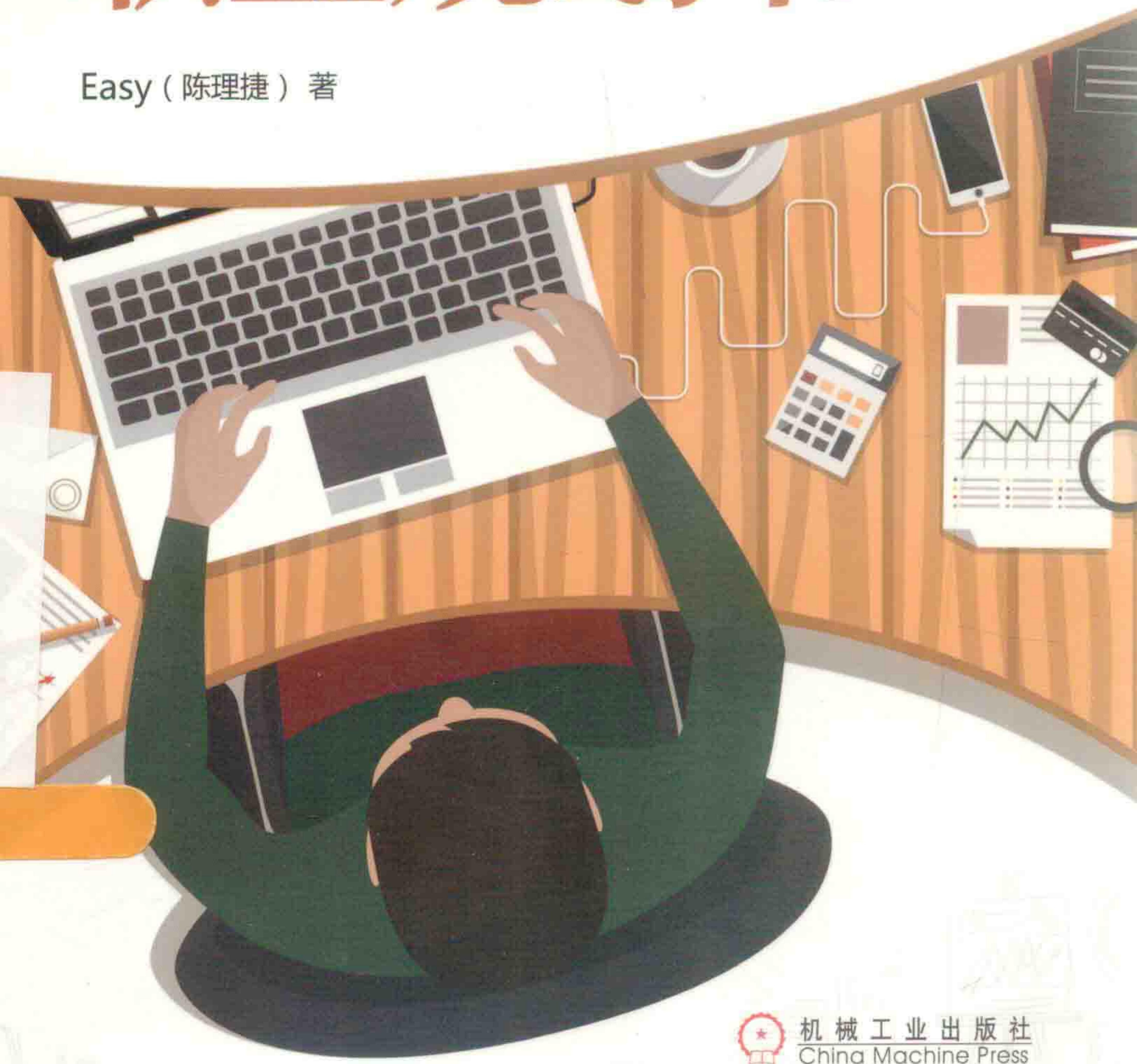




高端人才拍卖网站JobDeer.com CEO匠心打造  
带你绕过程序员路上的那些“坑”

