

## 12 | 流程和规范：红绿灯不是约束，而是用来提高效率

2019-03-23 宝玉

软件工程之美

[进入课程 >](#)



**讲述：宝玉**

时长 14:29 大小 13.27M



你好，我是宝玉，我今天想与你讨论流程和规范的价值，以及如何参与制定好的流程规范。

不知道你所在的软件项目中是不是也有各种流程规范，例如：

开发人员不能直接在生产环境修改代码操作数据库，必须在本地先测试验证后，由运维操作；

代码需要 Review 通过才能合并主分支；

代码需要遵守各种规范，像命名、格式，还有缩进用几个空格还是 tab 的细节问题；

遇到 Bug，先提交到 Bug 跟踪系统。

在我经历的项目中，或多或少都会有各种各样的流程规范，而且越是大的、正规的项目团队，流程规范越多。

然而很多人对于流程规范并不是很理解，甚至觉得是一种约束。

## 为什么要有流程规范？

从某种程度上来说，流程规范确实是一种约束：约束了我们如何做一件事，约束了我们用什么标准做事，约束了我们用特定的顺序做事。

既然如此约束我们，为什么还要有流程规范呢？

## 提升团队效率

**从个体来看，因为流程规范的存在，确实可能存在效率降低的情况，但从团队的角度来看，好的流程规范反而是提升效率的。**

这其实很像我们生活中的红绿灯，用一个简单的规则：红灯停绿灯行，来约束车辆行人按照指示灯行进。

从单个车辆来看，看似是因为红绿灯的存在而影响了效率，但是从整体来看，因为红绿灯的存在，有效避免了拥堵，反而是提升了大家出行的效率。

其实红绿灯除了能提高效率，还有其他好处：

红绿灯这样好的管理交通的经验，形成流程规范后，就可以全世界共享这种先进的经验；

红绿灯不再到处依赖于人指挥交通，而变成了让红绿灯的规则来指挥交通。

软件项目中的流程规范是不是也有这样的效果呢？

以代码审查的规范为例，对于技术高的程序员来说，代码审查可能会耽误一点时间，但对整个团队来讲：

即使是水平高的程序员，也可能会有错误，代码审查可以降低出错的概率，保障质量；

对于水平低的程序员，可以通过代码审查学习和成长，代码被高水平程序员审查后，可以有效提高质量。

软件项目中这样的例子还有很多，类似的还有像遇到 Bug 要提交到 Bug 跟踪系统，还需要配合重现步骤说明，看起来繁琐，但是却让 Bug 可以有效跟踪，让开发人员可以重现和定位，从而高效的修复 Bug。

## 将好的实践标准化流程化，让大家可以共享经验

我们知道，在运动项目上，有些运动员特别有天分，总能拿好的成绩，而这些运动员的动作，会被反复的研究学习，最终形成标准化动作。而其他天分一般的运动员，按照研究出来的标准动作练习，也能取得非常好的成绩。

软件工程也是这样，早些年的软件项目，就是个人英雄主义盛行的时代，项目的成败极其依赖于个别厉害的项目经理或者技术高手，而这种牛人，总是稀缺的存在。

所以后来很多编程高手写代码的方式，甚至写代码的格式，也会被研究，最终形成一套套的代码规范。其他水平一般的程序员，按照代码规范，也能写出不错的代码。

代码规范还有个好处，就是大家写出来的代码看起来差不多，换个人接手别人的代码，也能很快上手。

如果我们站在流程规范的角度看软件工程的开发模式，它也是源自实践过程中，有些厉害的项目经理发现了好的、可以提升软件质量的开发实践，不断总结改进，最后变成了流程，让普通的项目经理按照这一套流程，也能做出不错的软件。

你看瀑布模型也好，敏捷开发也好，最后落实下来，不就是开发过程中一个个的流程规范么？所以瀑布模型我们需要各种阶段评审，敏捷开发需要每天开站立会议，需要每个 Sprint 有计划会、评审会。

