XP系列丛书

规划极限编程

Planning Extreme Programming

Kent Beck Martin Fowler 著
曹济 译
北京 SPIN 审校

Foreword by Tom DeMarco

XP 系列丛书

极限编程

Planning Extreme Programming

Kent Beck
Martin Fowler
著

曹 济 译

北京 SPIN 审校

人民邮电出版社 Addison Wesley Pearson Education 出版集团

图书在版编目（CIP）数据

规划极限编程 /（美）贝克（Beck, K.），（美）弗劳尔（Fowler, M.）著；曹济译. —北京：人民邮电出版社，2002.6
（XP 系列丛书）

ISBN 7-115-10379-8

Ⅰ. 规… Ⅱ. ①贝…②弗…③曹… Ⅲ. 软件开发 Ⅳ. TP311.52

中国版本图书馆 CIP 数据核字（2002）第 043149 号

XP 系列丛书

规划极限编程

◆ 著 Kent Beck Martin Fowler
译 曹 济
审 校 北 京 SPIN
责任编辑 俞 彬

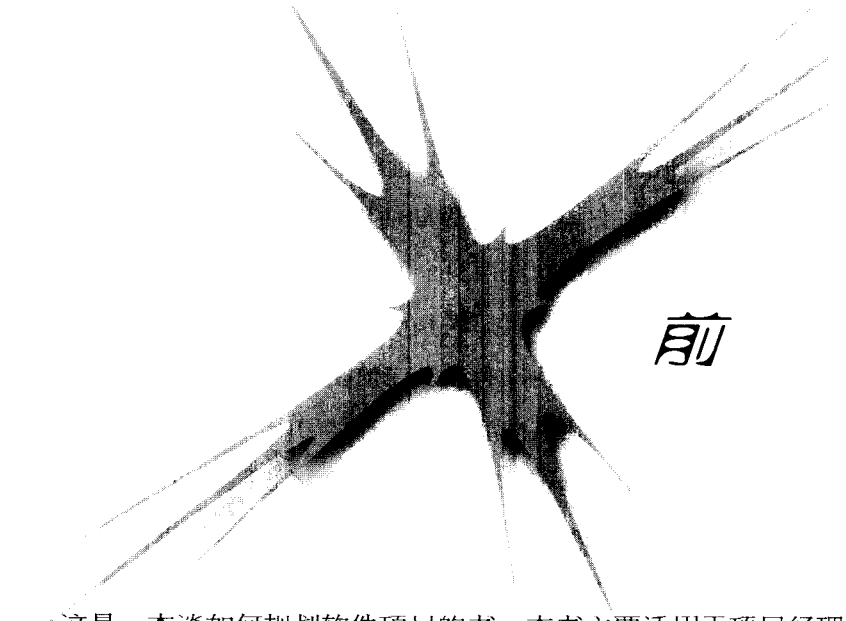
◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 http://www. ptpress.com.cn
读者热线 010-67180876
北京汉魂图文设计有限公司制作
北京顺义振华印刷厂印刷
新华书店总店北京发行所经销

◆ 开本：800×1000 1/16
印张：10
字数：159 千字 2002 年 6 月第 1 版
印数：1-5 000 册 2002 年 6 月北京第 1 次印刷

著作权合同登记 图字：01 - 2001 - 5029 号

ISBN 7-115-10379-8/TP • 2930

定价：23.00 元



前　言

这是一本谈如何规划软件项目的书。本书主要适用于项目经理——那些必须制订计划并针对实际情况跟踪计划进展情况的人。本书也适用于程序员和客户，因为在计划和开发软件的过程中，他们起着至关重要的作用。

计划并不是讨论如何预测未来。如果你为某一软件的开发制订了计划，开发工作并不会完全按照计划进行，确实如此。如果开发出来的软件和计划的软件完全相同，客户甚至会不满意，因为当软件完成的时候，他们又不想要原来计划的内容，而是与之不同的内容了。

