

自我营销的方式：博客 播客 视频 给杂志投稿 书籍 代码营 技术会议

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。

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。

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。

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。

个人推荐“技术简历服务”，<http://simpleprogrammercom/ss-resumewriter>，别忘了报本书的名字。)

，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

“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。

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

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。

试着加入本地的软件开发者小组，每周或者每月参加他们的聚会。换个场景，到咖啡馆去工作

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，我们会沉浸其中，也有动力继续。我把这称为冲量效应[1]**。

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

要更好地了解激励，可以查看Daniel Pink的《驱动力》（Drive）[Riverhead Hardcover, 2009] 一书。

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，

你可以在<http://wwwf6scom/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://wwwycombinatorcom>)。Y Combinator帮助过许多著名的创业型企业，如Dropbox。

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://wwwf6scom/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://wwwycombinatorcom>)。Y Combinator帮助过许多著名的创业型企业，如Dropbox。

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://wwwf6scom/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://wwwycombinatorcom>)。Y Combinator帮助过许多著名的创业型企业，如Dropbox。

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

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。

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。

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

软件开发人员如何自我营销”的课程（<http://devcareerboost.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

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

在社交媒体上发布消息，给名单上的一小部分人发送邮件，看看会有什么反馈。一旦得到反馈，更新你的邮件模板，发送给更多人

制作一个电子邮件模板，使用该模板给上面那份名单里的每一个人发邮件

关于如何收费的最后一条建议是，如果没有任何潜在客户跟你讲“不行”或者“你的收费太贵了”，马上提高费率！不断给你的收费加码，直到你听到“不行”为止。客户愿意为你的服务付的价格可能会让你大吃一惊

我提供的咨询服务中有一项是关于创建自动化测试框架的。当我和潜在客户就这项服务议价的时候，我会列出构建自动化测试框架需要多少钱，也会说明如果软件出错的话修复的成本有多高昂。然后，我会讲自己在构建自动化测试框架方面的丰富经验，并且我确切知道要做什么。我还会向潜在客户展示，如果以每小时300美元的价格雇用我其实要比雇用一个以前没有写过自动化测试框架的普通开发人员便宜得多。我会告诉他们，我1小时的指导可以节省他们20小时的也许是南辕北辙的工作。我没有撒谎。投标之所以如此高效，是因为我真的相信这是真的。关键点在于，我强调自己的服务完全值这个价钱

全职雇员的薪酬	自由职业者的薪酬
每小时50美元 - 无	每小时100美元 - 自主就业税 - 办公设备 - 会计或者记账员 - 非计费时间
实际收入=每小时50美元	实际收入=每小时50美元

自主就业税”——是的，这是政府因为你为自己工作而额外征收的费用。

吸引式营销

国内通常需要提前一个月提出离职申请

你的劳动合同中规定你所创造的一切都属于你的公司（哪怕是在自己的业余时间做的

推荐一本关于这个主题的好书——Cal Newport的So Good They Can't Ignore You [Business Plus , 2012]

如果想辞职为自己打工，需要一个切实可行的计划。你必须要想方设法积攒够足够多的收入来支撑自己，直到你的新生活步入正轨。

许多软件开发人员犯的一个巨大错误就是，当无法达到标准时，他们不是完善自己、迎接挑战，而是降低标准。

一篇关于“如何说不”的很好的文章（<http://simpleprogrammercom/ss-no>） ，

一些专业人士和外行的差异

专业人士	外行
遵守自己的原则	让干什么就干什么
专注于正确完成工作	专注于完成工作
不惧怕承认自己错了，不会文过饰非	不懂装懂
持续稳定	无法预测，不可靠
勇于承担责任	回避责任

对于你的老板和经理，你现在的“曝光度”如何？下一周，你可以采取何种具体方式来提升自己的“曝光度”？目前你正在自己学习些什么？确定要自学的最有价值的东西是什么，制订一份下一年的自学计划

