

加速度

# 2021 DevOps 状态报告



报告中文版由中国DevOps社区组织翻译

译者：余晓菁、朱婷、王伟、陈兆康、袁本刚

审校：姚冬、王英伟、胡帅、高俊宁、崔龙波

本报告仅供学习使用，版权为原报告所有

赞助商



Deloitte.



第一章

摘要

关键发现 06

第二章

我们如何对比？

软件交付及运维效能 09

第三章

我们如何改进？

云	15	安全	24
SRE 和 DevOps	19	DevOps技术能力	26
文档	21	新型冠状病毒（COVID-19）	29
		文化	31

## 第四章

# 哪些人参与了调研？

个人和企业统计概况 34

## 第五章

# 总结

## 第六章

# 致谢

## 第七章

# 作者

## 第八章

# 研究方法

## 第九章

# 扩展阅读

---

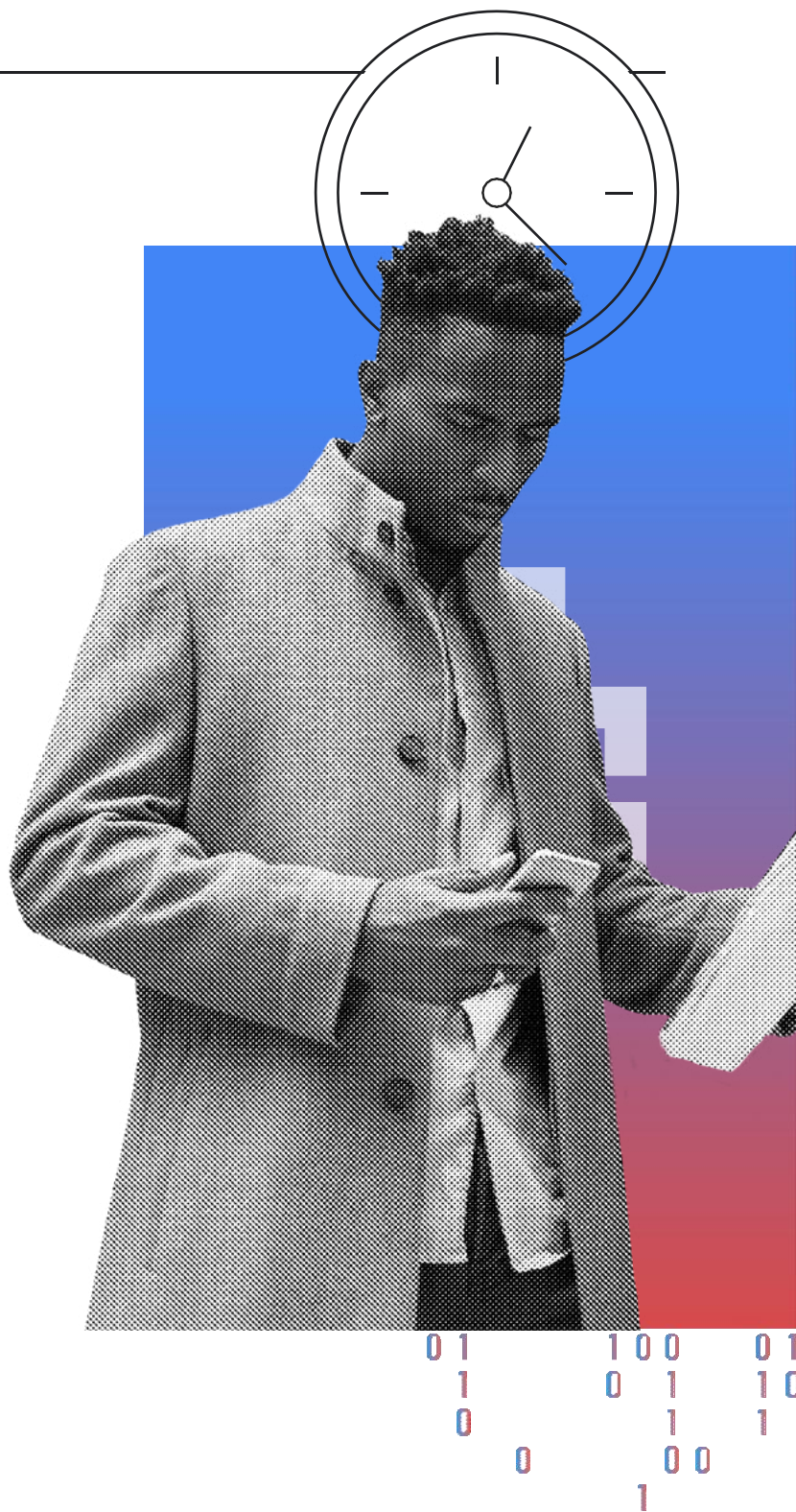


## 第一章

# 摘要

今年的《DevOps加速状态报告》，由谷歌云的 DevOps 研究和评估（DORA）团队发布，它代表了对全球32,000多名专业人士7年来的研究和数据。

我们的研究调查了驱动软件交付、运维和组织效能的能力和实践。藉由缜密的统计方法，我们力图了解通向卓越技术交付和强大业务成果的实践。为此，我们就开发和交付技术的最有效途径提供了数据驱动的解读。



我们的研究仍旧表明，软件交付和运维表现的卓越推动了技术转型中的组织效能。为了让团队能以行业为基准，我们使用聚类分析来形成有意义的效能分类（如低、中、高或精英效能者）。在您的团队了解他们当前相对于行业的表现后，可以利用我们预测分析的调研结果，以提高关键成果的实践和能力作为目标，并最终改善相对处境。今年，我们强调实现可靠性目标的重要性，在软件供应链中集成安全性，创建高质量的内部文档，以及充分发挥云的潜能。我们还探讨了积极的团队文化是否能缓和由于新冠疫情带来的远程工作的影响。

