

05 | 价值流分析：关于DevOps转型，我们应该从何处入手？

2019-10-22 石雪峰

DevOps实战笔记

[进入课程 >](#)



讲述：石雪峰

时长 16:31 大小 15.13M



你好，我是石雪峰。

关于“DevOps 如何落地”的问题，向来是关注度很高的，所以，从今天开始，我会用 16 讲的篇幅跟你聊聊这个话题的方方面面。作为“落地实践篇”的第 1 讲，我先跟你聊聊 DevOps 转型的那些事儿。

相信你一定听说过持续交付吧？现在，几乎每家实施 DevOps 的企业都宣称他们已经有了一套持续交付平台，或者是正在建设持续交付平台。但是，如果你认为只需要做好持续交付平台就够了，那就有点 OUT 了。因为现在国外很多搞持续交付产品的公司，都在一门心思地做另外一件事情，这就是 VSM 价值流交付平台。

比如，Jenkins 的主要维护者 CloudBees 公司最新推出的 DevOptics 产品，主打 VSM 功能，而经典的持续交付产品 GoCD 的 VSM 视图也一直为人所称道。那么，这个 VSM 究竟是个啥玩意儿呢？

要说清楚 VSM，首先就要说清楚什么是价值。简单来说，**价值就是那些带给企业生存发展的核心资源**，比如生产力、盈利能力、市场份额、用户满意度等。

VSM 是 Value Stream Mapping 的缩写，也就是我们常说的价值流图。它起源于传统制造业的精益思想，用于分析和管理一个产品交付给用户所经历的业务流、信息流，以及各个阶段的移交过程。

说白了，VSM 就是要说清楚在需求提出后，怎么一步步地加工原材料，进行层层的质量检查，最终将产品交付给用户的过程。通过观察完整流程中各个环节的流动效率和交付质量，识别不合理的、低效率的环节，进行优化，从而实现整体效率的提升。

这就好比我们在餐厅点了一道菜，这个需求提出后，要经历点单、原材料初加工（洗菜）、原材料细加工（切菜）、制作（炒菜），最终被服务员端到餐桌上的完整过程。但有时候，厨师已经把菜做好摆在窗口的小桌上了，结果负责上菜的服务员正在忙，等他（她）忙完了，才把菜端到我们的餐桌上，结果热腾腾的锅气就这么流失了。

对软件开发来说，也是如此。由于部门职责的划分，每个人关注的都是自己眼前的事情，这使得软件交付过程变得碎片化，以至于没有一个人能说清楚整个软件交付过程的方方面面。

所以，通过使用价值流图对软件交付过程进行建模，使整个过程可视化，从而识别出交付的瓶颈和各个环节之间的依赖关系，这恰恰是“DevOps 三步工作法”的第一步“流动”所要解决的问题。

我简单介绍下“DevOps 三步工作法”。它来源于《DevOps 实践指南》，可以说是整本书的核心主线。高度抽象的“三步工作法”，概括了 DevOps 的通用实施路径。

第一步：流动。通过工作可视化，限制在制品数量，并注入一系列的工程实践，从而加速从开发到运营的流动过程，实现低风险的发布。

第二步：反馈。通过注入流动各个过程的反馈能力，使缺陷在第一时间被发现，用户和运营数据第一时间展示，从而提升组织的响应能力。

第三步：持续学习和试验。没有任何文化和流程是天生完美的，通过团队激励学习分享，将持续改进注入日常工作，使组织不断进步。

关键要素

你并不需要花大力气去研究生产制造业中的价值流分析到底是怎么玩的，你只要了解有关 VSM 的几个关键要素和核心思想就行了。那么，VSM 中有哪些关键要素和概念呢？有 3 点是你必须要了解的。

1. **前置时间**（Lead Time，简称 LT）。前置时间在 DevOps 中是一项非常重要的指标。具体来说，它是指一个需求从提出（典型的就创建一个需求任务）的时间点开始，一直到最终上线交付给用户为止的时间周期。这部分时间直接体现了软件开发团队的**交付速率**，并且可以用来计算**交付吞吐量**。**DevOps 的核心使命之一就是优化这段时长。**
2. **增值活动时间和不增值活动时间**（Value Added Time/Non-Value Added Time，简称 VAT/NVAT）。在精益思想中，最重要的就是**消除浪费**，也就是说最大化流程中那些增值活动的时长，降低不增值活动的时长。在软件开发行业中，典型的不增值活动有很多，比如无意义的会议、需求的反复变更、开发的缺陷流向下游带来的返工等。
3. **完成度和准确度**（% Complete/Accurate，简称 %C/A）。这个指标用来表明工作的质量，也就是有多少工作因为质量不符合要求而被下游打回。这里面蕴含了大量的沟通和返工成本，从精益的视角来看，也是一种浪费。

