

第13章 创建你的第一个产品

作为一名软件开发人员，你拥有成为一名企业家得天独厚的优势，不仅能提出概念或新想法，还能自己把它创造出来。正是出于这个原因，许多开发人员选择创业、选择自己创造产品，而别的企业家不得不雇用他人来实现自己的想法。正如你所知，开发定制软件可是价格不菲。

作为软件开发人员，你不仅能够开发软件产品，还可以开发书籍和视频这样的信息产品。

在本章中，我会帮助你了解，创建自己的第一个产品，并开始漫长而又崎岖的创业之路你需要知道的。不过我要警告你，你即将踏上的这条道路坎坷不断。

找到受众

许多软件开发人员一开始就深陷创业者最常犯的错误之中——在为产品找到客户之前就构建好产品。从构建产品开始可能有些道理，不过你要避免落入这个陷阱之中，否则你只是冒险为一个不存在的问题创造了一个解决方案。

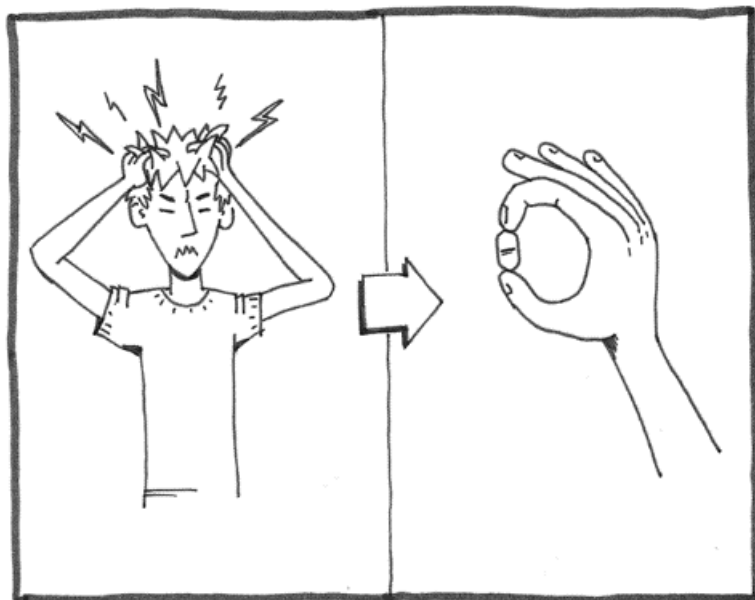
人类创造出的每个产品（包括这本书在内）都是为了解决某个特定的问题。没有要解决的问题的产品毫无意义，毫无意义的产品自然也就不会有人用，也就意味着你不会赚到钱。有些产品专门解决一些特殊群体遇到的特定问题。例如，一款能够帮助牙医管理自己的病人的软件产品，或者一本能帮助软件开发人员学会如何使用.NET Unity框架的书。还有一些产品解决普遍的问题，例如，无聊——娱乐产品（如电视节目和视频游戏）可归入此类。无论一个产品要解决何种问题，以及哪些人群有这个问题，必须在创建产品之前就要明确识别。

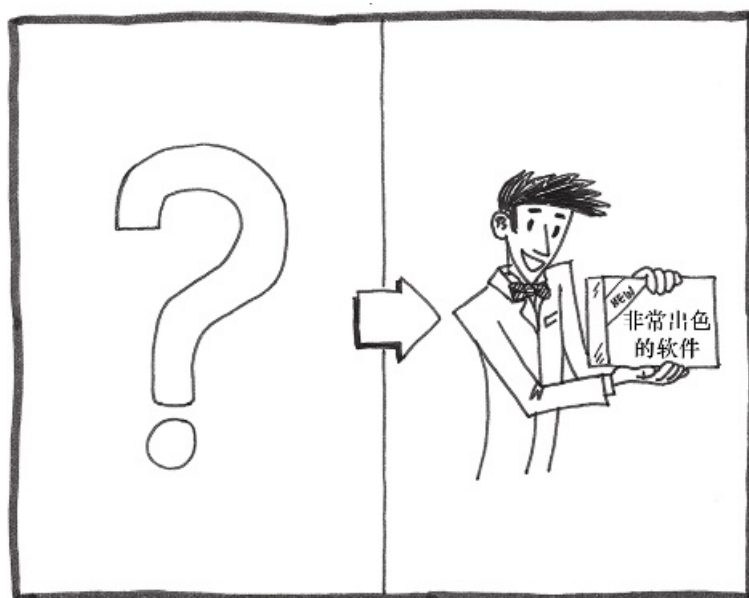
如果你想开发出一款产品，第一步应该是筛选出一组特定的受众，他们也是你的解决方案的目标用户。针对这些人你要解决的问题是什么，你可能已经有了总体概念。不过在很多情况下，你要多做些调研，找出要么没被解决的问题，要么没有被很好解决的问题。

去目标客户常去的地方，与用户参与的社区交流，了解一下普遍存在哪些问题。你能从中看到的痛点有哪些。

我开始注意到这样一个趋势——软件开发人员向我咨询如何打造业界声誉，如何提高自己的曝光率，或者如何增加关注度。许多访问我博客的软件开发人员也会问我相关的问题。我看到这里有一个实际问题——软件开发人员有学习自我营销的需要。（在这个案例中，我的受众是直接通过我的博客告诉我他们的问题的，所以事情就变得简单了。这是要拥有博客的另一个原因。）

我决定创建新产品来解决这一问题。我创建了一个名为“软件开发人员如何自我营销”的课程（<http://devcareboost.com/m>）。由于这个产品解决了我的目标客户一个非常具体的问题，所以我投入时间进行创建之前我就知道它一定会成功。（我还有另外一种可以提前验证产品是否成功的方法，稍后会谈到。）





产品需要客户

而许多开发人员是反着来的。他们在尚未有受众的时候就创建产品，然后再四处推销，努力寻找受众。当你以这样的方式做事情时，你要冒很大的风险，因为执果索因往往更加困难。

在我创建“软件开发人员如何自我营销”这个课程时，我的受众已经事先找到我，并告诉我他们的问题。要想让产品轻松卖出去，这是一个非常好的启动方法；构建产品再找受众，则与之相反。我们会在第二篇讨论如何自我营销。如果你利用其中提及的方法让自己成名，被粉丝围绕，并产出被追捧的内容，你会发现你已经拥有了愿意购买你创建的任何产品的客户。

许多名流使用这种方法创建和销售产品。他们已经拥有了一大群受众，并且对受众的需求和问题了然于胸。当他们面向受众发布新产品的时候，自然就会成功。以格林·贝克^[1]为例。抛开政治观点不论，正是由于受众庞大，这家伙才能成为《纽约时报》最畅销书作家。他没有必要去寻找受众，因为他创造了受众。他做出来的每一样东西自然而然地就有人等着去买。

如果你想让自己的产品也同样成功（虽然也许在规模上达不到），首先打造一个成功的博客，使用播客、演讲、视频和其他媒体来发展自己的受众。接下来，一旦你有了受众，你就能够向这些受众销售自己的产品。你之所以购买这本书可能就是因为你已经是我博客的粉丝，或者因为关注我的其他工作而无意中发现这本书，或者你之前听过我的播客。这就是发展自己的受众的威力。

测试市场

一旦你明确了产品的受众，并明确了如何用它解决用户的问题，在开发产品之前你还有一步工作需要完成。你应该通过测试市场来验证你的产品，看看你的潜在客户是否真的愿意为它买单。

记得在上一节我讲过，在创建“软件开发人员如何自我营销”这个产品之前，我还采用了另一种方法验证自己的成功。这是一个小秘密：甚至在我正式开始这项工作之前，已经有人为其付费了。

你可能会问我是怎么做到的。简单说就是：我要求他们这么做的。在我计划创建这个产品的时候，我决定在投入数月时间做这件事情之前就广而告之——我将要创建一个什么样的产品，并且为产品还没创建之前就付费的用户提供大幅折扣。这看似有点儿疯狂（在某种程度上说还真是这样），不过要想在投入全部时间开发产品之前，证明是否会有人真的愿意为它付费，这是个好办法。我还知道，如果我能让软件开发人员（我的目标客户）提前3个月（或更早）付费，那等产品正式上线时，销路就不成问题。

所以你可以这么做：创建一个简单的销售网页，讲述你将要创建的产品以及它将解决什么问题，讲述你的产品会包含哪些内容以及产品上市的确切时间，给个折扣价让感兴趣的人可以提前预订产品，保证产品一发布他们就能拿到。你还要提供退款保证，让潜在客户知道如果你的产品不能如期交付或者不能令他们满意，他们可以得到退款。

但是如果预订数量有限怎么办？碰到这种情况，你可以决定是否要修改产品或者作品，因为你不是在解决正确的问题，或者你可以直接退款给那几位预订者，向他们道歉，告诉他们感兴趣的人不多。这不是一件有趣的事儿，但总比花了3个月或更长的时间做好产品之后才发现没有人愿意买要强得多。

在我把预售网页上线的第一天，我的生意就开张了，卖出了7份程序副本。这给了我足够的信心，我知道自己可以继续前行，也知道自已并不是在浪费时间。我也有一群非常有兴趣的客户，我经常向他们征求反馈意见以帮助自己提升正在开发的产品。

从小处着手

我反复强调不要贸然辞职，踏上创业之路，但是我要再啰嗦一次：从小处着手。太多崭露头角的创业者为自己的第一个产品设置了非常激进的目标，不顾一切地追求新的梦想。

你必须明白并意识到，自己的首次创业可能会失败，很可能第二次、第三次也是如此。直到经过足够多的失败，你才可能真正成功。如果你倾尽所有投入创业，如果你为一次创业成功赌上自己的整个未来，你可能会把自己置于绝境——没有资源，甚至没有意志再去尝试一次。所以别这么做。从小处着手，作为副业开发你的第一个产品。

如果你想尽可能缩短自己的学习曲线，你就需要尽量缩短开始行动到看到成果之间的周期。大型产品的问题就在于，如果你没有走得很远，没有在开发产品上投入相当的精力，你就不可能看到实际的成果。

开始

也许本章的所有内容听上去都不错，但是你还是不知道如何开始。别担心，我在开发自己的第一个产品的时候，我也有相同困境。当时的我对自己应该如何创建我能创建的产品，以及如何销售我的产品毫无头绪。

我不说谎，我告诉你这其实很简单。你有相当多的东西要去学习，但是非常容易上手。如今，在线销售比以往更容易，并且还有大把的资源帮得上你。

我是通过阅读该主题的几本书开始的。你可以看看Ramit Sethi的博客（<http://www.iwillteachyoutoberich.com/>），他是这方面的专家，帮助过许多怀揣着“成功企业家”梦想的创业者走向成功。

我还建议你去读读Eric Ries的《精益创业》（*The Lean Startup*）[Crown Business, 2011]，从中获取一些关于如何创建小企业，以及如何起步的想法。

不过大量的创业技能最终还要通过尝试和失败获得。某种程度上，你必须去做自己认为正确的，找出为什么它不管用，然后再去尝试不同的东西。大多数创造出成功产品的创业者也是这样做的。

采取行动

- 整理出一批能够为自己将要创建的产品进行调研的目标受众。
- 从这些受众中挑选出一位用户，看看他参加了哪些集会（无论是线上的还是线下的）。加入他们的社区，倾听他们的问题。看看你能不能从中挑选出一两个潜在领域去开发产品，解决他们的痛点。
- 排查是否有人已经解决了这个问题，你并不想进入一个竞争已经白热化的市场。

[1] 格林·贝克（Glenn Lee Beck）生于1964年10月，美国当红的电视台与电台主持人、政治评论家、水星广播艺术的创始人和首席执行官。他的广播节目在美国各地超过260个频道播放，以“格林·贝克”为名的电视节目也曾曾在CNN、福克斯等电视频道开播，他还有6本书在《纽约时报》上被评为“最畅销书籍”。不过，即使在美国，对贝克的评论也是毁誉参半：他的支持者称赞他坚定维护美国的传统价值观，而他的批评者则诟病他一贯的阴谋论观点和煽动性言论。以上内容摘自维基百科。——译者注

第14章 你打算开始创业吗

对许多软件开发人员而言，最引人入胜的梦想之一莫过于自己创业。创业拥有巨大的潜在回报，但也极其危险。我认识不少软件开发人员，在创业项目上投入很多年，除了失败之外一无所获，境遇比他们刚开始创业时还要糟糕。

但是，如果你已经有了一个好点子，更为重要的是，有了让自己坚持到底的激情和干劲，你会发现冒着风险白手起家还是值得的。

在本章中，我们将深入探索创业到底意味着什么，如何去创业，以及作为创始人（初创企业的创建者的专属称号）必定要面对的潜在风险和回报。

创业的基础

一家创业公司就是一家新成立的公司，试图找到成功的商业模式并将其规模化，最终成为具有赢利能力的中型或大型公司。如果你现在创立一家新公司，本质上就是一家创业型公司。

现在，尽管从技术上讲，任何一家新成立的公司都被认为是一家创业公司，但是实际上还是存在两类不同的创业型公司。第一种创业公司在成立伊始就试图以获得外部投资者的投资来刺激公司快速成长。这种创业公司可能是我们最常听说的那类。许多成功的大型科技公司在开始创业的时候拿到投资人的资金，获得成长，并最终成功。与创业有关的术语和讨论实际上指的就是这类公司。

另一类创业就是自力更生创业。自力更生创业完全由其创始人提供资金支持。如果你创立了自力更生型公司，你不需要从投资人那里募集资金，也不在乎公司能否立刻做大。这类公司通常规模比引入投资人的创业公司小，但是失败的可能性也更低，因为他们通常开销更小，同时由于没有放弃公司的多数股权，创始人在业务上也更有控制权。

鉴于本书已经有相当篇幅讨论了如何开创自己的自力更生型企业，所以这里我们主要谈的是试图获取外部投资来得到成长的创业公司。从现在开始，我说到“创业”的时候，指的都是旨在吸引外部投资的创业公司。

不做大，毋宁回家

大多数创业者的目标就是把公司做大，吸引外部投资的全部原因也就是希冀公司规模能够快速扩张。多数创业公司的创始人都有所谓的退出策略。典型的退出策略就是说，当公司成长到一定规模的时候希望公司被人收购，这样，创始人和投资者都可以获得丰厚的回报，公司未来发展所要面临的风险可以在很大程度上得到规避。

在公司初创的时候就要缜密思考公司的未来，这是极其重要的。你可能想要创建一个能够长期持有的公司，但是你必须认识到，愿意向你的公司注资的投资人都希望能够最终套现获利，得到属于他们的投资回报。

还有，被收购并非获得高额回报的唯一途径。另一个常见的退出策略是上市。当公司上市时，向公众出售公司的股权。出售股票也可以让创始人和投资者获得巨额回报。

无论你的整体退出策略是什么，了解这一点非常重要——获得外部投资的创业公司通常都抱有希望未来有一天可以获得巨额回报的目的。所以，如果你为人保守，你并不适合创立这类公司。创业型公司通常就是为了等待“全垒打 [1] **”的机会。

你可以想象，抱有这种心态可能会有巨大的回报潜力，但随之而来的也会有巨大的风险。大多数创业公司都失败了。有一些估计表明，在获得了外部投资的创业公司中，有高达75%的公司以失败告终（<http://simpleprogrammer.com/ss-startupsfail>）。不知你怎么看，我觉得非常可怕。在创业之前，你需要谨慎思考。有可能在投入多年的辛苦生活和艰辛劳作之后，最后无奈关门，除了那些血淋淋的经验教训，你两手空空。

创业的典型周期

创业已经成为亚文化现象，关于如何创业的书也是汗牛充栋，我在这短短的一章中只能是蜻蜓点水。不过在本节中，我会尽力循序渐进地为你呈现关于典型的创业过程的最佳概述。

通常，当你着手创建一家创业公司的时候，你对自己要创建怎样的公司已经有了想法。通常，这家公司要有些独一无二的知识产权，使得它的那些规模更大的竞争对手难以进入并简单地复制你正在做的。很好的创业候选是能够申请专利或受保护的新技术和新方法，而糟糕的创业候选则包括餐厅或其他缺乏独创、很容易被复制的服务。好的创业项目要有规模扩张的潜力——想想Twitter、Dropbox和Facebook等。

一旦你有了一个想法，你必须决定是做一个单独的创始人还是希望作为联合创始人。尽管二者各有优缺点，但总的来说，大多数创业企业都至少两个联合创始人。如果你想加入创业加速器或者孵化器（接下来我们会讨论），你可能需要至少一位联合创始人。

创业加速器

要想在创业启动时获得额外的帮助，申请加入创业加速器计划是个好方法。加速器是帮助创业型企业启动项目的计划，通过向创业公司注入小额资金来换取公司的一些股份。你可以在<http://www.fbs.com/accelerators>上找到加速器的大清单。最受欢迎的创业加速器计划是Y Combinator（<http://www.ycombinator.com>）。Y Combinator帮助过许多著名的创业型企业，如Dropbox。

通常，申请加入创业孵化器计划要经历一个漫长的过程，但它非常值得一试。加速器项目是一个强度非常高的项目，通常只持续几个月，旨在帮助创业型企业度过启动阶段的难关。大多数加速器由成功的企业家创设的，他们已经创办了一到两个自己的创业公司，可以为刚起步的创业者提供良好的建议和指导。加速器通常也帮助创业公司准备为投资者做路演以获得资金，他们还经常为自己旗下的创业公司安排演示日。在演示日，创业者有机会向潜在的投资人演示自己的项目。

就我个人而言，如果没有被加速器项目接受，我至今也不可能开始创业。竞争太激烈了，而且与完全依靠自己单打独斗相比，加入孵化器计划的优势非常巨大。我曾经是一家创业公司的联合创始人，公司也加入了几个加速器计划，但经过一番深思熟虑，我决定退出公司，因为我发现我还没到创业的时候，还不想经受残酷的创业生活的洗礼。

获得投资

无论你是否加入创业加速器计划，对创业公司而言，第一个重要里程碑就是获得第一笔资金，可以说这是创业生死攸关的大事。第一笔资金通常被称为种子资金，一般天使投资人会投给早期创业者。天使投资人通常是个人投资者，他们在公司的早期阶段注资。这种投资风险很高，但是也可以带来高额回报。现在，天使投资人不会不图回报地给你的公司投资，他们通常会要求拥有公司的部分股权。

地雷：我该如何处理股权

涉及放弃新公司的股权时，请务必要小心。股权是你创业的命脉。没有股权，你的辛勤工作就无法获得回报，你也无法回报投资者。对于要出让多少股权、让给谁这样的问题，你要非常谨慎。