为了作出有意义的改进，团队必须采用持续改进的理念。利用标杆来衡量你们当前的状态，基于调研过的能力来确定制约因素，并尝试改进以缓解这些制约因素。实验将涉及各种成功和失败，但无论在哪种情况下，团队都可以根据学到的经验采取有意义的行动。

# 关键发现

## 01 表现最好的群体持续成长，并继续抬高标准。

在我们的研究中，精英效能组织现在占了26%。并且他们的生产变更前置时间减少了。行业继续加速发展，团队也从中看到了这样做的重要好处。

## 02 SRE 和 DevOps 是互补的。

一些团队采用了站点可靠性工程（Site Reliability Engineering，简称SRE）所概括的现代化运维实践，汇报了更好的运维效能。将交付和运维的卓越放在同一优先级考虑的团队报告了最好的组织效能。

## 03 更多的团队在使用云，并看到这样做的显著好处。

团队继续将工作负载转移到云上，那些利用云的所有五种能力的团队看到了在软件交付和技术运维（SDO）效能以及组织效能上的提高。多云应用也在增加，这样团队就可以利用每个提供商的独特能力。

## 04 安全的软件供应链必不可少，同时也是效能的驱动因素。

鉴于近年来恶意攻击的显著增加，组织必须从被动的方式转向主动和诊断性的措施。在整个软件供应链中集成安全实践的团队能够快速、可靠、安全地交付软件。

## 05 良好的文档是成功实行DevOps能力的基础。

我们首次衡量了内部文档的质量，以及有助于提高这种质量的实践。拥有高质量文档的团队能够更好地实施技术实践，并在整体上表现得更好。

## 06 积极的团队文化可以缓解在挑战性环境下的倦怠感。

团队文化对团队交付软件以及满足或超越其组织目标的能力有很大的影响。在新冠疫情期间，具有生成性（generative<sup>1,2</sup>）文化的开放性团队的工作倦怠程度较低。

---

1 从Westrum的类型学组织文化来看，生成性团队文化是指高度合作、打破竖井的团队，他们从失败引导探究，并且共担决策风险。

2 Westrum, R. (2004). 《A typology of organizational cultures》.BMJ Quality & Safety, 13[增刊 2], ii22-ii27页.



## 第二章

# 我们如何对比？

你是否好奇你的团队效能行业内的位置？本节介绍业内DevOps效能的最新评估基准。

我们研究了团队如何开发、交付和运维软件系统，然后将受访者分为四个效能群组：精英、高、中、低效能者。通过与每个群组的表现相比较，你可以看到自己团队在本报告中所描述的调查结果中的位置。









# 软件交付和运维效能

为了满足不断变化的行业需求，企业必须快速、可靠地交付和运行软件。你的团队越快地对软件进行变更，你就能越快地向客户提供价值，运行功能验证，并收到有价值的反馈。通过七年的数据收集和研究，我们已经开发并验证了四个衡量软件交付效能的指标。自2018年以来，我们已经将第五个指标纳入进来，来衡量组织的运维能力。

在所有五个方面都表现出色的团队会表现出卓越的组织效能。我们把这五项指标称为**软件交付和运维（SDO）效能**。请注意，这些衡量标准是基于系统级的统计结果，这有助于避免软件衡量标准的常见误区，如将彼此关联的功能孤立起来，以整体结果为代价进行局部优化。

软件交付效能指标	精英	高	中	低
 <b>部署频率</b> 对于你所负责的主要应用或服务，你的组织多长时间将代码部署到生产环境中或发布给最终用户？	按需发布（每天多次部署）	介于每周一次到每月一次之间	介于每月一次到每6个月一次之间	小于6个月一次
 <b>变更的前置时间</b> 对于你所负责的主要应用或服务，你的变更前置时间是多少（即，从提交代码到代码成功运行于生产环境中需要多长时间）？	不到1小时	1天到1周之间	1个月到6个月之间	6个月以上
 <b>服务恢复时长</b> 对于你所负责的主要应用或服务，当发生影响用户的服务事件或缺陷（如计划外中断或服务受损）时通常需要多长时间来恢复服务？	不到1小时	不到1天	1天到1周之间	6个月以上
 <b>变更失败率</b> 对于你所负责的主要应用或服务，对生产环境的变更或向用户发布的变更有多大比例导致服务下降（例如，导致服务受损或服务中断）并随后需要补救（例如，需要热修复、回滚、向前修复、补丁）	0%-15%	16%-30%	16%-30%	16%-30%