# 程序员必读的 职业规划书

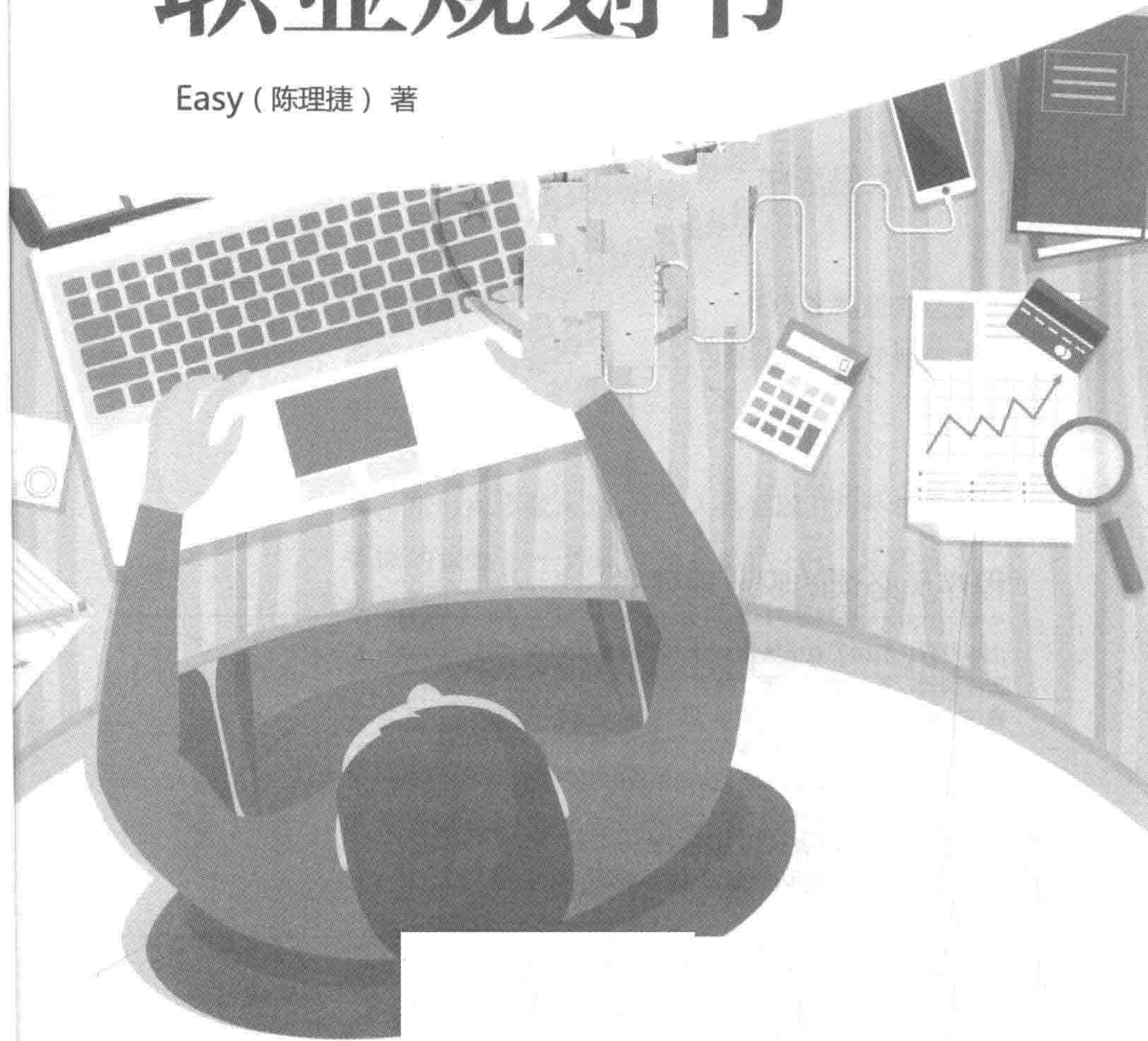
Easy (陈理捷) 著



机械工业出版社  
China Machine Press

# 程序员必读的 职业规划书

Easy (陈理捷) 著



## 图书在版编目 (CIP) 数据

程序员必读的职业规划书 / 陈理捷著 . —北京：机械工业出版社，2015.4

ISBN 978-7-111-49754-7

I. 程… II. 陈… III. 程序设计—工程技术人员—职业选择 IV. C913.2

中国版本图书馆 CIP 数据核字 (2015) 第 059222 号



## 程序员必读的职业规划书

出版发行：机械工业出版社（北京市西城区百万庄大街 22 号 邮政编码：100037）

责任编辑：王彬 秦健

责任校对：殷虹

印 刷：三河市宏图印务有限公司

版 次：2015 年 4 月第 1 版第 1 次印刷

开 本：147mm×210mm 1/32

印 张：3.625（含 0.25 印张插页）

书 号：ISBN 978-7-111-49754-7

定 价：29.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

客服热线：(010) 88378991 88361066

购书热线：(010) 68326294 88379649 68995259

投稿热线：(010) 88379604

读者信箱：hzjsj@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版 本书法律顾问：北京大成律师事务所 韩光 / 邹晓东

## 为什么我们要自己做职业生涯规划？

记得电影《社交网络》里边，CFO 在知道自己的股权被稀释时说了一句话：“我以为那些律师是我的律师。”