与许多人一样，我们也欣赏艾森豪威尔的名言："在战争中，我一直认为计划是毫无用处的，但制订计划却是必不可少的。"[1]因此本书的内容不是关于计划的，而是谈如何制订计划的。由于制订计划意义如此重要，因而在开发软件的过程中，需要每天制订一些计划。

如果你按照本书的建议去做，那么你每天需要多做一项新工作——制订计划——但我们并不打算对此道歉，因为如果不制订计划，软件开发必然会脱离轨道。

---

1. 理查德·尼克松，《六次危机》（纽约： Touchstone 出版社，1990 年）。

本书涉及的范围很窄，所选内容都是经过深思熟虑的。它包括怎样为XP 项目的软件开发制订计划并跟踪开发过程。本书的内容基于我们担任顾问和讲师的经验，以及日益壮大的先期使用 XP 人员的经验。

因此，本书不是一本项目管理百科全书。书中没有涵盖项目经理的日常工作，例如员工业绩评估、员工招聘和预算安排。本书没有涉及由很多程序员一起来做的大型项目，也没有涉及任何在其他软件开发过程中如何制订计划的内容，或者如何为其他活动制订计划的内容。我们认为本书中包含了每个人都可以使用的法则和技术，但我们专注于我们所了解的那部分流程——引领团队中的所有成员朝同一个方向努力；及早发现我们工作中的变化；使得项目重新保持和谐的状态。

XP（Extreme Programming）是一个实践体系，一个软件开发人员群体正在逐步解决快速交付优质软件的难题，然后使它满足不断变化的业务需求。

XP 不只是关于制订计划的，它还涵盖了小型团队软件开发的所有方面——设计、测试、实现、部署和维护。但是，制订计划是解决 XP 难题的关键一环。（有关 XP 的概述，请参阅《解析极限编程——拥抱变化》。买这本书时，请购买本套丛书中的所有其他书籍。）

XP 处理长期项目的方法是将其分解为一系列独立的、为期一周到三周的小型项目。每次迭代过程中：

- ✧ 客户选择要添加的特性。
- ✧ 程序员把这些特性添加上，以便他们可以圆满地进行部署。
- ✧ 程序员和客户编写并维护自动测试，以表明这些特性是存在的。
- ✧ 程序员对系统的设计进行改进，以便恰如其分地支持系统中的所有特性。

如果没有仔细的计划，整个流程就会支离破碎。

- ✧ 团队必须选择可能实现的最佳特性。
- ✧ 对于不可避免的挫折，团队必须尽量采取积极的态度。
- ✧ 团队成员不能承担过多的工作，不然工作速度就会下降。
- ✧ 团队也不能在工作中偷懒，不然客户就吃亏了。
- ✧ 团队成员必须清楚开发的进度，并准确无误地把进度报告给各方，这样每个人都能相应地调整各自的计划。

日常计划人员的任务是使团队在所有这些领域内保持正确的方向。

制订项目计划的想法是迫于无奈的。作为顾问，常常有人把几乎无法继续进行的项目交给我们处理。这些项目要么是事先未做任何计划，要么是陷进了制订了太多计划的怪圈里。

结果产生的想法是采用我们能够想到的最简单可行的计划。但首先要记住如果你忘记了软件是由人编制的，那么世间制订计划的所有方法，包括这些方法，都救不了你。最后，如果能使开发人员工作时思想集中、心情愉快且积极主动，他们就能交付优质的软件。