## 交付能力的四个指标

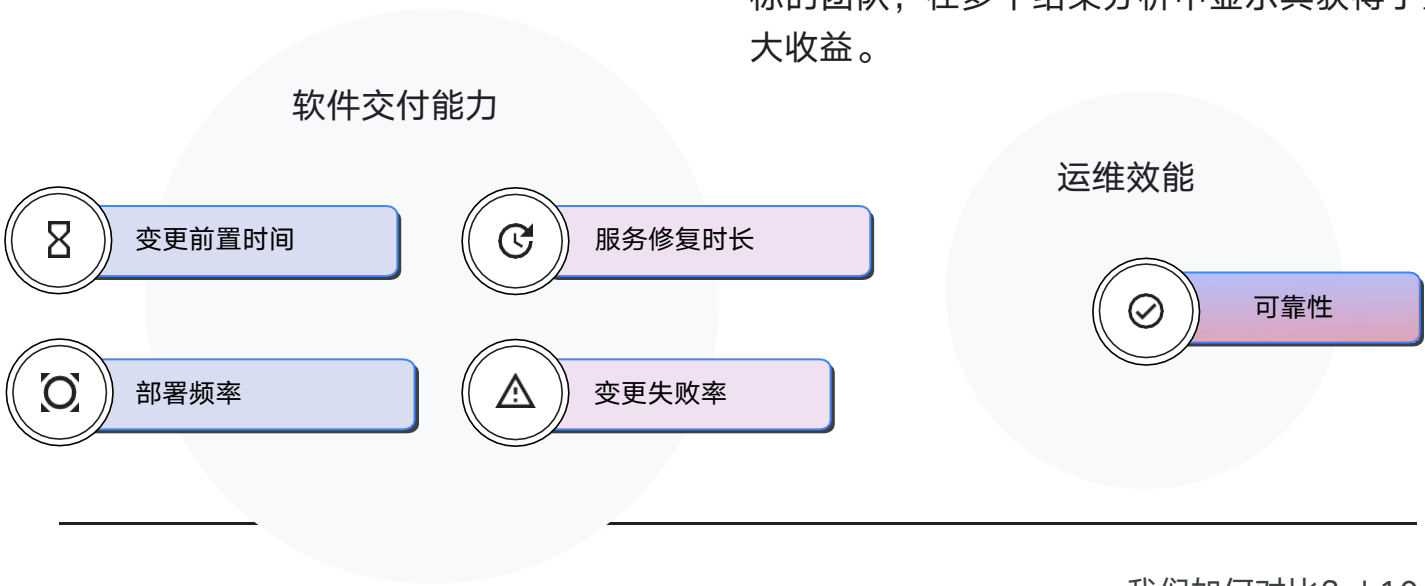
软件交付能力的四个指标可以从吞吐量和稳定性两个方面来考虑。我们用**代码修改的前置时间**（即从代码提交到在生产环境中发布的时间）和**部署频率**来衡量吞吐量。我们用事件发生后**服务恢复时长**和**变更失败率**来衡量稳定性。

对四个软件交付指标的聚类分析结果再次显示出了四种不同群组的表现：精英、高、中、低。其在吞吐量和稳定性指标上的统计结果也呈现出显著差异。与往年一样，我们的精英效能企业在所有四个指标上都表现得非常好，而低效能企业在所有领域都表现得非常差。

## 第五项指标：从可用性到可靠性

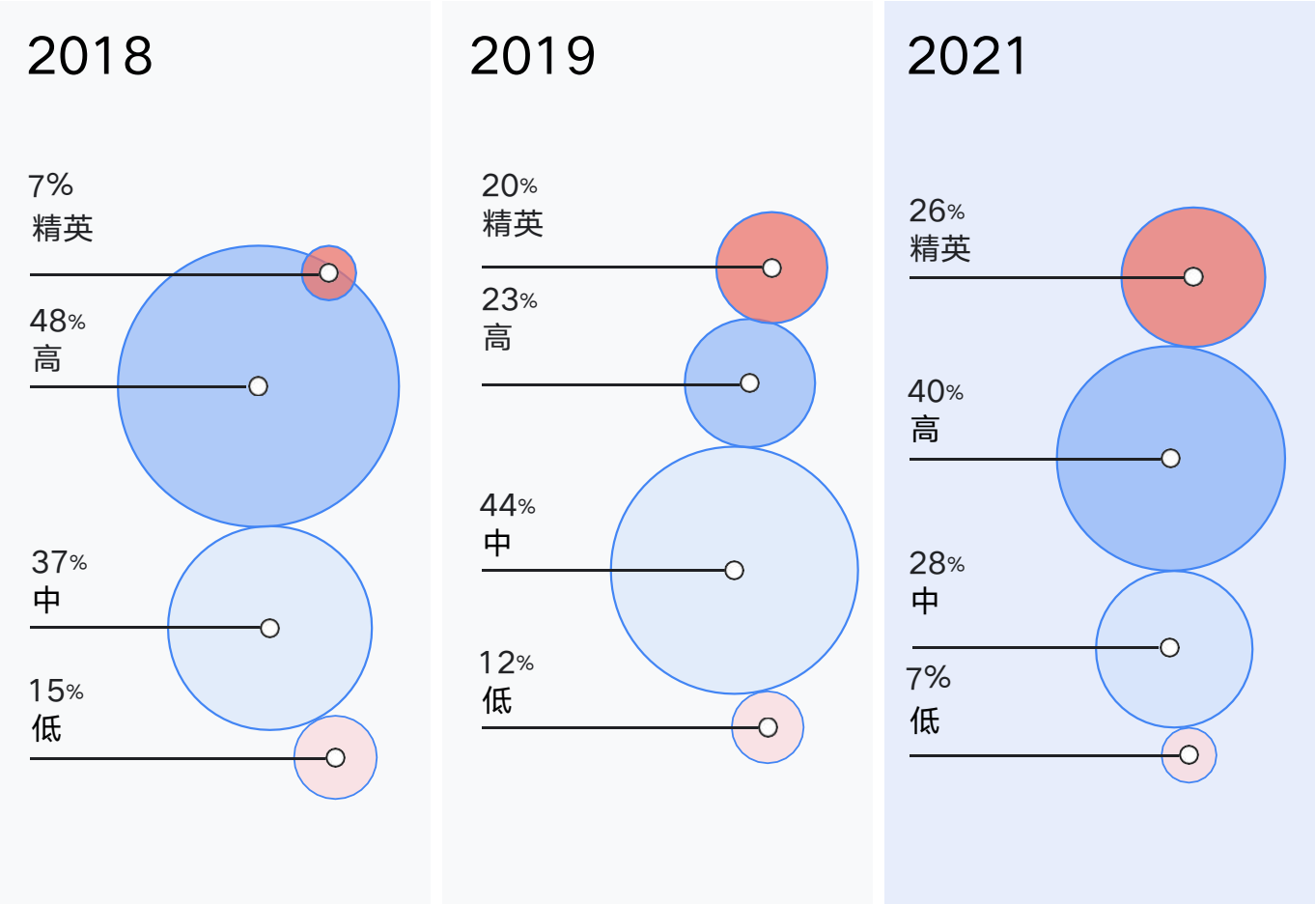
第五个指标代表了**运维效能**，是对现代运维实践水平的一种衡量。度量效能的主要指标是**可靠性**，即一个团队对其操作的软件的承诺和预期的兑现程度。在以往我们度量的是可用性而不是可靠性，但由于可用性是可靠性工程的一个具体的专注点，我们将评估范围扩大到可靠性，从而使可用性、延迟、性能和可扩展性得到更广泛的体现。具体来说，我们要求受访者评价他们达到或超过可靠性目标的能力。我们发现，具有不同级别交付能力的团队，在他们评价运维效能时，如果优先考虑了可靠性，评测结果的更好。