其实我们大多数人对 HR（人力资源）几乎都存在类似的误解——你以为她是你的 HR，其实她只是公司的 HR。她们关心的是如何编个理由用老板给的那点钱留住一个高性价比的人才，而不是关注真正有助于你发展的职业路线图。昨天还含情脉脉和你讨论人生的知心姐姐明天就可能变成拿着劳动合同逼你主动离职的凶婆娘。和人性无关——这就是她的工作，越专业的 HR 越擅长。

所以，你要自己来做这件事情。

我曾经花了两个月时间专门研究求职这件事。

那是 2009 年，我关掉自己的工作室后，打算重新找份工作。因为之前工作室还算挣钱，所以不是很着急。

60 天时间里，我读了很多书，画了很多图，见了很多人，面了很多试。根据梳理好的节点，最后我拒了腾讯的 offer（录用通知书），去了新浪做云计算。

那时候 SinaAppEngine 团队只有我一个员工，国内还没有几家提供云服务的；2013 年 9 月我离开新浪时，SinaAppEngine 的日访问量已经超过 8 亿次，云计算已经成为主流技术。

找工作是件非常重要的事情，它直接影响你 1 ~ 2 年，间接影响你 3 ~ 5 年的人生。一个潜在的机会可以让你少奋斗很多年，而一次冲动的离职会让你和千万财富失之交臂。

每每看见优秀的程序员因为太专注工作，在一个位置长期得不到发展（很讽刺），我都想冲上去，告诉他应该如何求职、如何跳槽、如何规划自己的职业和人生。换工作不是一场说走就走的旅行，而是一个深思熟虑的结果，是一项复杂的系统工程。建议大家每次换工作花 1 ~ 3 个月（的业余时间）来准备，不要嫌麻烦，只要试一次，你就会知道这是值得的。

但我也是一个内向的前程序员，不擅长与陌生人聊天，于是我决

定写一本书。这本书从 2013 年年底开始规划，陆续花了一年时间。并不是因为内容太多，而是因为期间进行了大量的重写工作。到 2014 年 10 月，整个体系才算清晰，于是我开始补充内容。