在实践中，企业往往将需求作为抓手，来串联打通各个环节，而前置时间是需求管理的自然产物，采集的难度不在于系统本身，而在于各环节的操作是否及时有效。有的团队也在使用需求管理工具，但是前置时长大多只有几秒钟。问题就在于，他们都是习惯了上线以后，一下子把任务状态直接从开始拖到最后，这样就失去了统计的意义。

需要注意的是，关于前置时间，有很多种解释，一般建议采用**需求前置时间**和**开发前置时间**两个指标进行衡量，关于这两个指标的定义，你可以简单了解一下。

需求前置时间：从需求提出（创建任务），到完成开发、测试、上线，最终验收通过的时间周期，考查的是团队整体的交付能力，也是用户核心感知的周期。

开发前置时间：从需求开始开发（进入开发中状态），到完成开发、测试、上线，最终验收通过的时间周期，考查的是团队的开发能力和工程能力。

对于增值活动时长，我的建议是初期不用过分精细，可以优先把等待时长统计出来，比如一个需求从准备就绪，到进入开发阶段，这段时间就是等待期。同前置时间一样，很多时候，研发的操作习惯也会影响数据的准确性，比如有的研发喜欢一次性把所有需求都放到开发阶段，然后再一个个处理掉，这就导致很多实际的等待时间难以识别。所以，如果完全依靠人的操作来确保流程的准确性，就会存在很大的变数。**通过流程和平台的结合，来驱动流程的自动化流转，这才是 DevOps 的正确姿势。**

举个例子，研发开发完成发起提测后，本次关联的需求状态可以自动从“开发中”变成“待测试”状态，而不是让人手动去修改状态，这样就可以避免人为因素的影响。通过代码，流水线和需求平台绑定，从而实现状态的自动流转。

关于完成度和准确度，在使用 VSM 的初期可以暂不处理。实际上，我见过一些公司在跑通主流程之后，着手建设质量门禁相关的指标，比如研发自测通过率，这些指标就客观地反映了 VSM 的完成度和准确度。关于质量门禁，在专栏后面我会花一讲的时间来介绍，你一定不要错过。

方式

关于 VSM 的关键要素，知道这些就足够了。那么作为企业 DevOps 转型工作的第一步，我们要如何开展一次成功的 VSM 活动呢？一般来说，有 2 种方式。

1. 召开一次企业内部价值流程梳理的工作坊或者会议。

这是我比较推荐的一种方式。对于大型企业而言，可以选取改进项目对象中某个核心的业务模块，参加会议的人员需要覆盖软件交付的所有环节，包括工具平台提供方。而且，参会人员要尽量是相对资深的，因为他们对自身所负责的业务和上下游都有比较深刻的理解，比较容易识别出问题背后的根本原因。

不过，这种方式的实施成本比较高。毕竟，这么多关键角色能够在同一时间坐在一起本身就比较困难。另外，面对面沟通的时候，为了给对方保留面子，大家多少都会有所保留，这样就会隐藏很多真实的问题。

所以，一般情况下，像团队内部的**敏捷回顾会**，或者是**版本发布总结会**，都是很合适的机会，只需要邀请部分平常不参会的成员就行了。

2.内部人员走访。

如果第 1 种方式难以开展，你可以退而求其次地采用第 2 种方式。通常来说，企业内部的 DevOps 转型工作都会有牵头人，甚至会成立转型小组，那么可以由这个小组中的成员对软件交付的各个环节的团队进行走访。这种方式在时间上是比较灵活的，但对走访人的要求比较高，最好是 DevOps 领域的专家，同时是企业内部的老员工，这样可以跟受访人有比较深入坦诚的交流。

