

10

用故事大小值估计工作

欢迎来到敏捷之岛！我们在工作坊的一个练习中会用到这个地图，用来阐明人类给出正确估值方面的能力，如图 10-1 所示。

首先，我们跟学生们展示这幅敏捷群岛的地图，问他们“马上回答！从 Fowler 到 Beck 你需要花多少时间？”工作坊参与者们通常都会卡壳，这是个很难回答的问题，因为你不知道你是用飞的还是划的，也不知道要讲英里、里格（league）¹还是纳米。

我们接着又问，需要多少时间才能完成从 Beck 到 Jeffries 的第二

¹ 译者注：里格是长度单位，1 里格等于 3 英里，参看 <http://www.collinsdictionary.com/dictionary/english/league>。纳米，nanometer，也即十亿分之一米，参看 <http://zh.wikipedia.org/wiki/纳米>。

段行程，这时有意思的事情发生了。

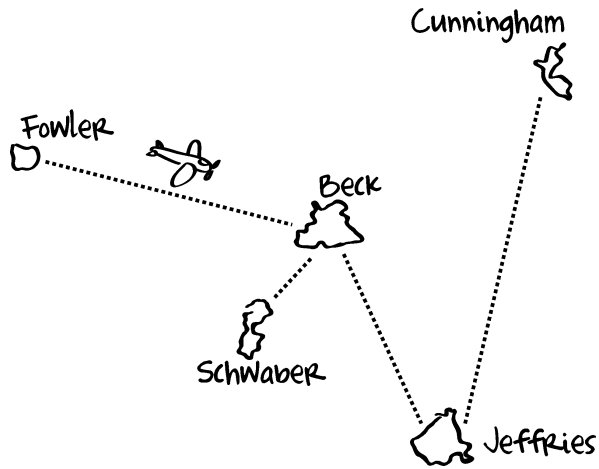


图 10-1

“我没法说需要多久，”有个学生很聪明，回答总是很响亮，“但是 Beck 到 Jeffries 看起来是 Fowler 到 Beck 距离的 $\frac{2}{3}$ 左右，所以我猜它花的时间应该也在 $\frac{2}{3}$ 的样子。”

结果证明，参与者们给出的相对答案总是异常地准确，虽然他们都不知道度量单位是什么。也就是说，所有人凭直觉就知道 Beck 跟 Jeffries 的距离只有 Fowler 跟 Beck 距离的 $\frac{2}{3}$ 。不管你是坐船还是坐飞机，行程都是 $\frac{2}{3}$ 的长度。

要查明“多久”只需做一件事，完成一段行程看看它到底要花多少时间即可。数据到手后，相对大小就能派上用场了，配合首段旅行的数据，可以对其他行程长度进行有效的预测。

要估值干嘛

我们要估值究竟想干嘛？毕竟得花时间才能生成估值。直接开工

干活岂不是能完成更多些？

生成估值的真正目标是要提供进度的可预测性度量。如果可预测性具有实际的业务价值，那就值得花功夫争取。碰上最终期限的情况，例如产品需要赶在假日购物季上架，那我们就得做出一个艰难的决定，专注于某些特性地同时也得推迟甚至放弃一些特性。这种情况下，做到所有相关特性的大小心里有数是非常重要的，这样才能做出重要的业务决策。我们生成估值就是为了要支撑业务决策，而它最终将创造更多价值。

相对大小 vs. 时长估值

传统的估算方式是询问开发人员需要多长时间才能完成工作。这种方式有两个问题：第一，他们并不真正知道要花多少时间；第二，不管怎样他们都会给你一个答案。

事实证明，虽然我们对相对大小拿捏得很准，但对所需时长的估计却相当糟糕，数量巨大的超期项目充分地反映了这一弱点。

我们拙于猜测事情需要花多长的时间，但并非没有好消息，我们擅长比较两个事物并判断其大小关系，借此我们仍可得到有效的估值。更棒的是，我们还很擅长判断两相比较的大小倍数。换句话说，虽然绝对大小我们不擅长，但相对大小我们做得很棒。