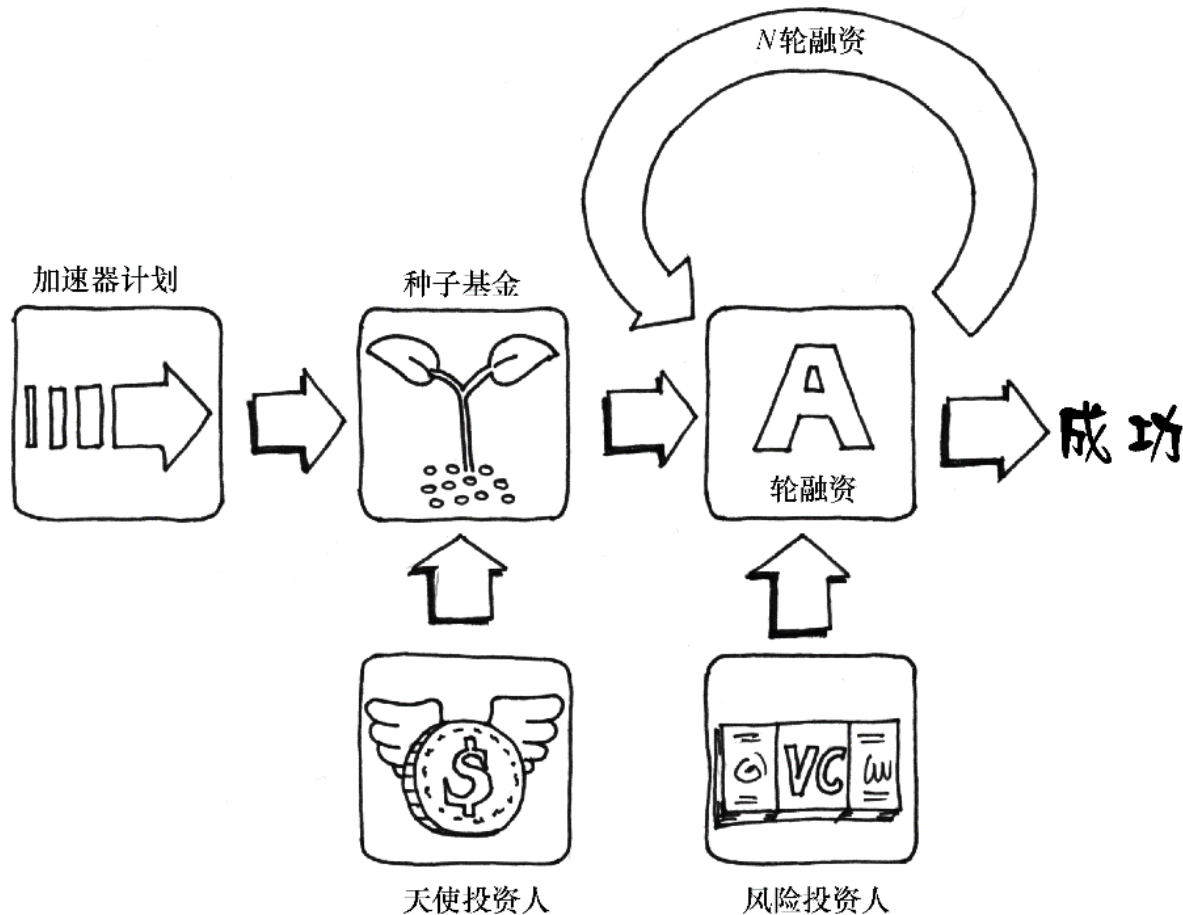
许多创业公司的创始人发现，把股权出让后给不思进取的合伙人后，自己就面临一个大烂摊子，后者不仅对公司毫无贡献，还榨取公司的宝贵资产。

所以在做与股权相关的决策时，请务必谨慎。你要明白，当你出让公司股权的时候，自己放弃了什么。出让股权不可避免，你多少总会出让一些，但是你必须确保在出让之前已经深思熟虑。

一旦创业公司获得了种子基金，便是启动之时。实际上，你应该在此之前就已经启动了。不过一旦获得了种子基金，你就可以雇用一些员工，开始扩张。可以预见的是，大多数创业企业在这个阶段并不能实现盈利。事实上，当你烧光了最初的种子资金，建立你的商业模式并证明这种商业模式时，你可能会深陷泥潭。

一旦你烧光了种子基金，如果你的项目仍然可行，那将会引来一次重大投资。种子轮^[2]之后的第一轮融资通常被称为A轮融资，风险资本通常会在这一轮介入。如果你听说过“见风投”，它通常是指把自己的公司推销给风险投资人（VC，以下简称“风投”），希望从他们那里得到一大笔投资，使公司得到成长。风投通常会向创业公司注入大笔资金以换取公司的大量股份。不必惊讶，A轮融资过后，风投持有的公司股份可能比你还多，特别是当你公司的联合创始人不止一位的时候。

A轮融资完成后，大多数创业公司还需要经过几轮融资，因为它们的初始资金几近耗尽，艰难地实现盈利和扩张。基本上你就是在不断重复这个过程以获得更多投资，直到再也得不到更多的投资，最后成功实现盈利，或者被收购。



获得资金的几个阶段

当然，这只是一个简化了的过程，不过还是希望本章能让你对创建创业型公司有一个良好的认识。

采取行动

- 研究一下一两家你心仪的创业公司的历史。注意它们是如何开始以及如何获得融资的。
- 这些公司创立时是一个创始人还是多个创始人？
- 这些公司的创始人还成功创立过其他公司吗？
- 这些公司什么时候获得融资的？它们获得了多少资金？
- 这些公司启动的时候是否加入了加速器计划？

[1] 全垒打：棒球运动中，全垒打是一种打者可环绕所有垒包一周的安打。除了打者跑到终点本垒时可以得到一之外，所有已经在垒包上的跑者每人皆可得到一分。棒球进攻队伍的球员必须依序经过四个垒包才能得分，在其他情况下，打者上垒成为跑者之后，必须藉由其他队友的帮助推进才能得分。但是若击出全垒打，计分板上分数立刻就增加了，因此全垒打一向是棒球运动中最为人津津乐道的一环，当然也比较罕见。以上摘自《维基百科》。——译者注

[2] 种子轮也被称为天使轮。——译者注

第15章 远程工作的生存策略

今天，越来越多的软件开发团队允许它们的开发人员在自己家里远程办公。有些团队甚至完全是虚拟团队，连真正的办公室都没有。如果你决定成为一名独立咨询师或者创业者，可能你会发现自己也是独自在家办公的状态。

虽然远程工作看起来可能像是一个梦想成真的故事，但是穿着睡衣工作的现实或许并不像想象中那么令人向往。在家办公的人必须要面对许多困难和挑战。在本章中，你会对在家办公是什么样子，以及如何处理在家办公要面对的问题，如孤立无援、形单影只和自我激励等，有一个更为清晰的了解。

做“隐士”面临的挑战

得到第一份在家办公的工作时，我激动异常。我想象不出有比这更好的事情——一大早从床上爬起来，穿过客厅，坐在舒适的椅子上工作。尽管我仍然认为在家办公很棒，但是很快我就发现在家办公有许多自己没有想到的挑战。

挑战1：时间管理

首先，我们面临的最明显的挑战就是时间管理。在家办公的时候，你会面临各种各样在办公室里不会遇到的干扰：如果你打开自己的Facebook页面，在上面耗一整天，也不会有人注意到；快递员上门送来包裹，你会想“嗯，也许我应该吃点东西”；你的孩子或配偶找你问一个问题，或者从你这里窃取“就一分钟”。在意识到这些之前，你的一整天就这样过去，毫无产出。

许多刚刚开始在家办公的人都觉得他们可以在零散的时间工作并且在他们可以工作的时候把事情做完，从而解决这一问题。他们觉得自己可以在白天享受美好生活，把工作放在晚上完成。这种想法无疑就是灾难，因为每当夜幕降临时，总有新的东西令你分心，或者你已经精疲力竭，不想再坐到电脑前了。

真正解决这个问题需要缜密的时间管理。你可以在你想工作的任何时间再去工作，但是你必须设定每周的日程表并且坚持下去。日程表越有规律、越是规划得当越好。我的妻子和朋友们经常跟我开玩笑，问我都已经在家为自己工作了，为什么还要坚持“朝九晚五”的典型工作模式，不过这份日程表确实能够保证我专注、严肃地对待自己的工作。我们不能过分相信自己可以不受外界干扰或者能够智慧地管理时间。我们需要提前做好计划，否则会屡屡经受不住诱惑。相信我，我了解这一点，我自己就经受过一长串的失败。

挑战2：自我激励

这里我只想说说自我激励的方式。如果你总是无法做到自律和自控，你也许应该重新考虑是否要在家办公。在做好时间管理之后，自我激励也许就是在家办公的人面对的最大“杀手”。这个与时间管理密切相关，但是即使你能有效地管理自己的时间，迟早也会有不想做任何工作的倦怠感。

当你在办公室工作的时候，一旦陷入这种情绪，一想到被“炒鱿鱼”的威胁，你马上就不治自愈。如果被老板发现你在工作时间趴在桌子上睡觉或者在手机上玩游戏，你可能立刻就得抱着纸箱走人。但你在家里工作的时候可没有一双窥探的眼睛时刻监督着你。你肩负着自我激励和自律的责任，当你的所有激励都不管用的时候，“自律”会让你继续工作。（要更好地了解激励，可以查看Daniel Pink的《驱动力》（*Drive*）[Riverhead Hardcover, 2009]一书。）

就像我之前说的那样，如果你缺乏自律，我真的认为你将注定失败。我可以教你所有自我激励的技巧，但是打开电视、玩游戏、浏览一整天Facebook的诱惑实在太太大。此外，如果你有一些自律精神，那么继续阅读。如果你愿意付出努力，还有可能处理好自我激励的问题的。

当你感到倦怠时，日程表和常规是非常重要的倚靠。前面已经讲过，我就不再赘述，不过一定要确保你已经建立了日程表和常规。当你不想工作的时候，有一个必须工作的固定时段可以帮你保持足够的激情去完成工作。常规也是一样的。如果可以，你培养一些常规。当激情不再的时候，习惯可以帮到你。有许多次我感觉太累了，晚上都不想去刷牙，可是习惯迫使我一定要刷了牙再睡觉。

你还要尽可能排除干扰和诱惑，让它们远离你的工作环境。如果电视机就摆在你身旁，当你觉得无聊的时候，“打开看看吧”的诱惑会格外强大。别想着靠意志力去战胜诱惑，这个教训适用于生活的许多方面。相反，排除诱惑，你会生活得更轻松自如。（更多内容我们会在第六篇中谈到。）

当你感到实在没有动力的时候，有一种我常用的简单方法（事实上，我没有告诉任何人，但我现在一直在用）——坐下来，设定一个15分钟的计时器，开始工作。在这15分钟里，你必须工作。你不能让自己分心，必须专注于手头的任务。15分钟纯粹且专注的工作结束后，你可能会发现继续下去变得容易些了。事实证明，一旦我们专心致志地工作一段时间，我们就会沉浸其中，也有动力继续。我把这称为冲量效应^{[1]**}。

挑战3：孤独感

刚开始在家办公的时候，我觉得就像是一种解脱。没人打扰你，你可以安静地坐在那里做自己的工作。这是真的。我刚开始在家办公的时候，我马上发现以前在办公室工作的时候那些无意义的对话浪费了自己多少时间。当我学会在家专注地工作之后，我发现我可以在更短的时间里完成很多工作。

但是，过了一段时间之后，那份寂静与安宁令人焦躁不安。你可能会发现自己一直紧盯着窗外，寻找任何生命的迹象。

“哦，看，一个人正在遛狗，也许我应该跑出去和她谈谈。”（别忘了先穿上裤子哦……我可不会说这是我的经验之谈。）好吧，也许我这么说是有点儿悲剧色彩，但是当你日复一日、周复一周孤独地坐在自己的办公桌前的时候，这一定会对你产生负面影响。

大多数在家办公的软件开发人员并没有意识到，长期缺乏社交会使自己变得孤独。毕竟不管怎么说，作为一个群体，我们过的是一种隐居式的生活。请相信我，如果不能在一年左右的时间里找到社交生活，你一定会感到自己要疯掉了。

想想吧，监狱里惩处罪犯的最严厉的手段——关禁闭。对任何人来说，单独隔离一两天真的是一项残酷的刑罚，因为人毕竟是社会型生物。

那么怎样才能治愈这种孤独感呢？这里我有一个简单的答案：走出去！确保你每周都能走出家门，让自己有机会去认识除配偶和孩子以外的其他人。试着加入本地的软件开发小组，每周或者每月参加他们的聚会。换个场景，到咖啡馆去工作。我每周去健身房三次，我当然还会推荐一些健身活动。我还发现，参加会议和其他社交活动让我有机会面对有兴趣的参加者侃侃而谈极客话题。这种倾诉的欲望通常都积累了好几个月呢。

你还可以利用其他一些资源来帮助你不至于脱离社会太久。Skype电话和Google Hangouts能够让你与自己的同事轻松交谈，甚至见面。

如果你能克服以上三项挑战，你就会成为一名成功的远程工作者，但是如果不能克服，你得再考虑一下自己是否适合在家办公。一些不能有效处理上述问题的远程工作者选择利用所谓的“联合办公空间”来找出解决方案。你可以把这些空间看作是小型办公室，里面坐的都是远程工作的人和创业者。这有点像在普通的办公环境下工作，只是你的同事并不是真的和你在一起工作。

地雷：我想做远程工作者，但是我找不到远程工作的岗位

很长时间内，我一直想找一份能让自己在家里办公的工作，但就是找不到。它们不会那么轻而易举得到，通常要经过激烈的竞争。如果你正在寻找一份远程工作但一无所获，我建议你做以下两件事。

（1）你可以看看你当前的工作是否可以变成远程工作。也许值得一试。你可以问问，每周是否可以在家办公一到两天。你要找个好理由，比如你可以完成更多工作，也能更专注。如果你有此机会，一定要真正展现出在家办公的效率更高。

（2）你可以开始关注允许远程工作的公司，或者完全分布式工作的团队，开始与这些公司建立联系。这需要花一些时间，但是如果你专注于那些你知道的允许远程工作的公司，就可以增加在这些企业获得工作的机会。认识在那里工作的开发人员，与招聘经理聊聊，表达出你对他们公司的兴趣，当有岗位空缺时，马上申请。

采取行动

- 对自己做一个客观的评估，针对上面描述的内容，想想你自己该怎样应对时间管理、自我激励和孤独感这三项挑战？
- 如果你已经在家办公或者计划在家办公，制订一个你每周都会坚持的时间表。你要决定自己的工作时长以及哪天是工作日。

[1] 此处原文为“momentum”，译为“冲量”是因为高中物理课本告诉我们，冲量等于作用力乘以时间。这里用“冲量”比喻专心工作一刻钟就会产生继续工作下去的动力。——译者注

第16章 假装自己能成功 [1]

作为一名软件开发人员，你在自己的职业生涯中必定会遇到自己无法处理的情况。时不时冒出来的挑战和障碍经常令我们猝不及防。在这种情况下，如何应对将是决定你能否成功的主要因素。

很多人在面对逆境的时候都会选择知难而退。但是另外一些人在面对同样的挑战时，会勇敢地站出来直面挑战。所有选择留下来与困难做斗争的人，是真的对自己成功和克服困难的能力信心十足吗？肯定不是，但他们中的许多人都有一个共同点——假装自己能成功。

“假装自己能成功”的真实含义

“假装自己能成功”是很常见的说法，但它又是一个被用滥了的说法。不同的人对其确切含义有不同的理解。因此，很有必要澄清一下我在本章中使用这一说法的真实含义。

当我说“假装自己能成功”的时候，并不是暗示你去撒谎或者假装拥有自己尚不掌握的知识和技能。相反，我的意思是，在你做事之前，可以暗示自己“我已经成功完成任务”。当我说“假装自己能成功”的时候，我是在说你要如心理暗示那样行事：

- 你仿佛已经具备了成功完成任务的技能和天分；
- 你仿佛已经成为自己想成为的那类人；
- 仿佛战斗已经结束，而你大获全胜，因为你深知如果自己坚持不懈，胜利就在眼前；
- 你仿佛已经对即将踏上的未知道路驾轻就熟。

当你以这样的心态做事的时候——注意，关键词是“做”——你终究会实现你为自己展示的蓝图。看起来像是魔术或者怪力乱神，但是事实上，我们的精神非常强大。在本书第七篇中我们还会深入讨论如何使用精神力量来控制和塑造你的现实世界。这里，你只需要知道，如果你真的能像心理暗示的那样行事，那你也可以让自己的内心相信那一点。你的内心将会凝聚起巨大的力量，促使美梦成真。

“假装自己能成功”就是这样起作用的。你说服自己的身体和内心去努力，使梦想成为现实。“假装自己能成功”是不自信的对立面。你要在做任何事情的时候都充满自信，即使是在自己的能力远远不到的时候，因为你有一种自己能够克服一切障碍的信念。

付诸实践

所谓“假装自己能成功”，就是要有意识地让自己身处逆境，迫使自己学会拼搏。正是这种在生活中“假装自己能成功”的心态，带领自己勇敢走向未知领域，坚信挑战与机会并存。如果你想学会“假装自己能成功”，你就必须拥有破釜沉舟的勇气。

你是否曾面对这样的困境——感觉对自己要做的事情一无所知？你感觉非常不自在，可能有些窘迫，甚至觉得有些无能为力？这种情况下你该如何应对？你会为自己的失败和表现不佳找借口吗？

面对这样的窘境和障碍，感到恐惧、尴尬，进而找借口，都是很自然的反应。但是，如果你能克服这些倾向，并看到真相，那你现在所面临的新情况和挑战都会变得“不过如此尔尔”，你也就学会了“假装自己能成功”。

你还记得第一次写代码或者第一次学某种编程语言吗？那时候你觉得它很难，不是吗？或许到现在它还是有点儿难。但是，不管你现在身居何处，回顾过去，曾经的艰难与那些看起来的“不可能”，你现在也都从容应对。现在我们要做的，不过就是把那些将来看起来很简单的任务和工作提前一些，假设现在就是将来。

对软件开发人员来说，“假装自己能成功”通常最适用的场合就是在面试的时候。通晓某一岗位所需的所有技术的专家几乎不存在。不同的技术太多，你很难全部掌握，因此当你去面试一个岗位时，这个岗位所需的某些技能是你尚未掌握的。

这里的关键词是“尚未”。许多开发人员参加面试时表现得紧张、很胆怯，表明他们缺乏自信，因为他们一旦工作中遇到自己没掌握或没见过的技术，他们就会怀疑自己完成工作的能力。他们相当短视，因为他们只会以现在的视角看问题。当下转瞬即逝，还没仔细看上几眼就成了过眼烟云。我们更应该着眼于未来。

的确，在眼下你正在面试的这个时间里，你可能并未掌握出色完成工作所需的全部技术。但是你并非打短工的，公司也不是要短期雇用你。其他开发人员面试工作的时候也会欠缺某些技能或经验，只是领域不同而已。正因为如此，你要散发出自信和能干的气场，知道自己曾经面对挑战迎难而上，并且更有理由相信自己将来也会那么做。

不要将这与撒谎混为一谈。我不是建议你向未来的雇主展示自己不具备的技能，声称自己具备尚不具备的能力。相反，我建议你坦率、诚实地面对自己的能力或不足，同时保持“我以前已经克服过那些困难”的心态和姿态，因为你清楚地知道，唯一横亘在“现在的你”和“未来的你”之间的是时间。

你的自信（注意，并非自大）是富有感染力的。如果你抱有“我可以”的心态，当你真正信任自己，既不夸大其词也不妄自菲薄的时候，知道自己最终能实现心中所想，那么你会发现别人也会对此深信不疑。抱着这种心态走向面试官吧，你会见识到“假装自己能成功”的威力。

- 诚实地评估一下自己。面对困境，你的心态如何？当遭遇新问题或不熟悉的问题时，你会怎样做？回想一下，上一次面对困难或者不熟悉的形势时，你是怎么做的？
 - 你该怎样树立一种自信而不自负的心态？自信与自负之间的区别是什么？为了强化“假装自己能成功”的能力，你现在应该做什么？
 - 小练习：有目的地将自己置于困境，演练一下自己既定的应对策略。
-