说实话，我不能保证这本书里的观点、结论都是对的，但我详细描述了自己思考的方式、使用的工具，买鱼送渔，期望能对你有所帮助。

本书分为三个部分，第一部分介绍原理，第二部分介绍准备工作，第三部分介绍实践操作。各部分之间有较强的逻辑关系，建议依次阅读，循序渐进。

如果你喜欢本书，请推荐给你的朋友。

另外，感谢邓新星同学为本书绘制插图。

Easy  
2015 年元旦  
于北京中关村

## LeanOS ◎ 基础平台

SAE ◎ 基础平台  
BAE ◎ 基础平台  
GAE ◎ 基础平台

AWS ◎ 基础平台  
QingCloud ◎ 基础平台  
AliCloud ◎ 基础平台

OpenStack ◎ IaaS  
Docker ◎ 容器技术  
Ubuntu ◎ 发行版  
CentOS ◎ 常用Shell命令

vminstat ◎ 性能分析  
sar ◎ 服务配置  
dstat ◎ 运行环境

Perl ◎ 开发  
Lua ◎ 开发  
Nginx ◎ WebServer

PHP基础 ◎ 核心语法  
PSR规范 ◎ 规范  
Composer ◎ 包管理  
PEAR ◎ 包管理

ThinkPHP ◎ 框架  
Yaf ◎ Web  
Laravel ◎ 框架  
Yii ◎ API  
Swoole ◎ API

PHPUnit ◎ 单元测试  
SimpleTest ◎ 单元测试  
xdebug ◎ 调试工具  
SocketLog ◎ 框架

HHVM ◎ 可替代实现  
V8 ◎ 引擎  
Express ◎ 框架  
Koa.js ◎ 框架  
NPM ◎ 包管理  
Karma ◎ 测试工具  
Mocha ◎ 测试工具  
Tib.js ◎ 可替代实现

PHP扩展 ◎ Go  
Java ◎ Go  
PHP ◎ C  
C ◎ Go

MySQL索引 ◎ B树  
Redis ◎ 高级查找  
MongoDB ◎ NOSQL  
MySQL ◎ SOL  
PostgreSQL ◎ 数据库  
SQL注入 ◎ 安全  
Hadoop ◎ 并行分析  
Mahout ◎ 可替代实现

REST API设计规范  
OAuth ◎ HTTPS  
HTTP2.0/SPDY ◎ HTTPS  
SMTP ◎ HTTPS  
HTTP ◎ 协议

API设计规范 ◎ 第三方服务  
邮件推送 ◎ 第三方服务  
微博 ◎ 开放平台  
微信 ◎ 支付接口  
支付宝 ◎ 支付接口  
易信 ◎ 支付接口  
服务端 ◎ 域名

API-用于微服务自动切用环境  
API设计规范 ◎ 第三方服务  
邮件推送 ◎ 第三方服务  
微博 ◎ 开放平台  
微信 ◎ 支付接口  
支付宝 ◎ 支付接口  
易信 ◎ 支付接口  
服务端 ◎ 域名

前端、翻转 ◎ 正则表达式  
类型转换 ◎ 正则表达式  
状态机 ◎ 字符串  
最终一致性 ◎ 数组  
链表、链表 ◎ 数组  
散列、散列 ◎ 字符串  
堆、堆 ◎ 数组  
优先队列、优先队列 ◎ 数组  
哈希表 ◎ 数组  
LRU ◎ 数组  
压缩算法 ◎ 数组  
时间复杂度 ◎ 数组  
性能 ◎ 数组

MySQL索引 ◎ B树  
Redis ◎ 高级查找  
MongoDB ◎ NOSQL  
MySQL ◎ SOL  
PostgreSQL ◎ 数据库  
SQL注入 ◎ 安全  
Hadoop ◎ 并行分析  
Mahout ◎ 可替代实现

REST API设计规范  
OAuth ◎ HTTPS  
HTTP2.0/SPDY ◎ HTTPS  
SMTP ◎ HTTPS  
HTTP ◎ 协议

API设计规范 ◎ 第三方服务  
邮件推送 ◎ 第三方服务  
微博 ◎ 开放平台  
微信 ◎ 支付接口  
支付宝 ◎ 支付接口  
易信 ◎ 支付接口  
服务端 ◎ 域名