窍门在于使用一个两步流程。第一步，给所有的工作项都分配相对大小值。大小值表明需要完成的工作有多少。

第二步，完成几个工作项，度量它们实际花费的时长。得到这个

实测数据之后，结合指定给其他条目的相对大小值使用，预期的进度可见性也就有了。这也正是大多数 scrum 团队的做法。

首先，他们估出故事相对其他故事的大小值，大小为“2”的故事是大小为“1”的故事的两倍。团队可以把最小故事的名字用做单位，如果最小故事是“登录页面故事”，那么就可以用如下方式给所有更大的故事估大小，例如“2个登录页面”、“3个登录页面”、“8个登录页面”等等。这个命名法用用没问题，只是有点笨拙。敏捷开发人员们设计了一个通用单位，可以替代登录界面使用，他们称之为“故事点”。有些开发人员比较懒，进一步简化到干脆就叫“点”。本书中我们会交替使用故事点和点，呃，取决于我们写作时的懒惰程度。关键在于，要记住团队使用故事点做单位，指定的是故事的相对大小值。

总而言之，故事点是用来度量完成故事所需工作量的相对单位。

第二步（要记得我们在此描述的流程有两个部分），团队工作 1 个 sprint 的时间并完成一些故事。此时再描述他们平均每个 sprint 完成工作速率的时候，就可以使用故事点了。“平均每个 sprint 故事点”说起来太拗口，所以就有了“速率（velocity）”的说法。如今，不需要讲“我们团队平均每个 sprint 完成 144 个故事点”，只要说“我们的速率是 144”即可。它就是那样的简单、朦胧。

你的速率就是平均每个 sprint 所完成的故事点数量。

斐波那契数列

我们先暂停不谈软件开发，到芝加哥找房子去。我们的第一站是北面的郊区小镇斯科基。站在漂亮的街边，我们面前是两栋房子，一

栋是 20 世纪 50 年代造型，单层牧场住宅（ranch house），另一栋是殖民风格双层乡村豪宅（colonial McMansion）。你能只看外表就知道哪栋房子有更多楼层吗？你会说，当然，很明显。¹

但我们是市区蚁族，不是郊区蚁族，于是我们掉头前往时髦的湖岸高速区²，那可是奥普拉居住的地方。现在我们看到的是一些高层建筑，其中一栋 46 层高，另一栋 47 层高。站在路边仰头看，你真能看出哪一栋的楼层更多吗？也许不行。

这个故事诠释了有关人类感知的一个重要事实，东西越大，我们对它们在大小上细微差别的感知能力也越差。估计故事大小时，我们无法真的说出 53 点故事和 57 点故事之间有多大差别，但这却阻止不了工程师们为此而争论不休！

为了制止软件团队继续这种无意义的争论，我们使用斐波那契数列的数字估计故事的大小。用最简单的术语说，斐波那契数列由整数构成，每一个数都等于前两个数的和，1、2、3、5、8、13、21、34、55，以此类推。

公元前 200 年的时候，印度学者发现了这个数列，但直到 13 世纪才由西方命名，用以纪念意大利商人列昂纳多·斐波那契³，他普及阿拉伯数字的事迹广为人知。（因为他，你不必在小学三年级时还要做

¹ 译者注：Skokie，美国伊利诺伊州库克城的小镇，http://en.wikipedia.org/wiki/Skokie,_Illinois。Ranch House 和 McMansion 名称请参看《城市·环境·设计》杂志第 50 期文章“135 种住宅类型”。

² 译者注：原文 Lake Shore Drive，沿着密西根湖而建，贯穿芝加哥。美国著名主持人奥普拉·温弗里曾在此拥有住宅，但因私密性不够而从未入住，翻译此书时已出售。可以参考 http://en.wikipedia.org/wiki/Lake_Shore_Drive，<<http://zh.wikipedia.org/wiki/奥普拉·温弗里>>。

LXXXVIII 乘以 XLII 的计算¹，如果你想感谢他，我们会等你。）

在研究兔子生长数量的时候，列昂那多首次观察到这个数列，发现它频繁地出现在自然界中。海贝壳的螺旋、蓟的叶子和树的分叉都符合了斐波那契数列。很多人造事物也用到了这个数列，例如财务分析工具、乐曲和 20 世纪 80 年代那些致人迷幻的分形海报（trippy fractal poster）。

然而我们更关注的是，用斐波那契数列的数字来表示“大小”时，配合其数字增长速率，人类仍能轻易地察觉到其中的差异。任何人一眼就能分辨 1 层和 2 层的房子，也能分辨 21 层的建筑和 34 层的建筑，或是长 13 尺和长 21 尺的帆船，或是重 34 磅和 55 磅的狗，或是装有 89 只和 144 只兔子的笼子，诸如此类。