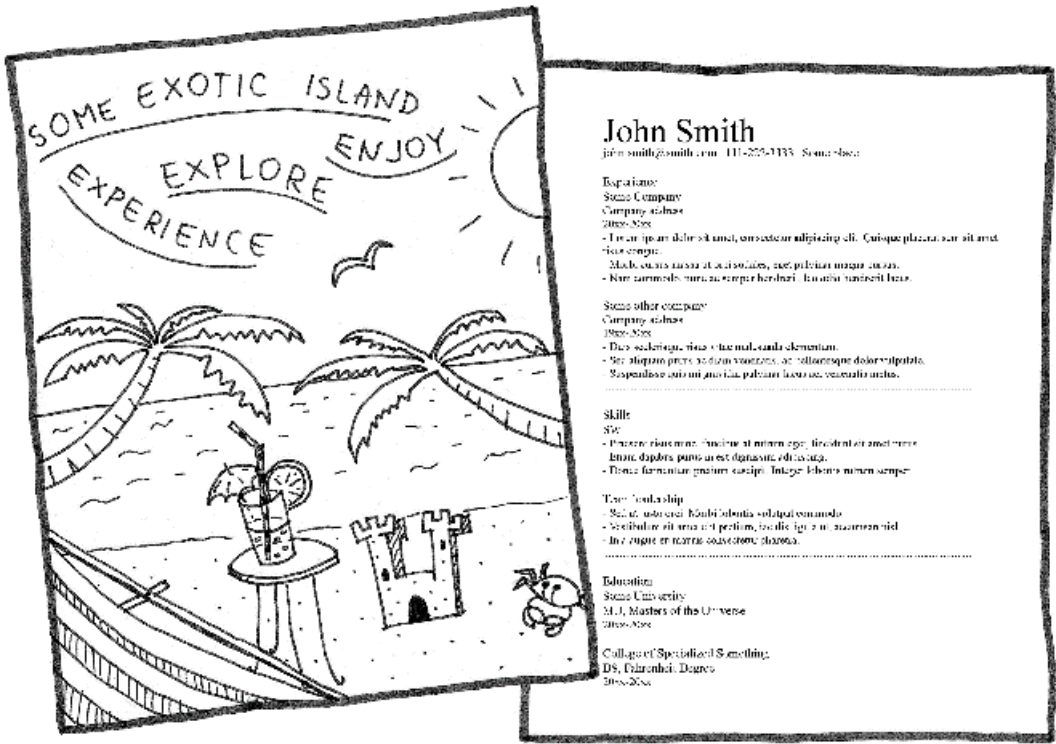
[1] 此处原文为“Fake it till you make it”，直译为“在真正成功之前，先装作成功”，这是一种成功心理暗示。——译者注

第17章 单调乏味的简历——如何修改

你外出度假的时候，可曾看到过酒店服务台上塞满的五颜六色的介绍本地景点的小册子？你有拿起过其中的一本并翻阅吗？大多数小册子只有三页，全彩印刷，设计精美。毫不夸张地说，这些精心设计的小册子足以让你心甘情愿花上100美元去玩玩帆船，或者租上一辆水上摩托。

现在，对照一下普通软件开发人员的简历——一种字体，双倍行距，多达五页，语法错误、拼写错误和没有章法的句子随处可见，充斥着“勇于担当”和“结果导向”这些词汇。

没错，简历和旅游小册子一样，都是广告，最终目的无非就是让人们心甘情愿掏钱——一个是让你花上100美元参加度假活动，另一个是让招聘经理掏出6万美元、8万美元或者更高的价钱“租”一个软件开发人员一年。



典型的简历无法与广告小手册相媲美

有的人为了卖出去100美元的东西，不惜工本对广告内容精雕细作，有的人却想凭借粗制滥造的广告卖出去要价6万多美元的东西，在我看来这实在是不可思议。现在，别误会我，我可没说你的简历是一堆垃圾，我想说的是，如果你的简历跟大多数软件开发人员的简历一样乏善可陈的话，你得花一点儿功夫。

你不是专业的“简历写手”

你的简历如此糟糕，其实另有原因。很简单，你不是专业的简历写手，不以写简历为生。我打包票，那些制作出精美的帆船广告小册子的姑娘小伙子们，是以设计宣传册或其他广告物料为生的。

也有很多职业拓展的书籍和课程教你怎么写出漂亮的简历。这两种我都不推荐。为什么呢？因为你并不需要成为一名专业的简历写手。在你的职业生涯中，写简历只是少数几次，完全没有必要为此投入许多。同时市场上有成千上万的专业简历写手，他们写出来的简历可能远超你的预期。

你可以这么想这个问题。你任职的公司的CEO并不会去写软件。当然，他可能会坐在电脑前打开IDE（集成编程环境）学习如何写代码开发出运营公司所需的软件。但是，更合理的是雇你来做这件事。那么，为什么你不去雇一位专业的简历写手，反而要浪费自己的时间去掌握他们的专业技能呢？

雇一个简历写手

希望看到这儿的时候我已经说服你了，你需要雇一个专业人士来帮你撰写简历。那么，你该怎么做呢？

外面有很多专门代写简历的人，在网上搜一下，比比皆是。问题是，挑选哪一个你必须得谨慎。给一个软件开发人员写简历可比给其他行业的人写简历难多了，因为我们的工作中有太多的专业术语和技术名词了（参见表17-1）。（如果你想找一个好的写手，我个人推荐“技术简历服务”，<http://simpleprogrammer.com/ss-resumewriter>，别忘了报本书的名字。）

表17-1 怎样寻找靠谱的简历代笔者

熟悉技术行业	不能雇一个不知道该如何充分推销你的开发技能的专业简历写手
可以给你展示简历样本	想知道你能得到怎样的工作的最好的办法就是，看看简历写手已经写好的简历上的工作

我得提醒你，优秀的简历写手可不便宜，但是物有所值，毕竟一份赏心悦目的简历可以帮你快速找到一份薪水丰厚的工作。请人代写一份专业的、有品质的简历大概需要花费300~500美元，的确很贵，可是想想看，如果这笔钱只相当于你将来薪水的2%~3%，那你就很容易决定掏钱了。

另外，在请一位简历写手之前，你一定要确保准备好了他所需的所有信息。请记住，进去的是垃圾，出来的也是垃圾。你可不想花巨资请人写出来的简历里谬误百出，就因为你懒得找出上一份工作经历的确切日期，或者因为你没有向写手准确地描述你的技能和职责。当你找到一位简历写手的时候，你主要是要请他做两件事情：

- 为你的经历做一个文字优美引人入胜的广告，令你看起来光彩照人；
- 让这份广告看上去装帧精美，格式赏心悦目。

你请他们来可不是做研究助理，或者核对你的个人信息的，所以你需要给他们提供尽可能多的信息。他们会把这些信息整理、提炼出来，从而可以高效地把你推向市场。

地雷：我觉得雇人写简历并不对

反对我提议的“雇人写简历”的最常见的理由就是这个了。很多人认为雇人写简历这种行为是错误的，带有欺骗性。他们觉得自己的简历就应该自己写才对。我能理解这种观点，我也鼓励你来写自己的简历，但是，雇人写简历跟雇人设计你的网站或者装修你的房子有什么区别呢？事实上，许多名人都聘请枪手为自己写书，然后在作者一栏署上自己的大名。我的观点是，这并不是什么大不了的事（像你认为的那样）。不能因为你一直都觉得开发人员应该自己写简历，就真的应该自己写简历。你不必到处嚷嚷你有专业的简历写手给你写简历。如果你真的觉得不舒服，那么就自己写简历吧，然后雇人“润色润色”。

比别人多做一点点

本章的标题明确指出传统的简历非常单调乏味，事实的确如此。虽然常规的简历对于任何希望找到好工作的软件开发人员而言都非常重要，但是它并非向潜在雇主提供相同信息的唯一途径。

你可以（也应该）把简历中的信息发布到网上。你应该有LinkedIn个人主页，这一主页包含来自简历上的信息，你还应该做一份在线简历，以便能给别人发送简历链接。申请Web开发人员的职位却没有在线简历，就像专业木匠没有自己的专属工具一样。

简历的格式也值得再三推敲。试着给自己的简历添加点独特的创意，以吸引眼球的形式呈现。你可以使用简历代写服务来生成独特的简历，也可以请专业的平面设计师把简历设计得非常流行。

我看过一位视频游戏程序员的简历，他的在线简历其实就是一个可以玩的视频游戏（<http://simpleprogrammer.com/ss-interactiveveresume>）。我敢肯定他不难找到工作。<http://simpleprogrammer.com/ss-beautiful-resumes> 上列出了许多漂亮、充满创意的简历，你可以从中得到一些启发。

对软件开发人员来说，简历不一定要花哨好看，但一份专业的简历非常重要。如果你想对自己十年前写好的充满错误拼写和蹩脚语句的Word文档简历进行一下删减，我建议你三思而行。如果你想找一份新工作，你能做的最好的投资就是写一份专业的简历。

如果不想雇专业的简历写手该怎么做

我能理解你还是宁愿自己写简历。也许你还没有准备做这方面投资，或者你觉得这本来就是自己该做的事情。

如果你选择自己创建简历，表17-2中有一些提示，可能会对你有帮助。

表17-2 改善简历的提示

提示	好处
把简历放到网上	确保雇主能够很容易访问你的简历。如果你申请Web开发的职位，这一点尤其重要
简历要有创意	简历样式不落俗套，别人一眼扫过去就会被深深吸引
使用“行动-结果”的描述	你的简历应该展现你都做了哪些工作及相应的结果。这样你的潜在雇主既能了解你会干什么，还能了解你能取得的成果，以及雇用你会给他们带来的收益
校对	即使是聘请专业的简历写手，你也要通篇校对。简历里有错别字或拼写错误会让人觉得你粗心大意

采取行动

- 不管你现在是否正在找工作，把你的简历给一些招聘人员，询问他们的意见。招聘人员看过大量的简历，他们是能告诉你是否需要改进你自己的简历的最佳人选。
- 查阅分析一些专业的简历代写服务，看看他们提供的简历样本。与之相比，你自己写的简历如何？

第18章 请勿陷入对技术的狂热之中

我不知道你是不是是一位有宗教信仰的人。无论你信仰何种宗教，我肯定你会同意我的看法，历史上许多最血腥、最残酷的战争，在某种程度上都是为宗教而战的。

这里我并非是在抨击宗教，也并非以某种方式表明某种宗教与生俱来的善与恶。我只是希望你能清楚地意识到，恪守教条的信仰往往有很大的煽动性。

软件开发也是如此。软件开发和技术的虔诚很容易像信仰生命起源或至高无上的神的存在一样。尽管一般我们不会因为某个人喜欢iOS超过Android就杀死他，但是我们确实有一种倾向，总想在没人注意的时候给对方点颜色看看。

我坚信，如果你能让自己不成为某种技术的信徒，你会在职业生涯之路上走得更远。在本章中，我们将探讨为什么会是这样的情形。

我们都是技术的信徒

这是真的，你还是承认这一点吧。你对某种技术或编程语言存在偏见，认为它是最好的——至少大多数程序员是这样的。这很正常。我们总是对自己做的事情充满热情；只要有激情、有热情，就很容易变得极度感性。来看看职业运动员。

对技术虔诚的一大问题是，我们中的大多数崇拜某项特定的技术，只是因为自己熟悉这种技术。我们很自然地会相信自己选择的是最好的，然而这会让我们经常忽略任何反对意见。我们不可能充分了解现存的所有技术，从而给“哪项技术最好”作出最英明、最睿智的判断，于是我们倾向于选择我们了解的技术并先入为主地认为它是最好的。人生多艰，无暇他顾。

尽管这一行为的起因合理又自然，但其实具有破坏性和局限性。当我们武断地只根据自身经验就固执己见时，很容易只与理念相同的人打交道而排斥其他人，最终我们只会固步自封、墨守陈规。我们自以为找到所有答案，却只是裹足不前。

在我职业生涯的相当长的一段时间里，我都是操作系统、编程语言甚至文本编辑器这些技术的忠实信徒。然后我才逐渐意识到，我没必要只是选择最好的而贬低其他的。

天生一物必有用

不是所有的技术都是“伟大的”，但多数被普遍应用的技术至少是“好的”。一样东西如果不好，就不会为人所知或使用，也不会成功。诚然，环境是随着时间而变化的，但重要的是，我们必须认识到：至少在历史的某个时间点，每项技术都被看作是那个时代里“好的”甚至是“伟大的”。

有了这样的视角会有助于你理解：在很多情况下，解决问题并不只有一个好的或是最好的方法。同理，最好的编程语言、框架、操作系统，甚至是文本编辑器不会只有一种。你可能会喜欢某项技术多过其他，或者用某种编程语言的效率要高过其他，但是称其最好并无必要。

发生在我身上的转变

让我自己相信这一点经历了一段漫长的艰难时间。我花了无数时间与人争论为什么Windows比Mac好得多。我为C#和其他静态类型语言大声疾呼，我认为它们可以把Perl或者Ruby这些动态语言甩开几条街。现在我讲起来都有些惭愧，我那时甚至会严厉斥责那些不这么认为的开发人员——他们怎么敢相信那些我都从来不用的技术？

令我大开眼界的是在我第一次担当一个Java项目的项目组长的时候。在此之前，我一直是.NET开发人员，专注C#语言。（好吧，其实也不尽然。在.NET技术出现以前我是C++的信徒。）我不能容忍用Java语言工作的想法。同优雅的C#相比，Java就是不入流。我甚至都不会使用Lambda表达式，我怎么可能喜欢用Java写代码呢？

最终我还是决定接受这份工作，因为这个机会实在是太好了，而且我认为这不过就是一个合同，我只要忍上区区一年而已。后来，我才发现，接下那份工作是我在职业生涯中所做出的最杰出的决策之一。使用自己一度厌恶的技术工作，让我从不同的视角审视所有的技术。事实证明，Java一点都不糟糕，我也能够理解为什么有些开发人员喜欢用Java而不用C#。

在做Java项目的那几年里，我学到的东西比之前工作中学到的都多。突然之间，我有了一个巨大的装满各式工具的工具箱，能用来解决任何问题，我再也不像以前那样死守着少数几个被过度使用的工具。

从那时起，面对其他编程语言，即使是动态语言，我都会抱有与面对Java时一样的开放心态，使用从中学到的任何知识和技能，使自己成为更好的程序员。我也放弃了对操作系统和框架的偏爱，努力尝试新东西之后再作评价。如果没有这段经历，我可能不会写这本书，又或者，我会写一本《为什么C#是最好的语言，其他语言都糟透了》。

不拘一格

不拘一格是本章的要点所在。没有理由去强烈坚持自己选择的技术就是最好的，而轻视甚至无视其他技术。如果固执己见，最终受损失的是你自己。

另一方面，如果你愿意对技术保持开放的心态，而不是固守自己已经了解的技术，声称它是最好的，你会发现有更多的机会为你敞开大门。

采取行动

- 列出你钟爱的所有技术，或者你觉得更胜一筹的技术。
- 针对这张列表里的每一项，想想它为什么吸引你。你都做了哪些比较来确定它确实是“更胜一筹”？你真的使用过它们的竞争对手吗？
- 挑选一门你厌恶的技术，找几个喜欢它的人，开诚布公地讨论为什么他们喜欢这门技术。更进一步，你也试着用用它。

第二篇 自我营销

营销就是一场争夺人们注意力的竞赛。

——Seth Godin

在软件开发行业，营销的名声并不好。一般人提起营销人员都会皱眉头，因为很多营销人员会急功近利地使用不诚信的手段。似乎每天都有无良的营销人员为了一己私利而兜售新骗局。

其实在现实生活中，营销本身并没有那么糟糕。你的营销手段决定了你的营销对象是受益还是受损。营销需要人们的关注，以便让人们关注你，关注你的产品。优秀的营销会将人们的需要或者期待与能够满足此愿望的产品或服务关联起来。所以，营销追求的是“实现价值在先，要求回报在后”。

第19章 针对“码农”的营销基础课

你有去过酒吧夜店看过驻场乐队的演出吗？那些乐队的演唱水平丝毫不亚于原唱的艺术家们。你有没有想过，为什么这些乐队就只能在小小的夜店里做驻唱演出，而有些音乐才华并不比他们高多少的流行乐队就可以在全世界巡回演出，创造着一个又一个白金唱片？

很明显，两个乐队都很有才华，但是在生活中，仅仅拥有才华是远远不够的。伟大的音乐家和超级巨星之间的真正区别无非就是营销而已。对于有才华的人来说，营销就是一个“乘数效应”——你的营销越好，你的才华才能表现得越发淋漓尽致。这就是对于软件开发人员来说，学习营销技能至关重要的原因。

自我营销意味着什么

营销的核心在于将一些人所需要的所期待的产品或者服务与产品或服务本身连接起来。所以“自我营销”也就是把希望得到你提供的产品或者服务的人和你自己连接起来。尽管营销经常声誉不佳，但如果你能以正确的方式营销自己并没有错。

自我营销的正确方式就是为他人提供价值。我们会在第21章对此进行深入的讨论。这里，你需要了解，成功进行自我营销的关键在于：如果想让别人喜欢你，想和你一起工作，你就必须要为他们提供价值。想想看，像Scott Hanselman^[1]这样的牛人是怎么做的。Scott通过他自己的博客（<http://www.hanselman.com>）、演讲和播客为开发人员提供了相当多有价值的东西。在我们进入细节讨论之前，我们先从实践的角度谈谈自我营销是什么。作为一名软件开发人员，你怎样自我营销？

无论你是否意识到，其实你每时每刻都在营销自己。当你试图说服他人接受你的想法时，本质上，你就是在把自己的想法推销给他们。正如我们在关于人际关系的第4章中讨论过的，我们知道如何包装一个想法往往比想法本身更重要。

当你申请一份工作时，从本质上讲，你的简历就是推销自己服务的一份广告。甚至于，你在社交媒体或自己的博客上发布的内容（如果你有的话），其实也是在为你自己和你提供的服务和产品在做某种推销。

问题在于，即使我们都在推销自己，我们中大多数人并不是有意识地在做这件事情。我们将机会拱手相让，任由他人和环境来定义我们。

自我营销无非就是学习如何控制好自己要传达的信息，塑造好自己的形象，扩展信息送达的人群。当你营销自己的时候，你就是在积极地管理自己的职业生涯，有目的地选择好如何塑造自己，以及将塑造好的自己主动推送给那些对你感兴趣的人。这些人或者想听到你的想法，或者想雇用你，或者想购买你提供的产品或服务。

想想一部新拍的电影大片是如何策划广告宣传的。通常，各种爆料漫天飞舞在各种广告媒体上。电影预告片描绘了一幅具体且清晰的画面，传达着特定的信息。这一信息又通过各种广告渠道被不断传播放大。

自我营销为何如此重要

在本章的第一个例子里我讲到了，音乐才华相当的两支乐队，一支在酒吧驻场而另一支是流行巨星，他们之间的成就差别为何如此悬殊？我将产生这种差异的原因归咎为营销。成就巨大的摇滚乐队通常在营销上要比酒吧驻场乐队做得更好。

当然，我们无法确定酒吧驻场乐队在“自我营销”方面到底做得怎么样，但是如果我们假设音乐才华都差不得的话，那么排除掉纯粹运气的因素，市场营销就是他们可以控制的因素了。所以，自我营销并不能确保你一定成功，但是它却是你可控的重要元素。

你也可以在其他领域找到相同的模式。以专业厨师为例，许多才华横溢的厨师烹饪出的菜肴非比寻常，而他们中的大多数人都默默无闻。但是也有像Gordon Ramsay^[2]*和Rachel Ray^[3]*这样的明星大厨赚得百万美金，这并不是因为他们比别的大厨更有天赋，而是因为他们深谙营销之道，通过营销来充分发挥自己的优势。

