个人都更加自我。
- **避免角色的僵化。**一个 **XP** 团队包含了多种角色：程序员、项目经理，教练等等，每个角色承担了不同的职责。角色的明确化有助于团队成员培养新的习惯，但是也不必拘泥于此，我们的目标是让每个人为团队的成功做出最大的努力。因此，项目经理也可以参与测试，教练也可以参与开发。

澄清

本节所列的观点旨在澄清一些以往人们对 **XP** 的误解，相信这将有助于读者更加全面的理解 **XP**。

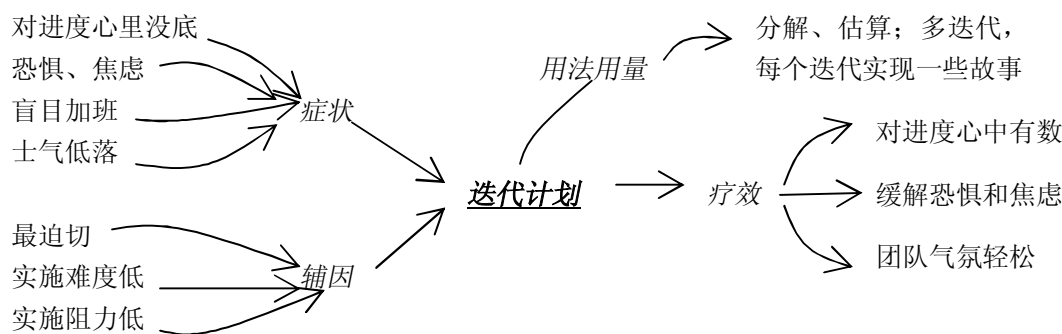
- **遗留系统中也可以引入 **XP**。**现实中，人们接触遗留系统的机往往会多一些。一些人对于遗留系统能否实施 **XP** 往往持怀疑态度：也许遗留系统的架构设计开始就是复杂的，也许人们已经习惯了传统的瀑布模型。不妨在整个软件生命周期的开头和结束阶段保持瀑布，然后将 **XP** 嵌入其中。在瀑布风格的开发过程中部分的应用 **XP** 实践，仍然可以获益。我接触的一个产品研发团队，就采用了这种混合的开发模式。
- ****XP** 不只关注技术问题，它也试图解决人际关系问题。****Kent** 在他书中的开篇就将 **XP** 称作“**social change**”，即表明了软件开发是一种人际协作。程序员们往往着迷于重构、测试驱动这些听起来很“酷”的 **XP** 实践。不过，历经数年的发展，**XP** 要比有些人想象的更丰富，也更细致。比如：也许你难以想象，**XP** 在某种程度上对 **HR** 的工作也提出了要求。**XP** 团队对新人的能力要求并没有很高的门槛，相反，能否与他人紧密合作才是更重要的。因此，有些敏捷团队在招聘新人时会让他参与到现有团队中，进行一段时间的考查。总之，**XP** 不止于提供实践技巧，它也思考着日常软件开发中我们所碰到的许多问题、困惑，以及诱惑，并且尝试着解决的办法。
- 对于“**是否是一个 **XP** 团队**”这样的问题，不存在二元回答。有些敏捷 **Fans** 们也许会特别在意这一点。其实，即使我们实施了书上的全部实践，也依然不能证明这就是纯正的 **XP** 团队，读者不妨回想一下前面那个园艺师的例子。相反，即使没有实施所有的 **XP** 实践，只要真实的体现了价值观，那就是有益的 **XP** 实践。而且，掌握了价值观和原则，实践是可以在结合实际环境的过程中创造发挥的。这对那些担心 **XP** 太过极端而驻足却步的观望者们，或许是个福音。

收获

本节内容涉及团队改进中常用的一些分析问题的方法，是笔者在阅读书本和实践体验时的一点收获。

实践图示法

在为团队引入某项 XP 实践时，我们可以借助图示的方法对实施情况做出分析。下图便是在我为某个项目团队引入“迭代计划”时所绘制的。图形的优势在于形象直观、简单易记，人们可以很好的将意图寓于图中，传达给他人，并在此基础上再行讨论和分析。



图：“迭代计划”图示

Kent 在《解析极限编程》中提供了一个关于“Energetic Work”实践的图示例子，有兴趣的读者不妨看一看。

对可控变量的分析

现实中，团队改进成功与否会受很多因素的影响。这些因素中，有些是我们能控制的，有些则是我们无法左右的。类似一个控制系统，我们首先找出存在哪些控制变量，然后分析哪些在当前情况下是可控的，这对认清形势对症下药很有帮助。比如对一般项目而言，典型的控制变量包括成本、时间、质量、范围。对不同变量的调节会有不同的效果，我们的日常工作不外乎固定某些变量，调节其他变量。

尾声

接受 XP 的观点也许会让人们感觉失去掩蔽，这让有些人感到不习惯，因为他们已经习惯于在“艰苦”环境下施展凌波“危”步了。隐藏错误保护自己，有时看起来是必要的，但其实这是时间和精力巨大浪费。不设防才是真正的安全，无论面对客户还是团队同伴，想象一下不再需要绞尽脑汁遮遮掩掩的感觉。从这个意义上讲，也许，XP 很适合某一类性格的人群。

在习惯以 XP 的方式思考问题之前，也就是成为真正的“园艺师”之前，我们还要学会逐渐提高自己的洞察力，找出自身无意间流露出来的与价值观不一致的行为。比如：作为 XP 团队的一员，你以 XP 为借口而不愿意写必要的文档，当上司询问时，你理直气壮的回答“没必要写，我们是 XP 团队”。不经意间，你已经忽视了作为 XP 价值观的沟通。XP 实践成功的关键在于，与我们的价值观协调一致的行动，要达到这样的一致性并非一日之功。

最后，这里所说的价值观是指真实的价值观，而非挂在嘴边的口号。在实际价值观与 XP 价值观不一致的组织中，XP 是无效的。