与以前的报告一样，我们将精英效能企业与低效能企业进行比较，以说明各项具体能力的作用。今年我们试图说明企业在运维效能指标方面的影响。在所有的交付能力类别（从低到精英）中，我们看到优先达成或超过其可靠性目标的团队，在多个结果分析中显示其获得了重大收益。



行业持续交付的加速度

每年，我们都会看到行业不断发展，并以更快的交付速度和更好的稳定性交付软件。第一次报告中，我们的高效能和精英效能企业占到了受访者的三分之二。此外，今年的精英效能企业再次提高了标准，与以前的评估相比，他们缩减了变更前置时间（例如，从2019年的不到一天提高到2021年的不到一个小时）。此外，与前几年中等和高效能企业能够做到的水平相比，第一次报告中只有精英效能企业将其变更失败率降到最低。



## 吞吐量

### 部署频率

与往年一致，精英效能组报告显示，其经常按需部署，每天进行多次部署。相比之下，低效能组报告每6个月部署不到一次（每年不到两次），与2019年相比，效能再次下降。从常规化的年度部署次数来看，精英效能组每年1460次部署（按每天4次部署×365天计算）；低效能组每年1.5次部署（平均两次部署和一次部署），这一分析结果表明，精英效能组部署代码的频率比低效能组高973倍。

### 变更前置时间

变更前置时间为从代码提交到在生产环境中成功部署该代码的时间。精英效能组报告的变更前置时间少于一小时，与2019年相比，这是一种效能的提高，当时我们精英效能组报告的变更前置时间不到一天。而与精英效能组相比，低效能组需要的变更前置时间超过6个月。精英效能组的前置时间为一小时（“不到一小时”的高度保守估计），低效能组的前置时间为6570小时（通过每年8760小时和6个月4380小时的平均计算），精英效能组的变更前置时间比低效能者快6570倍。

## 稳定性

### 服务恢复时长

精英效能组报告的服务恢复时长少于一小时，而低效能组报告的时间超过六个月。我们按保守的时间范围计算：精英效能组是一小时，低效能组是一年（8,760小时）和六个月（4,380小时）的平均值。根据这些数字，精英效能组服务恢复时长比低效能者快6,570倍。与2019年相比，精英效能组服务恢复时长保持不变，而低效能组则有所增长。

### 变更失败率

精英效能组报告的变更失败率在0%-15%之间，而低效能组报告的变更失败率为16%-30%。这两个范围之间的平均值显示，精英效能组的变更失败率为7.5%，低效能组的变更失败率为23%。精英效能组的变更失败率是低效能组的三倍。今年与2019年相比，精英效能组的变更失败率保持不变，低效能组的变更失败率有所提高，介于两者之间的其他组织数据则有所降低。



## 精英效能

将精英效能组与低效能组进行比较，我们发现，精英效能企业有以下能力特质：

973x

更加频繁  
部署代码

6570x

更短的前置时间  
从代码提交到部署

是的，你没看错。  
这不是随意编造  
的数据！

3x

更低的变更失败率  
(低于1/3的变更失败)

6570x

更快的故障修复时间

## 第三章

# 我们如何改进？

我们如何改进SDO和组织效能？  
我们的研究以证据指导，帮助你关注那些提升效能的能力。

今年的报告考察了云、SRE实践、安全、技术实践和文化的影响。在本节中，我们将介绍这些能力中的每一项，并注意它们对各种结果的影响。对于那些熟悉DORA的DevOps研究模型的人，我们已经创建了一个在线资源，用于托管今年的模型和所有以前的模型。<sup>3</sup>