别以为软件开发领域有什么不同。你也许是世界上最有天赋的软件开发人员，但是如果没有人知道你的存在，你也只不过是浮云一片。当然，你总能找到工作，但永远不会发挥出你的全部潜力，除非你能学会“如何自我营销”的技能。

在你职业生涯的某个时刻，你可能发现自己的技术水平已经能够与顶级开发人员并驾齐驱。其实许多软件开发人员经过大约10年的职业生涯都能达到这个水平。一旦达到这一水平，要想再提升会变得异常困难，因为你已经“泯然众人矣”。你的个人才智已经变得无关紧要，因为与你竞争的软件开发人员水平基本相当。

但是有一个办法可以让你脱颖而出。学习“如何自我营销”你就能够一鸣惊人，就像摇滚明星或者知名大厨一样，不仅收入更高，还能拥有更多的机会。

地雷：我真的不是什么专家，我没有什么东西可以营销

即便你不认为自己是专家，也并不妨碍你从现在开始自我营销。事实上，试图找出自我营销的方法，可以让你成为专家，专门从事某一特定领域的软件开发工作。

基本上每一个开发人员都是有些能耐的——可能你观察事物的视角比较独特，或者可能你与其他软件开发人员的背景不尽相同，又或者你的兴趣爱好与客户或者其他软件开发人员相似。只要营销得法，即便是“菜鸟”或者“业余爱好者”的身份都是你的优势所在——很多人都喜欢向只比自己稍微优秀一点点的人学习，因为这些人才是可望而又可及的。

关键是，不要让“不是专家”成为放弃自我营销的借口。无论你身处自己职业生涯的哪个阶段，你都可以从营造和传播自己的品牌中获益匪浅。

如何自我营销

我希望我已经让你相信自我营销至关重要，现在你可能想知道该如何自我营销呢？你怎样才能成为软件开发界的Gordon Ramsay？

我不会骗你说这很容易。成功从来就不是一蹴而就的，那样即使“成功”也无法长久。但是，“自我营销”确实是每一个开发人员都能做得到的——“天下事有难易乎？为之，则难者亦易矣”。在这里我将扼要介绍一下所有的关键概念，在下面的章节中，我们会一个接一个地详细论述。

自我营销要从打造能代表你自己的个人品牌做起。你不可能将所有的东西全部呈现出来，因此，对于自己成为怎样的人、给世界呈现怎样的形象，你要谨慎决定。如果你也想在有人多次接触你或者你的产品时营造出一种亲切感，个人品牌能帮到你。

一旦你拥有了自己的品牌风格，也知道要传达怎样的信息，你就需要找到传达上述信息的方式。尽管有多种媒介可供你使用，但对于软件开发人员，最突出也是我个人推荐的还是博客。我认为博客就是你在互联网上的大本营。这是一个你完全能够控制信息的地方，不像在其他的平台上你还要仰人鼻息。

我当时采用了Pat Flynn (<http://www.smartpassiveincome.com/>) 的策略。Pat是我非常推崇的企业家，他有一个策略叫做“无处不在”。该策略的基本思想就是，无论你身在何处都要营销。无论何时都要让你的目标受众轻轻一扫就有机会看到你。你可能会出现在他们的Twitter时间线上，他们也可能在听你的播客，又或者他们看到了你的在线视频。他们目光所及，总能与你相遇。

表19-1展示了一些可供选择的自我营销的方式。

表19-1 自我营销的方式

方式	描述
博客 播客 视频 给杂志投稿 书籍 代码营 技术会议	你自己的博客，以及你在别人的博客上发表的特邀文章 创建自己的播客频道，同时接受其他播客频道的访谈 在YouTube这样的视频网站上上传自己的分享内容和课程主题视频 给软件开发相关的杂志写文章 著书或者自出版 大多数代码营都允许参加者自由发言 拓展社交网络的好方法，如果能在研讨会上发言尤佳

这个策略需要时间，需要持之以恒。随着时间的推移，你写的每一篇博客，你采访的每一期播客，还有你写下的每一本书、每一篇文章，都有助于营销自己，提升你的个人品牌的认知度。最终你在这个领域就成为了权威，拥有了追随者。这些声望转化为更好的机会，最终成就你的事业。

我在前面提到过，将在后续章节中更深入地讨论这个话题。这里，我想强调的是：所有这一切都取决于你的能力——能给其他人带来怎样的价值。自我营销的基本机制是，要想让人们追随你、倾听你，你就要带给他们价值：你能为他们的问题提供答案，甚至是给他们带去欢乐。如果你在持续自我提升的同时并没有给他人带来价值，那么你终究不会走得太远，因为每个人都会离你而去。

采取行动

- 如果你还没有博客，开通一个吧。想想看，你会专注在哪些主题上？
- 给你的新博客至少想出20篇文章。
- 现在做一个时间表，列出你开通博客的确切时间，为博客创建内容。

[1] Scott Hanselman（斯科特·汉塞尔曼），业界知名的Web技术专家，曾担任在线金融系统提供商Corillian的首席架构师，目前在微软工作。以上这段描述摘自他的个人博客<http://www.hanselman.com/blog/AboutMe.aspx>。另外，斯科特还为本书作序。——译者注

[2] Gordon Ramsay（戈登·拉姆齐），出生于格拉斯哥，堪称英国乃至世界的顶级厨神，因其在各种名人烹饪节目的粗鲁与严格，以及追求完美的风格，而被媒体称为“地狱厨师”。以上摘自必应搜索。——译者注

[3] Rachel Ray（蕾切尔·雷），出生于纽约，美国电视名人、商人、名厨和作者。她主持了3个美食系列电视节目——《30分钟大餐》《蕾切尔·雷美味游记》和《一天40美元》，还推出了推出一本美食杂志《蕾切尔·雷的每一天》。在2006年，雷的电视节目赢得了3个艾美奖。以上摘自维基百科。——译者注

第20章 打造引人注目的品牌

品牌无处不在。无论身在何处，你总能看到百事、麦当劳、星巴克、惠普、微软.....不胜枚举。

品牌不仅仅是形象。大多数人会把品牌与商标联系起来，比如说，麦当劳那个著名的“金色拱门”商标。但品牌不只是商标，更是一项承诺。品牌树立了客户对你的期望，而且这些期望也必须能够实现。

在本章中，我们将讨论构成品牌的要素有哪些。我还将展示怎样创建自己的品牌，从而让你的营销成果引人注目。

什么是品牌

我们来谈一谈当今的流行品牌吧。我们以星巴克为例。星巴克是一个为大众所熟知的知名品牌，第一眼看上去，那个众所周知的商标图案似乎就是“星巴克”的品牌，但事实远非如此。星巴克的商标图案只是其品牌的视觉符号而已，并非品牌本身。

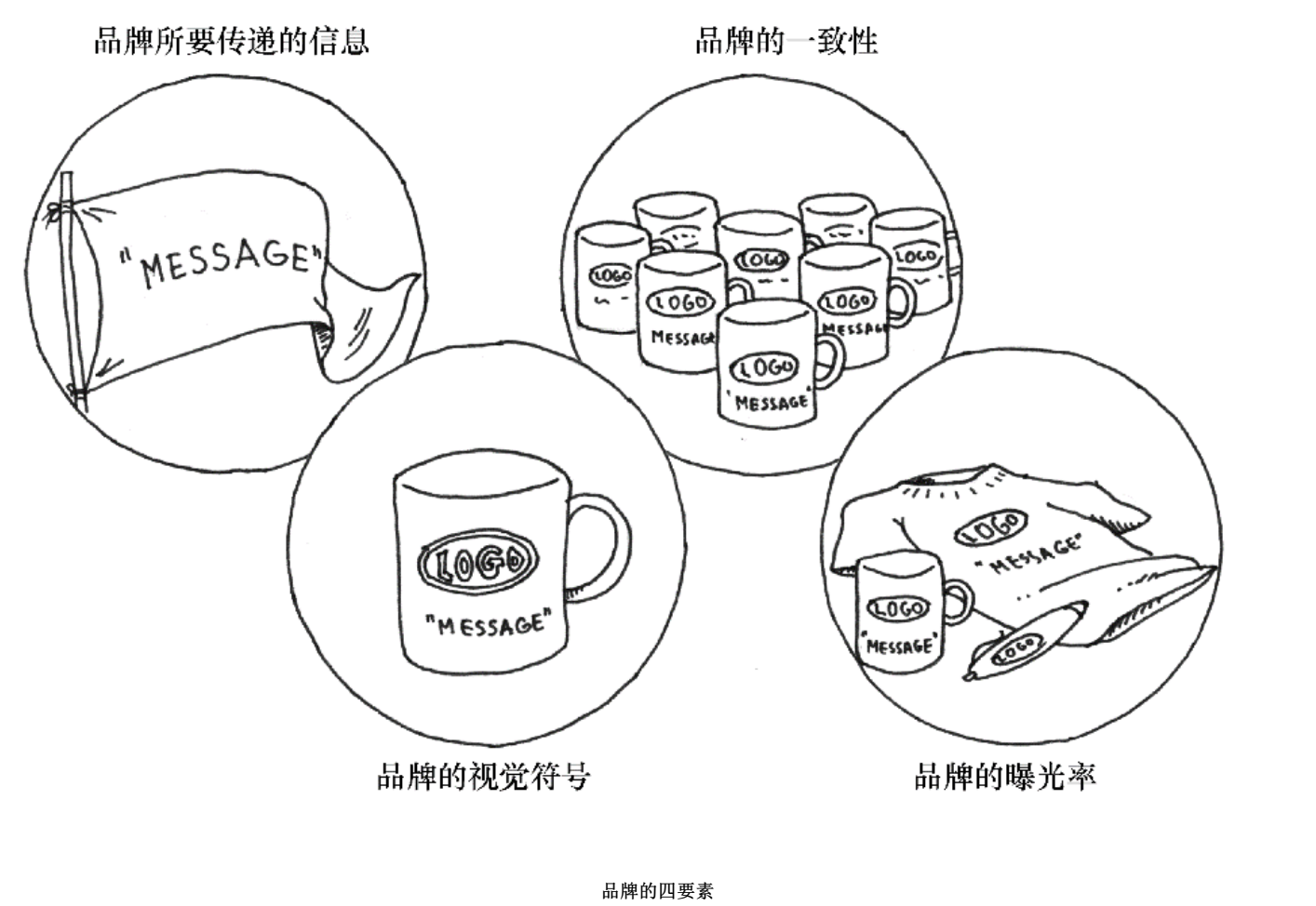
当你走进星巴克的时候，你希望看到什么、听到什么？你期望光线是什么样的？你期望的室内陈设和家具又是什么样的？现在你不妨闭上眼睛想想星巴克的店内布局，你的感觉又如何？

当你走向柜台点一杯饮品的时候，你期望咖啡师的形象是什么样的？你期望咖啡师通过什么方式和你打招呼并问你哪些问题？你熟悉他们的菜单吗？你对他们的定价和饮品的品质有预期吗？

现在，你明白了，品牌并不只是商标。品牌是对产品或服务的一整套预期。商标仅仅是品牌的视觉符号而已。品牌的关键并不在于视觉元素，而是品牌带给你的感受，是你与品牌互动时的预期。品牌即承诺：承诺按照你预期的方式交付你所预期的价值。

构成品牌的要素有哪些

要打造一个品牌，你需要四个要素——品牌所要传递的信息、品牌的视觉符号、品牌的一致性和品牌的曝光率。要构建成功的品牌，这四大要素缺一不可。我们将逐个介绍，以便让你了解如何使用这些概念来打造个人品牌。



首先，而且也是最重要的，就是品牌所传递的信息。缺乏信息的品牌也就没有了目的性。信息就是你想传达的，以及品牌所承载的情感诉求。在你创立个人品牌时，你需要一条能代表品牌的核心信息。你的品牌是干什么的？你是干什么的？例如，我的个人品牌Simple Programmer（简单程序员）是围绕着“化繁为简”这个信息做文章：我把各种复杂的概念分解后，转化为人人可以理解的简单概念。

其次，品牌需要视觉符号。尽管视觉符号不是品牌本身，但它对品牌非常重要。显而易见，一个品牌应该有一个标识，即能够代表品牌的简单的视觉符号。好的品牌会让自己的视觉符号无处不在。一套能够代表品牌的色彩和风格有助于提升品牌辨识度，加速品牌信息的传播。

即使你用自己的名字作为品牌，你也可以为它创建个出众的标识。我的好朋友John Papa就是绝佳的典范。如果浏览他的博客（<http://johnpapa.net>），你会看到他用自己的名字设计了一个标识。

再者，品牌要有一致性。品牌所包含的信息、品牌的视觉形象固然非常重要，但是一旦失去了一致性，就无法构建品牌认知度，或者更糟，你在不断地背离品牌。假设你走进了麦当劳，结果每个地方的菜单和价格都不一样，麦当劳的品牌价值将会消失殆尽。在你走进麦当劳的时候，你对自己的体验是有预期的。如果这种体验一直在变，不能保持一致，那品牌也就变得毫无意义。

许多开发人员在创立个人品牌的时候都犯了缺乏一致性的错误。他们要么不能传递一致的信息，要么在传递信息的方式上缺乏一致性。个人品牌的一致性越强，就越能被更多人关注，也更容易被记住。

最后一个要素——品牌曝光度。假设其他三个要素你做得都很好，但有的人只见过一次你的品牌，对你也没有太多意义。即使你的标识再好看、再吸引眼球，但人们只能看到一次，对你而言又有何用？品牌的核心在于建立一套预期，在人们看到你的标识或听到你的名字的时候，他们就能马上知道你是谁、你代表着什么。

通过写博客、写文章、发表演讲、创建视频分享、做播客内容等一切手段，你需要竭尽所能传播你的信息，扩大你的知名度。你的品牌传播的频率越高、传播的范围越广，目标人群记住它的概率才会越高，你被人们记住的概率才会越大。

创建属于自己的品牌

当你着手创建自己的专属品牌时，第一步就是要定义你想传达的信息。要想明确品牌所要传递的信息，你需要决定自己希望呈现什么。你不可能覆盖所有人群，因此需要缩小自己的受众范围，选择自己的细分市场。这与第7章提及的专业化概念非常相近。基本上，你需要选择一个规模更小的目标市场，或者非常符合你的品牌的独特领域。

在这方面我最佩服的软件开发人员之一当属Marcie Robillard，又名“数据报表女孩”（DataGrid Girl）。Marcie选择ASP.NET DataGrid（数据报表）控件为其细分市场，并以“数据报表女孩”作为自己的品牌来推广。这一细分市场非常窄但又充分聚焦，非常适合她。她做了很多次演讲，也在.NET Rocks这样的流行播客频道上传内容。我敢肯定，她还占据了关于ASP.NET DataGrid控件话题的大量搜索流量。

选择某个细分市场，然后以它为核心建立你的品牌，越有针对性越好。如果能充分聚焦，你就可以直接向受众传达信息，也能更轻松地建立品牌的认知度。

确定细分市场并非易事。你可以选择最令自己有激情的领域，然而激情可能会随着时间消退。通常，从战略角度出发是做出决策的最佳方式。在开拓某个细分市场的时候，你有什么优势？花点儿时间想一想。如果需要更换细分市场，也不要犹豫。

创建品牌的步骤

- 明确要传达的品牌信息。
- 挑选细分市场。
- 创建品牌口号。
- 创建电梯内销售概要^[1]。
- 创建视觉符号（即标识）。

一旦你确定了自己的细分市场，就可以打磨要传达的信息了。你也需要创建一个口号，言简意赅地描述自己的品牌。例如，我的口号就是“化繁为简”。看到我的这句口号的人能迅速理解我是做什么的。

接下来，你应该创建所谓“电梯内销售概要”。电梯内销售概要是一段能够快速描述你是谁、能做什么的宣传文字，乘坐电梯的工夫就可以浏览完毕。想想在晚宴上，或者就在电梯里，当有人问起来“你是做什么的”的时候你该怎样回答人家。

为了打造你的“电梯内销售概要”，你得好好想想自己能提供什么价值。你有哪些独一无二的特质？大家可以预期从你这里获得什么？在你的“电梯内销售概要”里你一定要讲明你是做什么的、你能带来的独特价值是什么。

有了这个经过深思熟虑的“电梯内销售概要”，将会保证你与自己的品牌保持一致，并能够保证在与他人谈及自己的品牌时，你一直传递着相同的信息，或者不论你使用何种媒介推广个人品牌，你都在呈现自己的品牌。

只有在明确了自己品牌的核心内容之后，你才可以创建品牌的视觉符号。你创建的视觉符号应该代表目前为止你为自己的品牌所定义的一切。视觉符号有助于传递品牌信息，并作为你的品牌所代表的视觉提示。

你没有必要花一大笔钱为自己的品牌设计视觉符号，但我也建议你做这件事，除非你具有出色的图形设计能力。我为自己的多个产品和服务创建了品牌，借助我个人非常喜欢的服务Fiverr（<http://simpleprogrammer.com/ss-fiverr>），我以每个5美元的价格搞定了其中几个品牌的标识。能以如此低价获得如此服务，着实令人吃惊。你也可以使用oDesk（<http://simpleprogrammer.com/ss-odesk>）这样的服务，找自由职业者来为你搞定这项工作。我已经成功使用这两种方法以低廉的价格完成了标识设计和其他设计工作。

大致的过程就是这样。如果你在前期花了些时间创建了一个清晰、一致的品牌，并且这个品牌高度聚焦在某个细分领域、承载着经过深思熟虑的信息，你就会在竞争中遥遥领先。但就像我之前说的，只有品牌信息和视觉标识是远远不够的。你需要坚持提升品牌的一致性和曝光率来打造一个有影响力的品牌。在接下来的几章中我们将一起了解如何给品牌增加一致性，以及如何通过博客、社交网络、演讲以及其他各种媒介传播品牌信息。

采取行动

- 列出一些你熟悉的流行品牌，选择1~2个进行深入研究，试着确定一下这些品牌传递的信息是什么，看看它们是如何使用商标和其他视觉元素来传达该信息的。
- 使用头脑风暴的方法为自己的个人品牌列出细分想法。先列出至少10~15个想法，然后再缩小到2~3个，最后选定一个作为你的个人品牌。

[1] “电梯内销售概要”是指用两到三句话概括一个产品、一个提案或一个项目，向潜在的买家或赞助人做介绍。以上摘自百度百科。——译者注

第21章 创建大获成功的博客

作为一名软件开发人员，可以用来推销自己的最佳媒介之一就是博客。我坚信每一个在乎自己职业生涯的软件开发人员都应该投资创建一个博客。

使用面对面的直接交往方式，你只能遇到有限的人，所以你需要另一种方式来推销自己，拓展自己的社交圈子。想想自己在过去的一年里认识了多少技术行业的专才。这一数字可能接近几百甚至上千，然而一个成功的博客能让数十万人认识你。

博客是推销自己的一种既廉价又简单的方式，对于让自己声名鹊起极具价值。成功的博客每天能够吸引数百甚至数千的访客，这可以给你带来很多机会，从工作岗位到咨询项目，甚至是产品销售的目标客户。

坦白地讲，我职业生涯的大部分成功要归功于我的博客。如果我没有创建一个博客并且使它成功，你也不可能读到这本书。

为什么博客如此重要

当你申请一份工作的时候，你的简历通常大概只有两页的篇幅。当你接受面试的时候，你通常会跟面试官聊上一两个小时。以如此简短的简历和如此短暂的面试来评估一名软件开发人员的技能非常困难，所以雇主以此判定某个人是否适合某个工作岗位也颇具难度。

然而试想一下，如果一位软件开发人员拥有一个定期更新的博客会怎么样。他的博客可能包含了丰富的相关信息，包括代码示例，还有对软件开发各个方面的深入分析。与任何其他方式相比，阅读一位软件开发人员的博客能让我对他了解更多。