不要只学软件开发。如果你把目标设定为更高级别的岗位甚至是行政岗位，你还需要学习领导力、管理和商科的相关知识

我当然推荐主动发送周报，不过还有其他许多方式能让你在所在的机构中更加引人注目。其中最好的一种方法就是做一个关于团队当前正面临的主题或者问题的演讲。选一个自己能介绍的主题，然后向团队展示这一主题。你甚至可以做成“午餐+学习”的形式，在午餐时间进行分享，

而且随着时间的推移，你还能在团队中逐步树立“及时雨”的名声。最终，这样的声誉可能会令你成为团队领导或者其他管理职位，只要你愿意走这条路。

在接受工作之前与为这家公司工作的开发人员聊一聊是个好主意，你能够更真切地体会在这家公司工作的感受。

表8-1

在小公司、中等公司和大公司工作的一些优点和缺点

公司规模	优点	缺点
小公司	承担多种职责，灵活的角色转换，对工作的影响较大，回报潜力大	没法安静地坐下来写代码，工作在众目睽睽之下，稳定性差
中等规模公司	工作稳定，加班较少，变化很慢低下，没有机会使用前沿技术	
大公司	完备的流程和规范，培训机会多，大型有影响力的项目	充斥着官僚主义作风，可能只负责代码库的一小部分，很难得关注

在中等规模的公司里，缓慢而稳健的做事风格通常能占得先机。

中等规模的公司往往要比大公司还稳定，因为大公司往往还有大裁员或者周期性重组。如果你喜欢稳定，那你会发现中等规模的公司最适合你。

从长期发展和工作环境的角度去评价可能对你更为重要。

专业领域: Web开发栈, 嵌入式系统, 特定的操作系统, 移动开发, 框架, 软件系统

律师这一行术业有专攻，他们通常在从业一开始就走上了专业之路，成为刑事律师、交通事故律师或者房地产律师等

你要负责税务、法律咨询、销售、健康保险以及各种与企业经营有关的事情。

独立咨询师通常有自己的公司，依照合同为客户工作，但并不与任何一个客户绑定。

我宁愿雇用这样的开发人员：知道的东西可以少一点，但是明确知道要做什么，以及怎样去做。

在我的例子中，我找到了已经在这家公司工作的开发人员的博客，并与他们建立了联系。

我开始关注为这家公司工作的开发人员的所有博客，并开始在他们的博客文章下留下经过深思熟虑的评论。慢慢地，许多在这家公司供职的开发人员开始认识我了，并通过我给他们博客做出的评论知道了我是谁。他们中的一些人甚至开始读我的博客

如果你意识到某个人就是所谓的“苛性碱”，不要试图去改变他们，也不要试图去和他们打交道，就让他们停留在自己的轨迹上，你所要做的只是尽量限制自己与他们互动。你会发现“苛性碱人”所过之处痕迹明显，他们似乎总是卷入某种形式的悲剧之

中，总是有不幸的事件发生在他们身上，他们老是把自己扮成受害者。如果你发现这种迹象，赶快跑——有多快跑多快。

：尽管我们为自己的智慧感到骄傲，但我们依然是情感动物。我们就像那些穿着西装、打着领带、四处游荡的小孩，假装自己已经长大，其实任何轻微的伤害都能让我们号啕大哭，或者大发雷霆，我们只是已经学会了如何控制和隐藏这些情绪。

更好的办法是从对方的心态出发提出建议，阐明为什么采用你建议的方法实现该功能对老板非常有用。理由可能是“让软件更稳定”，或者“能让软件按时交付”

人性的弱点

[笔记]这是一本好书

如果我想航海去中国，我没必要立刻知道要驶向的港口的精确的经纬度。我可以跳上船，向着中国的航向前进。随着离目标越来越近，我就能获得更具体的数据。启航后我需要的就只是自己离中国越来越近，还是越来越远。