无论哪种方式，你都需要识别出几个关键问题，缩小谈话范围，避免漫无目的地东拉西扯，尽量做到有效沟通。比如，可以建立一个问题列表：

在价值交付过程中，你所在团队的主要职责是什么？

你所在团队的上下游团队有哪些？

价值在当前环节的处理方式，时长是怎样的？

有哪些关键系统支持了价值交付工作？

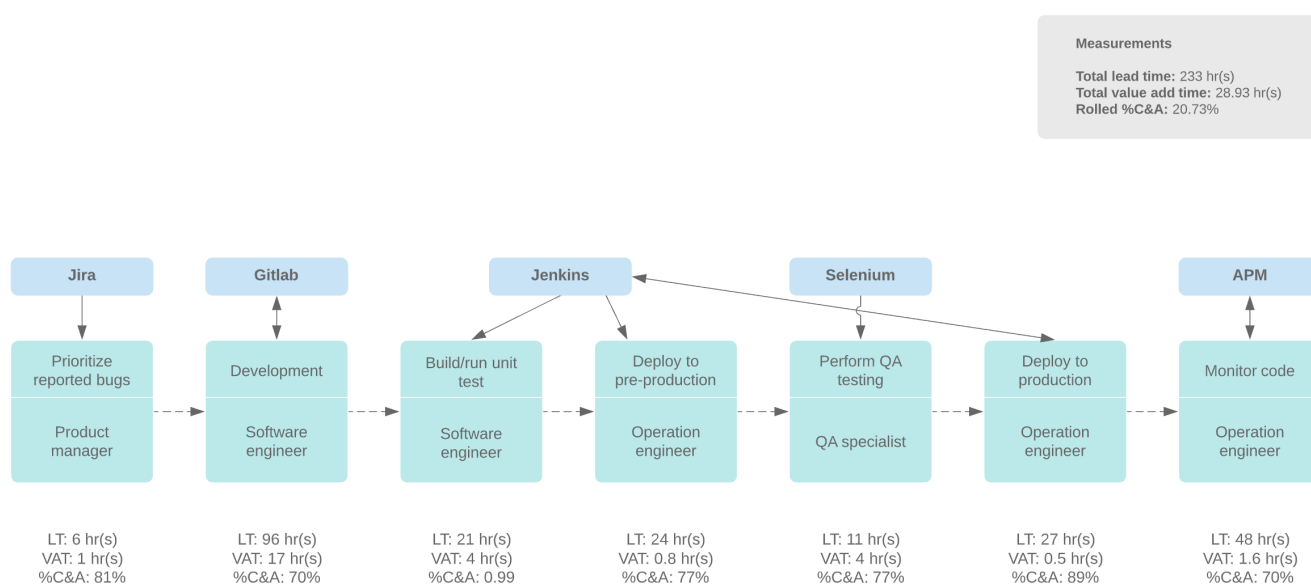
是否存在等待或其他类型的浪费？

工作向下游流转后被打回的比例是多少？

为了方便你更好地理解这些问题，我给你提供一份测试团队的访谈示例。

关键问题	参考答案
在价值交付的过程中，你所在团队的主要职责是什么？	我们是软件质量的把关人，负责交付软件的功能质量验收工作，避免缺陷流入线上环节。
你所在团队的上下游团队有哪些？	我们的上游是开发团队，下游是运维团队，另外还跟项目管理团队和安全团队有关系。
价值在当前环节的处理方式，时长是怎样的？	从迭代需求开始，我们就会参与需求评审，并进行测试设计。研发提测后，我们就会按照测试计划和案例对功能进行验收，如果验收通过，就给出测试报告并流转下游，如果存在问题，就提出缺陷。一般来说，平均测试流转周期为2.5天。
有哪些关键系统支持了价值交付工作？	包括研发提测系统，测试管理平台，缺陷系统。
是否存在等待和其他类型的浪费？	开发提测后一般都需要开发人员在测试环境部署，有时候开发人员不在就要等待，另外每次申请测试环境也需要向环境部门申请，往往几天才能下来。
工作向下游流转后被打回的比例有多少？	这个要看缺陷漏测率和线上缺陷逃逸率指标，我们有相关的数据，大概漏测率0.15%。

通过访谈交流，我们就可以对整个软件交付过程有一个全面的认识，并根据交付中的环节、上下游关系、处理时长、识别出来的等待浪费时长等，按照 VSM 模型图画出当前部门的价值流交付图，以及各个阶段的典型工具，如下图所示：