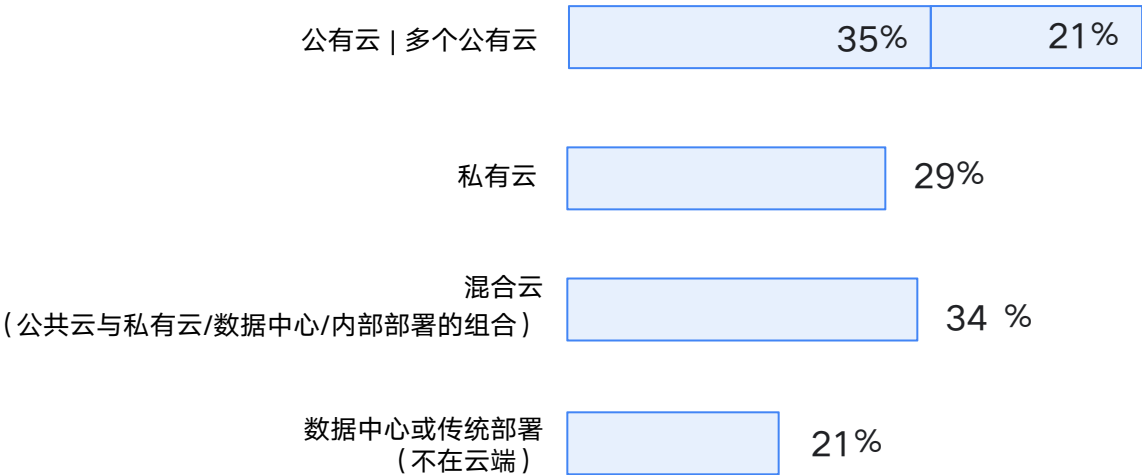
---

<sup>3</sup> <https://devops-research.com/models.html>

# 云

与《加速度：2019 DevOps状态报告》一致，越来越多的组织正在采纳多云和混合云解决方案。在我们的调查中，受访者被问到他们的主要服务或应用程序在哪里时，公有云的使用率正在上升。56%的受访者表示使用公有云（包括多个公有云），比2019年增加5%。今年我们还专门调研了多云的使用情况，21%的受访者表示部署到多个公共云。21%的受访者表示不使用云，而是使用数据中心或传统部署方案。最后，34%的受访者报告使用混合云和29%的受访者报告使用私有云。

## 采用



## 利用混合云和多云推进业务成果

今年，我们看到混合云和多云的使用在增长，对业务成果产出产生了重大影响。使用混合云或多云的受访者超过组织效能目标的可能性是未使用混合云或多云的受访者的1.6倍。我们还看到了SDO的强大影响，混合云和多云的用户在部署频率、变更前置时间、恢复时间、变更错误率和可靠性方面有优异表现的可能性高出1.4倍。

## 为什么是多云？

与我们2018年的评估类似，我们要求受访者报告他们选择多个公有云提供商的理由。今年，我们调研受访者使用多家供应商的主要原因，而不是选择所有适用的理由。超过四分之一（26%）的受访者这样做是处于每个云提供商的独特优势。这表明，当受访者选择其他供应商时，他们会在当前供应商和备选供应商之间寻找差异。迁移到多云的第二个最常见原因是可用性（22%）。毫不奇怪，采用多家云提供商的受访者达到或超过其可靠性目标的可能性是那些不采纳多云方案的1.5倍。

## 使用多个供应商的主要原因

利用每个提供商的独特优势	26%
可利用性	22%
灾难恢复	17%
法律遵从性	13%
其它	08%
谈判策略或采购要求	08%
对一个提供者缺乏信任	06%



## 基准变化

### 如何实施云基础设施很重要

从历史上看，我们发现并非所有受访者都以相同的方式采纳云。这导致了云应用在有效推动业务成果方面存在差异。我们通过关注云计算的基本特征（由国家标准与技术研究所（NIST）定义）来解决这一局限性，并以此为指导。使用NIST对云计算的定义，我们调查了核心实践对SDO效能的影响，而不仅仅是调查采用云对SDO的影响。

第三次，我们发现真正重要的是团队如何实现他们的云服务，而不仅仅是他们在使用云技术。精英效能者符合NIST云计算所有基本特征的可能性是一般员工的3.5倍。表示正在使用云基础设施的受访者中，只有32%同意或强烈同意他们符合NIST定义的云计算的所有五个基本特征，比2019年增加了3%。总体而言，NIST云计算特性的采纳增加了14 – 19%，其中快速弹性增长幅度最大。

73%

的受访者使用按需自助服务，比2019年增长16%

74%

的受访者使用广泛的网络接入，比2019年增长14%

### 按需自助服务

消费者可以根据需要自动预制计算资源，而无需提供商进行任何人工交互。

### 广泛网络接入

功能广泛可用，可通过多个客户端（如移动电话、平板电脑、笔记本电脑和 workstation）访问。

## 资源池

提供商将资源汇集在多租户模型中，物理和虚拟资源按需动态分配和重新分配。客户通常无法直接控制所提供资源的确切位置，但可以在更高的抽象级别，如国家、州或数据中心等进行位置选择。

73%

的受访者在使用的资源池，  
比2019年增长15%

## 快速弹性

可以弹性地预制和释放功能，以根据需求快速向外或向内扩展。可用于资源调配的消费者功能似乎是无限的，并且可以在任何时间以任何数量进行分配。

77%

的受访者在使用的快速弹性，  
比2019年增长18%

## 可度量服务

云系统通过在与服务类型（如存储、处理、带宽和活动用户帐户）相适应的抽象级别上利用度量功能，自动控制和优化资源使用。可以监视、控制和报告资源的使用情况以提高透明度。

78%

的受访者使用可度量服务，  
比2019年增长16%

# SRE 和 DevOps

当DevOps社区在公开会议和对话中崭露头角时，正在谷歌内部形成一场志同道合的运动：站点可靠性工程（SRE）。SRE和类似的方法，如Facebook生产工程规程，包含许多激励DevOps的相同目标和技术。2016年，当第一本书<sup>4</sup>《站点可靠性工程》出版时，SRE正式加入公众讨论。自那时以来，该运动不断发展，如今，全球SRE从业人员在社区就技术操作实践进行合作。

