

美国公司管理层的洗脑技巧

美国公司管理层的洗脑技巧

在美国工作过的人，都对美国公司管理层的心理战术深有感受。很多人都感觉跟上司之间有某种难以言表的问题，总是感觉自己的价值不受尊重，然而由于上司说话客气，所以又不好发作，难以说清楚这到底是怎么回事。

其实美国公司的管理者，使用着一些强大的意识控制（mind control）技巧，经常给你念经洗脑，使得你的头脑失去防御能力，最后沦为任他们摆布的奴隶。经历了这么多工作之后，我想把他们常用的技巧总结一下，以便广大员工对此提高警惕，采取相应的对策。

我这里所指出的技巧，似乎在网上都搜不到。如果你在 Google 搜索，只能找到一些冠冕堂皇的题为『如何管理职工』之类的文章，把管理层说得简直无比善良，尊重人才，体贴又公平。当你真正理解了他们才发现，根本不是那样的。

建立对比参照物

管理层跟你杀价，最重要的手法就是“比较”。这作法跟你妈妈买菜时杀价是一样的原理：选择另外一个人作为你的“参照物”，在需要的时候，把这个人抬出来跟你比，以达到某种目的。这个人一般是上司认为水平比你高，或者职称比你高的某个人。这个人也可以就是上司自己，比如他可以跟你讲述他曾经做出的，你无法核实的“业绩”。为了方便，下文把这个用来跟你比的人称为“小强”。

管理层首先假设小强比你强，价值比你高。虽然事实也许不是这样，但他们总是会想方设法让他显得比你强，比如夸他以前做过多么了不起的事情，让你在不知不觉中接受这个假定。夸别人的时候总是不心虚的，所以把他吹上天也没有自大狂的嫌疑。他到底做没做那些事情，你也无从对证。等你开始点头，他们就可以随时在你面前提起他的名字，借此来达到某种目的。

举个例子，上司可以让你去做一些高手不愿意做的杂活，比如为团队的代码配置一个测试工具。作为一个编程高手，刚进公司代码都还没碰呢，就被叫去干这种杂活，所以你对此一点不感兴趣。而且这工具由于设计很差，问题很多，耗费你很多时间，却收效不多，非常没有成就感，显然这不是一个聪明人愿意做的。你心想，公司里那么多大学刚毕业的，干嘛不让他们去折腾这些，却让我做这个事情。

当你对此提出异议的时候，上司就会苦口婆心对你说：“这种杂活是我们每个人都应该会的。你看看小强，人家编程水平可以说是超一流了，然而当年我们的连续集成（CI）系统，那么无聊的东西，都是他设置的。”这样你就没话可说了。然而上司没告诉你的事情是，小强当年是自己对那 CI 系统感兴趣，所以他就去做了，那并不是管理层强制分配给他的任务。而对于你，这是强加给你的任务，你不感兴趣也得做。而且小强设置的 CI 系统，比你现在折腾的这个东西容易配置，没那么多毛病，而且对于团队意义挺大。你的情况跟小强的其实没有可比之处，然而这种突如其来的“对比心理压制”，让你措手不及，只好默认接受。

这里我要指出很多人的一种误区，他们认为杂事必须得有人做，你不做我不做，难道让老板做？所以老板让任何人做任何事都是可以的。这种想法是不对的。公司里自然有不同水平的人，杂事一般都应该交给水平比较低的，或者刚毕业的新手去捣鼓，或者正好有人感兴趣，那当然最好。当然对于新手我们也应该尊重，如果他折腾了一段时间，跟你反映这个东西问题太多不好用，你就应该考虑放弃这个东西，也许有其它更好的选择。水平高的人请进来，真的不是你叫他做什么都可以的。他完全应该拒绝给你端茶倒水，给你跑腿买机票，对不对？所以他也可以拒绝做他觉得不符合自己身份和职责的事情。我见过有些老板，自以为有钱能使鬼推磨，不尊重人才，所以到最后没人愿意给他做事。