当然，我们刻意地严格使用这一组数字，并不是说某个给定的故事正好值 21 个点。我们说的是，相比 13 和 34，它更接近 21。这已足够准确，能给我们有效的进度可预测性。你可以回忆一下，有效的进度可预测性就是我们一直以来所追求的东西。它正是我们不厌其烦地做估算的原因。

图 10-2 是一套典型的斐波那契卡片集，在故事时间会议中使用起来非常顺手。

你可以网上购买，也可以用索引卡自己做。

¹ 译者注：斐波那契的介绍可参看 <<http://zh.wikipedia.org/wiki/斐波那契>>。LXXXVIII 和 XLII 均为罗马数字，分别为 88 和 42，关于罗马数字请参看 <<http://zh.wikipedia.org/wiki/罗马数字>>。

1	2	3	5
8	13	21	34
55	89	144	?

图 10-2

请注意，我们数列末尾的数字是一个问号。这张卡片的意思是“我们真的不知道”。受“它是 55 还是 89？”这种问题困扰时，你不用求助这张问号卡。遇到那种情况，别太自以为是，选个数就行（提示：选择那个大的）。在你不确定故事应该是 2 或是 144 的时候，就得使用问号卡了。缺少某些关键信息，就会出现这种情况。那样的话，你可以先试着明确缺失的信息，问自己“如果我们想估计这个条目的大小，有哪些问题需要解答？”

你也可以新建一个特殊类别的“研究探针”¹故事，给原来那个不明确的故事标上问号。

研究探针可能简单到只要“问客户要更多信息”，也可能会需要写点代码以便更多地了解某个不熟悉的技术。

团队估算游戏

要帮助团队运转起来并给出有效估值，我们认为**团队估算游戏**是最佳办法。用此方法就像是在玩游戏，但它的成果却很有价值，能为用户故事指定故事点估值。

¹ 译者注：spike 是极限编程的一大准则，可以参看 <http://www.extremeprogramming.org/rules/spike.html> 和 <http://c2.com/xp/SpikeSolution.html>。

一般来说，团队用此方法能在 1 小时内估值 20 到 60 个故事。我们的朋友兼同事 Steve Bockman 发明了这个游戏。如下是团队玩此游戏的过程：

团队估算游戏第一部分：集体排队

Frank 是团队的 scrum master，他已经在团队室的墙上清理出一块很长的空间，团队也已经聚集在面前。Brad 是产品负责人，他带来了产品列表中的一叠共 30 个用户故事，团队将要用团队估算游戏估计它们的大小。

“Kira，不如你先来？”Brad 边说边把故事卡递给她。Frank 手里拿着一卷蓝色美纹纸胶带，也撕下一小片递给了她。

Kira 开始了这个游戏，她取出最上面的那张故事卡，大声地念出来，然后将卡片贴到了墙壁中间位置。接着她把剩下的卡片递给了 Kai。

Kai 继续从最上面挑了一个故事并念给所有人听。“我认为这个比 Kira 刚刚贴的那个更大。”Kai 说完就把他的故事贴在了 Kira 那个故事的右边。

接下来是 Mark。在他看来，他读的是一个小故事，因此他把它放在了其他故事的左边，如图 10-3 所示。



图 10-3

Jeff 接着又从中拿出一个故事。“这个也挺小的。”他犹豫片刻后，把 Mark 的小故事继续向左移，给他的故事留出位置。“但还没有 Mark 贴的故事那么小。”

团队轮流地把故事贴上去。第三次轮到 Kira 的时候，她并未挑选新故事。相反，她调整了已在墙上的一个故事的位置，把它移到了更右边的位置。“相信我，”她说，“这个故事的遗留代码是个烂摊子，我们得把它全改成线程安全这故事才能用！”

很快，所有的故事都已上墙，但团队仍继续轮番上阵。不继续贴新故事，他们只调整顺序，一次移动一个，这个过程有时候很安静，有时候人们也会讲几句。

新一轮开始，“通过。”轮到 Malay 时他说，示意他已经很满意故事的顺序。Justus 同样也通过。Kira 和 Mark 各自又移动了一个故事，接着就都通过了。终于在某一轮的时候，他们全都通过了。团队估算游戏第一部分结束！