也许不可避免地会出现了让人费解的问题。SRE和DevOps之间有什么区别？我需要一个还是另一个？哪一个更好？事实上，这里没有冲突；SRE和DevOps是高度互补的，我们的研究证明了它们的一致性。SRE是一门学问优先考虑跨职能沟通和心理安全的学科，这些价值观是本课程的核心精英DevOps团队典型的以效能为导向的生成性文化。SRE从其核心原则出发，提供了实用技术，包括服务级别指标/服务级别目标（SLI/SLO）度量框架。正如精益产品框架规定了如何实现我们的研究所支持的快速客户反馈周期一样，SRE框架提供了实践和工具的定义，这些实践和工具可以提高团队持续履行对用户承诺的能力。

2021年，我们扩大了对运维的调查，从服务可用性分析扩展到更一般的可靠性类别。今年的调查引入了几个受SRE实践启发的项目，以评估团队：

- 根据面向用户的行为定义可靠性
- 利用SLI/SLO度量框架根据错误预算对工作进行优先级排序
- 使用自动化减少手动工作中断性警报
- 为事件响应定义协议和准备演练
- 在整个软件交付生命周期中纳入可靠性原则（“可靠性左移”）

在分析结果时，我们有证据表明，在这些现代运维实践中表现出色的团队产生更好的SDO效能的可能性是其他团队的1.4倍，报告更好的业务成果的可能性是其他团队的1.8倍。

在我们的研究中，大多数团队都采用了SRE实践：52%的受访者报告在一定程度上使用了这些实践，尽管不同团队采用SRE实践的深度差异很大。数据表明，使用这些方法可以预测更高的可靠性和更高的整体SDO效能：SRE推动了DevOps的成功。

---

<sup>4</sup> Betsy Beyer等人编著的《Site Reliability Engineering》（O'Reilly Media, 2016）。

此外，我们还发现，一个共同承担运维责任的模型（反映在开发人员和运营商联合授权为可靠性做出贡献的程度上）也可以预测更好的可靠性结果。

除了改进客观的效能衡量标准外，SRE还提高了技术从业者的工作经验。通常，承担繁重操作任务的个人容易精疲力竭，但SRE有积极的影响。我们发现，一个团队越多地采用SRE实践，其成员越不可能经历倦怠。SRE还可能有助于优化资源：通过应用SRE实践达到可靠性目标的团队报告说，他们要比不实践SRE的团队花更多的时间编写代码。

我们的研究表明，无论SDO表现如何，从低效能到精英效能，团队都可能从SRE实践的增加中获益。团队的效能越好，他们采用现代运维模式的可能性就越大：精英效能者报告使用SRE实践的可能性是低效能者的2.1倍。但即使是最高级别的团队也有增长空间：只有10%的精英效能受访者表示，他们的团队已经全面实施了我们调查的每一项SRE实践。随着跨行业SDO效能的不断提高，每个团队的运维方法都是DevOps持续改进的关键驱动力。

52%

的受访者报告使用了  
SRE实践

---

精英效能组织报告使用SRE实践的可能性为低效能者的2.1倍。但即使是最高级别的团队也有增长空间：只有10%的精英效能受访者表示，他们的团队已经完全实施了我们的每一项SRE实践。



# 文档

今年，我们考察了**内部文档**的质量，即团队工作的服务和应用程序的文档，如手册、README文件，甚至代码注释。我们通过以下程度来衡量文件质量：

- 帮助读者实现他们的目标
- 是准确、最新和全面的
- 可查找、组织良好且清晰。<sup>5</sup>

记录和访问有关内部系统的信息是团队技术工作的关键部分。我们发现，大约25%的受访者拥有高质量的文档，这种文档工作的影响是显而易见的：文档质量较高的团队看到更好的软件交付和运维（SDO）效能的可能性是其他团队的2.4倍。文档质量好的团队比文档质量差的团队更快、更可靠地交付软件。文档不必是完美的。我们的研究表明，文档质量的任何改进都会对效能产生积极和直接的影响。

今天的技术环境中越来越复杂的系统，以及这些系统不同方面的专家和专业角色。从安全到测试，文档是在这些专业子团队之间以及与更广泛的团队共享专业知识和指导的关键方式。

我们发现，文档质量可以预测团队在实施技术实践方面的成功。这些实践反过来又可以预测系统技术能力的改进，如可观测性、连续测试和部署自动化。我们发现文档具有质量的团队包括：

- 实施安全措施的可能性提高3.8倍
- 达到或超过其可靠性目标的可能性高出2.4倍
- 实施站点可靠性工程（SRE）实践的可能性提高3.5倍
- 充分利用云的可能性提高2.5倍

---

5 在现有研究的技术文档上获悉的质量指标，如：

—Aghajani, E.等(2019).《Software Documentation Issues Unveiled》.2019年IEEE/ACM第41届国际软件工程大会文集，1199-1210页。 <https://doi.org/10.1109/ICSE.2019.00122>

—Plösch, R., Dautovic, A., & Saft, M.(2014).《The Value of Software Documentation Quality》.国际质量软件大会文集，333-342页。 <https://doi.org/10.1109/QSIC.2014.22>

— Zhi, J.等(2015).《Cost benefits and quality of software development documentation:A systematic mapping》. Systems and Software期刊，99期[C]，175-198。 <https://doi.org/10.1016/j.jss.2014.09.042>

## 如何改进文档质量

技术工作包括查找和使用信息，但文档质量依赖于编写和维护内容的人员。2019年，我们的研究发现，访问内部和外部信息源有助于提高生产率。今年的研究进一步考察了所访问文档的质量，以及对文档质量有影响的实践。