当然，实际交付流程相当复杂，涉及到多种角色之间的频繁互动，是对 DevOps 转型团队的一种考验。因为这不仅需要团队对软件开发流程有深刻的认识，还要充分了解 DevOps 的理念和精髓，在沟通方面还得是一把好手，能够快速跟陌生人建立起信任关系。

价值

话说回来，为什么 VSM 会是企业 DevOps 转型的第一步呢？实际上，它的价值绝不仅限于输出了一幅价值流交付图而已。VSM 具有非常丰富的价值，包括以下几个方面：

1. 看见全貌。

如果只关注单点问题，我们会很容易陷入局部优化的怪圈。DevOps 追求的是**价值流动效率最大化**，也就是说，就算单点能力再强，单点之间的割裂和浪费对于价值交付效率的影响也是超乎想象的。所以，对于流程改进来说，第一步，也是最重要的一步，就是能够看见全貌，这样才能从全局视角找到可优化的瓶颈点，从而提升整体的交付效率。

另外，**对于全局交付的建模，最终也会体现到软件持续交付流水线的建设上**，因为流水线反映的就是企业客观的交付流程。这也就很好理解，为啥很多做持续交付流水线的公司，现在都延伸到了价值流交付平台上。因为这两者之间本身就存在一些共性，只不过抽象的级别和展现方式不同罢了。

2. 识别问题。

在谈到企业交付效率的时候，我们很容易泛泛而谈，各种感觉满天飞，但感觉既不可度量，也不靠谱，毕竟，它更多地是依赖于个人认知。换句话说，即便交付效率提升了，也不知道是为啥提升的。

而 VSM 中的几个关键指标，也就是前置时长、增值和不增值时长，以及完成度和准确度，都是可以客观量化改进的指标。当面对这样一幅价值流图的时候，我们很容易就能识别出当前最重要的问题和改进事项。

3. 促进沟通。

DevOps 倡导通过团队成员间的沟通和协作来提升交付效率，但客观现实是，在很多企业中，团队成员基本都是“网友关系”。即便都在一个楼里办公，也会因为部门不同坐在不同的地方，基本上只靠即时通讯软件和邮件交流。偶尔开会的时候能见上一面，但也很少有深入的交流。如果团队之间处于你不认识我、我也不认识你的状况下，又怎么有效协作呢？

另外，很多时候，在我们开展 VSM 梳理的时候，团队才第一次真正了解上下游团队的职责、工作方式，以及让他们痛苦低效的事情。这时，我们通常会设身处地地想：“只要我们多做一点点，就能大大改善兄弟团队的生存状况了。”实际上，这种**同理心对打破协作的壁垒很有帮助**，可以为改善团队内部文化带来非常正面的影响。实际上，这也是我推荐你用会议或者工作坊的方式推进 VSM 的根本原因。

4. 驱动度量。

我们都认可数据的力量，让数据驱动改进。但是，面对这么庞杂的数据体系，到底哪些才是真正有价值的呢？VSM 就可以回答这个问题。

在 VSM 访谈的时候，我们要问一个团队的交付周期、准确率等指标问题，如果你发现这个团队支支吾吾，只能给出模糊的回答，这时你就要注意啦，这里本身就大有问题。因为这就表示当前环节的度量指标不够清晰，或者指标过于复杂，团队不清楚关键的结果指标。

另外，如果数据的提取需要大量时间，比如需要采用人为统计算数的方式，那么这就体现了这个环节的平台建设能力不足，无法自动化地收集和统计数据，甚至有些关键数据还没有沉淀到数据系统中，只能通过人工本地化的方式进行管理。

这些都是 DevOps 转型的过程中需要解决的问题，可以优先处理。可以说，**VSM 是一场团队协作的试炼**。收集 VSM 数据的过程本身，就需要平台间的打通和数据共享，以及自动化的推进，这有助于度量活动的开展。

5. 价值展现。

对于企业而言，任何投入都需要有产出。要实现 DevOps 的转型，企业需要投入大量的精力。那么如何让高层领导明白企业交付效率改善所带来的价值呢？价值流梳理就是一种很好的方式。因为 VSM 从价值分析而来，到价值优化而去，本身就是在回答 DevOps 对于企业的价值问题。