Kent Beck

俄勒冈州 Merlin

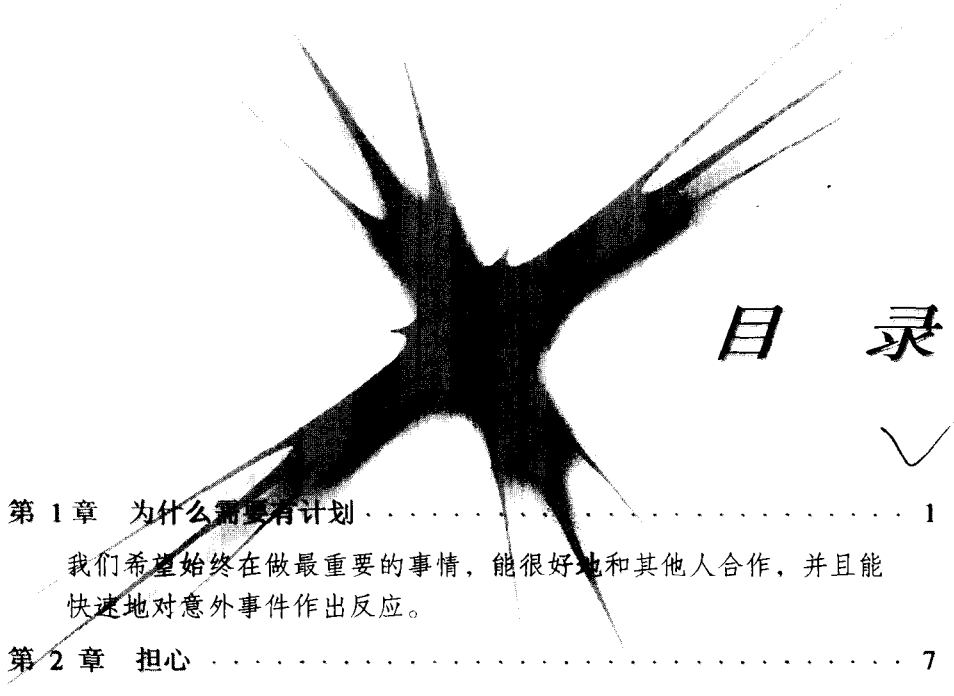
Martin Fowler

http://www.martinfowler.com 2000 年 7 月

马萨诸塞州 Melrose

我有一个绝妙的计划。

—— Baldrick，Blackadder



目 录

第 1 章
为什么需要有计划

所有人拟得最好的计划都是在快结束的时候前功尽弃。
– Robert Burns, "To a Mouse"

我们希望始终在做最重要的事情，能很好地和其他人合作，
并且能快速地对意外事件作出反应。

在 Kent 大约十岁时，他第一次去爱达荷州走廊地区用假蝇钓鱼。那天虽然一条鱼也没钓到，可是他们在小河里尽情地玩了个够，过得非常愉快。在回家的路上，他们在茂密的森林里跌跌绊绊地走了半个小时，后来发现迷路了。Kent 开始感到惊慌——呼吸急促，眼前发黑，浑身发冷。后来有人提议做个计划——他们要一直向山上走，直到找到那条通到他们熟悉的运木材的公路为止。一眨眼的功夫，最初的惊慌就消失得无影无踪了。

当时，Kent 深深体会到了计划的重要性。如果没有这个计划，他不知会做出什么蠢事来，也许还会患上紧张性精神症。有了计划以后，他又恢复了常态。

软件开发中的计划起着同样的作用。如果给你的工作期限很紧，但在制订计划之后，你发现可以在规定的期限里完成任务，这样虽然刚开始工

作时会感到有点紧张，但还是能够尽可能好地进行下去的。毕竟，时间足够了。正是这种状态能使计划在最大程度上实现。惊慌只会导致疲劳、出错，无法和别人正常交流。

但是我们也遇到过造成麻烦的一些计划。这些计划可能是个无底洞，把时间都吞了进去，而有人可能想用这些时间做些实实在在的事情。计划可用作鞭策人们的棍棒，最糟糕的是，它们可能会把问题掩盖起来，等到发现的时候已为时太晚。

1.1 为什么要做计划

我们做计划的目的不是为了预测未来。业务和软件飞速变化，对它们进行预测是不可能的。即使能够预测出三年内的事情，也没有多大用处，因为这三年内我们会需要做出很多更改。

要做的工作内容越清楚明了，就越应该问一问为什么要这样做。在处理重要的软件开发项目时，必须制订一些计划。因此，在开始规划一个项目以前，你必须了解为什么要完成这个项目。如果连这一点都不清楚，你怎么能知道是否能胜任呢？