我们的研究表明，以下做法对文档质量有显著的积极影响：

**记录产品和服务的关键用例。**关于系统的文档很重要，用例允许读者将信息和系统投入工作。

**为更新和编辑现有文档创建明确的指导原则。**大部分文档工作都是维护现有内容。当团队成员知道如何进行更新或删除不准确或过时的信息时，即使系统随时间变化，团队也可以保持文档质量。

**定义所有者。**拥有高质量文档的团队更有可能明确定义文档的所有权。所有权使得明确编写新内容和更新或验证对现有内容的更改的责任成为了可能。具有高质量文档的团队更有可能声明文档是针对他们所处理的应用程序的所有主要功能编写的，明确的所有权有助于创建这种广泛的覆盖范围。

## 具有高质量文档的团队

3.8x

更有可能实施  
安全实践

2.4x

更有可能达到或超过  
他们的可靠性目标

3.5x

更有可能实施  
站点可靠性工程  
(SRE) 实践

2.5x

更有可能充分利  
用云

**将文档作为软件开发过程的一部分。**创建文档并在系统更改时进行更新的团队拥有更高质量的文档。与测试一样，文档创建和维护也是高效能软件开发过程的一个组成部分。

**将文档工作作为效能考核和晋升的考虑因素。**识别与总体文档质量相关。编写和维护文档是软件工程工作的核心部分，这样处理文档可以提高其质量。

我们发现支持有质量文档的其他资源包括：

- 关于如何编写和维护文档的培训
- 代码样本或不完整文档的自动测试
- 指南，如文档风格指南和面向全球读者的写作指南

文档是成功实现DevOps能力的基础。更高质量的文档增强了对单个DevOps能力（如安全性、可靠性和充分利用云）的投资成果。实施高质量文档的实践通过更强大的技术能力和更高的SDO效能获得回报。

# 安全

## [左移]并在整个过程中集成

随着技术团队的不断加速和发展，安全威胁的数量和复杂性也在不断提高。根据Tenable的2020年威胁展望回顾报告<sup>6</sup>，2020年，超过220亿条机密个人信息或业务数据记录被泄露。安全不能是事后考虑或交付前的最后一步，它必须整合在整个软件开发过程中。

为了安全地交付软件，安全实践必须比恶意行为人使用的技术发展得更快。在2020年的SolarWinds和Codecov软件供应链攻击中，黑客破坏了SolarWinds的构建系统和Codecov的bash上传脚本<sup>7</sup>，暗中将恶意代码嵌入到这些公司数千名客户的基础设施中。考虑到这些攻击的广泛影响，业界必须从预防性方法转向诊断性方法，软件团队应该假设他们的系统已经受损，并在供应链中构建安全性。

与之前的报告一致，我们发现精英效能者擅长实施安全实践。今年，达到或超过可靠性目标的精英效能者在软件开发过程中集成安全性的可能性是其他人的两倍。这表明，那些在保持可靠性标准的同时加快交付速度的团队已经找到了一种集成安全检查和实践的方法，而不会影响他们快速或可靠交付软件的能力。

除了表现出较高的交付和操作效能外，在整个开发过程中集成安全实践的团队达到或超过其组织目标的可能性比其他团队高出1.6倍。支持安全性的开发团队看到了显著的业务价值驱动。

---

达到或超过可靠性目标的精英效能者在软件开发过程中进行安全集成的可能性是其他人的两倍。

---

<sup>6</sup> <https://www.tenable.com/cyber-exposure/2020-threat-landscape-retrospective>

<sup>7</sup> <https://www.cybersecuritydive.com/news/codecov-breach-solarwinds-software-supply-chain/598950/>

---



## 如何使它正确

强调安全性的重要性并建议团队优先考虑安全性是很容易的，但这样做需要对传统的信息安全方法进行一些更改。通过利用以下实践，你可以集成安全性、改进软件交付和运维效能，并提高组织效能：

**安全测试。**根据需要进行安全测试是要求自动化测试过程的一部分，包括应使用预批准代码的区域。

**将安全审查整合到每个阶段。**将信息安全（InfoSec）集成到整个软件交付生命周期的日常工作中。这包括让InfoSec团队在应用程序的设计和架构阶段提供输入，参加软件演示，并在演示期间提供反馈。

**安全检查。**对所有主要功能进行安全审查。

**构建预先批准的代码。**让InfoSec团队构建预先批准、易于使用的库、包、工具链和流程，供开发人员和IT操作员在工作中使用。

**尽早并经常邀请InfoSec。**在规划和应用程序开发的所有后续阶段中融入InfoSec，以便他们能够尽早发现与安全相关的弱点，从而使团队有足够的时间修复这些弱点。

如前所述，高质量的文档推动了各种功能的成功，安全性也不例外。我们发现，拥有高质量文档的团队在整个开发过程中集成安全性的可能性是其他团队的3.8倍。并非组织中的每个人都有密码学方面的专业知识。通过记录在案的安全实践，这些专业知识在组织中发挥最大价值。

### 安全实践

安全测试	58%
将安全审查整合到每个阶段	54%
安全审查	60%
构建预先批准的代码	49%
尽早并经常邀请InfoSec	63%

# DevOps技术能力

我们的研究表明，通过采用持续交付进行DevOps转型的组织更有可能拥有高质量、低风险和成本效益的流程。

具体而言，我们衡量了以下技术实践：

- 松耦合架构
- 基于主干的开发
- 持续测试
- 持续集成
- 使用开源技术
- 监测和观测实践
- 数据库更改的管理
- 自动化部署

我们发现，虽然所有这些实践都改善了持续交付，但松耦合架构和持续测试的影响最大。例如，今年我们发现，达到可靠性目标的精英效能者采用松耦合架构的可能性是低效能同行的三倍。

