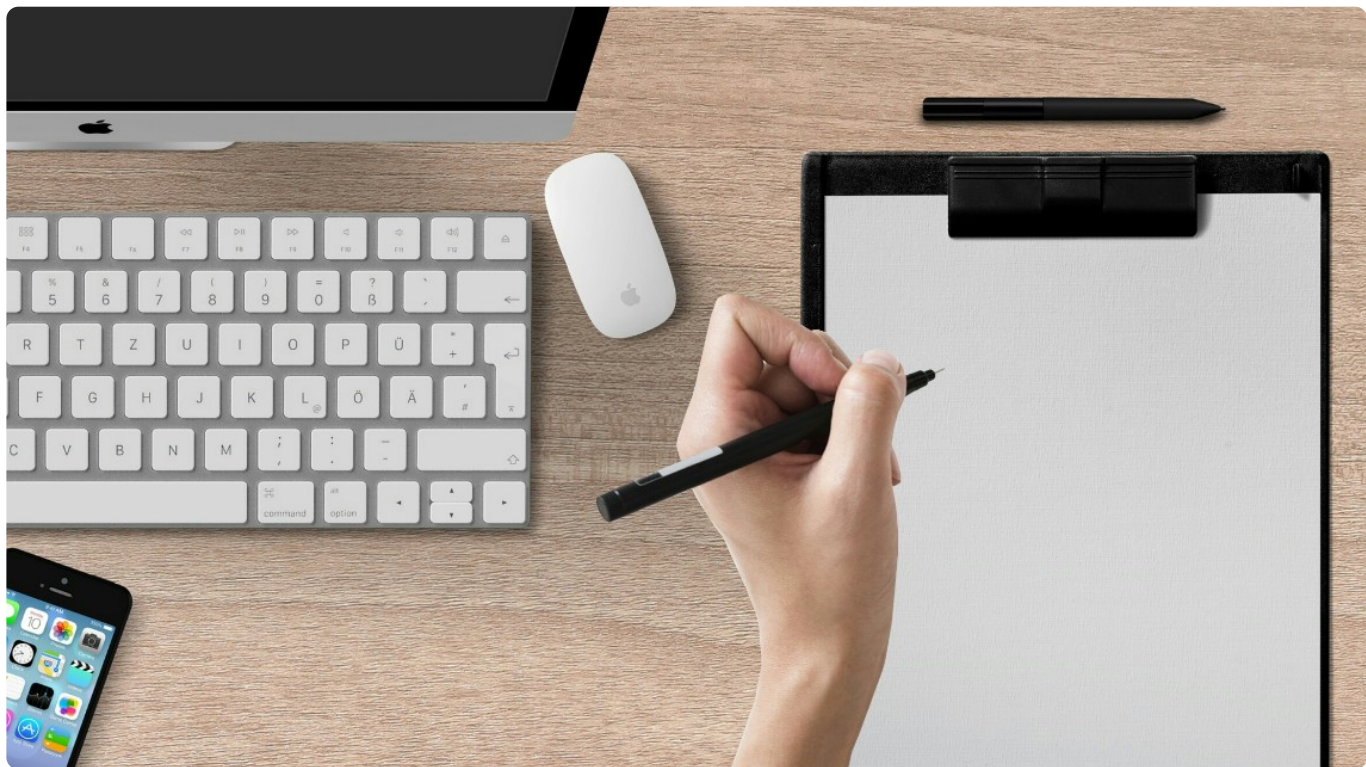


17 | 测试左移：测试如何应对新的开发模式？

2019-09-30 葛俊

研发效率破局之道

[进入课程 >](#)



讲述：葛俊

时长 12:04 大小 11.07M



你好，我是葛俊。今天，我们来聊聊测试这个话题。

为什么需要测试左移，测试右移？

测试可以保证产品质量，重要性不言而喻。但，要做好测试也比较困难，需要克服很多挑战。尤其是，持续交付、敏捷开发等开发模式为传统软件测试方式带来了更大的时间压力。

我们先来看看下面这种熟悉的测试方式都有什么问题：测试人员接到项目后参与需求评审，然后根据需求文档写用例、准备脚本，等开发提测之后正式开始测试、提 Bug、回归，测试通过后就结束了，项目交给运维上线，之后投入下一个项目继续重复这样的流程。

这样的流程看似没什么错，但有两大问题：

测试人员非常被动。当需求质量、开发质量较差的时候，测试人员只能被动接受。

但同时，测试又被认为是质量的责任人。如果因为需求质量、开发提测质量差而导致上线延期，大家通常会首先怪罪测试团队。

这些问题，在新的开发模式下愈发严重。因为这些新的开发模式有一个共同点，就是要缩短产品的交付周期，对自动化的要求越来越高，能够专门留给传统竖井流程中测试环节的时间越来越短，自然更难保证质量。