需要做计划的原因有以下几条：

- ✧ 我们需要确保始终在做最重要的工作。
- ✧ 我们需要和其他人通力合作。
- ✧ 当意外事件发生时，我们需要了解前两项的因果关系。

第一条是为什么要制订计划的明显原因。如果对系统的某项功能忙活了大半天，最后却发现它实际上没有什么用，因为系统的下一个版本根本不需要这项功能，还有什么比这更令人沮丧的呢？做了某一项工作就会忽略另一项工作，如果另一项工作更重要一些，我们可能就完不成任务了。

比如说，现在是两点钟，我们正在波士顿。我们想驱车去阿卡迪亚，但还想理理发，到自由港去买一些野营用的用品。上次我们马不停蹄地开车赶到阿卡迪亚，用了五个小时。这样我们有几个选择。如果直接赶到阿卡迪亚，七点钟能到那儿。如果要在路上停下来吃顿饭，假如吃一个小时的话，八点钟能赶到。要是再理理发，又需要一个小时，这样的话九点钟能到，去自由港又需要一个小时。我们商量了一下，看看对我们来讲什么是最重要的。如果我们想吃饱，想带上野营用品，并且不想太晚到那的话，我们可以把外表搁在一边，去掉理发这一项。计划能帮我们更好地做出选择。

其他人之所以希望我们做计划，是因为我们需要协调合作。我们的妻子打电话来约我们到 Bar Harbor 吃晚餐。现在是两点钟，如果我们现在起程，路上在停一下，大概八点钟能赶到那里。软件开发也充满了这样的协调：销售计划、财务周期或管理政策。制订计划能使我们了解什么是合理的。

但是制订计划的作用只和估算的作用差不多（计划以估算为基础），而估算出来的结果总是比事实相差一步。如果我们路上遇到了严重的交通堵

塞，那么世界上所有的计划都无法帮助我们按时赴约。现实世界有破坏计划的可恶习惯，正如 Burns 先生在本章开头所指出的。

虽然现实世界常常扰乱我们的计划，但做计划仍然是有用的，因为有了计划，我们就可以预见意外事件造成的影响。我们两点出发，路上遇上了堵车，都五点了还没赶到波特兰。通常到这里只需一个半小时，所以根据以往的经验（和计划），我们打电话让朋友把晚餐推迟到八点半，并放弃了去自由港的计划。计划使我们既可以调整手头的工作，又可以和其他人进行配合。关键的一点是，一旦知道了意外事件会造成什么影响，就要相应地调整计划。我们的妻子希望在五点的时候就能知道我们被耽搁了，而不是到八点才得知，如果由于去自由港耽误了时间，不能和妻子共进晚餐，会让人感到很懊恼。（我们甚至不愿考虑这种可能性，相比之下，软件开发的失败不过是小事一桩罢了。）

1.2 制订计划时需要些什么

人们制订的计划有长有短：你可能计划你每天的活动；团队订出它几周的任务；开发部门和业务部门为来年制订计划；高级管理人员为大公司开发计划。如果你正从波士顿驱车前往阿卡迪亚，你不会考虑路上的每一个拐弯，但会先想好该走哪一条路，什么时候从一条路转到另一条路上。你不大可能一分不差地到达目的地，可是有时候，我们需要打个电话，为不能及时赶到说声抱歉。

为了更好地进行协调合作，你必须清楚地知道计划进行到什么地方了，这是至关重要的。在旅途中，这相当简单。你可以测量里程，并考虑到路况，然后粗略地画出一个时间表，把到达重要地点的时间标出来。如果到波特兰的时候已经很迟了，你可以很容易地对到达 Bar Harbor 要延迟的时间做出类似的估计。软件的实质特性与以上所述的旅途特性完全相反。由

于软件开发具有随意性，因此很难确定你已完成了 70% 还是完成了 30%。就像你正在旅途中，不知道自己是走了 30 英里还是 300 英里一样。没有任何东西做参照，你会觉得心里很不踏实。如果很晚你还没有到约会地点，而你的约会伙伴又不知道你什么时候能到，她也会感到不踏实的。