## 借助流程规范，让项目管理从人治到“法治”

在《[10 | 如果你想技术转管理，先来试试管好一个项目](#)》这篇文章中我就提到过，管理就是管人和管事，而管人，就要借助流程规范来管理。

因为如果在项目管理中，过于依赖人的管理，项目经理就会成为瓶颈，大事小事都需要项目经理来决策。再说项目经理也不能保证每次决策的正确性，如果决策失误，会很可能导致一些冲突。

**而好的项目管理，不需要直接管人管事，而是管理好计划和流程规范；项目成员不需要按照项目经理的指令做事，而是遵循计划和流程规范。**

我以前工作过的一个项目组，一个项目持续了好多年，中间人换了一批又一批，甚至有时候连项目经理都空缺，而项目一直井然有序的进行着，没有出什么问题，靠的就是多年积累下来的适合项目组的流程规范。

就像在《[06 | 大厂都在用哪些敏捷方法？（上）](#)》这篇文章中描述的那样：项目成员日常从看板就可以知道要做什么任务，代码审查、自动化测试可以有效保证质量，项目文档可以保证新人加入时能快速上手，结对编程可以保证新人遇到问题可以得到直接的帮助。

还有一个常见场景就是需求变更，产品经理想加一个紧急需求，这通常是让项目经理为难的事情：加吧，影响项目进度，开发人员有意见；不加呢，可能客户或者产品经理有意见。一个不小心就两边都得罪了。

如果你有一个大家认可的需求变更流程，就不再需要靠项目经理一个人决定该不该加需求，而是通过流程，来大家一起决策是不是要加这个流程。

所以你看，**流程规范，看起来是约束，实际上你用的好的话，不仅可以提高团队效率，还可以将好的实践标准化流程化，让大家可以共享经验，还可以有效的管理项目。**

## **如何制定好流程规范？**

在项目管理中，难免要去制定流程规范。即使你不是管理者，也可以提出合理的流程规范，帮助把项目管理好。

有一个科学的制定流程规范的方法，可以让你更好地制定出好的流程规范。

## **制定流程规范的四个步骤**

对于流程规范的制定，可以通过四个步骤来开展。

## 第一步：明确要解决的问题

要制定一个流程规范，第一步就是明确你是要解决什么样的问题。项目中很多问题，都可以思考是不是能通过流程解决。

比如说有程序员在生产环境操作，误删了数据表，造成了严重问题。如果只是对程序员进行处罚，寄希望于小心谨慎避免类似问题，那么下一次还有可能会有类似的事情发生。

如果说在流程上规范起来，例如：数据库操作之前先备份数据库，事先写好 SQL 语句，需要有人审查，测试环境先测试通过，最后在生产环境执行，那么就可以避免以后再出现不小心删除数据表的事情发生。

## 第二步：提出解决方案

对于问题，也不用着急马上就想着用流程规范，可以先思考解决的方法，有了方法后再进一步思考是否能提炼流程规范。

那么方法和流程规范有什么区别呢？

相对来说，方法更有针对性，可能只适用于特定场景或者特定人，而要将方法上升到流程规范，则需要有一定的普适性，能变成具体的步骤或者标准，让每个人都能执行。

比如说服务器部署后出现问题，高手可能就直接上服务器操作，直接修改代码编译解决，这是一个解决方法，但这不能成为一个流程规范，因为换一个水平不行或者对代码不熟悉的人来做，可能会搞出更大的问题。这时候回滚操作就是一个相对普适的方法，可以变成一个部署后出现问题的流程。