API-用于微服务自动切用环境  
API设计规范 ◎ 第三方服务  
邮件推送 ◎ 第三方服务  
微博 ◎ 开放平台  
微信 ◎ 支付接口  
支付宝 ◎ 支付接口  
易信 ◎ 支付接口  
服务端 ◎ 域名

前端、翻转 ◎ 正则表达式  
类型转换 ◎ 正则表达式  
状态机 ◎ 字符串  
最终一致性 ◎ 数组  
链表、链表 ◎ 数组  
散列、散列 ◎ 字符串  
堆、堆 ◎ 数组  
优先队列、优先队列 ◎ 数组  
哈希表 ◎ 数组  
LRU ◎ 数组  
压缩算法 ◎ 数组  
时间复杂度 ◎ 数组  
性能 ◎ 数组

MySQL索引 ◎ B树  
Redis ◎ 高级查找  
MongoDB ◎ NOSQL  
MySQL ◎ SOL  
PostgreSQL ◎ 数据库  
SQL注入 ◎ 安全  
Hadoop ◎ 并行分析  
Mahout ◎ 可替代实现

REST API设计规范  
OAuth ◎ HTTPS  
HTTP2.0/SPDY ◎ HTTPS  
SMTP ◎ HTTPS  
HTTP ◎ 协议

API设计规范 ◎ 第三方服务  
邮件推送 ◎ 第三方服务  
微博 ◎ 开放平台  
微信 ◎ 支付接口  
支付宝 ◎ 支付接口  
易信 ◎ 支付接口  
服务端 ◎ 域名

API设计规范 ◎ 第三方服务  
邮件推送 ◎ 第三方服务  
微博 ◎ 开放平台  
微信 ◎ 支付接口  
支付宝 ◎ 支付接口  
易信 ◎ 支付接口  
服务端 ◎ 域名

API设计规范 ◎ 第三方服务  
邮件推送 ◎ 第三方服务  
微博 ◎ 开放平台  
微信 ◎ 支付接口  
支付宝 ◎ 支付接口  
易信 ◎ 支付接口  
服务端 ◎ 域名

API设计规范 ◎ 第三方服务  
邮件推送 ◎ 第三方服务  
微博 ◎ 开放平台  
微信 ◎ 支付接口  
支付宝 ◎ 支付接口  
易信 ◎ 支付接口  
服务端 ◎ 域名

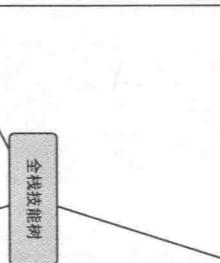
API设计规范 ◎ 第三方服务  
邮件推送 ◎ 第三方服务  
微博 ◎ 开放平台  
微信 ◎ 支付接口  
支付宝 ◎ 支付接口  
易信 ◎ 支付接口  
服务端 ◎ 域名

API设计规范 ◎ 第三方服务  
邮件推送 ◎ 第三方服务  
微博 ◎ 开放平台  
微信 ◎ 支付接口  
支付宝 ◎ 支付接口  
易信 ◎ 支付接口  
服务端 ◎ 域名

API设计规范 ◎ 第三方服务  
邮件推送 ◎ 第三方服务  
微博 ◎ 开放平台  
微信 ◎ 支付接口  
支付宝 ◎ 支付接口  
易信 ◎ 支付接口  
服务端 ◎ 域名

API设计规范 ◎ 第三方服务  
邮件推送 ◎ 第三方服务  
微博 ◎ 开放平台  
微信 ◎ 支付接口  
支付宝 ◎ 支付接口  
易信 ◎ 支付接口  
服务端 ◎ 域名

API设计规范 ◎ 第三方服务  
邮件推送 ◎ 第三方服务  
微博 ◎ 开放平台  
微信 ◎ 支付接口  
支付宝 ◎ 支付接口  
易信 ◎ 支付接口  
服务端 ◎ 域名