任何软件计划方法都必须努力使计划的进度一目了然，这样从事项目的每个人就都能真正了解项目的进度。这就是说，你需要把一些重要的事项标出来，也就是那些不能蒙混过关、能够明确代表项目进度的事项。这些重要事项还必须是项目涉及到的每个人（包括客户）都能理解和信任的。

计划是关于如何确定项目中各事件可能的发展过程以及一些无法避免的变化会造成什么样的后果的。不同的项目规模需要有不同的计划，但是所有计划都必须既简单易行又易于随时更新。由于复杂的大型计划制定和维护的成本都过高，因此不能起多大作用。由于计划涉及到协作，所以对于受到计划影响的人，都必须能够理解计划——这也是我们强调简明性的另一个原因。

计划必须真实可信，迄今为止的所有信息都应该是这样。计划应该令那些企图通过提交与事实不符的进度报告来蒙混过关的人难以得逞。

1.3 计划的陷阱

上一段中给我们提供了一个关于为什么计划也可能是一个陷阱的线索。这是因为之所以要制订计划还有另一个原因：那就是用于表明事件在他们的控制之下。

但是控制事件是一个矛盾的说法：你不能控制事件，你只能控制自己的反应。即使是后者，你的控制范围也是有限的，事件会改变计划。一旦你遇上了交通阻塞，你的晚餐或者自由港购物计划都会受到影响。你无法装出什么也没发生的样子继续按计划行事，那样就太愚蠢了。

可是，我们遇到过很多这样的蠢事。如果事情没有按照计划进行，制订计划的人会担心受到指责，于是就说计划进行得很顺利。但他心里清楚已经偏离了计划，但如果计划足够复杂，他甚至能把这一情况掩盖起来。对计划的制订人来讲，最重要的是要向外界说，一切都在按原计划进行。

这样，计划就偏离了事实，变成了一个幻想。更糟的是，制订计划的人还投入精力来维持这一幻想，不让它破灭。于是开发人员渐渐失去了开发的动力。如果所谓的计划只是一个幻想，为什么还要照着它去做呢？

也许最后项目本身能解决问题。有时候是这样的。也许有些人遇到了比这更糟的问题。但更常见的情况是，幻想和现实之间的距离不断变大，直到幻想破灭。这时事情会变得很糟糕：客户会很生气，因为她是根据这个幻想来制订她自己的计划的，而且做出的是她永远无法实现的承诺；程序员也很生气，因为他们确实对工作尽心尽力了，但现在却因为没有完成不可能完成的事，没有把幻想变成事实而受到责骂。

新的情况不断出现，计划也应随之而改变。如果一切都完全按照计划进行，通常不是什么好兆头，很可能发生了问题。对于一个项目来讲，可能发生的最糟糕的事情就是计划和现实之间相背离，所以不要掉入这个陷阱里。保持计划的真实性，并始终做好修改计划的准备。

第2章 担心

是勇气！是什么使一个奴隶成为国王？
是勇气！是什么使船桅上的旗帜飘扬？
是勇气！是什么使大象能在朦胧的黎明或在昏暗的黄昏挺起象牙，勇敢冲锋？
是什么使麝鼠保卫它的麝囊？
是勇气！是什么造就了世界第七大奇迹斯芬克司？
是勇气！为什么他们得到而我得不到？
– Cowardly Lion，The Wizard of OZ

软件开发是有风险的，有关人员非常担心什么都可能会出错。为了有效地进行开发，我们必须承认这一事实（这些担心）。

为什么需要软件过程？原因和我们需要法律、政府和税收一样：担心。《独立宣言》中写到：

这些权力包括生命的权力、自由的权力和追求幸福的权力。为了保证这些权力，政府是来自于人民的，政府的正当权力是由人民赋予的。

虽然这些话可能过于深奥，不好理解，但这些措辞是可靠和必要的。