没有明确的方向，你走的每一步都是徒劳的。不要随心所欲地生活，不要随遇而安地行走在职业生涯的漫漫长路上

大多数成功的公司都会开发出让客户主动上门购买的产品或服务，它们才不会一个接一个地追逐客户

作为一个软件开发人员，你只有真正专注于一类客户，才能找到非常好的工作

产品营销做得越好，你就能给服务定越高的价格，也越有机会吸引更多潜在的客户。

从被一纸“卖身契”束缚住的仆人转变为一名拥有自己生意的商人。在起步阶段就具备这种心态会改变你对职业生涯的思维方式，将此铭记在心，并积极主动地管理自己的职业生涯

事实上，许多真正的公司也就只有一个大客户，这个客户给它带来了大部分的收入

因为只有你开始把自己当作一个企业去思考时，你才能开始做出良好的商业决策

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

我最喜欢的一本书——Napoleon Hill的《思考致富》（Think and Grow Rich

那么在公司内部给自己寻找一位导师无疑是明智之举。你的老板或者你老板的老板这样的资深人士是导师的不二人选，你很可能会提前接受晋升所需要的各种教育。此外，与高管做朋友对你的职业生涯毫无害处。

的老板或者你老板的老板这样的资深人士是导师的不二人选，你很可能会提前接受晋

升所需要的各种教育。此外，与高管做朋友对你的职业生涯毫无害处。

你最好的选择就是去自己认识的人中找，自己的朋友的朋友、家人的朋友等。如果你愿意做一点儿功课，再四处打听一下，无论你努力追求的目标是什么，你极有可能在由家人和朋友组成的关系网中找到适合做你导师的人。

如果一个人能够对很多人产生好的影响，能够帮助他们实现目标，那么他也更有可能为你做同样的事情。

有很多你以为自己明白了的知识点，其实并没有摸透。于是你会将那些以前自己没太明白的东西联系起来，并且简化自己大脑中已有的信息，将它们浓缩并经常复习

你在第3步中定义的成功标准。试着把自己正在学习的内容与最终目标关联起来。你掌握的每个模块，都应该以某种方式推动你向着终极目标前进。

我们很少能有足够的时间把一本书从头读到尾。这些资料只是帮你自学，基本上你可以以解决在动手操作中发现的问题为主要目的。

假设你已经在调试光圈的过程中积累了一些问题，这时你可以通过阅读相关主题的资料来回答这些问题。你可以通览已有的资料，仔细查找与光线和其他通过操作发现的问题有关的内容。

假设你已经在调试光圈的过程中积累了一些问题，这时你可以通过阅读相关主题的资料来回答这些问题。你可以通览已有的资料，仔细查找与光线和其他通过操作

骤7到步骤10的目标是通过“学习—实践—掌握—教授”（LDLT）的方式真正领会知识你还应当看看评论，试着找出质最高的资源。

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，试着找出质最高的资源。

果五位不同的作者都把内容都分解为相同的模块和顺序，那我就会遵循这样的方法制订自己的学习计划。

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。

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。

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。

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

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。

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。

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，

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

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。

这样你才能弄清自己到底要学什么，找出最好的学习方式

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

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。

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。

，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地了解

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。

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。

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

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

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

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

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。

你的生存之道就是不必太在意。别害怕被人看作傻瓜

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

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。

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，你现在就开始写并坚持下去，随着时间的推移你的写作技能终将会得到提高

可以与像Pluralsight (<http://simpleprogrammercom/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，

好的，有像Toastmas (<http://www.toastmastersorg/>) 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。

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发表演讲。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

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

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

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

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

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

在博客文章的结尾呼吁人们在Twitter上关注你，无伤大雅。

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。

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是更加轻松自如，而当他们真遇到障碍的时候，总是会有好多人伸出援手。

你要把它视为对未来的一项投资。

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。

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，如果你表达的和你传递的信息不能帮到其他人，那么每个人都将会无视你。

不要努力成为一个成功的人，而要努力成为一个有价值的人。——阿尔伯特·爱因斯坦

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://newsycombinator.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。

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

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。)

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。

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

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。

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。