另外一个例子。评职称的时候，你做了一个很好的成果，然而上司可以对你这样说：“我觉得你的成果还没有到让你升职成为 principal 的资格。当年小强做了那个什么什么，比你这难度还要大，人家当时才升了一级而已……”小强那东西的价值和难度，其实谁也不清楚，但这样跟你一比，就让你沉默接受了他给你的评定。

把人拿来互相比，是一种不尊重人的作法。管理层就是通过这种比较的方式，来压低人的价值。在 A 面前说 B 更强，在 B 面前说 A 更强。因为 A 和 B 信息不通，结果 A 和 B 的价钱就都给压下去了。这就跟你买菜的时候，跟 A 家的菜农说：“B 家这个菜才卖 2 毛钱一斤。你怎么卖 3 毛 5，也太贵了！”而其实 B 家的价钱是 5 毛，或者 B 家的菜不新鲜……菜农都知道这种技巧，所以他们会对你说：“一分钱一分货。你嫌贵就去 B 家买好了！”

我觉得广大受过高等教育的人才，也应该有这种一口价的霸气。只有这样，我们程序员才能维护自己这个行业的权益，不受刻薄的买菜大妈们的欺负。

激将法

上司可以用参照物来实现“激将法”，使得你更加卖命的工作。常用的做法就是，把小强做的东西吹得神乎其神，拿来跟你做的东西比。做技术的人很多都争强好胜，你想要做出胜过小强的东西，所以你就更加拼命地工作。等你做出来，就让公司赚了。最后你仔细一看小强做的东西，其实做得很水……

我在 Google 做 PySonar 的时候，上司就经常使用这种激将法。他经常会对我说：“你看这个某某 PL 人，多牛啊，做了一个 DSL，要是请到我们这里来该多好！”或者说：“你看这个 Facebook 的开源项目静态分析工具，貌似也可以做 Python 的静态分析呢，还能支持好多其他语言！”如果你不仔细看看，也许就会更加拼命的工作，想在这群人里面立于不败之地。然而你仔细看看，就发现那 DSL 其实是完全没必要的东西，反而让问题变复杂了。而 Facebook 那东西，完全就是一个玩具，根本不可能进行深入的静态分析，不过是一个像 lint 一样的 style checker 而已。说支持这么多种语言，其实大部分语言的目录里全都是空的，只有 PHP 的目录里有点东西！

我看透了上司的激将法，这就是为什么在 PySonar 的第一版里面，我留了一手。Google 使用的版本，其实比 PySonar2 要弱很多。最强的版本，我只留给自己。

树立虚假威信

很多管理层的人，从跟你面试时开始，就已经开始树立自己的高大形象和威信了。只要你对公司表达一点点的欣赏，他们就会想办法在心理上压倒你。比如，当你表达对某公司的尊敬时，他就会开始说：“对的，这是一个成熟的公司。我在这已经工作了 14 年了，这表示我喜欢这个公司。当年，我是 A 系统的构架师，我还是 B 系统的构架师，我跟某某牛人一起工作……”

等你进了这家公司，从人事部（HR）管理系统里面一查，才发现这人的履历里面根本没有什么 A 系统和 B 系统构架师的经历。后来从同事那里听说，他只不过是参与制作了 A 和 B 里面的一小部分，根本不是什么构架师，而且他参与的那部分代码，直到许多年后的今天还有 bug。

还有一种虚假威信不依赖于公司，而是在公司外面的“社区”里形成的。比如在 JavaScript 的社区里，这种现象极为严重。有些人写了一块很小的代码，GitHub 上的星比较多，然后就被无知的 JavaScript 程序员们捧上了天。参加各种肤浅的会议，把大学一年级就该学到的东西（closure 一类的）当成宝贝一样宣讲。还有些人开始写书，当然这书完全没有价值，Amazon 上全是一两颗星的差评。然而这种人总能以“书作者”的身份自居，在某些肤浅的公司占据管理位置。