团队现在已经按照从小到大的顺序把故事从左到右全都排列整齐了。大家都认同所需工作量最少的故事被放到了最左边，而大家认为所需工作量最大的故事被放在最右边。

细心的朋友可能已经发现，这个游戏有可能会陷入一个无限循环。Mark 可能把故事放到右边，而 Kira 又把它移回到左边。Mark 在下一轮的时候可能又把它移到右边，然后就一直这样地重复下去。虽然理论上的确存在无限循环的情况，但我们玩这个游戏已经数百次，这种事情还从未发生过。

团队估算游戏第二部分：你的数字是？

作为团队估算游戏第二轮的准备，Frank 制作了一组斐波那契卡片。整组卡片每一张上都有一个斐波那契数字，从 1 到 144。

Mick 先开始。他起身走到墙边，指向最左边的故事，摆出一幅幸运之轮里 Vanna White 的样子¹。“女士们先生们，这个差不多就是我们能看到的最小的故事了。”他把斐波那契卡片“1”贴在了这个故事上方的位置。

接着是 Justus。他手持卡片“2”，看着墙上的故事，寻找墙上的故事开始变成“1”那个故事两倍大小的地方。他选好位置，然后把卡片“2”放在了从左边数过来第四张故事卡的上方，如图 10-4 所示。



图 10-4

继续几轮之后，所有团队成员都已经放置过一张斐波那契卡片，就放在他们认为大小开始突变的那个故事的上方。

再次轮到 Kira，她有些犹豫，接着指向了两个故事。“你知道，”她说，“我想我们可能得调换这两个故事的顺序。我认为这个是 8，而那个

¹ 译者注：Wheel of Fortune，美国的一档电视节目，形式和国内的“幸运 52”比较像，可参看 [http://en.wikipedia.org/wiki/Wheel_of_Fortune_\(U.S._game_show\)](http://en.wikipedia.org/wiki/Wheel_of_Fortune_(U.S._game_show))；Vanna White 是这个节目的主持人，请参看 http://en.wikipedia.org/wiki/Vanna_White。

是 13。”她利用自己的轮次调整了这两张故事卡的顺序，然后把数字卡片组递给了 Mark。

Mark 把卡片“21”贴在一个故事上方。接着轮到了 Malay，他摇了摇头，拿掉了 Mark 刚刚放上的卡片“21”。“我觉得这个实际上应该是个 34，”他提到了斐波那契数列中紧接着最大的数字。

他把卡片“21”换成了卡片“34”。

“他是对的。”Jeff 也说。

Jeff 帮助 Malay 移动故事卡，在最后一个大小 13 的故事和第一个大小 34 的故事之间创建出一块空白地带。团队在第一轮放置故事卡片的时候，就放得比较松散，留出了足够的空间可以移动，因为他们知道这样有助于第二部分的时候进行调整。

Malay 把卡片“21”就贴在故事所在行这个空白地的上方，示意尚无此大小的故事，如图 10-5 所示。如果所有人都觉得大小排列很让人放心，纷纷选择轮空，那么游戏到此就结束了。

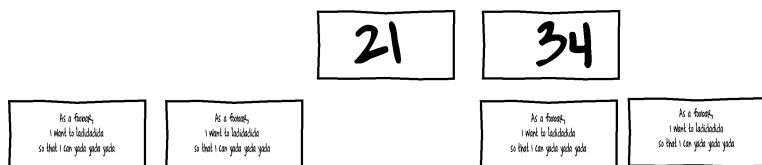
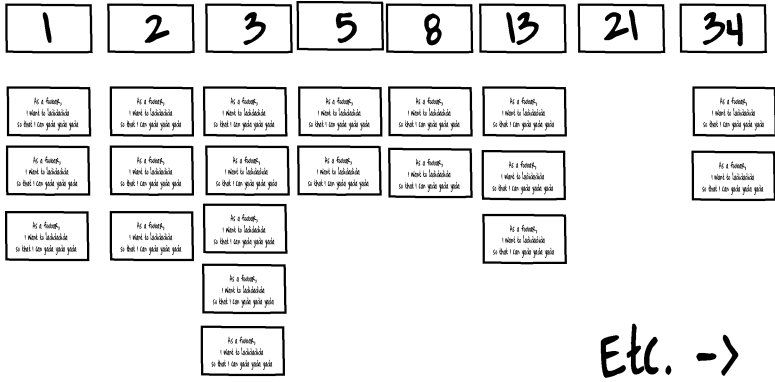