在提出解决方案，制定开发流程时，可以参考借鉴软件工程中，大家公认的好的实践。比如说：

**敏捷开发的流程：**虽然你的项目不一定采用敏捷开发的方式，但是敏捷开发中一些好的流程是可以借鉴的，例如参考我之前文章提到的像看板、站立会议、持续集成，这些好的工作流程，都可以借鉴。

**代码规范：**其实很多公司都公开了他们的代码规范，可以直接基于这些规范制定团队的规范。例如说前端的有 Airbnb 的代码规范 [Airbnb JavaScript Style Guide](#)，Java 的有



[Google Java Style Guide](#) , .Net 的有[.NET Guide](#), 等等。

**源代码管理流程：**现在的源代码主流是 git，而基于 Git 的代码管理已经有很多成熟的流程规范可以参考。例如阮一峰老师写过的《[Git 使用规范流程](#)》、《[Git 工作流程](#)》和《[Git 分支管理策略](#)》，或者 Github 官方出品的《[Understanding the GitHub flow](#)》，Gitlab 官方推荐的《[Introduction to GitLab Flow](#)》。

**部署流程：**十年前，每日定时构建还是很时髦的部署流程，而现在，主流的部署流程已经变成了持续部署，每次代码合并到主分支都可以触发一次自动部署，这样一有问题，就能马上知道发生在哪个环节。

像这样的好的流程还有很多，在我们专栏会介绍一些。如果平时多留心，你也可以学到很多。

### 第三步：达成共识，推广执行

在流程规范提出后，还需要得到大家认可，只有大家认可，达成共识，才能共同遵守，保障制度的执行。

对于大家都认可、很重要的流程规范，一定要让大家严格遵守，必要的时候需要配合一些奖惩制度，以保障其执行。

比如说流程规范的执行和绩效考评挂钩，对于没有执行的需要私下沟通提醒，严重的需要批评教育。否则流程规范会形同虚设，没有太大的意义。

### 第四步：持续优化，不断改进

流程制定后，在实际执行的时候，难免发现一些不合理或者不科学的地方，这时候就需要对其进行调整。

还有一些流程规范，随着时间推移，可能已经不能适合要求了，也需要考虑改进甚至抛弃，不然反而会成为一种阻碍。

比如说以前采用瀑布模型开发时，项目经理因为需要了解进度，所以每个项目成员要写日报，如果有站立会议了，日报这种形式就可以完全被站立会议替代，没有再存在的必要。

通过以上四个步骤，你就可以将日常项目中遇到的一些问题，用流程规范的方式逐步管理起来，在实施的过程中再不断优化改进，淘汰不合适的流程规范。

## 将流程规范工具化

如果说，以前我还是人为去推动一些流程规范的执行，近些年，我越来越感觉到，**应该尽可能借助技术手段来推动甚至替代流程规范。**

例如说代码规范，以前代码规范的执行，主要靠反复的教育宣传和代码审查中一个个去检查。而现在，借助 VSCode 这种强大的 IDE，以及 ESLint 这种代码检查工具，可以方便的检测出不符合规范的代码，甚至于可以帮你直接格式化成满足代码规范的格式。

还有像保证代码质量的问题，早些年必须依赖测试人员大量手工的测试，而现在借助 CI (Continuous Integration, 持续集成)、自动化测试和 Git，可以保证代码必须在通过测试以后，才会合并到主分支，从而很好的保证了代码的质量。

就像任正非的《全面提升软件工程能力与实践，打造可信的高质量产品》公开信中题记的这一段话说的：

“软件工程”和“质量工程”需要依靠架构技术，而不是依靠 CMM 和 QA 管理流程。一切工程问题，首先要思考能否通过技术解决，当前技术无法解决的问题，暂时由管理手段代劳，同时不停止寻找技术手段。

“软件工程”不要过于依赖流程和管理手段，要思考怎么通过技术手段去解决问题。

## 总结

流程和规范，就像红绿灯一样，不是一种约束，而是牺牲一点个体利益，提高团队效率；流程和规范将好的实践标准化流程化，让大家可以共享经验；流程和规范，让项目管理从人治变成“法治”。

要制定好项目规范，先明确要解决的问题，然后提出解决方案，看是否可以通过流程规范来解决，有了方案后需要团队成员一起达成一致，最后再推广执行。在执行过程中需要持续的优化，不断改进。