然后呢，为了维护自己的虚假威信和地位，为了建立“业绩”，他们会在公司里疯狂的瞎指挥。总是要提出一些跟其他公司都不一样的“新作法”，显得自己有创造力，结果每次都帮倒忙。为了有人支持自己，他必须招收更多的同类进入公司。最后导致整个部门都被这种人占领，各种假大空一拥而入，光拿钱不做事，或者帮倒忙。真正有本事的人受不了这些人的排挤，陆续离去，形成“反向淘汰”。这样的公司不走向失败才怪。

虽然你心知肚明，但为了自我保护，不能告诉上司你已经看透了他。所以他也假设你不知道，于是在以后的日子里，他就可以利用这些虚假的资历，占据对话的有利地位，对你指手画脚。

虚构奋斗经历

管理层想给你一个比以前公司还低的职称（title），你对此提出异议，他们有一种忽悠技巧可以让你接受它。上司可以这样说：“你知道吗，小公司的职称都是乱来的，不算数的。我当年在小公司，那可是 VP engineering。进这大公司时，人家只给了我个 manager 的职位。可是因为公司尊重人才，再加上我的出色表现，很快就升到了现在的职位。所以关于这个职称你不用担心……”

等你进了这家公司一查 HR 系统，发现他进公司时的职位哪里是 manager 啊，一进来就是 director of XYZ，领导几百个人。上司为自己制造这种虚假的“卑微”和“奋斗”经历，目的是让你接受不公平的待遇，让你拼命工作而已。

夸大职称职责

为了压低你的职称和薪水，管理层可以故意夸大高级职称所需要担当的职责。比如你想升职的时候，他可以对你滔滔不绝的念经：“你想要 senior 职称？我们这可是大公司，我们的 senior 级别可不比小公司的。我们的 senior 需要完全从头设计和实现高难度的新功能，要能指导和教育低级别的员工，要能按时保质保量的完成艰巨的任务。如果你想要升到 principal，那你就需要超强的系统编程能力。你看小强，他在六个月之内实现了从编译器，操作系统，一直到数据库的所有东西！你还需要能够从公司利益的角度出发，想法把一个几亿美元的产品，做大到几十亿美元的年利润。我们的 senior 和 principal 级别，可不是像小公司一样闹着玩的！”

你还真能被他们说无语，可是回头一想就发现不对劲。后来你发现，这公司的 senior 职工，根本就没有任何他们所宣传的能动性，一般都是打打小补丁而已，朝九晚五的干活，你不叫他做事就完全不管，测试失败了还在继续提交代码，各种耽误你时间。而小强六个月做出来的什么编译器，操作系统，数据库，都是玩具级别的，或者其实根本就是上大学时做的期末大作业。六个月我能把工业级的编译器，操作系统，数据库全都做出来，我还能把几亿美元的产品变成几十亿美元的，那我干嘛要来你们公司工作？我自己开个公司，每年赚个一亿就够了！

管理层的这种作法，就像在地摊上买顶 20 块钱的帽子。挑了半天，终于决定买了，却只掏出 10 块钱，跟你讨价还价：“你知道我以前 20 块钱能买什么样的帽子吗？那可是金丝貂皮镶边，顶上还嵌了钻石，蓝宝石……而且还是买一送 50% 折扣券……”所以你就只有跟他说，我不做你这种人的生意！

制造领域壁垒

公司为了压低你的价钱，还可以故意制造和强调“领域壁垒”。本来程序已经写得出神入化了，各种语言，编译器，高级的算法，数据结构，网络协议，WEB，……都会。你从来没有觉得一个程序员应该局限于某一个子领域，不管什么技术，拿起来就应该会。然而当你面试之后，也许就会收到这样一封 offer 信：

“亲爱的 XX，我们很高兴的 offer 你软件工程师的职位。虽然你的背景显示你缺乏并发系统编程的经验，我们觉得你是一个很有潜力的人选。我们决定给你 NNN 的工资，……”