即便这是创建并维护博客的唯一理由，这理由也足够了。但并非仅仅如此。拥有博客不仅能帮你找到一份不错的工作，还能让你成为更好的软件开发人员和传道者，能带给你许多意想不到的机会。

想想那些技术牛人，像Scott Hanselman、Bob Martin大叔，还有Kent Beck，他们都有博客。

如果你是一名自由职业者，或者你有兴趣做兼职（参见第12章），你会发现，一个成功的博客可以给你带来很多客户，比你不得不外出找到的客户多得多。自己主动上门的客户，更愿意付你更高的薪酬，也更容易给你工作。

如果你博客的流量足够大，你可以把它作为推销自己产品的平台（参见第13章）。如果有稳定的博客访问用户，那你不妨围绕他们的兴趣开发产品，将流量直接转化为客户。

不要忘记一个成功的博客为你带来的业界声望。许多著名的软件开发人员直接靠成功的博客赢得口碑。Jeff Atwood是个绝佳范例。他是Stack Overflow和Stack Exchange的创始人之一。他的博客“Coding Horror”（代码的荣耀，<http://codinghorror.com>）大获成功，收获了大量受众，直接促成了Stack Overflow的成功。他的博客也让他认识了自己的搭档——另一位成功的博主Joel Spolsky（<http://joelonsoftware.com>）。

即使你对上面提及的博客能带给你的所有好处都打了折扣，有一个好处是你无法轻易抹杀的——提高你的沟通技巧。组织自己的思想，并将其转化为文字，是一项颇具难度却也极具价值的技能。定期写作能帮助你打磨此技能，有了很好的沟通能力会让你在生活的诸多领域受益。此外，如果你能约束自己定期更新博客，你也就在持续刷新自己的技能，保证自己处于自己所在专业领域的前沿。

作为软件开发人员，学习如何写博客实际上都能帮你写出更好的代码，因为你能更轻松地传达自己的意图。博客还能帮你更好地传达自己的想法，令想法更有说服力。

创建博客

你已经相信自己需要博客了吧？太好了。那么下一个问题就是如何开始。

当下搞个博客很容易。使用Wordpress（<http://wordpress.com>）或Blogger（<http://blogger.com>）这样的免费服务，5分钟之内你就可以创建一个博客。但是，在签约同意接受这些服务之前，你还需要思考几件事情。

免费服务是搭建博客最简单、最便宜的方法，但未必是最好的方法。免费服务的一个典型问题就是，你无法更灵活地掌控自己博客的主题和布局。你可以做一些定制，但无法给博客添加付费广告、购物车或别的功能。这些功能目前对你而言可能不是很重要，但是随着你的博客越来越有名，你或许希望能给博客增加一些从免费服务无法获得的功能。

幸运的是，有一种很方便的方式来替代这种完全免费的托管平台。你可以找到许多付费服务，它们能够托管基于主流的Wordpress.org软件（<http://simpleprogrammer.com/ss-wordpress>）搭建的博客，每月仅需8~10美元。（顺便说一句，我强烈推荐你使用Wordpress.org搭建自己的博客。它被广泛使用，拥有庞大的生态系统。如果你选择Wordpress.org，你很容易就能够找到各种的插件来扩展你的博客，找到主题定制博客的外观。）这些付费的托管服务以低廉的价格，让你享受更多的灵活性。

你可能一开始会用免费的Wordpress.com服务来搭建博客。（切勿把它与Wordpress.org混淆。后者是你搭建可托管在付费托管服务上的博客的真实软件。）在一段时间内，把博客建在Wordpress.com还是不错的，但是最终你还是希望通过添加插件来定制你的博客，也希望增加广告。可能最终你还是不得不迁移到付费托管平台。这一过程有点儿麻烦，所以还不如一开始

就使用便宜的付费托管服务。

如果你决定采用付费的托管服务，你可以找到很多提供一键安装Wordpress.org软件的服务，几分钟内就可以搭好运行了。与免费服务相比，没难多少，但你能有更多的权限去定制自己的博客。

你也可以将自己的博客托管在虚拟专用服务器（VPS）上。VPS基本上就是一个能托管博客的完整的云端操作系统。在所有的付费服务中它是最便宜的，但也是难度最大的。我目前就将自己的博客Simple Programmer（<http://simpleprogrammer.com>）托管在VPS上。但是如果你刚刚起步，我不建议你这么做。

如果你还是决定用免费的托管服务，我有一条忠告：务必注册自己的域名。免费的托管服务默认会给你的博客分配一个地址，但这个地址是他们的域名的一部分。你应该注册自己的域名并以此来代替默认的博客域名，即使这样你需要支付额外的费用。因为，你的博客流量的很大一部分可能是来自Google这样的搜索引擎。

Google给特定网页和域名进行“网页排名”时，主要依据是有多少网站链接到该域名。如果你计划未来把自己的博客切换到付费托管服务上，你需要先确保你能够把搜索引擎的相关性（即网页排名）一起带走，所以一定要确保一开始就有自己的定制域名。（你也可以以后再解决这个问题，但是不值得这么折腾。一开始就正确地做事要容易得多。）

创建博客的步骤

- 决定要使用哪种托管服务：免费的、付费的还是VPS。
- 设置或安装博客软件。
- 配置或者自定义所有主题。
- 开始写吧！

打造成功博客的秘诀

好吧，现在你已经设置好自己的博客，也写了几篇文章，接下来怎么办呢？要是没什么人读你的博客，那也是无济于事，所以你应该了解如何获得流量。毕竟，本章不就是论述如何打造成功博客嘛。

打造成功博客的最大秘诀有且仅有一个——持之以恒。我和许多成功的博主都谈论过，他们都有一个共同点——写了很多博客。我认识的一些最成功的博主每天都要写博客，而且已经坚持了许多年。

别担心，你其实不必每天都写博客（刚开始的时候，一周写上两三篇博客也无妨）。最重要的是，定好一个计划，然后坚持不懈。博客更新的频率将决定你的博客成功的速度。我强烈建议以每周至少一篇的速度更新博客。以这个频率，你每年就会增加52篇博客。这非常关键，因为正如我前文所说，你的博客流量的很大一部分（甚至是绝大部分）都来自Google这样的搜索引擎。博客文章写得越多，来自互联网搜索的流量也就越多（只要文章看起来像那么回事儿，而不是一大堆词汇的随机堆砌）。

我敢肯定，如果你每天写一篇博客且坚持好几年，想不成功都难。尽管如此，只是持之以恒还不足以使你的博客大获成功，你还应该确保你写的都是高质量内容。为什么要重视博客内容的品质？原因有两个。第一个也许也是最重要的一个原因是，博客内容的质量越高，越能吸引人持续浏览你的博客，或者通过RSS和电子邮件订阅阅读。在你给读者提供更多有价值的信息的同时，你也就成功积累了自己的目标客户。

重视博客内容品质的另一个重要原因是为你的博客提供更有价值的链接。大多数搜索引擎判断网页品质时都是基于有多少其他页面链接到这个页面上。网站内容的品质越高，就越有可能被社交媒体分享，并被从其他网站链接到该网站上。链接到你博客的网站越多，能给你的博客带来的流量越多——道理就是这么简单明了。你也希望能写出让别人愿意阅读并分享的内容。

在你被这一切吓倒之前，我要强调一点：别担心。你的博客不需要完美。在起步之初，你的博客很可能会很糟糕。但是，随着不断尝试贡献出优秀的内容，而不是将自己脑子里的东西随意丢到网页上，不考虑格式和结构还有一大堆拼写错误，你终会成功。每周只发布高质量的内容，随着时间的推移，你的博客的品质会越来越高。

有价值的内容可以以不同的形式呈现。分享你的经验也许能帮助阅读你博客的人，而一个有趣的故事也许能让他们感到些许快乐。

持之以恒地坚持写作，坚持不懈地产生高品质的内容，如果你做到了这两点，基本上你就成功了。我是怎么知道的呢？因为我一直都在给软件开发人员做演讲，每当我问软件开发人员“谁开了博客并且坚持每周更新的请举手”时，一屋子100个开发者，运气好的时候我只能看到一个人举手。持之以恒地撰写优质内容，能让你轻松跻身开发人员的前1%之列，至少在自我营销这方面。

地雷：可是我真的不知道该写些什么

许多想要开博客的人要么从来就没有开过，要么开了之后不久就很快放弃了，因为他们要么不知道该写什么，要么发现自己实在没什么可写。

解决这个问题的最好方法是提前头脑风暴出各种不同的想法，随时更新可能的博客主题的清单，这样你总是保持一堆话题可供选择。

同时，不要太担心你的文笔如何，不要太在意别人的想法。有时候你只是要写一篇博客让自己的博客有内容，并不知道这篇博客会是自己点击量最高的文章。我写过不少自己觉得很差的文章，却成为最热门的文章。

要想弄清写什么，还有一个技巧，就是与别人就某个话题展开对话，甚至辩论。我经常发现自己写的最好的文章一般是先前曾与别人讨论过的。找一位朋友，就某个话题展开辩论，你会发现这个话题得写好几页。

当然，还有一些其他的方法可以让你的博客成功。接下来我们来讨论这些方法。

提升访问量

刚开始开博客的时候，基本上就是无人问津。你无法从搜索引擎获得更多的流量，也没有哪个网站会链接到你的博客上。你该怎么做呢？

我要推荐的第一个策略就是开始评论其他人的博客。找到撰写类似主题的博客的开发人员，在他们的博客写下有意义的评论，只要有机会就链接到自己的博客上。（通常在为了评论而进行注册时，你的个人主页会包含你的博客链接，所以你甚至都不用亲自在评论中链接到自己的博客。）

要想让这一策略奏效，你需要付出不少努力，但是这种方式也能帮你与欣赏你的评论的博主建立联系。（请不要做出只包含你博客链接的垃圾评论，那不会给对话增加任何实际价值。）坚持每天都在不同的博客上发表评论，假以时日，你就能看到来自你访问并留下评论的博客的流量了。你的评论越有见地，人们越有兴趣查看你在自己博客上所说的。（你也可以撰写博客文章，作为对别人的某篇博客文章的回应。这是获得流量非常有效的一个策略，特别是，如果他们也添加了你的博客链接的话。）

另外一些在初始阶段获得流量的好办法就是：在社交网络上分享你的博客文章，在你的电子邮件签名的底部以及所有的在线个人主页中添加你的博客链接。这种方法可能不会产生如你预期的流量，但仍然值得一试。

你还应该让分享内容的方式更简单，以方便其他人分享你的内容。如果你使用Wordpress.org，可以使用各种插件给自己的内容添加分享按钮。Wordpress.org软件甚至内置了一些分享功能。你甚至可以在博客底部添加一个“一键式”按钮，让你的博客的读者直接分享博客的内容或者订阅你的博客。

最后，如果你足够勇敢并且认为自己的文章足够优秀或者富有争议，可以把自己的文章提交（或者请别人提交）到Reddit（<http://reddit.com>）或者Hacker News（<http://news.ycombinator.com>）这样的社交新闻网站。当然最后还要提醒一句：有些挂在这些网站的人纯粹心怀不善。我曾经把自己的文章分享到Hacker News，差点被怒气冲冲的评论员们生吞活剥，他们就是故意找茬儿。你得脸皮够厚，承受得起这类恶毒的辱骂。但是，如果某天你的一篇文章在这些网站大热，那你每天都能收到数万的访问和回链。总的来说，还是值得的。

我可不能保证你能够一鸣惊人

我很想说“如果你遵循本章中我教你的每一件事，我保证你会成功”，但是很不幸，我无法做这样的保证。我只能说：“如果你遵循这些建议，你更有可能获得成功。”让你的博客大获成功，需要一点运气和时机，但是一个博主没有持续写出高水平的内容就能获得成功，这几乎闻所未闻。

采取行动

- 你喜欢的开发人员博客有哪些？看一看你读的这些博客，试着列出这些博客的更新频率和每篇文章的平均长度。
- 如果你现在还没有博客，马上行动。今天就注册一个，然后创建你的第一篇文章。制订一个时间表，强迫自己将来坚持写博客。
- 承诺自己会坚持写博客至少一年时间。要实现目标需要时间和努力。大多数人坚持一年左右的时间就会看到相应的效果。
- 创建一个流动的博客文章主题列表。每当有新想法，就把它添加到这个列表中。在需要写新博文的时候，文章主题就能够信手拈来。

第22章 你的主要目标：为他人增加价值

不要努力成为一个成功的人，而努力成为一个有价值的人。

——阿尔伯特·爱因斯坦

当你营销自己的时候，如果你所做的一切都是为了自己的利益而不给别人带来真正的价值，那么即使你所做的都正确无误，那也无济于事。你可以写博客、在社交媒体上分享自己的内容、在大会发言、著书立说，尽你所能来推广自己，但是，如果你表达的和传递的信息不能帮到其他人，那么每个人都将会无视你。

人们最关心的还是自己。没人想听到你的成功故事，也不想知道为什么他们要帮你获得成功，但是他们肯定想听到你会怎样帮他们获得成功。所以，要想让自我营销的所有努力奏效，基本的方法就是帮助他人获得成功。

Zig Ziglar说得好：“如果你能帮助足够多的人们得到他们想要的东西，你就会得到自己想要的东西。”这是一个基本的策略，你应该将其用于自我营销。它比其他任何方法都更有效。

给人们想要的东西

要想给人们想要的东西，要先知道他们想要什么。但是这并非易事，因为如果你问他们，他们可能会撒谎。他们并非故意撒谎，而是他们也只有一个模糊的概念，并不确切地知道自己想要什么。就像新娘在找心目中的完美婚纱时，只有切实看到了才会知道“这就是我想要的”。

想要弄明白他们的真实想法，你得全凭自己。你必须要学会通过现象看本质，然后找到方法来提供自己的价值。如果你已经深谙此道，对你来说这个就是驾轻就熟。但是，如果你还疏于此道，那么你要去探究一下人们对什么感兴趣。在网络论坛上与你选定的细分领域相关的话题有哪些？你认为行业的整体趋势是什么？以及，或许是最重要的，人们都在害怕什么，对此你又该如何应对？

你提供的内容应该直接瞄准你所选定的研究领域，为该领域带来价值。你可能会对某一构架或者某一技术的某个方面特别感兴趣，但是如果你的受众对此并不感兴趣，那也是白搭。另一方面，你通过自己的博客或其他你创建的内容触动了别人的神经，你很快就能发现受众的兴趣点。如果你能通过自己产生的内容解决一个真实的问题，或者能让别人关注你文章的内容，那你就为他人创造真正的价值。你也就给他们想要的。

把你工作成果的90%都做成免费的

有些人心存疑虑，为什么我将我提供的所有内容都设置成免费的。我每周写三篇博客，发布一个YouTube视频、两个播客还有其他一些内容，这些都是完全免费的。我坚信你提供的所有内容，90%应该完全免费。为自己的辛勤工作收费，无可厚非。但是你会发现，当你免费给人们提供实际价值时，回报更大。

免费内容比付费内容更容易被分享。你撰写博客、制作视频或者播客，然后将这些内容免费提供，与收费内容相比，人们更可能会分享和传播这些免费的内容。分享免费内容就跟在Twitter上发链接或发邮件一样简单。与付费内容相比，免费内容能让你触及更多的目标受众。

通过提供免费内容，你让人们有机会无需投入资金就能了解你生产的内容多么有价值。你可能暂时没有收费的打算，但是一旦有此计划，你很容易说服人们去付费，因为他们早已通过你提供的免费内容知道你提供高品质的内容。免费还会让人们对你心生感激之情，他们也希望能够通过购买你开发的产品或服务来回报你。

虽然免费做这些事情看起来是在浪费时间，但是你要把它视为对未来的一项投资。通过为人们创造价值并且免费提供这种方式自我营销，你就赢得了为他人提供价值的声誉，也为自己的未来创造了机会。这种声誉的价值是无法衡量的。它能帮你赢得更好、更高薪水的工作，获得更多的客户，或者成功地发布一款产品。

奔向成功的快车道

每次当你开始着手做某件事情的时候，无论是创建博客、截屏视频还是别的活动，你应该从它是如何为他人创造价值的角度来看待它。正如我坐在这里写这本书的时候，我时常都在想，我写下的每个字是否能让你获益。我该如何向你传递这些对你有帮助的信息？我如何才能为你提供价值？

你很容易落入这样的陷阱：一直谈论自己并试图证明自己价值连城。然而，你会发现，能解决他人的问题，真正能够帮到他人，你更容易获得成功。跟别人唠叨为什么自己是世界上最好的Android开发者，对你并无益处，但是如果你能帮助别人解决他在开发Android应用时遇到的问题，他就会认为你很棒。

你应该把这种态度应用在所有自我营销的媒介上。在下面几章中，我们将讨论如何利用多种媒介来营销自己。但是，如果你不知道如何通过为他人解决问题和提供价值来把自己与目标客户关联起来，你就不会成功。

赠人玫瑰，手有余香

也许你会认为，应用“思利及人”的做事方式就可以获得成功是值得质疑的。但是，事实证明，最富有创造力的人也是最乐于助人的人。为什么呢？我个人认为这是多种因素组合的结果。你帮助别人越多，面对的问题和情况越多，就能结交的人也越多。总是帮助他人解决问题的人，在解决他们自己的问题的时候会变得更加轻松自如，而当他们真遇到障碍的时候，总是会有好多人伸出援手。

相信这一点的人并非只有我一个。我读过一篇有趣的文章，讲述沃顿商学院一位31岁教授的故事。他极为高产，也乐于助人（<http://simpleprogrammer.com/ss-giving-secret>），在自己的研究领域——组织心理学方面建树颇丰。他所做的大量研究表明：帮助他人实际上就是在帮助自己获得成功。

采取行动

- 什么样的内容会让你觉得最有价值？有没有哪个特别的博客会让你每周都去阅读，或者哪个播客的内容如此有价值让你欲罢不能，以至于一集都不想落下？
- 你可以提供给自己的受众或你的细分市场的最大价值是什么？你认为什么样的内容对于你的受众群体而言是最有价值的？

第23章 善于运用社交媒体

如今，社交媒体在人类生活中占据很大的比重。Facebook、Twitter、Google+以及LinkedIn成为人与人之间重要的联系纽带，也是共享信息的重要手段。身为希望营销自己的软件开发人员，你需要在这些社交网络上占有一席之地，你需要学会如何通过分享的内容和分享的方式塑造自己的形象。

最近，“通过社交媒体进行品牌营销的重要性”成为社交媒体专家们的讨论热点。尽管我也认同其重要性，但是我并不认为它像某些人对你宣扬的那么有效。不管怎样，如果你想在受众面前最大限度地提升曝光度和参与度，那你就需要了解使用社交媒体来推广自己品牌的基础知识。

在本章中，我会帮助你梳理社交媒体的使用策略，带你领略一个个主流社交网络，并且展示一些运用社交媒体传播自己信息的方法。

培育你的社交网络

要想用好社交媒体，首先你要给自己积累足够多的粉丝，或者至少是让人们进入你的社交网络。如果你拿着大喇叭大声呼喊却无人听到，那也无济于事。

有多种不同的策略来帮助你构建你的社交网络，你要根据具体的社交网络来决定自己该怎么做。不过对于大多数社交网络来说，最容易做到的就是关注他人，或者邀请他人加入你的网络。方法很浅显，但大多数开发人员还是守株待兔，等着其他人来关注和互动。记住，如果你对别人感兴趣，他们会对你更感兴趣。