板本更新 ◎ 2014年10月27日版

## 目录 · CONTENTS

### 前 言

### 原理篇

- 价值论 /2
  - :: 使用价值 /2
- 供需 /5
- 信息透明度 /7
- 跳槽不是什么 /9
  - :: 跳槽不是为了追求价值最大化 /9
  - :: 跳槽不是找工作 /10
  - :: 跳槽不是加薪的砝码 /10
- 跳槽到底为什么 /13
  - :: 求职的本质 /13
  - :: 跳槽的意义 /14
  - :: 跳槽的原则 /14
  - :: 跳槽的节奏和路线图 /15

## 准备篇

- 自我认识和自我实现 /18
- 程序员职业的四大象限 /19
  - :: E 象限 /19
  - :: S 象限 /21
  - :: B 象限 /23
  - :: I 象限 /25
- 职业路线图 /26
- 市场需求的分析 /27
  - :: 定量分析 /27
  - :: 定性分析 /39
- 根据需求调整自己的定位 /40
  - :: 站在风口不一定能飞起来，但站在冰山上必然会沉下去 /40
  - :: 学会观察技术潮流 /41
  - :: 投资新兴市场和细分市场 /44
- JobDeer 职业画布 /46
  - :: 如何构架价值 /47
  - :: 如何传递价值 /48

:: 预估收益 /48
• 构建个人品牌 /50
:: 公司品牌不是个人品牌 /50
:: 一生相随的个人品牌 /50
:: 程序员如何做个人品牌 /52
:: 发展粉丝和虚荣无关 /53
• 学会沟通和写作 /54
:: 条理性 /54
:: 逻辑性 /55
• 走完分享的最后一公里 /57
:: 渠道 /57
• 开始你的开源项目 /59
:: 通过开源项目转型 /59
:: 开源项目不是遥不可及的 /60
• 提升架构能力 /62
:: DRY /62
:: 正交性 /63

## 操作篇

• 求职材料 /66
------------

• 简历的本质 /67
• 简历要说什么 /68
:: FAB /68
:: 给论据但别给论题 /68
:: 对比体现成长 /69
• 工具和模板 /71
:: 书写工具 /71
:: 简历模板 /73
• 求职邮件 /77
:: 邮箱 /77
:: 文件名 /77
• 求职渠道 /78
• 人脉：最优途径 /79
• 竞拍：遍历潜在机会 /80
:: 什么是人才竞拍 /80
:: 哪些情况不适合竞拍渠道 /82
• 猎头：求职中的隐私保护 /83
:: 使用小号 /84
:: 使用来电拦截工具 /84

- 直投：绕过 HR /85
  - :: 直投类招聘网站 /85
- 常规渠道 /86
- 面试准备 /87
- 知识补全计划 /88
  - :: 遍历简历知识点 /88
  - :: 技能树 /88
- 常见面试问题 /90
  - :: 为什么要离职? /90
  - :: 遇到过的最大挑战以及解决办法有哪些? /90
  - :: 你希望三年后成为一个什么样的人? /91
  - :: 你还有什么要问我们的? /91
- 知己知彼 /92
- 准时和礼节 /93
- 离职 /94
  - :: 离职访谈 /94