在极端的情况下，比如在持续部署的模式下，所有测试都是自动化的，已经完全没有留给测试人员专门进行手工测试的时间了。与此同时，测试的能力和质量又是这些开发模式成功的关键。否则，即使可以频繁地构建产品，质量不过关价值也为零。在我看来，《[持续交付：发布可靠软件的系统方法](#)》这本关于持续交付经典图书，更像是一本介绍测试的书，也是因为这个原因吧。

所以，在快速开发模式的挑战下，测试左移、测试右移就应运而生了。这些测试模式，能让测试人员拥有更多主动权，以及更多的时间进行测试。那，到底什么是测试左移和测试右移呢？

什么是测试左移和测试右移？

测试左移和右移，就是把测试的范围从传统测试的节点中释放出来，向左和右扩展。

向左扩展，就是让测试介入代码提测之前的部分。比如，扩展到开发阶段，在架构设计时就考虑产品的可测试性，并尽量进行开发自测等。另外，测试可以更进一步扩展到需求评审阶段，让测试人员不只是了解需求，更要评估需求的质量，比如分析需求的合理性以及完整性等。

类似的，测试右移，是让测试介入代码提测之后的部分。比如，测试人员在产品上线过程中，利用线上的真实环境测试。另外产品上线之后，测试人员仍然介入，通过线上监控和预警，及时发现问题并跟进解决，将影响范围降到最低。这样一来，测试人员不但有更多的时间进行测试，还能发现在非生产环境中难以发现的问题。

从测试右移的概念中可以看出，它与我们下一篇文章“蓝绿红黑灰度发布：这些五颜六色的发布到底怎么用？”中的各种部署模式强相关，所以我会下一篇文章中与你介绍。而在今天这篇文章中，我会与你详细介绍测试左移的原则和实践。

测试左移的原则和实践

在我看来，要做好测试左移，有 3 个基本原则：

调整团队对测试的态度；

把测试添加到开发和产品需求步骤中；

频繁测试，快速测试。

接下来，我们具体看看这 3 个原则吧。

测试左移原则一：调整团队对测试的态度

调整团队对测试的态度，打破竖井的工作方式，是测试左移的前提。**一个有效的办法是，按照功能的维度管理团队，让整个功能团队对产品负责。**也就是说，如果产品质量出现问题，不只是测试团队“背锅”，而是会影响整个功能团队的绩效。同时，让质量问题的直接责任人承担更多的责任，来进一步增强团队成员的责任心。这种利益绑定的办法，虽然简单但非常有效，只不过出现质量问题时要记得进行根因分析，以避免再次出现类似问题。

另外，还要改变团队成员对测试工作的认知。传统的工作方式中，我们通常认为发现 Bug 最重要，但其实为了提高产品质量，更重要的是预防 Bug。所以说，在测试左移的过程中，我们应该更聚焦在预防 Bug 上。

测试左移原则二：把测试添加到开发和产品需求步骤中

测试左移的第一步，是把测试工作融入到开发步骤中。常用的办法是，让测试人员参与到开发阶段的方案设计中，了解开发的实现方式。因为很多开发人员可能只熟悉他负责的那一部分，而测试人员往往对全局更加了解，所以测试人员要评估改动范围以及是否有遗漏的模块和系统。

另外一个比较彻底，也很有效的方法是，我在[第 8 篇文章](#)中提到的全栈开发。当时，我们讨论全栈开发的场景主要是，通过运维团队提供工具和支持，让开发人员尽量参与到运维工作中去。对于测试来说，也是同样的道理。我们可以让测试团队转型，进行工具开发，并更多地去支持专项测试，比如性能测试、安全测试等，通过“使能”的办法，让开发人员完成功能测试，包括单元测试、集成测试等。

说到让开发人员完成部分测试工作，常常会听到很多质疑声。反对者认为，测试人员的心理模型跟开发人员不一样，他们更倾向于去找问题。而开发人员面对自己开发的产品，潜意识里就不愿意去找问题，比如，他们不愿意专门尝试各种边界输入进行测试，而把自己开发的功能搞崩溃。所以，开发人员和测试人员更适合分开。

这种观念，在 10 年前瀑布开发模式盛行时就深入人心。我曾经也非常认同，但在 Facebook 工作了 5 年后改变了看法。如果你能够把开发人员的责任界定得很清楚，谁开发的产品谁要保证质量，那么开发人员自然而然地就会去尝试做好测试，比如进行边界测试。而且，开发人员最了解自己写的代码，所以他能够最高效地对自己的代码进行测试。

当然，做全栈开发的同时我们仍会保留一部分功能测试人员，毕竟从竖井模式转变到全栈模式是一个循序渐进的长期过程。不过 Facebook 做到了极致，完全没有了功能测试人员，也就是我们所说的“去 QA”。