图 10-5

此时团队稍作收拾，移动故事卡片到斐波那契卡片下方排成列，如图 10-6 所示。所有“1”和“2”之间的故事全都收起来，在卡片“1”的下方排成一列，这些都是 1 个点的故事。下一列包含所有 2 个点的故事，以此类推。团队最终还是没有在“21”下面放任何故事，

因此这一列也还是空的。



Etc. ->

图 10-6

我们描述的是游戏最简单的形式。在自己的团队中使用时要注意，故事大小最小值并不是非从“1”开始算。如果玩家认为未来可能还会有更小的故事，比目前墙上最小的故事还要小很多，那他们可能会想给第一个故事标成“2”或“3”，而不是“1”。这样做的话，未来如果遇到比现在最小故事还要小的故事，还有数字可用。例如，某玩家在最左边那个最小故事卡片的上方放置卡片“2”，意思是说，他们认为团队未来可能会遇到只占这个一半工作量的故事。

我们把游戏教给和我们合作的那些团队，很多团队都告诉我们，他们从未见到过这种情况，项目刚发起居然整个团队都相信这个估计很准确。制定所有人都信服的计划就该用这方法！

团队估算游戏：规则

第一部分：集体排队

所有玩家轮番上阵，在自己的回合可以做如下任何一件事：

- ◆ 在墙上放置新的故事卡。
- ◆ 移动已在墙上的故事卡。只要保持其他卡片的顺序，为了给那张换位卡留出空间而滑动很多卡片也完全没问题。
- ◆ 移交行动权给下一个玩家。

卡片从左到右按照从小到大的顺序放置。值得把它们摆宽松些，后面你调整顺序的时候才方便。所有人都选择轮空时游戏宣告结束。

第二部分：你的数字是？

所有玩家轮番上阵，在自己的回合可以做如下任何一件事：

- ◆ 在某张故事卡上方放置下一张斐波那契卡片，表明此处的故事大小增加了。
- ◆ 把斐波那契卡片换到另一个故事的位置。（移动时必须维持数字的顺序，也即是说 1 在 2 之前，13 在 21 之前。）
- ◆ 像第一部分那样移动故事卡片。
- ◆ 移交行动权给下一个玩家。

所有人都选择轮空时游戏宣告结束，意味着故事的顺序和大小分配都已无需再做调整。

计划扑克

等你适应即时估计故事大小后，就已经准备好可以使用另一个巧妙技法，也即是 James Grenning 于 2002 年设计名为计划扑克的估算游戏。2005 年 Mike Cohn 的《敏捷估计与规划》让计划扑克变得流行起来，它是一个精心安排的游戏，估计任务时用于达成群体共识。它之所以得名扑克，是因为它的确用到了一组卡片，还是斐波那契卡片。而且，游戏的目标是达成共识，所以人人都是赢家。

游戏开始时，每个团队成员都拿到一组斐波那契卡片。引导者，例如你们的 scrum master，要大声地读出故事的内容。队员们各自选择一张卡片，代表他们对任务难度级别的最佳猜测，然后所有人同时揭露卡片。如果所有估值都相同，那就完工了，不需要再讨论。

如果估值比较分散，那么卡片数值最大和最低的玩家就获得了发言机会，辩护自己的选择，然后游戏再来一遍。如果新估值都相同，那故事就已经估好了。如果还有一两个意见不同，那就要问他们能否认同大多数人的估值。如果估值分散得很厉害，那就得继续讨论，游戏也得重新开始。事实上，经过多轮结构化讨论之后，不同估值会迅速地汇聚。

速率

等你们团队做完一到两个 sprint 后，对于你们在单个 sprint 内所能处理的故事点数目就能有更清楚的认识了。所谓团队速率也就是实现成就的速度。

产品负责人可以参考团队速率，为后续 sprint 选择故事。如果团

队速率是平均每个 sprint 完成 20 个故事点，产品负责人可以相应地做出选择，例如一个 8 点故事、两个 5 点故事和一个 2 点故事。或者也可以选择四个 5 点故事。无论哪种方式，他们都可以基于自己的判断做出决定，对剩余故事进行优先级排序，选定带入下一次 sprint 规划会议的故事。

速率已知的情况下，需要进一步拆解以放入单个 sprint 的故事就变得显而易见了。

绝不要把速率当做绩效指标使用，它可不是用来向管理层展示你工作有多快的，它是为了提高进度可预测性从而产出更高价值。

新团队的速率刚开始可能会增长，但最终还是趋于稳定，仅有小幅波动。