总结

在这一讲中，我给你介绍了 DevOps 转型的第一步——VSM 价值流图，包括它的来源、3 个关键要素，以及在企业中开展 VSM 的 2 种方式。最后，我介绍了 VSM 的 5 大价值，分别是看见全貌、识别问题、促进沟通、驱动度量和价值展现。

就像我们常说的，DevOps 转型是一场没有终点的旅程，VSM 的梳理也不会是一帆风顺的。因为对于企业价值交付流程的梳理，需要随着认知的深入不断地进行迭代和优化。不过，好的开始是成功的一半，当我们开始梳理 VSM 的时候，我们的着眼点就会慢慢调整到 DevOps 模式，并真正地开启我们的 DevOps 转型之旅。



思考题

最后，给你留一道思考题：你认为在公司内部梳理价值流的最大障碍是什么？在提取价值流图中的 3 个关键要素的数据时，你遇到过什么挑战吗？

欢迎在留言区写下你的思考和答案，我们一起讨论，共同学习进步。如果你觉得这篇文章对你有所帮助，欢迎你把文章分享给你的朋友。

DevOps 实战笔记

精要 30 计，让 DevOps 快速落地

石雪峰

京东商城工程效率专家



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 特别放送：成为DevOps工程师的必备技能（下）

下一篇 06 | 转型之路：企业实施DevOps的常见路径和问题

精选留言 (12)

写留言



桃子-夏勇杰

2019-10-22

需求提出时间的定义是什么？用户提出需求的时间，还是业务部门提出需求的时间，还是产品经理提交PRD的时间，还是提出就绪用户故事的时间？需求如何拆，规模多大？这些问题对前置时间的计算都有很大影响，想听一些这方面的实战经验。

展开

作者回复: 你好，你把下一讲的问题提前剧透了，以我们公司现有的实践来说，需求提出时间，具体指业务方提出需求的实践，以业务方在系统中录入原始需求为准。需求录入之后，会经过需求承接，产品设计，PRD评审，到准备就绪，这些就是需求管理层面的事情，再往后就进入了开发阶段。

至于需求拆分的粒度，根据不同业务形态也不尽相同，我们现行的指导原则是，在一个迭代周期内可以上线交付，平均统计下来3-5天居多。另外，除了需求粒度之外，还有一个比较重要的指

标，就是配置化需求比例，也就是不需要额外开发代码，仅需要调整配置就可以完成交付的功能。

1

2



阿硕

2019-10-23

石老师，您好，请问价值流图中的指标如何度量出来的呢？

作者回复: 你好，价值流图反应的是团队客观的软件交付流程，度量价值流实际上就是度量软件交付过程。一般来说，都是以需求作为抓手，统计在各个交付阶段的时长，在针对这个阶段的流转时长度量可改进的方向。所以，关键是把需求纳入统一系统管理，需求状态切换尽量通过自动化手段完成，让需求管理系统和各个环节系统进行打通，那么从需求管理平台就能一目了然的看清楚团队现状。

1

1



Brian

2019-10-28

一边学习，一边思考我们公司现在的项目，如果用VSM的各项指标去衡量的话浪费极为严重。开展VSM活动对于我们而言，难点在于首先要让大家认同，不认同就意味着不会积极配合，也就没有办法通过会议或者走访的形式去挖掘最底层的问题。

展开

1

1



工画师

2019-10-28

我是比较认同VSM中的价值流的。DevOps其实是与敏捷伴生的，但不是说没有使用敏捷的公司就无法开展DevOps，只不过落地实施会走样，当效率工具罢了。个人认为要想发展DevOps甚至VSM，需要先搞清敏捷与价值流动，从文化和思维模式上进行转变，否则单说DevOps有些跟风了，无法发挥出内在的价值，也就不会有质的飞跃。

展开

1

1



sugar

2019-10-25

目前问题还是比较多，看似上了很多平台、系统、工具提升效率，其实有时反而降低了效率，我觉得主要是文化及意识的问题，没有优先级，没个需求提出都说着急上，需要一个牵头人做整体的推动和沟通交流