对于需要手动操作的流程，可以思考是不是能采用技术手段自动化，通过技术手段去解决。

## 课后思考

你所在项目中有没有不合理的流程规范？或者欠缺流程规范导致混乱的情况？如果有，你觉得可以制定什么样的流程规范来改善？欢迎在留言区与我分享讨论。

感谢阅读，如果你觉得这篇文章对你有一些启发，也欢迎把它分享给你的朋友。

 极客时间

# 软件工程之美

## 重新理解软件工程

宝玉

Groupon 资深工程师  
微软最有价值专家



新版升级：点击「 请朋友读」，10位好友免费读，邀请订阅更有**现金**奖励。

© 版权归极客邦科技所有，未经许可不得传播售卖。页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇 11 | 项目计划：代码未动，计划先行

下一篇 13 | 白天开会，加班写代码的节奏怎么破？

## 精选留言 (20)

写留言



kirogiyi

2019-03-24

4

流程和规范相当于国家的法律法规，制定不好引起矛盾，执行不好引起反抗。

良好的流程和规范的制定，需要根据当前公司文化、发展方向、人才培养等几方面综合考



虑。从公司角度来看，长远发展、快速发展是第一位；从员工的角度来看，舒适度、积极性、成长空间、既得利益是第一位。但怎么来控制这两者之间的平衡，宝玉老师能否帮...  
展开 ▾

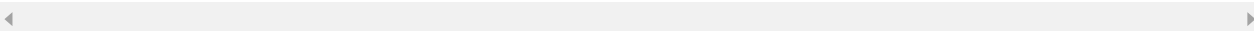
作者回复: 这是个好问题。

员工利益和公司利益其实大部分都是一致的，否则也无法走到一起。公司发展，员工才有机会成长，获得更多利益；员工成长，公司也会有更大发展。

作为管理者，不能一味站在公司利益不考虑员工利益，这样的管理者得不到拥戴；也不能一味站在员工利益不考虑公司利益，这样的管理者公司是不喜欢的。

皮之不存，毛将焉附。首先要带领团队为公司创造价值，然后才能有更多话语权，从而可以帮助员工谋取更多福利，帮助他们成长。

供参考。



纯洁的憎恶

2019-03-23

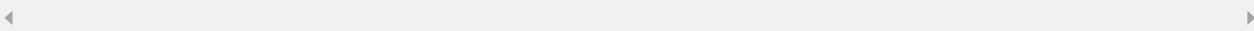
👍 4

流程规范是工业化大协作的基础，它不仅可以提高整体效率、细化分工协作、人治变法治，且标准统一、易扩展推广，还具备极大自动化潜力。

通过共商流程规范，让客户、产品经理、项目经理、开发等角色充分参与到需求变更过程，从而高效寻找到各方均能接受的平衡点。这个思路令我眼前一亮！感谢老师🙏...  
展开 ▾

作者回复: 🙏其实我没说啥，是你自己想到的思路😊

需求变更的话题，我会在后面章节（20 | 如何应对让人头痛的需求变更问题？）再细讲，帮你分析背后的动机，从而找到可能的解决方法，除了流程规范，其实还有其他办法可想。



hua168

2019-03-23

👍 2

我之前一个公司没有运维，一接手，那个乱呀:

- 1.放网站没有统一路径，还好网站目录放在一个大目录中，
- 2.到处乱挂NFS，空间不够就挂NFS来解决😓，而且没有放进开机启动里，弄的服务器出问题都不敢重启！😓
3. 每台服务器上面都不知道装哪些东西，要一个一个进程去查！😓...

