

## 55讲江湖路：刀剑相接，战场升级



回首自己的成长之路，通常每五年就会感觉碰到一个成长的瓶颈点。在传统 IT 行业的第一个五年后，我就感觉明显进入技术成长的瓶颈期；之后也算有点运气，通过转换到互联网行业升级到了新的技术维度。

又过了五年，站在十年后的一端，回望过去，刀剑相接，如梦似幻，我渐渐感知到突破这次瓶颈的道路，就意味着走向一个升级后的新战场。

### 刀剑相接：杀人术

天下风云出我辈，一入江湖岁月催。

你狠狠地敲下键盘的回车键，终于看见程序按预期输出了正确结果。长长吐了一口气，点上一支烟，环顾四周，独自一人，又是一个夜深人静的晚上。在一种搞定 Bug 的满足与空旷寂寥的忧伤中，你不禁迷惘。

记不清这是你修复的第多少个 Bug 了，甚至记不清这是你参与开发和维护的第几个系统了。就像一个剑客在这个江湖上行走多年，已记不清死在自己剑下的人有多少，拔剑，收剑，有人倒下，你继续行走，如今“杀人术”已成。

对一个程序员而言何谓“杀人术”？你选择了一门语言开始学习编程，就像一个刚入江湖的人选了学剑或刀。再弄几本江湖宝典，假想了一个项目开始练习，熟悉基本的使用套路。然后走入江湖，拜入门派，腥风血雨，数年后剑鸣空灵，刀啸云天，飞刀无影，“杀人术”终成。

这就是一个程序员的成长之路，你选了门武器，学了基本招式，然后进入江湖不停地在厮杀中成长。终于你能搞定各种各样的系统问题，了解不同系统的设计模式。每过数月或一年半载，你总会发现过去的代码写得不好，再重构上一遍，改进你的招式，数年后，终成江湖高手。

一个程序员修成“杀人术”大概需要多久？按照一万小时理论，如果你在某一领域每天持续学习和实践练习十小时，最快也要三年。但这三年是没算各种可能的中断的，比如：生病、偷懒、假期休闲娱乐等等，所以大部分人的平均时间可能需要五年。

五年成术已算理想，实际上我自身用了更长的时间，走了更多弯路。从 Basic 程序入门，后来 VB 再到 Delphi，然后 C 最后

Java，Java 也经历了几代变迁，但还算一脉相承。技术的发展，时代的变迁会让“杀人术”也在不停地演化。而今剑术已成，然拔剑四顾，却发现已进入枪炮时代，不免心下茫然。

经历了一万小时的杀人术训练与实战后，技能增长曲线已经进入了对数增长的平缓期，过于单一的技术维度成为了我们的瓶颈和焦虑的源头，该如何去突破这样的瓶颈点？

## 认知升维：化形

爱因斯坦说过：“我们不能用制造问题时同一水平的思维来解决问题。”

技能维度的瓶颈问题，经常会让作为程序员的我们陷入一种常见的平面思维方式。比如，一个程序员做了十多年桌面客户端开发，后来移动崛起，桌面式微，就颇感焦虑，这就是他所面临的技能维度的瓶颈。而他想尝试突破的方法，可能却是转到服务器的后端开发，因为感觉这个领域还一直比较长青。

然而这只是从一个领域的核心硬技能转换到了另一个领域，但这两个领域基本是独立的，关联性很弱，而且交叉的区域也很薄，也就意味着很多经验和能力要重新积累。这就是从问题本身的维度去寻找到的解决方案，而爱因斯坦说了，我们需要到更高的维度去寻找答案。而更高的维度就是认知的维度，所以首先需要的是**升维我们的认知结构**。

在我修行成术的过程中出现了好多新技术，当时我总想忙完这阵就抽空去学习了解下。但一过几年也一直没能抽出空去看，如今再去看时发现好些当年的新技术已不需再看了。五年成术是立足于一点，成立身之本；而下一阶段不该是寻找更多的点，而是**由点及线、由线成网、由网化形**。围绕一个点去划线，由一组线结成网，最后由网化成形，“化形”表达了一种更高级的知识和技能运用形态，比一堆离散的知识技能点有价值得多。

而对于认知升维，由点及线、由线成网、由网化形，其实走的是一种“升维学习”之道。这个过程几乎没有终点，是一个持续学习、不断完善的过程，最终结多大的网，成什么样的形，全看个人修为。一条线至少要两个点才能画出，那么第二个点的选择就要看能不能和第一个点连起来了，而这比在一个维度上去预测和乱踩点要有效得多。

其实这套道理在金庸设计的武学体系中也很是明显。这里就以大家最熟悉的《射雕》三部曲为例来看下。郭靖一开始师从江南七怪，后来又跟全真七子中的几位学过功夫。这在功夫里就是两个点，但没看出这两个点有何联系，最后郭靖江湖成名，终成一代高手靠的是**什么**？降龙十八掌。为什么有十八掌这么多，从小说里的描述表达了一个体系的意思，一个体系结网成形，最后的形态命名为降龙十八掌。