你也可以把自己社交网络的个人主页的链接放在你的在线介绍、博客文章的结尾或者邮件的签名档中，通过这些方式获得关注。简化人们联系你的方式，人们就会联系你。你也不要害怕去主动邀请别人。在博客文章的结尾呼吁人们在Twitter上关注你，无伤大雅。

要构建大型的社交网络肯定会需要一些时间，所以不要急于求成。那些宣称“可以在几天之内大幅度增加你的粉丝数量”的黑幕服务看起来很有诱惑力，但是，大多数情况下花钱买“伪粉”结果都是打水漂。这些“伪粉”实在是不值一提，因为他们并不代表真正的人群。

有效地运用社交媒体

你的社交媒体应用策略应该主要聚焦于构建稳定的受众群体，并逐步提升活跃度。你应该想方设法让人们从关注者变为粉丝，这样他们就会更加关注你的内容，会与别人分享，也会积极地推荐你。这也是你打造业界声望的方法。那么，你该怎么做呢？

我依旧将这个办法归结为“提供价值”。如果你持续不断地在自己的社交媒体上给别人分享和提供有价值的内容，那你必定会赢得尊重和声誉。但是，如果你总是发布不适宜的、攻击性的内容，又或者都是一些只与你自己相关的类似“早餐吃了什么样的鸡蛋”这样毫无营养的内容，那就等于是把人赶走。

那么，你应该在社交网络上发布什么内容来为他人带来价值呢？答案很简单：你认为有用或有趣的。你自己觉得有价值的东西，在很大概率上别人也会认同。要确保你发布的内容一直保持在较高的水准上。如果你以“江湖百晓生”而闻名，特别是在某个细分领域，那么人们自然会更多关注你在社交媒体上的分享，并且更愿意分享你的内容。

每周我都会把自认为有用的信息整理为一份内容集锦，藉此吸引人们在社交网络上关注我。这份集锦通常包括博客文章、新闻报道、励志名言、与软件开发相关的小技巧等，同时我还会提出一些问题来挑战我的关注者，促使他们与我对话。表23-1展示了我分享的内容类型。

表 23-1 通过社交媒体分享的内容

方式	内容
博客文章	转帖一些博客文章，或者自己的博客
新闻报道	转载一些有趣的文章，尽量与你的细分领域有关，或者与软件开发相关
励志名言	名人名言，特别是鼓舞人心、非常流行的名言警句
技巧、小窍门	任何你所了解的特殊技巧或者知识，别人会很欣赏
幽默故事	发一点儿幽默故事挺不错的，但是一定要确保确实好笑，另外千万别冒犯别人
吸引人的问题	这是一个很棒的吸引你的听众并和他们互动的方法

方式	内容
自己的一些推广活动	别太多，并且跟其他内容混合在一起

显然你要一直发布新博客文章或者自己的其他内容。不过如果你要销售书或别的产品，或者自己提供的咨询服务，你要谨慎地在自己的社交媒体上发布。正如你要把自己的产品/服务90%免费一样，你分享给自己的关注者的90%也应该是有价值的内容而非你的广告。

保持活跃度

社交媒体面临的一大挑战就是保持活跃度。如果你不能持续不断地在你的社交媒体上保持足够的活跃度，你将会丢掉大量粉丝。但是，保持活跃也会成为一种负担——每天都要管理和维护自己的Twitter、Google+、Facebook和LinkedIn以及其他各种社交媒体，同时还得从事现实生活中的实际工作。

除非你愿意每天都投入大量的时间，否则你不可能在所有的社会化媒体平台都保持很高的活跃度。那么，最有可能的情况就是，选择一到两个作为你最心仪的平台。

就我个人而言，其实我并不喜欢在社交媒体投入大量的时间。我觉得它很容易就把我每天的时间蚕食掉，所以我尽量对它退避三舍。但是我仍然需要保持活跃，那该怎么做呢？

我目前使用Buffer (<http://simpleprogrammer.com/SS-buffer>) 这一工具，当然能做同样事情的类似工具还有很多。Buffer能够让我一次性安排好我的社交媒体更新计划。当新的一周开始的时候，我会浏览所有我希望分享到自己的社交媒体上的各种内容，再创建一个各种内容的集锦。我会计划好在这一周内的不同时间发布这些内容。当我发现一些有趣的值得分享的东西时，我可能会临时增加内容，不过每周我都确保自己每天在每个社交网络至少发布两篇内容。同时我还设置：当我发表新的博客文章或YouTube视频时，这些内容被自动分享到我的所有社交网络上。

我强烈推荐你采用类似的方法来管理自己的社交网络，这样你每天就不用为了面面俱到而浪费大把的时间。我现在每周在社交媒体上只花费不到一小时的时间，确实相当高效。

不同的社交网络，不一样的功能划分

作为一名潜心钻研“自我营销”的软件开发人员，你应该在各大主流社交网络上占有一席之地，特别是更偏重技术和职业发展的社交网站。你可能也想要创建个人页面或个人主页，以便直接展示自己的品牌。要想同时维护好自己的个人账号和职业账号，工作量相当巨大。

我的首选推荐是Twitter，因为许多开发人员都在使用Twitter，它是一种能让你结识用别的方法认识不了的人的极好方式。你可以在一条推文中提及别人，甚至是相当有名的人，并且有较高的概率获得回复，因为Twitter使用起来很方便。某人可能会忽略你发给他的电子邮件，但是他会只花几秒来回复你的推文。我还发现，Twitter也是分享博客文章和科技相关新闻的好地方。Twitter对字数的限制能保持对话简短又切题。

接下来我会推荐LinkedIn。很显然，鉴于LinkedIn是面向专业人士的社交网络，你应该在LinkedIn上创建个人主页。在LinkedIn上你可以创建自己的在线简历，与其他专业人士建立联系。它是社交和专业内容的优质渠道，你的博客文章能够准确直达目标受众。你还能使用LinkedIn的群组功能直接与你的目标群体建立联系，他们要么对你的细分市场感兴趣，要么自己早已参与其中。

LinkedIn最为人津津乐道的功能就是你可以邀请联系人为自己的能力背书。这项功能很棒，你一定要善加利用。对于列在LinkedIn个人主页上的每一项工作，请务必找前同事或者经理为它们背书。这可能会让你觉得有点儿不自在，不过LinkedIn页面上的背书可能会与你原来的认知相去甚远。背书提供了社会证明，这对于塑造个人形象是一个强有力的工具。想想你最近一次在亚马逊上买东西的经历。你有没有阅读产品评论，找到评价最高的产品？那是在网上购物的基本方法，我还知道很多人也是这样在网上购物的。

Facebook和Google+的重要性都不及Twitter和LinkedIn，但我还是建议在上面都开一个账户。对于这些平台中的任意一个，你可以使用个人账号，也可以创建一个页面用作业务介绍或品牌介绍。在Facebook和Google+上你也可以找到有价值的群组，可以与你的目标受众保持联络，也能向有兴趣的人直接分享特定的编程语言或技术。

采取行动

- 你当前是如何使用社交媒体的？看看你使用社交媒体的时间线，找出别人根据你在社交媒体上的分享，对你和你的品牌产生的印象。
- 制订一份社交媒体改进计划。确定有哪些内容需要分享在不同的社交媒体上，把它们整合在一起形成一套每周内容分享策略。关注哪类分享最为流行。

第24章 演讲、报告和培训：做“说话的极客”

演讲和举办某种形式的培训是连接普罗大众和自我营销的最有效的方式之一。虽然这一方式在扩散速度上不及其他媒介，但是站在目标受众面前，直接面对他们侃侃而谈，却是最能打动他们的方式。

就我自己而言，再也没有比走上讲台发表演讲更让我兴奋不已的事情了。这种面对面地与听众直接交流、直接获取反馈的方式拥有特殊的魔力，是其他方式所不能比的。

即使你没有机会走上讲台上发表演讲，在工作中举办一些讲座对你的职业生涯也大有裨益。你可以创造绝佳的机会展示你的高效沟通能力，充分表达自己的想法，从而打动你的同事甚至你的老板。

唯一的问题在于，开口讲话并不那么容易。开始的时候你大概还会很踌躇——“我没有经验啊”“我很紧张很害怕啊”。走上讲台在公众面前开口讲话，确实不轻松，尤其是在你从来没有这样做过的时候。

在本章中，我将告诉你，为什么演讲和培训这类事情对你的职业生涯非常重要，我还会给出一些具体的操作性建议：如果你从来没做过，我会告诉你如何开始；如果你已经做过了，我将告诉你如何更上一层楼。

为什么现场演讲如此有力

你听过摇滚音乐会，看过乐队的现场演出吗？为什么非要去现场呢？完全可以买张唱片在家里悠哉悠哉地听上一曲啊。没准儿用你自己的耳机聆听CD的音效会更好呢。去听歌剧、看戏剧也是如此。为什么不能换成看电影？

要想解释清楚有些困难，在你参加现场活动的时候，你建立了非常个性化的链接，而这种感受无法通过听录音或者看录像获得。同理，让听众在现场聆听演讲会比其他媒介更能打动他们，即使它们的内容完全一样。

听过你演讲的人更容易记住你，也会觉得与你建立了某种个人关系。这就好比我们能够记住去看自己心仪的乐队现场演出的次数，但是记不住听他们唱片的次数。

演讲也是一种互动媒介，或者至少你能将其作为媒介使用。当你在大会发言时，你能够直接回答现场听众的问题，让他们参与到你的演讲当中。这种交互方式能够迅速建立起人群对你的大量信任感，能帮你收获粉丝，促进信息的传播。无巧不成书，在我撰写本章内容的时候，我看到有一位开发者正在发推文。他曾经参加过我组织的“如何自我营销”的讲座，现在他不遗余力地把我和我的博客推荐给别人。我想，要是他没有听过我的演讲，他是不会主动做这件事的。

许多你早有耳闻的知名软件开发人员也是通过演讲拓展他们的职业生涯的。我的朋友John Papa就是一个很好的例子。开始的时候他只是在小型会议上发表演讲，现在他穿梭于世界各地，畅谈各种技术。通过成为“演说家”而为人所知，进而给自己创造了很多机会。

开始演讲

好了，现在我应该是已经说服你了，演讲是如此重要，值得一试。不过你可能还在疑惑该如何开始，因为这还真有点儿棘手。

首先我要提醒你，如果你先前从来没有做过公开演讲，也没有建立起自己的名声，那你不可能去重要的大会上发表演说。不过你也没必要以这种方式起步。最好从小规模的场合做起，逐渐完善你的演讲技能。要想能在公众面前从容自如地发表演说，需要很长时间的刻苦练习。

一开始最好是在自己的工作场合做讲座。大多数公司都欢迎自己的员工分享各种话题，特别是与当前工作直接相关的内容。你可以介绍自己的团队正在使用的技术，或者给团队进行一些相关领域的培训。你无需把自己包装成为专家，只是热心希望通过分享自己所学而对他人有所帮助。（事实上，你会发现，你应该经常采用这种方法。太多人希望被包装成为“专家”，从此不再真诚谦逊。而作为一个脚踏实地、有缺点和弱点的真实的人，花更长时间与公众建立信任，也会让你看起来不那么古怪。）

另一条成为“演说家”的路径是代码训练营^[1]和用户组举办讲座。大多数大城市里都有各种软件开发人员组成的用户组，加入离自己近的用户组并不难。在加入用户组一段时间后，你可以询问组织者是否可以就某个特定的话题做一次讲座。大多数用户组都欢迎新人分享，因此只要你的话题有趣，就有机会出场。这样的机会非常好。你面对的是一个较小规模但是更宽容的听众群体。此外，这也是向当地的公司和招聘人员推销自己的好机会。

除了用户组，每年世界各地还有各种代码训练营。大多数代码训练营允许任何人就它们选定的主题发表演讲，不拘能力经验水平。抓住这样的机会，每年至少在一场代码训练营上演讲。因为参加这些活动是免费的，所以这种场合下演讲者的压力很小。你大可以放松，即使搞砸了也没什么大不了的。

最后，一旦你拥有了一些演讲经验，就可以向各种开发者大会提交演讲申请。当然，在这一领域会有一些竞争，往往还会经过一个“老男孩”系统的筛选^[2]，但是一旦你获得认可，每年你就可以获得很多演讲机会。而且，大多数这种会议，会为你全额报销差旅和其他费用。（我认识的许多软件开发人员都穿梭于世界各地参加这种大会。他们可能不会因为演讲而获得酬劳，但是他们能够去他们没有去过的地方，扩展自己的受众群体。如果你是自由职业者，这样的大场合也是拓展业务的好地方。）

地雷：我对在公众面前演讲怀有恐惧

没事的，很多人都有此症。公共演讲恐惧症是最常见的一种恐惧症。对此你可以做什么呢？好的，有像Toastmas (<http://www.toastmasters.org/>) 这样的组织，帮你在舒适的氛围中克服对公共演讲的恐惧，你可以加入他们。你也可以从较小规模场合的演讲做起，比如在会议上站起来发言，或者在一小群你熟知的人面前做讲座。当你对此越来越感到轻车熟路的时候，你就可以到更令人生畏的地方公开演讲了。

你要记住，作为人类，我们拥有良好的适应能力。只要你把一件事情重复足够多次，你自然就会接纳它。伞兵们第一次跳出机舱的时候都很害怕，但是经过多次成功跳伞之后，恐惧最终都会消失。如果你一直坚持在公共场合发表演说，你一定会应对自如，恐惧感终将消散。

做培训怎么样

做培训，不论是现场培训或录制培训，都是建立自己的声誉的另一种很棒的方式，甚至还能赚点钱。我本人正是通过提供网上课程取得了相当大的成功的。培训不仅能够给我带来收入，还让我获得了“行业专家”的美誉。

在过去，成为职业的培训讲师或是获得讲课机会，都挺难的。现在，几乎每个人都可以提供某种形式的在线培训课程。当然，你也可以坚持做传统的课堂培训，但是对于大多数开发人员来说，由于不打算以此为职业，因而网上视频培训是一个更简单也更灵活的解决方案。

刚开始的时候，最好的方式是在免费视频网站（如YouTube）上通过创建截屏视频的方式做分享。截屏视频就是在你演示如何操作的时候录制你的计算机屏幕。如果你能通过这些截屏视频一板一眼地教会其他开发人员某个概念，你就可以轻松赢得“精通某某领域的专家”的美誉。这种声誉可以帮你赢得更好的工作，如果你是自由职业者，“专家”的美誉还能带给你客户。

即使开始的时候你提供的是免费的培训课程——如前所述，提供免费培训是提升自己品牌的绝佳方式——最终你也可能想为自己创建的内容收费。关于如何针对你的视频培训内容收费，有多种方案可供选择。

首先，可以与像Pluralsight (<http://simpleprogrammer.com/ss-pluralsight>) 这样的专业为开发者提供培训的公司合作。我的大部分在线培训视频都是为Pluralsight制作的。不过也有其他公司会付费请你去制作内容，将部分利润以版税的形式付给你作为酬劳。这与写书类似。当你为这类公司制作内容的时候，你通常只负责制作内容，无需为市场和销售操心，因为你的内容会被整合进它们现有的课程体系中。通常情况下，这类网站都会有某种形式的试讲环节，因此并不能保证你一定会被接受，但是试一试总没有坏处吧。

如果你想走更为激进的“单飞”路线，那可以试着自己创建内容并直接销售。在这方面我已经获得了成功，一直在自己的网站上直接销售“[How to Market Yourself as a Software Developer](http://devcareerboost.com/m)”（“作为软件开发人员，你将如何营销你自己”，<http://devcareerboost.com/m>）培训课程。这种方法的唯一困难在于，你不得不亲自完成所有的营销工作，你必须找出内容发行的方法和相应的收款方式。

还有一种混合方式就是与在线教育公司Udemy (<http://simpleprogrammer.com/ss-udemy>) 开展合作。Udemy允许任何人在他们的托管平台上发布内容，但是他们会抽取利润中的很大一部分。你主要负责做自己的营销，为自己赢得客户。我知道有几位软件开发人员已经在此平台上获得成功。

采取行动

- 列出你所在地区的所有用户组清单，同时列出你能去发表演讲的代码训练营清单。挑选一个自己觉得比较有把握的专题发表一次演讲。
- 找出网上提供的那些针对软件开发人员的培训课程，无论是付费的还是免费的，关注并记录找出成功的培训师是怎么做的。
- 试着以截屏视频的形式创建自己的短时培训课程，然后在YouTube等网站发布。
- 创建一个列表，列出所有你可以发表演讲的主题。

[1] “代码训练营”是一种非正式的全免费社区组织，通常是按照地区来组建的。按照社区成员的建议，代码训练营不定期的举办讲座。讲座的主持人也是代码营的成员。最初的代码训练营由微软倡导组建。以上这段文字根据维基百科编译。——译者注

[2] 暗指要经过经验和技能考察。——译者注

第25章 著书立说，吸引追随者

如果你想要在写作方面有所建树，就必须要与你的读者建立密切的联系。衷心希望，你在阅读这些文字，能够感受到我与你在一起——这正是我想要做到的。我本可以以另一种截然不同的方式开始本章，但是我没有，因为我希望能通过直接对话的方式强化我们之间的联系。

如果我所做的一切都正确无误，在你阅读这些文字的此刻，你应该感到我正在与你谈话，而不是说教。文字不只是信息。文字就像一块功能强大的画布，将你的心声传递给其他人。有时，在你阅读我写的文字的时候，这种感觉可能会比我对你所讲的同样的话语更为真实。通过文字，你能传达自己的心声；如果你的心声很有趣，能够吸引到读者的注意，给他们提供价值，那他们就会与你建立联系，你终将征服他们。

为什么著书立说如此重要

你有没有听过“关于这个主题，他写过一本书”这样的话？书籍为它的作者带来非凡的影响力。一个人通过写书可以收获极大的信赖。如果你也想被看作业界值得信赖的人，你也应该写本书。同样，在软件开发的相关杂志上发表文章也是如此。大多数人都认为，如果某个人就某一特定主题写过书或者发表过文章，那么他们就可能是这方面的专家。如果你打算营销自己，被视为专家当然无任何坏处。

除了能让自己大名出现在装订成册的书脊上，一本书就像是一辆满载的货车，可以以非常具有针对性的、聚焦的方式传达你的信息。当人们坐下来读你写的书的时候，你会长期被他们关注。读完一本书通常需要10~15小时。你很难再找到别的媒介，能让人投入这么长的时间去倾听你的信息。通过一本书，你能将自己想要表达的信息完整无误地呈现给你的读者。

书籍和杂志都不赚钱

许多软件开发人员误解了写书的原因。他们想当然地认为大多数图书和杂志作者能够通过写作赚大钱。但是，事实很简单，你不会为了赚钱而去写书。写书是为了提高自己的声誉。

通过一本书可以获得一笔可观的收入是非常罕见的，而且通常作者只能获得收益的一小部分。大多数杂志只会为每篇文章支付少量的酬劳，而撰写和编辑这篇文章本身可能需要很长的时间。所以，不要指望可以直接通过写书或发表杂志文章致富，除非你碰巧非常非常幸运，轻松写出一本畅销书。

但是，没有直接收益并不意味着出书没有回报。正如我之前提到的，写书或在杂志发表文章的真正好处在于，通过出版，你能获得更广泛的知名度和声望。出版业类似于守门员，为内容的质量把关。如果你能突破守门员进球，你会发现这些已经出版的内容间接地带来了各种各样其他的赚钱机会。

那些出版过专著的作者会发现自己更容易获邀在各种会议发表演讲，能够建立自己在某一学科的权威地位，这些都能为你带来更多的客户和更好的工作机会。

获得出版机会

我不得不承认，这本书是我的第一本以传统方式出版的书籍。经过与其他许多作者的交流，我了解到出版书籍并非易事，尤其是出版你的第一本书。许多出版商并不愿意冒风险与完全不熟知的作者合作，作者无法如期交稿的风险很大，毕竟写书不是件轻轻松松的事。