“并发系统编程”是什么玩意？之前的工作里，线程什么的也不是没用过。你从来没有想到，这样一个不值一提的“技能”，有资格列举在你的简历上。可是现在管理层因此说你缺乏这方面的经验！

当然，NNN 是一个有点寒酸的工资。如果你不满意，管理层就会对你说教：“你知道吗，并发系统不是你想象的那么容易。我们的工程师在这个领域都有 30 年以上的经验，才能到 senior 级别……”

等你进去，一眼就发现那些所谓“30 年并发系统编程经验”的 senior 程序员，在很明显的“竞争条件”（race condition）的地方少放了几个锁，系统会在小概率的情况下出现灾难性的后果。他们不放锁的原因是因为一种误区，认为使用锁一定会降低程序的速度。而且由于过度使用并发，造成了很难察觉的性能瓶颈，本来想用多线程提高性能，结果适得其反。

其实 IT 界哪有这么多“领域”，每一个领域需要的知识，不过都是计算机科学：数据结构，算法，语言，再加一点工程经验。只能写某一个领域程序的人，根本不配叫做“程序员”。其实管理层早就知道你能胜任这个工作，故意的制造和强调领域壁垒，不过是压低你职称和工资的手段而已。

贡献至上论

很多公司和人，都流行一种“贡献至上论”。你发现跟自己水平相当甚至比你低的人，职称和工资却比你高，跟管理理论的时候，他们就会抬出“贡献至上论”。总之就是说，人的薪水是按他为公司做出的贡献来算的，而不是以能力或者水平。你才进公司一年，虽然已经显示出超人的能力，但是公司还没有得到你的贡献，所以现在的薪水是合理的。咋一看貌似一个合理的标准，有多大贡献拿多少钱，然而其实这里面有好几个陷阱。

首先我们从物理单位的角度来说明，为什么工资应该对应能力，而不是贡献。所谓“工资”都是时间工资，而不是总收入。比如，你的时间工资是“100 万/年”，这并不是说你一进公司就拿到 100 万，而是说你工作一年的时间，累积能拿到 100 万。所以“工资”对应的物理单位是“功率”，而不是“能量”。按照这个道理，工资对应的东西，应该是人的功率。人的功率就是他的“能力”，而不是他的“贡献”。粗略的公式是：贡献 = 能力 × 时间。比如 A 一个星期能完成 100 个单元的工作量，B 一个星期只能完成 10 个单元的工作，那么按理 A 的工资就应该是 B 工资的 10 倍。这跟 A 和 B 待在公司的时间，他们已经对公司做出的贡献大小，关系其实不大。B 也许在公司的时间比 A 长很多，已经做的贡献也更多，然而 B 已经拿到的那些收入，已经偿还了他所做出的贡献，所以 A 的工资并不应该因此比 B 的低。

然后再来想一下，商品的价格是怎么定的？物以稀为贵，对不对？能力高的人是稀少的，如果你不付足够的工资，能力高的人就可以不为你工作，你也许今后再也找不到另一个能力这么高的人。不管他是刚进公司，还是进去一段时间了，都是如此。能力高的人永远不会因为自己刚进公司，没有“贡献”，就接受低于普通人的薪水。如果你知道其他人的职称和薪水，就会发现他们其实一进去就很高，根本不需要已经做出什么贡献。

另外，“贡献”是一个很主观的东西。打个比方吧，资深外科专家给你做了心脏手术，你不想付钱，于是就说：“虽然你医术很高，然而手术之所以可以成功，主要是因为有家人的安慰我，让我得到心灵的抚慰和照顾。所以你的贡献其实很小，不应该付给你那么多的费用！”明白了吗？公司有许许多多的人想要晋升到你想要的职位，可是职位却是有限数量的。每个人都做了一点东西，可是最终决定谁做出了更大的贡献，是管理层的少数人决定的。就算你能力高超，