作者回复: 嗯嗯, 缺少顶层设计和实施框架, 的确容易野蛮生长, 这也是现在很多公司部门都在搞 DevOps的通病, 开发和运维之家的壁垒还没解决, DevOps各家工具之间的壁垒先整出来了。



manatee

2019-10-24

想请问老师, 关于价值流中的三个指标, 前置时间我理解就是从需求到上线的时间, 这个还好统计, 无非就是确定一个个阶段再分段统计时间, 那增值时间和完成度又如何在系统上实现呢。我的意思是这两个指标如何具体的去度量。就拿增值时间来说, 这个如何定义, 有没有什么标准, 又如何在系统上自动化的统计呢, 感觉完全没有思路, 请老师指点展开 ∨

作者回复: 你好, 其实我在文章中也有提到过, 稍微展开一下。

关于增值时间和不增值时间, 可以划分为两个层面。首先是从全局的维度统计等待时间, 如果你已经用了电子看板工具, 那么一个任务从开始到结束的周期内, 有多长时间处于等待状态, 比如待开发, 待测试, 待发布, 那这些都是典型的不增值时间。

其次, 从微观角度, 深入到各个环节内容, 比如开发环节, 统计那些除了真正写代码之外的活动时长, 比如构建打包, 比如提测任务, 比如没人评审的提交, 比如审批活动, 这些理论上都是不增值的时间, 可以尽量优化掉。

至于说开发本身的效率高, 这个不方便度量, 但我相信, 只要外部这些不增值的时间被优化掉, 其实就能解决大头的问题。

关于完成度和准确率, 完成度最简单的就是统计迭代周期内的任务, 有多少按时完成了, 有多少延期了, 比如按时提测率, 虽然刚开始的时候对于任务的工作估计会有很大偏差, 但随着团队磨合以及对于业务的理解, 这个估计会被不断的矫正, 更加贴近于真实情况。

准确率方面, 可以重点考察被下游打回的流程, 比如研发提测后, 测试发现自测不达标打回研发, 可以统计提测打回率, 再比如上线发布后, 发现问题出现回滚, 那就是上线失败率, 也可以进行统计。

关键点是, 指标贵精不贵多, 预期一次性度量10个指标, 不如把一个关注点的指标做透, 直到改进流程影响行为。



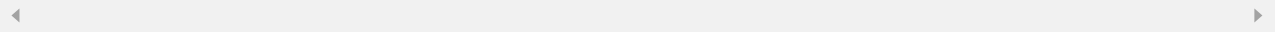
leslie

2019-10-22

VSM中“价值”: 应当是最难的, 现在的公司现有系统经常就在这个模块里的“识别问题”和“促进沟通”以及“价值展现”中一堆的问题; 故而做了一系列关于现状的沟通, 然后明白自己要想实现一些的梦想应当去哪里 😊

展开 ∨

作者回复: 期待你的梦想成真😊



johnny

2019-10-22

像我们这种没有devops体系的公司来说，可能关注更多的是如何落地。

老师以后会多讲一些devops所涉及环节具体实践吗？

比方说，devops会涉及到一个持续集成/持续交付的环节（会用到jenkins、容器等技术），怎么实现？

我的意思是老师会不会以Adminset、jenkins或者其它运维平台或工具为主线，讲解dev...
展开▼

作者回复: 你好，感谢你的留言，我在整理内容的时候，也在纠结工具本身的介绍要占据多大的篇幅，其实如果展开来说，每个工具都可以成为一个独立的专栏，所以权衡下来会采用实践为主，工具和案例为辅的方式，也就是先说明白为什么要做这个实践，实践的正确姿势是怎样的，再结合实践和工具列举一些小的案例帮助理解。至于工具本身，一方面网上有很多资料，另外一方面DevOps本身也不强依赖于某个具体的工具，所以我推荐还是以问题驱动学习的方式。你好，感谢你的留言，我在整理内容的时候，也在纠结工具本身的介绍要占据多大的篇幅，其实如果展开来说，每个工具都可以成为一个独立的专栏，所以权衡下来会采用实践为主，工具和案例为辅的方式，也就是先说明白为什么要做这个实践，实践的正确姿势是怎样的，再结合实践和工具列举一些小的案例帮助理解。至于工具本身，一方面网上有很多资料，另外一方面DevOps本身也不强依赖于某个具体的工具，所以我推荐还是以问题驱动学习的方式。