要想让自己有机会出书，最好的办法就是明确一个有市场需求的主题，同时也能够充分展示你作为该领域专家的学识。如果你已经为自己的品牌确定了细分市场，由于你已经拥有了自己的专业领域，并且没有太多竞争，你更容易做成这件事。你的主题越聚焦、越专注，越容易证明你的专业能力，但同时你的潜在受众群体也会越小，所以你需要找到适当的平衡点以吸引出版商。

你也应该提前做功课从而让自己在市场上立足。我建议你开设博客，给一些较小的杂志投稿。随着你在自己的专业领域树立起自己的声望，你可以积累更大的出版物。出版社和杂志社都愿意和拥有众多粉丝的作者合作，因为这意味着他们的名气就是发行量的保证。你的粉丝有多多，你对出版商的吸引力就有多大。

最后，你应当准备一份翔实的写作提纲（文章摘要），清晰地概括自己的写作目的，明确本书的目标读者，以及你为何认为这本书会成功，为何你是写作这本书的最佳人选。你的提纲写得越好，它被出版商接受的可能性就越大。

地雷：可是我不擅长写作

我也不行，可我还是写了本书。我在学校学习的那段岁月里，最差的科目一直都是英文。我的数学、科学甚至是历史都名列前茅，但是我的英文就是勉强中等。我从来没有想到过，自己职业生涯的大部分时间居然是在写作，就像我现在做的这样。

发生了什么呢？嗯，我也就是每天坚持写作而已。开始的时候主要就是写些博客文章。我最初的博客文章写得极差，但最终我越写越好。虽然我仍然没有成为海明威那样的大作家，但是我现在可以用自己写下的文字有效地传达我的想法和意见，至少大部分时候如此。

我的观点是：别担心自己不擅长写作。现在写作能力不行，这并不打紧。要紧的是，你从现在开始写并坚持下去，随着时间的推移你的写作技能终将会得到提高。

自出版

传统的出版方式并不是你的唯一选择。越来越多的作者发现可以通过自出版获得成功，尤其是自己已经拥有粉丝群体的作者。我自己自出版了好几本书，完全能靠自己销售得很好。尽管我没有大型出版商的资源和发行渠道，但我也没有他们的各种开销。我能够将写书赚到的收益几乎全数收入囊中。

自出版是非常棒的起步方式，你完全可以凭一己之力完成，很容易做到。如果你打算写一本书，但是还没有找到合适的出版商（当然既没有签订合同，也没有确定书稿的最后交稿日期），自出版也是一个好办法，能让你确定写什么专题更合适。

有很多种服务能够帮你出版自己的书，其中在程序员中广为流行的一个是Leanpub（<http://simpleprogrammer.com/ss-leanpub>）。这项服务让你能使用Markdown这一简单的格式化语言进行写作，Leanpub将书籍做好格式转换后上架销售，并收取相关费用。他们收取的费用仅占书的总价的很小一部分。

你也可以通过亚马逊的Kindle Direct Publishing（Kindle直接出版）计划将自己的书直接在亚马逊上出售，或者使用Smashwords（<http://simpleprogrammer.com/ss-smashwords>）、BookBaby（<http://simpleprogrammer.com/ss-bookbaby>）这样的服务来将自己的书发行到多个市场。这些服务甚至可以帮你将你的书转换为电子书格式。

我的两位好朋友都是自出版书籍的作者，他们每年都能从图书销售中获得10 000~20 000美元的收入。这算是一笔不错的额外收入，也是让你声名鹊起、提升声望的绝妙方法。当然，对于后者，传统出版方式的分量更重。

采取行动

- 查看亚马逊上与软件开发相关的畅销书列表，找出哪种类型的书销量最好。
- 在你写长篇大作之前，先从杂志文章这样的短篇着手。找出一些发行量不高的软件开发类杂志，提交一篇文章摘要。

第26章 百折不挠，越挫越勇

如果你真的想在自我营销方面获得成功，你就一定要学着克服我们大多数人都会有的一种恐惧——看起来像个傻瓜。

登上讲台在众人面前侃侃而谈，并非易事。写博客被整个互联网看到并品评，也并非易事。在播客里听到自己的声音，在视频里看到自己，可能会让你感到难为情。从某种程度上说，就连写书也需要一定的勇气——特别是如果你想把自己已知的一切都放进书里的时候。

但是，如果想通过自己的努力获得成功，你就必须学会不在乎别人怎么想。你必须学会如何无惧自己被别人看作傻瓜。

万事开头难

当我第一次站上讲台面对众人演讲的时候，我紧张得汗流浹背。我试图让自己的声音保持平稳，但是我的声音却一直在颤抖。我本想点一张幻灯片，可我的手不听使唤滑过了两张。你们知道后来怎么样吗？我挺了过来。我可能没有做到最好。虽然我没能用我的魅力打动观众，但是我总算是把这段时间撑过去了，演讲最终结束了。

第二次登台，我仍然是一塌糊涂，但是我不那么紧张了。我的手不再抖得那么厉害，我的衬衫也没有被汗水浸透。接下来的每一次都更轻松。现在当我站上讲台上时，我会手拿麦克风，在整个房间里从容信步；我的激情感染了整个大厅，我自己也是活力四射。在前几次演讲的时候，我从来没想过自己能做到这样。

事实上，万物都在变化之中。随着时间流逝，原本让你不适的东西你最终都会应对自如。你需要足够的时间和意愿去克服尴尬，直到它们不再让你感到局促不安。

在第一次做某件让自己不自在的事情的时候，你可能无法想象终有一天自己竟然觉得做起这些事儿来如此怡然自得。你会误导自己去认为“这不适合我”或者“别人在这个领域有天赋，而我没有”。你必须学会克服这种想法，认识到大多数人在第一次面对挑战时都需要克服这样的不适感，尤其是他们面对一群人的时候。

我跟你说实话吧，大多数人并没有这么做到这一天。他们早早地选择了放弃。他们太在意别人怎么看自己，没有拼尽全力去克服这些困难，克服掉那些不适感，让自己变得更加优秀。如果遵循本书的建议，你会在别人失败的地方获得成功，原因就在于此。许多开发人员并不愿意做你愿意做的事情。大多数开发人员并不愿意为了收获更多成就而在短期内被看作傻瓜。

被看作傻瓜又如何

或许你现在已经相信我说的，随着时间流逝，一切都会更容易。如果你忍住坚持下去，如果你坚持写博客，如果你坚持上台演讲或者制作YouTube视频，最终你不会再感到“这不舒服”，甚至会觉得“这很自然”。可是，当你的手还受控制地颤抖，几乎无法握住麦克风的时候，你该如何做到这一点呢？

很简单，别太在意。别在意自己站在大庭广众之下可能会哑口无言，别在意别人看了你的博客后觉得你完全错了并且很蠢，别在意别人会嘲笑你，因为你已经准备好和他们一起嘲笑自己。同样地，我知道这事说起来很容易，但是我们还是要剖析一下。

首先，如果你看起来像个傻瓜，那最糟糕的情形会是什么？你不会因为让自己出丑而受到身体上的伤害。无论你在讲台上演讲得多糟糕，也没有人真的在乎。诚然，站在讲台上抖抖索索、汗如雨下可能会让你看起来很可笑，而当这一切都结束的时候，没人会记住这些。

想想看，上一次你看到别人在拳击台上被击倒是什么时候？你还记得他是谁吗？当他下台时你有起哄吗？你有给他发邮件或者打电话让他知道他表现得太糟糕，浪费了你宝贵的时间吗？当然没有，那你还有什么可担心的？

如果你想成功，你必须学会收起自己脆弱的自尊心，勇敢走出去，别害怕让自己出丑。每一位著名的演员、音乐家、专业运动员和公共演说家都曾有过表现不佳的时候，他们自觉地选择了走出困境，尽力而为。成功终将会来。你不可能专注做某件事而毫无长进，你只要坚持足够长的时间就会有收获。你的生存之道就是不必太在意。别害怕被人看作傻瓜。

在我的职业生涯中，我一共错失了9 000多次投篮，输掉了近300场比赛。我本来有26次绝杀的机会却投球不进。我失败了一次又一次。这就是我能够成功的原因。

——迈克尔·乔丹

小步快跑

如果照我的想法，我会带你到游泳池边上，然后一脚把你踹进深水区，因为据我所知，这是学习游泳最快的方式。但我知道，并不是每个人都接受这种“要么大获成功要么一败涂地”的激进方式，你可能更想采用小幅前进的方式稳扎稳打。

如果你对演讲、写作或者其他我在本篇前几章讲过的营销方式心怀畏惧的话，那么仔细想想你能做到的、不会让你紧张的、最简单的事情，并以此起步。

开始阶段，一种很好的方式是在别人的博客下面写评论。我发现，即便是这样的任务也会吓倒一些开发人员。但是，写评

论确实是一种很好的起步方式，不需要你长篇大论，而且你可以加入到对话中而不需要重新开启话题。

准备好面对批评，但不要惧怕。总会有人不喜欢你的言论，或者不赞同你的观点。那又怎样？在互联网的世界里每个人都有权发表自己的观点，所以不要让这些打击到你。习惯些许辱骂是件好事，因为即使再完美的作品也会遭人抨击。你永远不可能取悦每一个人。

一旦你觉得自己变勇敢了那么一点点，那就开始写自己的博客吧。围绕自己熟知的话题写，或者干脆写“怎么做”。不要写已有定论的话题，因为这会引来互联网“巨魔”的攻击，他们会爬出洞穴，用言语的大棒将你打倒。你会发现写博客并不是那么糟，有人可能真的喜欢你写的东西呢！（不要被这冲昏了头脑呦。）

现在你可以进一步拓展。你可以给别人的博客撰写客座文章，也可以接受播客访谈。你甚至有可能加入Toastmasters这样的俱乐部来习惯公共演讲。许多从来没想过自己能够在众人面前演讲的朋友通过Toastmasters成为了优秀的演说家。

无论是像缓慢适应水温那样小心翼翼，还是一头就扎进深水区，这些都无关紧要。核心在于一直保持前进状态。你会感觉不适，你可能会受惊吓甚至被吓倒，但是这一切终将会过去。如果你愿意坚持下去，面对前方的种种挑战，如果你不介意短期内被看作傻瓜，你就会在大多数人失败的地方获得成功。我保证这一切的付出都是值得的。

采取行动

- 让自己勇敢起来。今天就是你的大日子。走出去做一些令你害怕的事情，大小都不要紧，迫使自己处于不自在的环境，并且提醒自己：“这没什么大不了的。”
- 现在，重复上一步，至少每周一次。

第三篇 学习

教育就是当一个人把在学校所学全部忘光之后剩下的东西。

——阿尔伯特·爱因斯坦

软件开发的世界是不断变化的。每一天都会有一门新的技术出现，昨天你学到的东西今天可能就毫无意义了。

在这个飞速变化的世界里，学习的能力是至关重要的。软件开发人员如果选择固步自封，忽视自己的技能发展，那么他们很快就会落在后面，错失未来，只能被派去维护过去的遗留系统。如果想摆脱这种宿命，你就需要学会如何学习。

在这一篇中，我的目标就是教你如何自学。我将带你了解我开发的快速掌握新技术的“十步学习法”。我使用这一方法在一年时间内开发了超过30个适合开发人员的长期培训课程。我还会给你一些靠谱的建议，帮你寻找导师，指导别人，以及如何释放自己的内在潜力像海绵一样地吸收知识。

第27章 学习怎样学习：如何自我教育

走进学校，接受良好教育，没有任何问题。但是如果你毕业后就停滞不前不再学习，那么你将在生活中处于非常不利的境地。事实上，如果你一直依赖他人来教你，从来没有掌握自我教育技能，这会严重限制你提升自己的知识和技能的机会。

软件开发人员可以学到的最重要的一项技能就是自我教育能力。在这个新技术发展日新月异的世界里，就连初级职位的Web开发人员都需要掌握至少三种编程语言，所以自我教育是一项必不可少的技能。

如果想成为最好的软件开发人员，你就必须学会如何自学。很不幸，学校并不会教你自我教育这项技能。你可以轻松反驳说教育体系的设计初衷是针对群体而非个人。无论如何，学会学习是自我教育的核心技能。

剖析学习过程

你是否思考过自己是如何学习的？学习的真正含义是什么？我们几乎都是下意识地倾向于学习自己感兴趣的东西。当别人给我讲一个精彩纷呈的故事时，我们通常不会做笔记，也不会记住确切情节；然而我们中大多数人在听到故事后，不费吹灰之力就能将它复述出来。

这也同样适用于我们做的事情。如果我告诉你该怎么做，你可能会忘掉，但如果你自己动手做一次，你可能就记住了。如果你能将自己所学的东西教给别人，你不仅能记住，还能理解得更深刻。尽管每个人的学习风格千差万别（更多信息参见<http://simpleprogrammer.com/ss-learning-myth>），但是通过动手实践和教会他人，我们能学得更好。与其他学习方式相比，主动学习是效率更高的方式。

教育的首要目标，并不在于“知”而在于“行”。

——赫伯特·斯宾塞

可以换个角度思考一下这个问题：你可能看遍了教你如何正确骑车的书，也可能看过别人骑自行车的视频，我也可以给你培训正确骑车的机械原理，但是，如果你从来没有骑过自行车，那么当你第一次骑的时候你一定会摔倒。你可能对自行车无所不知，熟悉骑车的机械原理，知道哪种自行车最好，但是直到你将自己所学的东西用于实践，你才算真正学会了骑自行车。

同理，许多软件开发人员拿起一本关于编程语言或框架的技术书，从头到尾地读上一遍，就想奢望他们能够吸收其中的所有信息吗？最好的情况可能是借助这种方法，你能快速积累该主题的全部信息，但你仍然没有真正学会它。

自我教育

如果你想学习一些东西，你应该做什么？好了，刚才我们讲过了，最好的方法就是付诸于实践，如果你也能承担将自己所学的内容教给别人的任务，那么你会理解得更深刻。所以，你在自我教育方面的努力，应该聚焦在如何让自己切实参与，并且尽早付诸实践。

我觉得学习知识的最好方式就是立即将其用于实践，即使你还不知道自己在做什么。如果关于某个主题你能够获得足够的知识能够操作，你就可以发挥自己心灵深处强大的创造力和好奇心。当我们能够在一件事情上尽情发挥的时候，我们的内心就更倾向于吸收更多的信息，思考更有意义的问题。

这似乎有些奇怪，但事实的确如此。玩耍是一项强大的学习机制，这一点我们在整个动物王国都看得到。动物的幼崽总是贪玩，通过玩耍，它们学会了许多赖以生存的重要技能。你看过小猫学捉老鼠的过程吗？人类也是通过玩耍、主动操作来学习的，即使在对我们所做的一无所知的时候。

再举一个例子。在我小的时候，我经常玩一种名叫“万智牌”的集换式卡牌对战。我被它深深吸引，乐此不疲。在游戏中，为了击败对手，你需要综合自己的智慧、运气和创造力，这让我非常着迷。

就凭这一点，我记住了游戏中出现的成千上万张卡牌。你随便抽一张牌，我都可以告诉你这张牌有什么属性、派什么用场。（就算到现在，我依然可以说出大多数牌的信息。）你以为我是在那里正襟危坐，努力背下这成千上万张牌吗？不，我不需要那么做。我只是玩，并且乐在其中。这种自然的探索和好奇心帮助我轻而易举地记住了那么多的信息。

通过发挥这样的能力，玩成为一个你可以大加利用的强大武器。它不仅能激励你，还能大大加快你的学习步伐。在你阅读某个主题的书之前，大致浏览后就开始实践。不用担心自己是否知道自己在做什么。乐在其中，你就会发现随着自己的实验和探索，自己在哪些方面发现了问题。

一旦你已经实践过，并积累了各式各样的问题，立刻回到书本当中。当你回去重读这些参考资料的时候，你有强烈的冲动去消化吸收其中的内容。因为你已经积攒了很多想要找到答案的问题，你对哪些内容更重要早已了然于胸。

然后，你可以把自己学到的新知识重新应用于实践。看看你学到的新方法是否能解决你已有的问题。你可以继续探索新领域，发现需要解决的新问题。以解决实践过程中发现的问题为目标，在向着知识前进的道路上重复这个循环，周而复始。通过这种方式获得的信息对你才是有意义的——“纸上得来终觉浅”啊。

最后，你可以将自己所学的打包教给别人。这一点是画龙点睛之笔，你应该随时准备与有兴趣倾听的人分享你新学到的知

识，你会为自己的发现激动不已——玩儿的力量就是这么巨大。教导他人也很容易，它可以简单到你与配偶之间就自己学到的新东西进行的一次对话，也可以是写一篇博客文章。关键在于，你要用自己的语言将这些信息组织起来，把你的思想表述给别人。

这就是我发明的“十步学习法”背后的逻辑。在接下来的几章中我将就此展开详细的介绍。我增加了一些正式步骤，也引入了预备步骤来帮你在开始之前就让学习系统化。不过本方法的关键指导原则是通过玩儿、探索以及将自己所学教给他人来学习。这一简便易行的方法更符合我们的天性——在某种程度上，抛弃了“填鸭式教学”的自主学习才是最简单和最纯粹的学习方式。

采取行动

- 你最近一次自学了什么？你的学习过程是怎样的？
- 你上一次对兴趣爱好感到激动是在什么时候？你对此兴趣爱好了解多少？你是经过刻苦学习才培养此兴趣的，还是通过玩耍自然而然地就学会了？

第28章 我的“十步学习法”

多年以来，我都承受着巨大的压力：快速学习新技术、新编程语言、新框架和其他能力。通常，这种压力是我自己造成的，我总是投入新事物，结果力不从心。但是，即便不考虑压力来源，这也迫使我开发出一个可重复使用的自学体系。

在接下来的几章里，我会带你了解我自创的进行快速学习的“十步学习法”。让我们先从了解这个体系的确切含义及其工作原理开始。

体系背后的逻辑

在我职业生涯的早年间，我学习知识的主要途径就是“从封面到封底”仔细阅读专著。只有通读全书后我才会将自己学到的知识应用于实践。使用这个方法，我发现，我确实能学到东西，但是效率很低，我还得经常回顾书的内容，来弥补自己在该学科方面存在的知识短板。

当我拥有足够的时间，且没有一个真实具体的目标的时候，这种学习方法很好。我最终学会了我想要学习的东西，而且从头到尾地读书学东西也并不难，只是要花时间。随着我开始有更紧迫的理由需要快速学习，我发现自己原来的方法就无法奏效了。通常，我并没有时间通读全书，而且我也发现书本里的很多内容更适合作为参考资料，而不适合实际学习。

于是，我迫使自己去寻找更好的自学方法，能在有限的时间内掌握所需内容。有些时候，我只有一周甚至更少的时间去吸收足够的信息以教给别人。我发现在这种情况下，我很自然地会先明确需要掌握哪些内容，再去寻找我能获得这些信息的最佳资源，同时也会忽略那些并非达成目标所必需的其他信息。

我发现，为了能够掌握一门技术，我需要了解以下三个要点。

1. **如何开始**——要想开始使用自己所学的，我需要掌握哪些基本知识？

2. **学科范围**——我现在学的东西有多宏大？我应该怎么做？在开始阶段，我不需要了解每个细节，但是如果我能对该学科的轮廓有大致的了解，那么将来我就能发现更多细节。