做的东西其他人都没法完成，然而你的贡献到了管理层那里，只是几行字而已。这些字是你的上司以及他的上司写出来的，至于他们怎么写，你是无法控制，甚至无法知道的。

这种贡献至上论导致的结果，是让有能力的人得不到合理的回报，甚至得不到发展才能的机会。由于你刚进公司还没有“贡献”，就算你能力很强，公司可以以此为借口给你很低的工资，然后给你一个完全无法施展才能的任务。你辛苦折腾了一年后发现，管理层说你的贡献没有很多，所以你的职称和薪水就停留在很低的水平。

所以，能力强的人必须反对这种“贡献至上论”。你的薪资水平，不应该是看你已经做出多少贡献，而应该是看你的能力。如果你是一个能力超强的人，你必须要求高的薪资。因为公司在单位时间内为你付出更多的钱，他们必须想办法利用你的能力，创造大的价值，这样你的才能才有可能得到发挥，你才会变得越来越强。

引用公司政策

当你不服上司的说法，他们说不过你，就会引用公司的政策文件。比如，如果你对职称不满意，他们就会说：“请你参考公司的职称标准，上面列举了 senior 和 principal 职称需要具备的能力和职责。我觉得你在现阶段还没有满足这些条件。”

所有这什么“公司规定”，其实都是借口，是“catch”，它们就像有些信用卡条款下面的小字，你不仔细看就以为小便宜就在眼前，结果根本不是的。那些小字写的是，你必须满足这样那样的条件才可以！

所有这些诱惑的作用，就是把你变成一头拉磨的驴子。在你的头顶上架个杆子，上面挂着食物。你想吃到这眼前的食物，可是你每走一步这食物就随着你往前挪一步，所以你就总也吃不到，只是不停地给别人拉磨。

总有人传言说，在这公司升职好快哦，看那谁进来那么低职位，很快就变成 principal 了。轮到你的时候，就发现完全不是那回事。等你去查，发现那个人升职根本就没那么快，好几年了才升了一级。于是你醒悟了，原来这些都是广告吧？

这些红头文件，貌似很容易拿出来，无可辩驳地压制任何人的意见，然而它们对于有能力的人其实是无效的。很多上司都会在你提出加薪的时候对你说：“我们公司都是每年五月的时候，所有人一起做评估，升职加薪都在那个时候，你就再等等好吗？”而实际情况却是，公司其实在任何时候都可以给人加薪。只要你对薪水不满意要走人，公司立马可以给你加薪。

“这是我们的决定”

你的团队遇到一个技术问题，有两种不同的解决方案。有个叫小强的 principal engineer 说应该采用 A 方案，而你觉得应该采用 B 方案。经过好几天，几十封 email 的全组讨论，你用深刻的技术分析和大量的证据，说服团队里的绝大部分人，应该采用 B 方案。A 方案不但不解决问题，而且导致系统过度复杂，容易出错，而且工作量巨大，延缓工期。然而结果却出人意料，你的上司最后发了这样一封 email:

“经过讨论，我们决定采用 A 方案。这是我们全组人的决定！详情请参考这篇‘XYZ 小组规范文档’（链接）对于此事，不得讨价还价！NON-NEGOTIABLE！”

你打开链接，发现这文档是上司今天刚写出来的，里面不但缺乏你在 email 讨论里面提出的最重要的观点，而且还有很多理解错误的地方。上司一意孤行，自己做了决定，却号称这是“我们的决定”。自己写了“规范”，完全没有经过队友们同意，却说是“我们小组的规范”。这就是所谓“强奸民意”吧。

最后你发现，你的上司总是听小强的，不管他说的是对的还是错的。而这个小强，就是团队的系统过度复杂的主要原因。显然你的上司有什么把柄落在了小强的手里……同事们也觉得很奇怪，然而大家都面面相觑，除了服从，没有其它办法。

待续……

当然管理层的招数不止这些，我也还在总结中。如果你有其它的经验，欢迎来信给我指出：hi@yinwang.org。