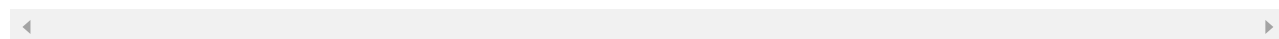
展开 ∨

作者回复: 你这已经做的很好了👍

我没运维经验，我的建议是：

你的运维，一定要尽可能脚本化自动化，比如说，脚本规定好放在什么目录，你不需要人工去放这些目录，每次部署的时候，将程序打包好（需要定义好格式），然后执行你的部署脚本，自动放到应该放的地方、自动更新配置文件。

自动化才是最好的流程！



**起而行**

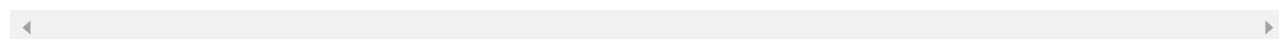
2019-03-23

👍 2

Python的缩进机制为代码设立了很好的规范。在开始，认为Python的缩进要求苛刻降低了灵活性。但后来，看到c++的大型项目中，大括号的风格各式各样，可读性低，难以辨认。这才知道，Python的缩进规范，能使得别人阅读源码时，能更好地接受他人风格。当然，另一方面，在大型项目上，也很少有人使用python

展开 ∨

作者回复: Python我用的少，现在写js代码，每次保存文件的时候vscode自动把代码格式的整整齐齐漂漂亮亮的，感觉特别好！



**起而行**

2019-03-23

👍 2

可能在学校的小组合作课设里，如何分工是个较大的问题。并且经常难以分工，比如3个人的小组，课设要求做数据爬取和数据分析，而组内三个人对这两项基础都是现学现用。这个时候做数据爬取的同学做完后，可能看不到数据分析的同学付出了多少，任务量是多少，有可能造成双方认为对方做得少了，有什么好的解决办法吗？

展开 ∨

作者回复: 在用《构建之法》当教材的学校，做项目，一个阶段后要换组的，这样可以去体验一下其他人做的事情。

我觉得这个可以借鉴一下，把课设分成几个阶段，到一个阶段了换一下分工，应该可以很好解决。

供参考。



titan

2019-03-25

👍 1

我曾经写过一篇文章：论企业管理的核心，请老师指正：

[https://mp.weixin.qq.com/s?](https://mp.weixin.qq.com/s?__biz=MzU2MzgxdYwNQ==&mid=2247483679&idx=1&sn=b62016eb0397d0ec6)

[\\_\\_biz=MzU2MzgxdYwNQ==&mid=2247483679&idx=1&sn=b62016eb0397d0ec6](https://mp.weixin.qq.com/s?__biz=MzU2MzgxdYwNQ==&mid=2247483679&idx=1&sn=b62016eb0397d0ec6)

展开 ∨

作者回复：已拜读，写的非常好👍

而且已经超越了项目管理，到了企业管理的高度。

其中的企业规章制度和项目的管理流程还是有很多相似相通之处



Charles

2019-03-24

👍 1

目前小团队更多的还是依赖于人，流程规范也有，像老师课程里说的很多规范没有执行到位，有点像摆设！比如有版本概念，版本目标也比较清晰，口头+issue，也有git工作流（改进版），但是总感觉版本细节任务不够清晰，时间估算可能不准，没有站立会议也没有日报几天才会了解进度更多靠自觉，经常版本延误！

...

展开 ∨

作者回复：站立会议我强烈建议你开一下，5-6人其实刚好的，超过7个人就要考虑分开开了，一起15~20分钟就结束了，组织的好，可以实时了解进度，及时发现障碍。

前期可以先试行一段看看效果，主持的时候不要发散：重点说做过什么、什么计划、有没有什么障碍。



小伟

2019-03-23

👍 1

好的流程规范需要项目管理的人（也可能是产品经理或技术经理）制定和发布，并设定奖惩机制，看到很多团队制定者都不遵守，后来形同虚设，所以关键是执行起来，反复迭代即可。

展开 ▾

作者回复: 是呀，制订了还不执行，那还不如不制定呀！

不执行，也要分析原因：

1. 是不是不具备可行性
2. 是不是没有严格执行

如果能改进就改进，不然就不如取消。



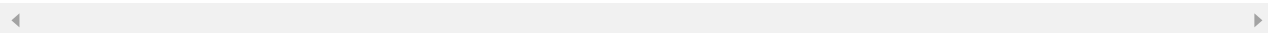
敖天羽

2019-04-23

👍

老师说的很对，目前我在推广中也遇到了一些问题。指正一点，VSC 不是 IDE 哦。

作者回复: 谢谢指正 🙏♂



Joey

2019-04-11

👍

请教宝玉老师：站在开发部门层面，如何做好研发质量管理？或者说有什么管理机制或手段可以分享下？

我们现在有的流程机制有：

- 1、瀑布模式下的需求、开发、测试、安全等各环节的过程管控；...