其实郭靖还学了另一个更有体系、形态更牛的武功秘籍——《九阴真经》。除了郭靖，《九阴真经》还有很多人看过、学过，有高手如：黄药师、王重阳等，也有一般人如：梅超风。高手们本身有自己的武功体系和形态，所以看了《九阴真经》也仅仅是从中领悟，融入自己的体系中甚至因此创造出新的武功形态。而梅超风之流则仅仅是学点其中的招式，如：九阴白骨爪，和之前自身所学其实没有太多关联，武功境界终究有限。

所以，**升维化形，化的正是技能模型，而这套模型基本决定了你的功力高低**。

再回到前面那位桌面端程序员的瓶颈问题，升一点维度看更泛的终端，桌面端不过是这棵技能模型树上的一个分枝。树并没有死，甚至更壮大了，只是自己这棵枝干瘪了些，所以可以去嫁接其他分枝获取营养，而非想要跳到另一棵树上，重新发芽开枝。

## 战场升级：十面埋伏

结网化形，走上升维之道，因而战场也变大了，但你的时间并没有增多，这就存在一个理论学习和战场实战的矛盾。

到底是应该更宽泛地看书学习建立理论边界，还是在实战中领悟提升？关于这点，你需要选择建立适当的平衡，走两边的极端都不合适。在学校的学习更多是在建立理论体系，而在工作前五年的成术过程则更多是偏实战。

再之后的阶段又可能需要回归偏理论，提升抽象高度，从具体的问题中跳出来，尝试去解决更高层次、更长远也更本质的问题。而从更现实的角度来看，你的环境也会制约你能参与实战的经历，导致有些东西靠实战可能永远接触不到，不去抽象地思考是无法获得和领悟的。

历史上关于理论和实战有很多争论，还留下了一些著名的成语。理论派的负面历史代表人物有：赵括。还有一个关于他的成语：纸上谈兵。他谈起军事理论来一套一套的，一上战场真打起来就葬送了数十万将士的性命，所以大家都会以赵括为例来批评没有实战经验支撑的理论靠不住。

但其实还有另一个更著名的历史人物，也是理论派出身，在真正拜将之前也没什么实战经验。并且也有关于他的成语，如：背水一战，这是他抽象地思考过很久的战法，但也是第一次上战场使用，一战而青史留名。

他就是韩信，历史上说他率军出陈仓、定三秦、擒魏、破代、灭赵、降燕、伐齐，直至垓下全歼楚军，无一败绩，天下莫敢与之相争。王侯将相韩信一人全任，一时国士无双，属于中国古代从理论到实战的谋战派代表人物。

韩信的对手项羽在历史上就是一个实战派代表人物，个人的“杀人术”相比韩信高出怕不止一个等级。但其实他和韩信根本不在一个维度上，韩信在最后面对项羽前，已通过众多大大小小的战斗去不断实证和完善了他的谋战理论。垓下之战项羽中十面埋伏，致其乌江自刎，更像是一场高维打低维的降维攻击。

所以，关于理论和实战的关系，从这个历史故事可以有所体会。而“**十面埋伏**”这样的技能维度显然比“霸王举鼎”要高出不少，而升维后的技能，也需要升级后的战场才发挥得出来。

技能的成长速度总会进入平缓阶段，并慢慢陷入瓶颈点，然后也许你就会感到焦虑；而焦虑只是一种预警，此时你还未真正陷入困境，但若忽视这样的预警，不能及时进行**认知和技能升维**，将有可能陷入越来越勤奋，却越来越焦虑的状态，结果走入“**三穷之地**”（包括如下三种“穷”）：

1. 结果穷：技能增长的边际收益递减；
2. 方法穷：黔驴技穷，维度过于单一；
3. 时间穷：年龄增长后你能用来成长的时间会变少，分心的事务更多，而且专注力会下降。

认知和技能升维带来新的成长收益，同时防止了单一维度的死胡同，而年长的优势正在于经验带来的理解力和思考力的提升。

最后，总结下今天的分享内容，在程序江湖上，从刀剑相接到战场升级走的是这样一条升维路：

- 刀剑相接的战场，我们靠“杀人术”也即硬技能求生存，但时间久了就会有瓶颈；
- 技能升维，需要认知结构先升维，“我们不能用制造问题时同一水平的思维来解决问题”；
- 升维后的技能，也需要一个升级后的新战场，走上理论结合实践的“谋战”之路。

在我的寻路过程中，我找到的就是这样一条技能升维之道，那你呢？

---

# 程序员进阶攻略

每个程序员都应该知道的成长法则

胡峰 京东成都研究院 技术专家



## 精选留言



森晨

所以说 打算入行的小伙伴要选一门会一直火的语言入门 就像降龙十八掌 然后再去苦练内功 算法与数据结构 就像九阴真经 这样你打出来的招式才厉害 那么问题来了 外功 java 和 python哪个更好?

2018-12-07 11:20

作者回复

两者的互补关系强于替代关系吧

2018-12-08 10:53



third

一切的战争，是认知维度的战争。

升级我们的认知结构，连点成线，以高维度打低纬度。

理论和实战，需要相互印证

2018-12-15 20:38