

新入职程序员生存之道

为什么想讲这个主题？

我觉得我只是说出了一个事实，那就是大家对于新团队的理解和认知远远低于事实情况，就像我们对于自己的身体一样无知。好了，下面我们假设进入了一家我们所谓的新团队，那么我们应该怎么办？我这里列出我所理解的几条规则。

一战成名

我们为什么需要一战成名？原因有两个，那就是一方面要在团队中建立威信，毕竟是技术团队，最有效的还是专家管理，就是我的技术能力比你强，你就自然没话好说。我个人持有一种偏见，那就是纯管理的经理，其职业发展道路是有限的，技术的底子有多厚，决定了管理上能走多高多远。团队中的人信不信任你，服不服你是你的想法能否落地的关键。第二的原因是需要让你的老大，产品和运营能够建立对你的信心，如果你的老大不挺你，产品和运营的同学不支持你，你的外部压力就会很大，导致没有施展的空间。

有一种架构师叫做新来的架构师，他也知道信任和信心的重要性，只是他选择的是一条非常难的道路，就是要引入一个新的架构，而这件事情劳命伤财，周期太长，风险巨大，不能快速看到效果。最好的方式是选择一个业务上影响很大好久都没有解决的问题，因为这样产品业务都能有所感知；第二就是快速起效的，能够很快看到效果的。我们不是要证明自己强，而是要证明自己对于业务的价值。

管理领导的预期

有些团队的负责人一上来就开始闷头做事，带着团队开始往业务目标冲刺。我建议先不要这样做，首先的一件事情是和你的上司谈，和你的对口的产品业务谈，谈什么？谈他们的预期，他们最希望解决的问题是什么，以及时间点。还有就是什么是可以缓做的，或者不做的。原因在哪里？首先一个原因是一个技术团队的产出品的好坏，CEO，产品经理，业务的人最有发言权，不管团队自己说的多好，他们不满意都是白搭。

产出是检验一个技术团队好坏的唯一标准，而这个标准是捏在外部的这些人手上的，不是在内部的。其次，团队内部的反馈帮助不大，内部的声音往往会有两种，一种是受害者心态，就是抱怨受到不公正待遇；另外一种是自己什么问题都没有。如果团队意识到问题的话，问题早就被解决了，往往是意识不到问题，或者不是真正认为他是个问题，或者他自己本身有问题，才无法让问题得到解决。

管理预期还有一个重要的作用就是要为团队减负，为调整留下空间。传统机械的管理模式是一看团队产出不行就采取高压的方式，结果虽然短期内看到产能提升了，但是质量却大大下降，为以后积累的大量的技术债务，这些债务的集中爆发才引起了团队之后的效能大降。我们在团队效能提升的过程中是需要大量还债，简化系统，需要把负担降下来，所以如果能够把外部压力降下来，就可以让我们进行更大范围内的调整。这就像是面临经济危机的时候，不应该收紧银根，反而应该货币超发，这些都是经济学的基本规律。

让不合适的人离开

经过了前面两项之后，你的能力也秀了一把了，老大也开始挺你了，外部的压力也缓解了，下一步就是要好好盘点一下手上的人了。不管今天我们面临的技术问题是长成什么样子，最终你会发现都是人的问题。人是组织的基本单元，人的思维和做事方式决定了组织的基因，要该组织的基因，必须从人开始入手。

对于团队中人的评估主要分成几个点：

1. 技术能力。好了，又回到了技术上了，技术管理中间很重要一点就是对于每个人的技术能力进行一个合理的评估，这又是一个技术管理者的一项基础能力。
2. 能量场。他身上到底是存在正能量还是负能量，凡是有抱怨的，必然是负能量场，需要特别留意。

对于这两者评估以后就自然会出现四个象限：

1. 技术好正能量，委以重任，加薪升职。
2. 正能量技术好，老黄牛类型，降低难度，多布置任务。
3. 技术好负能量，适当隔离，给予有难度的技术任务，紧盯考核。
4. 技术差负能量，马上离开。