展开 ▾

作者回复: 一点建议：

首先，人是其中最大的变数也是最重要的因素，要保证团队有一批靠谱的水平不错的人。多花一点时间精力在团队的招聘和培养上面。

然后要建设好流程制度上，从制度上保障质量。

比如说：

- 在关键环节要有严格的评审，需求评审，设计评审；

- 单元测试、代码审查都是行之有效的手段；
- 把基于CI的代码自动测试、发布流程要建立起来；
- 有规范的Bug跟踪和修复流程；
- 对于线上故障有合理的应急处理流程。

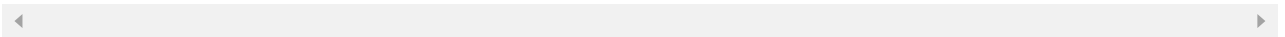
量化的数据是很好的参考，可以充分利用起来帮助及时发现风险。

给测试留有充足的时间，不要压缩测试时间。测试后期，要冻结需求，一边增加需求一边修改Bug质量是没法稳定下来的。

线上要有监控，对于关键的指标设置阈值和报警，比如：说http错误率超过5%报警。

项目结束后要有总结、反思和调整。

供参考！



**hua168**

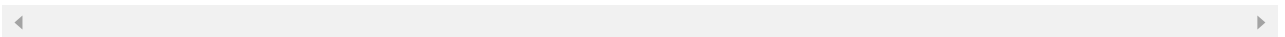
2019-03-28



我去猎聘、51job提前了解了一下技术管理方面的，运维主管/经理 看到有些要求会ITIL

作者回复: 那你可以学习一下，总不是坏事：)

但我觉得这一定不是主要的障碍，关键还是多积累项目管理经验。



**hua168**

2019-03-28



老师，

1.管理类要学ITIL的吗？

2.它的六个模块:即业务管理、服务管理、ICT基础架构管理、IT服务管理规划与实施、应用管理和安全管理。

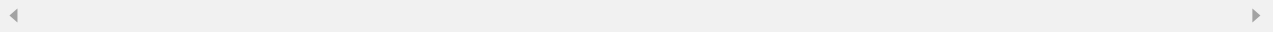
都要学吗？ ...

展开 ∨

作者回复: ITIL我真的不了解，不好妄加评论。



IT行业的管理一个就是软件工程，对软件项目过程的管理，一个就是项目管理，对项目人和事的管理。



**天之大舒**

2019-03-26

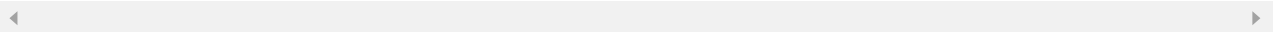


制度流程和规范的目的是减少沟通提供效率，成员行动一致，做事是步骤，结果检查有标准。

由于行业特性，我们系统的更新必须是到客户现场进行，之前由于没有制定整理更新包的步骤和现场更新的步骤，出问题时，导致准备更新包的同事和现场更新的同事的矛盾，客户也很有意见，认为做事不专业。后面我们制定了更新包准备清单、更新包验证清单， ...  
展开 ∨

作者回复: 谢谢补充，很好的案例。

如果这个更新包发布频率比较高，建议可以考虑是不是可以写个脚本自动化，应该能提高效率和减少出错。



**Felix**

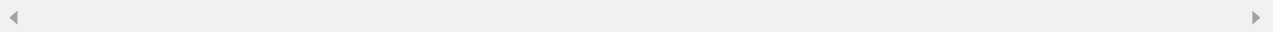
2019-03-26



流程和规范是最好的工作助手！

展开 ∨

作者回复: 是的，也要记住：看是不是可以用工具或自动化替代。



**小先生**

2019-03-25



流程规范，对于个人来说可能会影响效率，但对于整个团队来说，肯定是利大于弊的。

1 找出需要解决的问题 2 提出解决方案  
3 达成共识和推广 4 不断更新，改进

...