3. **基础知识**——不止在开始阶段，要想使用一项特定的技术，我需要了解基本的用户案例和最常见的问题，也需要知道自己学的哪20%就能满足80%的日常应用。

熟知了这三个关键点后，我可以高效地学习一门技术，无需通晓全部细节。我发现，如果我了解三项主题——如何开始，我能做什么，以及相关基础知识，那么我就能随着学习深入学会所需的其他知识。如果我想提前掌握所有知识，那只是在浪费时间，因为真正重要的内容会湮没在那些细枝末节中。这种新方法能让我关注重点。当我确实需要了解更多细节时，我可以利用参考资料来弥补这些不足。有多少次你从头到尾仔细阅读一本技术书籍，却发现自己实际用到的也只是书里介绍的技术的一小部分？

使用这种方法，我在很短的时间内学会了Go语言——仅仅几个星期而已。我专注于学习如何尽快用Go语言写代码。很快我就对这门编程语言以及它有哪些可用的库有了一个大致的了解。我希望对这门语言能做什么能有一个整体的了解。最后，我完成并掌握了基础知识。当我需要深入了解时，我只需要在这些基础知识的基础上进行扩展。

十步学习法体系

事实证明，掌握这三大要点可不像看上去那么轻松。学会“如何开始学习一门技术”可以说是一项挑战，而且往往很难找出那些所谓的“篇幅只占20%、然而有效性却达到80%”的内容。另外，我经常发现，要总结出一门技术的应用广泛程度的简短描述，是很难的。通常你需要阅读一整本甚至几本不同的书籍之后才能获得此信息。

为了解决这些问题，我又提前做功课，确保自己能够找出自己所需的信息，并以最合理的方式将它们组织起来，实施落地。

“十步学习法”的基本思想就是：要对自己要学的内容有个基本的了解——了解自己不知道什么就足矣。然后，利用这些信息勾勒出学习的范围，即需要学哪些内容，以及学成之后又会获得什么。依靠这些知识，你可以找出各种资源（不局限于书）来帮助自己学习。最后，你可以创建自己的学习计划，列出要去学习哪些相关课程，筛选学习材料，只保留能帮助自己达成目标的优质内容。

一旦完成这些工作，你对自己要学什么和怎样学都了然于胸，你就可以把控自己的学习计划中的每个关键点，通过“学习—实践—掌握—教授”（Learning, Doing, Learning and Teaching, LDLT）的过程，获得对该学科的深刻理解，同时你也向着自己的目标前进。

“十步学习法”的第一部分是研究，它是一次性完成的。但是从第7步到第10步则是重复的过程，贯穿于你的学习计划的各个模块。这个方法非常奏效，因为它迫使你提前明确自己的学习目标，也持续不断的激励你通过实践（而不仅仅是读书或听讲座）向着目标前进。



“十步学习法”体系

借助此方法，我得以在几天之内就全面掌握了一门编程语言。成千上万注册了我的“十步学习法”视频课程（<http://simpleprogrammer.com/ss-10steps>）的其他软件开发人员也取得了类似的成果。

这是快速学习的唯一途径吗？这是一种魔法系统吗？不，不是的。它只是通过“聚焦学习范围，让你关注重点内容”来学习的一种快速学习方法，它迫使你通过“自我探索”和“主动教学”的方式将真正重要的内容印刻在自己的大脑中。接下来的几章，我们将介绍该方法的实际操作步骤。你可以根据自己的情况对该方法进行裁剪，去掉你不喜欢或者认为无效的部分，保留对你有用的部分。最终，你一定会找到适合自己的自学方法，而你的未来将仰仗于它。

采取行动

挑选一项你已经烂熟于胸的技术，看看你能否明确以下几点。

- 如何开始用它？
- 该技术的应用广度如何？
- 利用你需要知道的20%发挥出80%功效。

第29章 第1步到第6步：这些步骤只做一次

对于“十步学习法”的前六个步骤，你需要集中精力完成足够多的前期调研，确保自己明确知道要学哪些内容，以及如何确认自己已达成目标。你还将学到如何挑选最好的资源来帮自己实现目标、制订学习计划。

这六个步骤只需要针对你想学的每个主题做一次。第7步到第10步则要针对在第5步所制订的学习计划中的每个模块循环往复。尽管第1步到第6步只需做一次，但是它们却是最重要的步骤，因为它们将对你未来的成败起决定性作用。在这六个步骤中，你要为自己实际的学习主题做好一切准备工作。“不积跬步无以至千里”，基础打得越牢固，目标越容易实现。



第1步到第6步只做一次

第1步：了解全局

学习始终是一项棘手的任务，因为在开始学习某些东西的时候，你对自己到底要学什么理解得并不透彻。美国前国防部长唐纳德·拉姆斯菲尔德曾说过“未知之未知”（unknown unknowns），即你根本不知道自己不知道。

大多数开发人员在打开一本新书开始阅读的时候，他们对自己所不知的一无所知。他们将“未知之未知”留到后面去发现。这一方法的问题在于，你要么学非所需，要么力所不及。在深入探究某个主题之前至少要对其有所了解，这一点非常重要。这样你才能弄清自己到底要学什么，找出最好的学习方式。

在这一步，你要做的就是了解自己将要学习的主题的全局。这个主题宏观上什么样？你能从中学到足够丰富的知识以了解自己所不知道的吗？以及自己所不知道的有多少？

假设你想学习数码摄影。你可能会先在网上搜索与这个主题相关的内容，浏览与数码摄影有关的博客和文章。短短几小时之内你就能对这个主题的全局及现有的子课题有一个清晰的认识。

要完成这一步，你需要对自己想要学的课题做一些基础性研究。通常你可以使用网络搜索来完成大部分研究。如果你碰巧有一本关于该主题的书，那么你就可以只读一下其中的介绍性章节，粗略浏览一下内容，但是不要在这一步上花费太多时间。记住，我们在这一步的目的不是要掌握该主题，只是对这一主题的相关内容有一个全局性的了解。

第2步：确定范围

现在，你至少对自己的学习主题及其全局有了一个大致的了解。下一步就是集中精力去明确自己到底要学什么。在任何项目中，明确项目的范围都是至关重要的，唯有这样才能了解项目的全局，做好相应的准备工作。与此相比，学习并没有什么不同。

让我们继续你要学数码摄影这一例子。此时，你想要理解的是“这一主题到底有多大”，以及“如何将其分解为更小的范围”。在一定的时间内，你不可能掌握关于数码摄影的一切知识，所以你需要决定学习的重点和学习的范围。如果你想了解如何拍摄人像照片，那这就是你的学习范围。

在学习过程中，大家很容易犯的一个错误就是试图解决太大的问题而把自己搞得不堪重负。例如，试图全面掌握物理学是不切实际的，因为这一主题太过庞大，也不够聚焦。你不可能在有限的时间里学会与物理学有关的一切——穷尽一生也不可能。因此，你要明确自己的学习范围。为此，你需要运用自己在上一步中获得的信息，让自己的关注点落脚到更小也更可控的范围。

让我们看看表29-1，看如何将庞大的主题分解为小而聚焦的主题。

表29-1 将一个大的主题分解成可控的子主题

初始的主题	可以分解为子主题
学习C#	学习C#语言的基础知识，掌握如何创建一个简单的控制台程序
学习摄影	掌握针对人像拍摄的数码摄影知识
学习Linux	了解如何设置和安装Ubuntu Linux，以及如何使用它的基本特性

请注意，在上面的例子中，是如何将类似“学习C#”这样的宏大主题缩小并聚焦到一个特定范围内的。我们从一个几乎无边界的主题中明确了一个清晰且聚焦的范围。你还会注意到，在这一步中，我们还为学习添加了一个理由，从而将其限定在一个范围明确的主题内。例如，你想学摄影，特别是学数码摄影，目的是能够拍摄人像。阐述学习的理由能够帮你明确学习范围，因为人们通常是为一些特殊的理由才去学一些东西的。

在这一步中，你可以充分利用自己在第1步中收集到的信息，找出自己的学习范围。同时也可以借助自己的学习理由来决定学习的范围。

在此过程中，你可能会受到诱惑，为了学习该主题下的不同子主题，你可能会扩张你的学习范围而不够聚焦，但是请务必抵制住这个诱惑，尽可能地保持专注。你一次只能学一样东西。你可以稍后再回头学习别的分支领域，但就目前而言，选择一个专注而聚焦的范围，潜心学习吧。

最后，在这一步中一定要注意：明确学习范围的时候要考虑时间因素。如果你只有一周时间，你需要本着实事求是的态度确定自己在这段时间内学到什么。如果你有几个月的时间，你也许能攻克一个更大的主题。你的学习范围务必大小适当，既能符合你的学习理由，又能符合你的时间限制。

第3步：定义目标

在全力以赴启动之前，明确“成功”的含义极为重要。如果不知道成功是什么样子，很难找准目标，也很难知道自己什么时候已经真正达到目标。在尝试学习任何东西之前，你都应该在自己脑海中清晰地描绘出成功的样子。当你知道自己的目标是什么的时候，你就可以更轻松地使用倒推的方式，明确实现目标所需的步骤。

现在，我们依然使用刚才那个学习数码摄影的例子。你可能已经明确成功的标准包括掌握数码相机的所有功能，能够清晰地描述这些功能是什么，并且了解何时以及为什么使用每一项功能。

这一步的目标是形成一份简明清晰的陈述，勾勒出你勤奋学习后的成功图景。根据不同的学习内容，这份陈述也各不相同。但是要确保其中包含具体的成功标准，从而能让你用来充分评估自己是否已经达成学习目标。

好的成功标准应该是具体的、无二义性的。不要对自己想要完成的任务进行含糊不清的描述。相反，要列出某一特定的结果，或者一旦实现自己所能达到的目标你应该能够做到的事情。表29-2展示了一些示例。

表29-2 “好的”和“不好的”成功标准

不好的成功标准	好的成功标准
我可以用我的数码相机拍出好照片	我可以使用我的数码相机里的所有功能，能够清晰地描述这些功能是什么，并且知道何时何地使用各功能
我学习了关于C#语言的基础知识	我可以利用C#语言的主要功能写出一个小的应用程序
我知道如何使用HTML去构建一个网页	我可以使用HTML5在网上为我自己创建一个主页，展示我的简历和我的代表性工作

你想从自己的学习经历中获得什么决定了你的成功标准是什么。请确保你能借此在学习结束后评估自己是否达成了目标。好的成功标准也能让你向着既定目标不断前进。

第4步：寻找资源

还记得学生会针对某个特定主题写报告吗？如果你写好了报告却只有一条参考文献，譬如你所有的信息都来自一本书，结果会怎样？你可能会拿到一个大大的“鸭蛋”。为什么现在很多人学东西的时候却与那时如出一辙？关于一个主题我们只读一本书，或者所有的研究只使用一个资源。

继续以数码摄影为例。你可能会从阅读相机的用户手册开始，但是不会就此止步。你可能还会查找各种专业的数码摄影的网站，甚至是自己所用的相机的网站。你也可能会在亚马逊网站上搜索数码摄影的相关图书，最后可能找到能寻求建议的专家。

[class="sigil_index_marker"] { color:#335500; }要尝试收集到多种多样的资源以帮助你学习，而不是只读一本关于这一主题的书。资源可以是多种多样的，不局限于书籍。事实上，如今，随着互联网的广泛应用，各种类型的内容随处可见，你几乎可以针对自己感兴趣的任何主题，找到大量的资源。

在这一步中，你会想找到尽可能多的与自己所选主题相关的资源。此时你无需考虑这些资源的质量。这一步与头脑风暴类似。稍后你会对你找到的这些资源进行过滤，去伪存真，但是目前还是想先获得尽可能多的不同类型的资源。

要做到这一点，最好的方法就是迅速打开电脑，开始搜索与自己的主题相关的信息。我通常会从亚马逊开始，看看能从中找到多少相关图书；然后我会用Google搜索，看看能不能找到视频、博客文章、播客和其他有用的内容。你甚至可以像“老派人”那样去拜访图书馆。重点就是你要找到各种不同的资源。你不想因为单一来源的信息而产生偏见，更愿意尽可能获取各种各样的信息。

信息来源

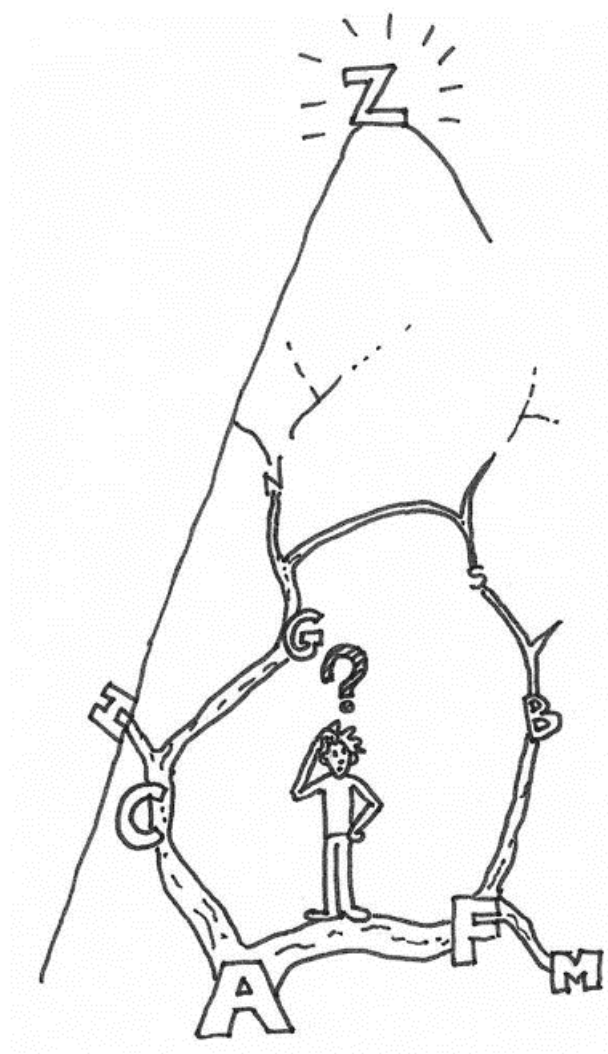
- 图书
- 博客文章
- 在线视频
- 专家，或者对你所想要学习的内容已经熟知的人
- 播客
- 源代码
- 示例项目
- 在线文档

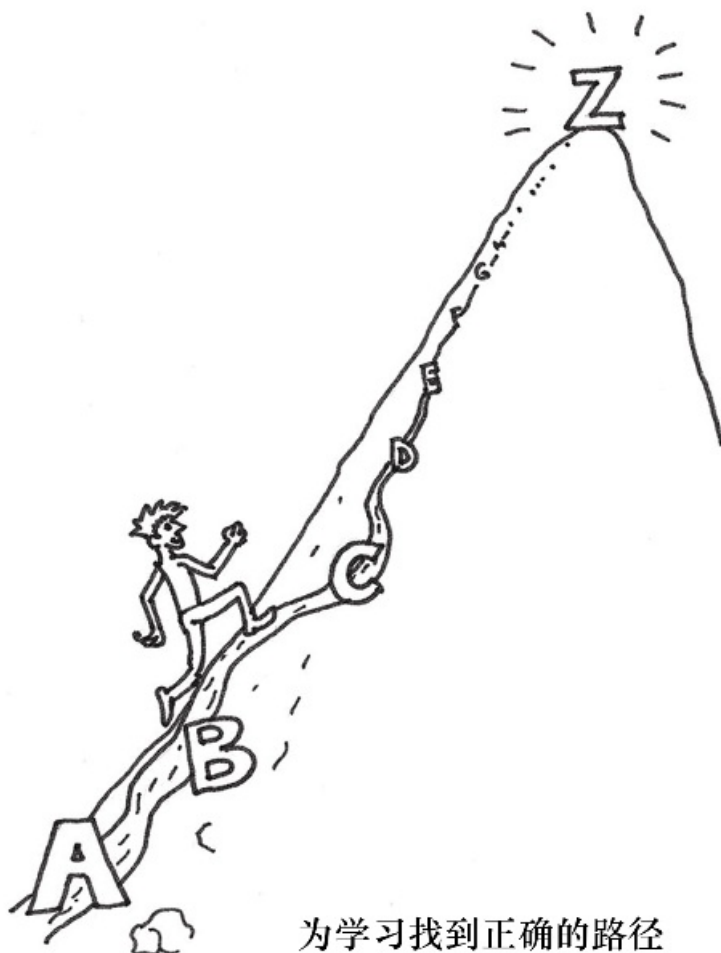
第5步：创建学习计划

你有没有注意过，大多数书籍都被分解成若干章，各章在内容上又层层推进？好的技术书都遵循这样的规律：打好基础，做好铺垫，然后逐个展开每一章的论述。

现在你已经掌握了一些资源，你可以借助这些资源对自己要学什么、以什么顺序进行有了想法。现在你应该对数码摄影到底要学什么有了很好的认识。你需要通览已有的数码摄影的相关资料，找到一种方法将这个主题分解成更小的部分。

对于大多数学科而言，学习是一个自然的过程。从A开始，前进到B，最后到达Z。这个顺序对你掌握随机的碎片化知识价值不大。你需要找出在最短的时间内从A到Z的正确路径，并且到达沿途的重要地标。





为学习找到正确的路径

为学习找到正确的路径

在这一步，你需要创建自己的学习路径。把它看作自己写书时候的大纲。事实上，当你大功告成的时候，你会发现自己的学习路径可能与一本书的目录非常相似。你基本上就是完成一系列的模块，直到最后达成目标。

打造自己的学习计划，一个好方法就是观察别人是如何教你感兴趣的主题的。就我自己而言，在这一步我通常会翻看自己在第4步中找来的图书的目录。如果五位不同的作者都把内容都分解为相同的模块和顺序，那我就遵循这样的方法制订自己的学习计划。

这并不意味着你只要复制一本书的目录就可以称其为自己的学习计划了。有的书的内容超出你的需求，有的书的结构很差劲儿。通览你收集到的全部资源，你就对自己需要哪些内容以及如何组合这些内容有更清晰的认识。

第6步：筛选资源

现在，你知道自己要学什么以及以什么顺序学，那么是时候决定要使用哪些资源来完成自己的学习任务了。回到第4步，你收集了与研究主题相关的所有资源。在第5步中你使用这些资源制订了自己的个人学习计划。现在是时候对这些资源进行筛选，挑选最有价值的几项来帮你实现自己的目标。

这时的你已经有了大关于数码摄影的书籍、博客文章和其他资源等，不一而足。但是，问题是你该如何利用这些资源。很多数据都是冗余，并非所有资源都适合你的学习计划。

为了研究一个主题要读10本书、50篇博客文章，并不现实，并且即使你真的这么做了，其中也有很大一部分是重复的。因此很有必要对现有资源进行筛选，从中选出最能帮助你实现自己的目标的资源。

这样想一想：在第6步中，你就是一位篮球教练，你需要精简队伍。当然，你想让所有人都去打比赛，但这显然是不可能的。你不得不将队员人数减少到自己可以管理的规模。

在这一步中，把你在第4步中收集的全部资源浏览一遍，找出哪些内容能够覆盖你的学习计划。你还应当看看评论，试着找出品质最高的资源。我在购书时，通常会浏览亚马逊上的评论，找出我认为最物超所值的一两本书。

一旦完成了这一步，你就可以准备前进到学习计划中的第一个模块。在你实现自己的目标之前，你还需要为每个模块重复第7步到第10步。

- 挑选你想要了解的一项课题，实际经演练一下上述这六个步骤。你可以从一些规模较小的课题开始，以便让自己习惯于这一过程。但是注意，一定要实际运用。如果你只是把这些步骤当作是阅读内容，那么它们对你不会有太大用处。