一般来说这个时候肯定会出现一种情况，那就是这个人离开了以后他的事情就没有人做了。这就是为什么我们要和产品业务达成预期的原因，很多工作是不需要做的，如果我们的外部压力不放一些的话，里面无法调整，这就是死锁。而且一个烂人同样占有一个名额，业务团队的期望值是跟着你人数走的，而不是看具体的能力走的。第二，真正重要的事情，就应该给更重要的人去干，不重要的事情就算不做了也没有人关心。第三，如果这个岗位没有其他的情况，那么可以考虑稍微缓缓再动，但是裁员最好的方式是一步到位。这样对团队的冲击反而最小。

稳住节奏

清理了人员，建立了规矩之后，我们必须再次回到对外，团队管理中对外管理是非常重要的部分，甚至可以占到管理中30%以上的时间，之前说过技术团队的衡量标准是团队的产出，而这个标准是外部团队定的，不是自己定的。同时还有一条，技术团队的输入也主要来自于外部，需求来自于外部。当然也有很多的需求来自于内部，比如技术重构，性能优化，架构升级，这些工作会占据外部需求开发的时间，可是最终又会让产出结果更好，这些都是需要跟外部去沟通的。基于这几条，我们需要把对外沟通的渠道和方法变成一种有效的开发节奏，这就是我所说的稳住节奏。

首先是明确迭代周期，传统的开发模式是需求固定，人员不固定，开发时间也不固定的状态；而现在scrum模式的方式是人员固定，开发时间固定，而需求不固定。这个怎么来理解，拿远洋运输来比喻可能最好了，传统开发就像是就是把一批货物从A运到B运完，不管用多少条船来运，运完为止，时间完全取决于船的大小。而迭代模式是从A到B点开了一条航线，就这么些运力，每周一班准时出发，需求要被切碎到一个个集装箱里面装好，这样集中发船，装船卸货的速度都是标准化的，这样效率就会很高，但是同时对于货物的切分和先后就提出了更高的要求。

所以回到技术团队和外部的沟通部分，基于明确的迭代周期，一周，两周或者是三周，一个月，这个根据业务形态来定，然后产品需要维护一个产品需求池，技术需要维护一个技术改进需求池，每次迭代周期需要有明确一次迭代需求的会议，什么放进去，什么不放进去。平时的时候还需要针对需求设计，还有开发过程中的需求改进进行沟通，到迭代结束的时候，还需要产品来进行确认验收，整个过程需要经过长期打磨，变得井然有序。这又是一个长长的磨合过程，不可一蹴而就。

引入新人

为什么这个时候才谈引入新人，最重要的原因是虽然新人很重要，但是我们不能完全寄托在新人身上，有的团队有这样的期待，就是牛人来了以后什么问题就解决了。基本上这个事情叫做可遇而不可求，首要任务还是在自己团队里面先折腾起来，把自己能够控制的部分先做到最好。招人是个长期的活，不是一个想招的时候捞一把，不想招聘的时候看也不看，一个技术经理最好每周都有面试，因为好人的窗口期是很短的，所以我们时刻要敞开，另外一方面也是保持时刻对外的一种开放视角，看看不同的人，不同的趋势，招聘是一种很好的学习。这就是为什么Facebook会说招人是公司最重要的事。

真金不怕火炼

如果你去问一个优秀的技术人员他什么时候的技术提升最快，他一定会跟你绘声绘色地讲一个故事，在哪一年的时候，当时业务的发展速度太快，导致系统的各种顶不住，然后用了各种方法来进行提高，解决了一个问题又解决了一个问题，直到系统变得稳定之后，反而有一种失落感。

我们把这种现象叫做跑得连鞋子掉了都没时间穿的情况，这是一种非常好的状态，在这种状态下，团队会跑的很快，而且不需要做过多的管理，自组织，配合就会体现出来，

跟不上的同学会自然掉退和被淘汰。由业务快速发展本身产生的驱动力会让团队的成就感和信心大大提升，战友间的感情也会特别深。

只可惜的事情是这种情况可遇不可求，如果一个公司十年上市的话，最多只有两三年时间能够有这样的黄金状态吧。但是这确实是团队锻炼最黄金的时间，让业务的压力来驱动团队跑起来。

做好离开的准备

如同公司转型一样，团队转型面临的风险也是巨大的，因为我们是在和习惯在做斗争，就像开始的时候说的，他不是治疗感冒药到病除，而是治疗癌症，是基因的扭转。如果你有幸成为这个扭转乾坤的人，那么恭喜你，这是人生宝贵的一段修行。如果不是，也别灰心，继续修炼吧

GitChat