## 常见问题

PART •

## 原理篇

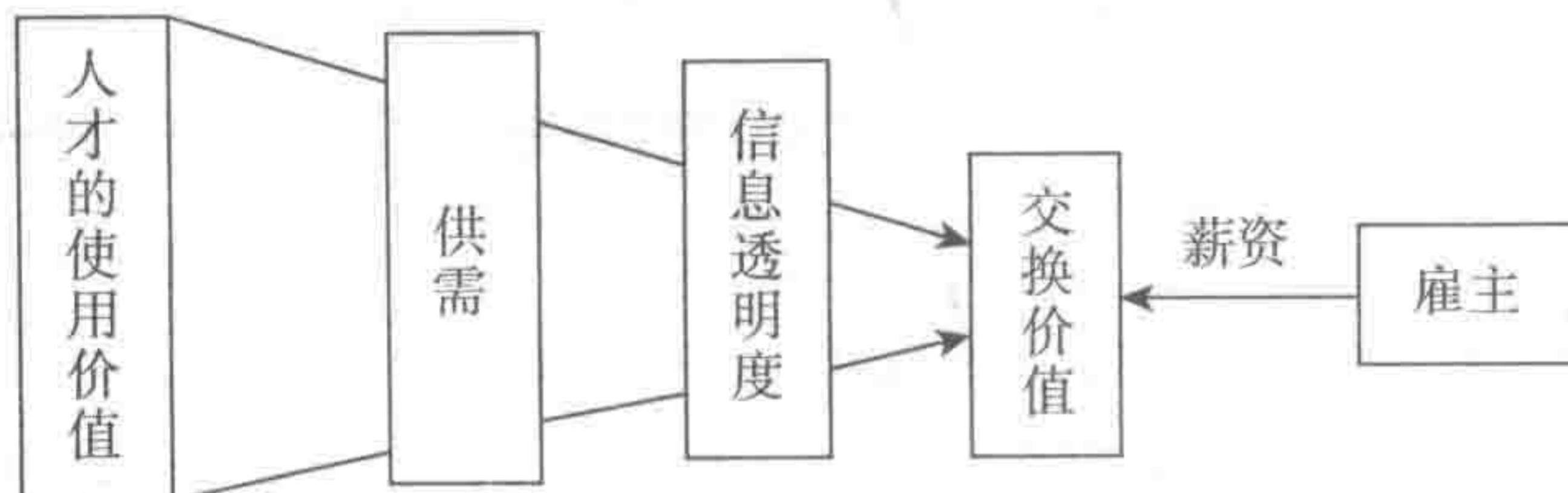
我之所以会去研究求职，以至于最后在人才这个方向创业，很大程度上是因为我觉得以前所有的求职指导都是经验论，缺乏逻辑性。

而程序员是一种逻辑动物，只有当他们理解求职到底是一种什么行为以后，才能做出有意义的行动。

我花了很久去寻找背后的理论支持，直到我重逢了价值论。本部分从价值理论开始，讲述我们求职行为的本质。



## · 价 值 论 ·



自从中国转向市场经济以后，市场规则就开始产生效果。虽然人才不完全等同于其他物品，但现在先让我们把人才也作为一类商品来看待。

### :: 使用价值

除了我们这些做人力资源相关行业的公司外，绝大部分公司购买人才都是为了使用，所以他们看中的是人才的使用价值。

说得更直白一点，这个使用价值就是指人才如何直接或者间接为公司挣钱。

使用价值不是独立存在的，而是相对于使用者存在。所以程序员的价值也是相对于公司而言的。

有的同学可能说，我的技术很好啊，又会机器学习又会编译原理，凭什么那些写 JavaScript 的人薪水比我高一倍？

谁让你在一家建站公司上班呢。对一家做网站的公司而言，机器学习和编译原理是不能为它带来收益的，而 JavaScript 写成的带有完美动画的交互组件却能实实在在地提高公司产品的销售量。