测试左移到了开发阶段之后，再往左移一步就到了产品设计阶段，在这里，测试人员除了解需求外，更重要的是评估需求的质量。

我推荐使用 BDD (Behavior Driven Development, 行为驱动开发) 的方法进行开发，促进团队在需求评审时更多地考虑测试。BDD 是通过特定的框架，用自然语言或类自然语言，按照编写用户故事或者用例的方式，从功能使用者的视角描述并编写测试用例，从而让业务人员、开发人员和测试人员着眼于代码要实现的业务行为，并以此为依据通过测试用例进行验证。

这里有一篇[关于 BDD 的文章](#)，推荐你阅读。

测试左移原则三：频繁测试，快速测试

测试左移的第三个重要原则是，频繁测试、快速测试。在测试左移之前，我们需要等待提测，比较被动，不能频繁测试。但测试左移到开发阶段之后，我们就有了很大的自由度去频繁运行测试，从而更好地发挥测试的作用，尽早发现更多的问题。

这里最重要的方法就是，我在讲持续开发时提到的几个关于验证的操作，具体包括：

- 规范化、自动化本地检查；

- 建设并自动化代码入库前的检查流程；

提供快速反馈，促进增量开发。

你可以再复习下[第 5 篇文章](#)中的相关内容。

另外，为了能够顺利、频繁地运行测试，我们还要提升测试运行的速度。给测试提速的常见办法包括：

并行运行。比如把测试用例放到多台机器上运行，用资源换时间。

提高构建速度。比如使用精准构建，因为通常构建之后才能运行测试。

精准测试，也就是只运行跟改动最相关的测试。可以建立需求与代码的关系，以及需求与测试用例的关系，从而在代码改动时重点关注与之最相关的测试用例。

分层测试，即不同情况运行不同测试用例集合。

减少不必要的用例。比如，识别不稳定的用例，对其删除或者优化。

在这里，**我再与推荐两个精准构建和精准测试的工具**。一个是，Facebook 使用并开源的 [Buck](#) 系统，可以用来提高构建速度；另一个是，Google 开源的 [Bazel](#)，支持精准构建和精准测试。

小结

在敏捷、持续交付等开发模式愈发流行的今天，产品的研发节奏越来越快，传统的开发提测之后进行测试，然后交给运维上线的测试模式受到很大的挑战。由此促成了测试左移和右移等新的测试方法，也就是测试向左扩展到产品设计、开发流程中，向右扩展到发布、生产流程中，从而在整个研发流程中持续关注测试，解决新模式给测试带来的挑战。

而测试左移，本质上是尽早发现问题、预防问题。基本原则包括：从人的角度出发，让产品、开发、运维人员统一认识，重视测试；从流程上，让测试融入产品设计和开发步骤中；快速测试、频繁测试。

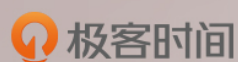
其实，软件开发行业早就达成了共识，问题发现得越晚，修复代价越大。《代码大全》这本书，从软件工程实践的角度说明了修复一个 Bug 的成本在产品需求分析阶段、设计阶段、开发阶段、测试阶段有着天壤之别。比如，在集成阶段修复一个 Bug 的成本是编码阶段的 40 倍。除了成本悬殊之外，在修复难度、引入新问题的可能性、沟通成本、团队状态等方面也有很大的影响。

在我看来，Facebook 正是成功进行了持续测试，让测试融入到了整个研发流程中，从而没有 QA 团队也能保证产品的高质量。

思考题

你认为测试左移和测试右移，会不会减少测试人员的工作机会呢？如果你是测试人员，又应该怎么面对这个新的测试模式带来的挑战呢？

感谢你的收听，欢迎你在评论区给我留言分享你的观点，也欢迎你把这篇文章分享给更多的朋友一起阅读。我们下期再见！



研发效率破局之道

Facebook 研发效率工作法

葛俊

前 Facebook 内部工具团队 Tech Lead



新版升级：点击「 请朋友读」，20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 16 | 高效上云：如何用云计算来提高效能？

精选留言 (3)

写留言



李双

2019-09-30

有学习一新思路

展开 ▾

💬 1

👍 1



刘丹

2019-09-30

测试左移和右移，减少了耗时长长的功能测试（例如从3个人变成1个人），增加了自动化测试（例如从1个人变成2个人），提升了测试的效率，总体上应该是减少了测试人员岗位数量。

作为测试人员，要用全栈开发的标准来要求自己，不仅要横向发展，也要竖向发展。例如在学习多种测试技能（功能测试、自动化测试、性能测试、安全测试）、完成本岗位工...

展开 ▾

💬 1

👍 1



CountingStars

2019-09-30

测试左移是 TDD/BDD 能充分发挥作用的地方了。右移大部分时间应该在预发布环境（共享生产环境数据）进行测试。

展开 ▾

💬

👍