展开 ∨

作者回复: 总结的不错👍



一路向北

2019-03-25



我们目前的开发可以说都是按照一种下意识的流程来做，一个原因是团队小，经常是一个人负责一大块事情，和别人之间的接口大多数情况下就是协议，另外一个原因是，大家心底里遵循的流程是比较固化了。

这里一个比较大的问题是，随意性比较强，特别是在遇到一些问题的时候，或者时间紧的时候，代码不review了，需求分析也精简了，甚至测试就只做增量部分了.....

展开▼

作者回复: 磨刀不误砍柴工，有些不能省，比如代码审查，至少要做到心中有数，不然将来也需要还账的。

时间紧也可以有时间紧的流程。比如说时间紧来不及写测试，代码审查发现的暂时不修改的代码，那么就应该创建一个Ticket后续跟踪，时间不紧的时候把来不及做的事情补回去。



alva\_xu

2019-03-25



另外，我查了一下，老师引用的那句任总的话，并未出现在他的公开信里，而是编辑的题记。

作者回复: 🙏谢谢指正，我让编辑同学帮助更正一下



alva\_xu

2019-03-25



我来谈谈对老师讲的几个点的个人看法和实践。

1，方法和流程规范的区别。

老师讲的很对，流程规范是在很多经验总结后形成的。从ITIL流程来说，这里的方法实际上可以理解为事件管理的范畴，就是发现了一个incident，想办法去解决，甚至用work around 的方法去解决。当相同的incident发现次数多了，在review的时候，事件就上升...

展开 ∨

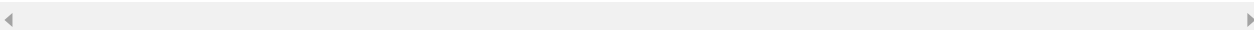
作者回复: 🙏 谢谢高质量的补充!

需求文档验收可以通过需求评审会议, 评审时开发和测试都要有代表参加, 一个是提出反馈, 另一个是及早了解需求。评审会议通常要开几次才能最终定下来。

测试用例通常是产品经理协助验收或者辅助确认。

原来我们在飞信时, 会有一个模拟生产环境的压力测试环节, 从生产环境同步真实数据过去, 规模按生产环境比例缩放。

还有的压力测试是直接在生产环境做的, 在半夜人流量少的时候。



张驰

2019-03-25

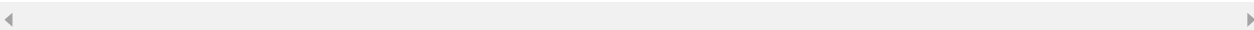


在日常工作中, 流程应该由谁来制定呢? 普通开发人员还是领导者, 亦或者是公司有这种专职专岗的人。往往很多人都能够发现问题, 甚至也有一些自己解决问题的方式方法, 但是要想具体流程化对公司整体产生作用, 往往感觉是有力无门, 没有一个好的渠道。

作者回复: 一个好的流程经常是跟问题切实相关的人员提出来的, 或者把问题反馈出来, 大家一起想办法, 最后由项目经理或者部门负责人帮助落实推广。

其实像敏捷开发每次迭代结束后的Sprint回顾会议就是一个很好的讨论问题的方式。

可以考虑参考Sprint回顾会议的做法, 定期有专门的会议可以讨论这样的问题。另外如果有组员之间的1-1会议, 也是个不错的讨论这样问题和解决方案的途径。也可以通过邮件、聊天工具讨论解决。



williamcai

2019-03-25



其实公司都有规范, 但是执行力有待提高

展开 ∨

作者回复: 反思执行力的同时，也要思考一下规范是不是制订的科学合理，有时候是规则本身的问题导致难以执行的