顺便说下，你既然提到了Adminset，也给你一些小的福利，关于这个工具的问题，我会收集起来邀请作者来跟大家一起互动哈。



delly

2019-10-22

以我所在的大型it服务公司的角度来看，我个人觉得价值流是挺难梳理的。因为照我的认知，基于itil的服务交付体系，大家的执行动力是以sla为驱动，工作模式基本都是等单子，看优先级，然后做单子。手里很多单的时候会按照优先级逐级去做，手里单子很少呢，有可能勤快点来个单子就做，也可能来了单子就等着，到快到deadline了才处理。每个人可能都处于在一个或多个时间区间内来回挪腾的状态，所以我的意思是这种情况下去弄清...
展开▼

作者回复: 你好，感谢你的回复，我觉得说的特别好。

没错，问题的关键就在于组织的运作不是面向效率建设的，换句话说只要符合现有的流程前提之

下，效率的高低其实并没那么重要。但是另外一方面，我们又看到ITILv4的指导原则，也体现出一种DevOps的特征，比如关注价值，关注现状，自动化，流程与反馈等等，所以你说的这种现状不会长期存在，迟早也会进行调整。除非IT服务变成了铁饭碗，也没有行业竞争的压力，那么当竞争的重心从服务全面性，到服务质量，再到运作效率的转变时，消除浪费也就在所难免了。我还是始终强调一点，如果公司想在不改变流程的前提下推行DevOps，那么大概率是会失败的。所以思想和流程的共同转变，再辅以工具技术的支撑，才有可能改变现状。

1



牧野静风

2019-10-22

看老师的专栏，对于企业技术管理，转型很有参考意义，除了软件交付上的流程更加通畅，高效，人员的水平提高也很有帮助，不仅仅在技术方面，很多软实力也是需要的。

作者回复: 你好，我今天跟团队还在学习谷歌的代码评审文档，不知道你看了没？其中感受特别深的一点就是国外这些所谓的优秀公司，在总结规范和实践的时候，对人的关注非常高。比如举个例子，在谷歌的代码评审实践中，特别提到了要如何给出评审建议，怎么样的话术比较容易让人接受，如果两方争执不下，要怎么处理的好，甚至要关注评审双方的情绪，等等。但是我们在做类似的事情的时候，很多时候都是指标先行，通过定义指标，来考核团队的落实情况，至于指标是否合理，人员的情绪是否有照顾到，说实话，并不太关心。这也是值得反思的一点吧。

1



蜗牛

2019-10-22

可能对于较大的，已有交付流程较为规整的企业来讲，价值流的整理会相对简单些.....但是对于某些初创企业来说，本身的交付流程就相对混乱，这个时候就需要梳理人本身有一个已有的价值流模型来做作为参考，结合公司的流程现状来梳理，规划调整。不知道我的理解石老师会怎么看？对于初创企业的价值流整理有什么好的建议？

展开

作者回复: 你好，我非常认同你的观点，价值流梳理听起来很炫，但归根结底梳理的还是软件交付流程，这个流程并不是说梳理价值流才出现的，而是只要公司在开发交付软件，这个流程就一直存在，只不过没有人全局的来看，或者把他提到一个比较高的维度而已。对于初创企业来说，压力大，人手紧张，再加上业务范围不确定，本身就是需要灵活变化的能力，这样看来，容错率更低的初创公司，对于DevOps的协作，自动化，全局优化更为适用。作为价值流梳理的副产品，公司内部持续交付流水线就是很多初创公司都在加强建设的方向，无论是基于云服务商提供的平台，还是内部自己搭建的平台，把这个事情说清楚了，价值流也就出来啦。

1



Robert小七

2019-10-22

目前金融行业中大部分的研发人员来自isv，而企业自有人员的研发能力不足，可能对价值交付流程不熟，对于不同的项目可能来自不同的isv，价值交付流程图也有很大区别，那么老师是如何对这类企业进行走访调查的？

作者回复: 你好，根据我这一两年的经历，我可以很确定的说，金融企业对于软件自我意识和把控诉求与日俱增，一方面自身软件交付团队和平台的建立有目共睹，另外一方面对于外包的质量，流程，交付节奏，甚至DevOps能力都再越来越严。

所以，即便每个项目合作方都不相同，软件开发流程也不相同，但是殊途同归，毕竟你总要有一种有效的衡量手段。而价值流中的几个核心指标，对于这一点来说很是很有帮助的。