同样是这个人，如果他去一家以大数据分析为核心业务的公司工作，那么他的价值就不一样了。

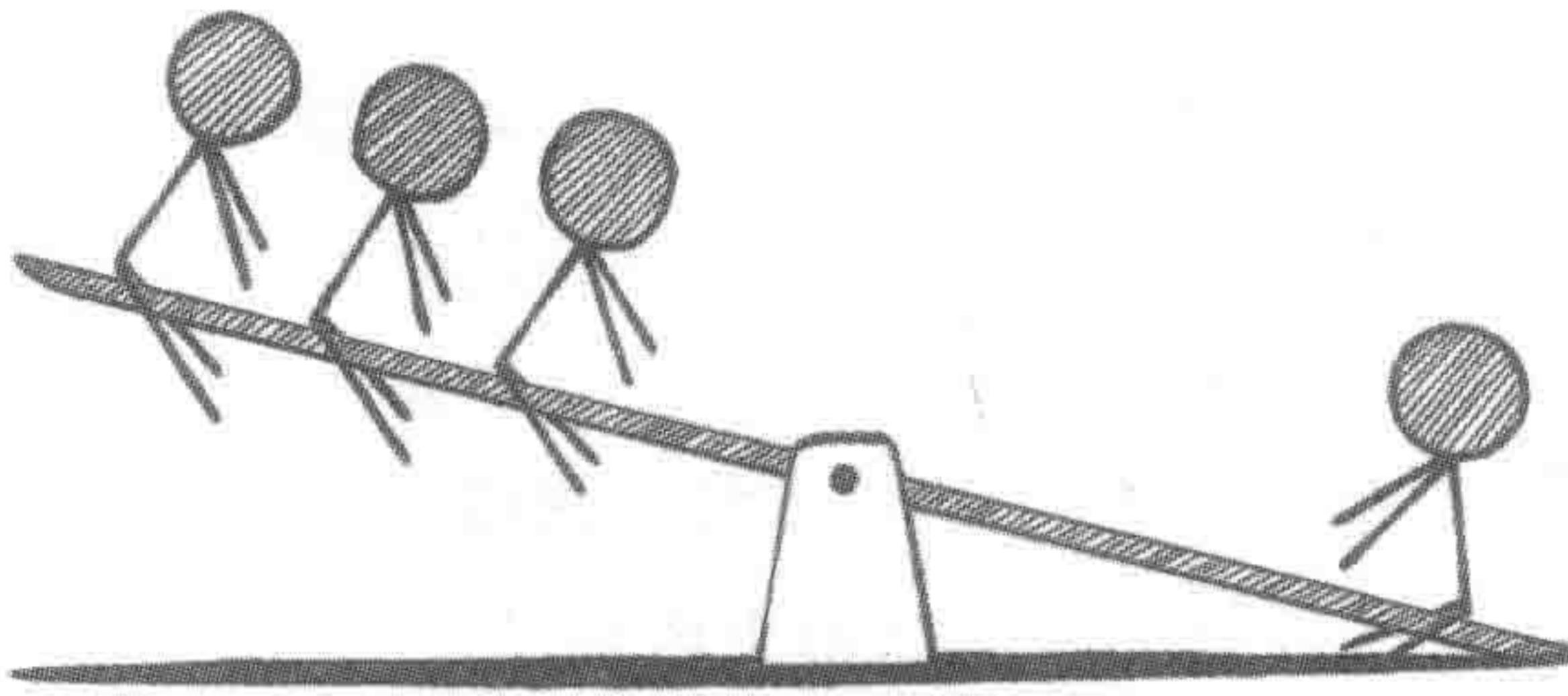
所以你的价值和你牛不牛无关，只和你能为你的雇主提供多少价值有关。（当然，大多数情况下，你能力越强提供的价值越多。）

这是基本规则。

增加自己使用价值的方式很简单，提升自己的业务能力就好。

**小提示**

对技术人员来讲，业务能力通常是指解决问题的能力。即遇到问题后，能迅速分析原因、定位问题、给出解决方案，最终搞定问题的能力。



## · 供 需 ·

有个 80 后自嘲的段子是这么说的：

读小学时，大学不要钱；读大学时，小学不要钱；还没工作时，工作是分配的；可以工作时，得自谋职业；没挣钱时，房子是分配的；能挣钱时，发现房价已经那么高了。

我不知道国内人才市场是什么时候市场化的，就从 80 后开始工作时算起吧。人才市场化意味着你有了选择公司的权利，同时也意味着公司有了选择你的权利。

这个时候就体现了交换价值。虽然交换价值以使用价值为基础，但它更容易受供需影响。

简单地说，当企业的职位空缺远少于找工作的人数时，人才的价格就会下降；而当企业的职位空缺比找工作的人数多时，人才的价格就会上升。

在过去相当长的时间里，我们都处于前一种情况。这意味着求职者要彼此竞争，而招聘方可以选择要价更低的候选人。大家在大学毕业找工作时应该深有体会。