## 松耦合架构

我们的研究继续表明，你可以通过减少服务和团队之间的细粒度依赖关系来提高IT效能。事实上，这是成功持续交付最有力的预测因素之一。使用松散耦合的体系架构，团队可以彼此独立地进行扩展、失败、测试和部署。团队可以按照自己的节奏前进，以较小的批量工作，积累较少的技术债务，并更快地从失败中恢复。

## 持续测试和持续集成

与前几年的研究结果类似，我们发现持续测试是成功持续交付的有力前提。达到可靠性目标的精英效能者利用持续测试的可能性是其他人的3.7倍。通过在整个交付过程中结合早期和频繁的测试，测试人员在整个交付过程中与开发人员一起工作，团队可以更快地迭代和更改其产品、服务或应用程序。你可以使用此反馈循环为客户提供价值，同时还可以轻松地结合自动化测试和持续集成等实践。

持续集成还改进了持续交付。达到可靠性目标的精英效能者利用持续集成的可能性是其他人的5.8倍。在持续集成中，每次提交都会触发软件的构建

并运行一系列自动测试，在几分钟内提供反馈。通过持续集成，您可以减少成功集成所需的手动和通常复杂的协作。

由Kent Beck和极限编程社区定义的持续集成（它的发源地）还包括基于主干的开发实践，下面将讨论。<sup>8</sup>

## 基于主干的开发

我们的研究表明，高效能组织更可能实施基于主干的开发，即开发人员以小批量工作，并经常将其工作合并到共享主干中。事实上，达到可靠性目标的精英效能者使用基于主干的开发的可能性是其他员工的2.3倍。低效能者更有可能使用长寿命的分支并延迟合并。

团队应至少每天合并一次工作，如果可能的话，每天应合并多次。基于主干的开发与持续集成密切相关，因此你应该同时实现这两种技术实践，因为当你将它们结合使用时，它们会产生更大的影响。

## 部署自动化

在理想的工作环境中，计算机执行重复性任务，而人类则专注于解决问题。实现部署自动化有助于你的团队更接近这一目标。

当你以自动化的方式将软件从测试转移到生产时，你可以通过实现更快、更高效的部署来缩短交付周期。你还可以降低部署错误的可能性，这在手动部署中更为常见。当你的团队使用部署自动化时，他们会立即收到反馈，这可以帮助你以更快的速度改进服务或产品。虽然你不必同时实施连续测试、连续集成和自动化部署，但当你同时使用这三种实践时，你可能会看到更大的改进。

## 数据库变更管理

通过版本控制跟踪更改是编写和维护代码以及管理数据库的关键部分。我们的研究发现，与表现不佳的同行相比，达到可靠性目标的精英效能者执行数据库更改管理的可能性高3.4倍。此外，成功的数据库变更管理的关键是所有相关团队之间的协作、沟通和透明度。虽然你可以从具体的实现方法中进行选择，但我们建议，无论何时你需要对数据库进行更改，团队都应该在你更新数据库之前聚集在一起检查更改。

---

<sup>8</sup> Beck, K. (2000). 《Extreme programming explained: Embrace change》. Addison-Wesley Professional（出版社）

## 监控和可观测性

与前几年一样，我们发现监控和可观测性实践支持持续交付。成功实现其可靠性目标的精英效能者拥有将可观测性纳入整体系统健康的解决方案的可能性是其他人的4.1倍。可观测性实践可以让你的团队更好地了解你的系统，从而减少识别和排除问题所需的时间。我们的研究还表明，具有良好可观测性实践的团队花在编码上的时间更多。这一发现的一个可能解释是，实现可观测性实践有助于将开发人员的时间从寻找问题的原因转移到故障排除，并最终返回到编码。

开发人员可以使用它来提供支持。开源技术更容易获得，成本相对较低，并且可以定制。达到可靠性目标的精英效能者利用开源技术的可能性是其他人的2.4倍。我们建议你在实施DevOps转型时，更多的使用开源软件。

## 开源技术

许多开发人员已经利用了开源技术，他们对这些工具的熟悉是组织的优势。封闭源代码技术的一个主要弱点是，它们限制了你在组织内外传递知识的能力。例如，你不能雇佣已经熟悉组织工具的人员，开发人员也不能将他们积累的知识转移到其他组织。相比之下，大多数开源技术都有一个社区。

更多关于专业DevOps能力的信息，参见DORA能力模型<https://cloud.google.com/devops/capabilities>

# 新型冠状病毒（COVID-19）

今年，我们调查了在新冠病毒流行期间影响团队表现的因素。具体而言，新冠病毒流行是否对软件交付和运维（SDO）效能产生了负面影响？团队是否因此倦怠？最后，哪些因素有助于缓解工作倦怠？

89%

的受访者因疫情而在家工作

首先，我们试图了解疫情对交付和运维表现的影响。许多组织优先考虑现代化，以适应剧烈的市场变化（例如，从亲自购买转向在线购买）。在“我们如何比较？”一章中，我们将讨论软件行业的效能是如何显著提高并继续提高的。高效能团队现在是我们样本中的大多数，精英效能团队继续提高标准，以更短的交付周期、更快的恢复时间和更好的更改失败率更频繁地部署。类似地，GitHub研究人员的一项研究表明，到2020年，开发者活动（即推送、拉取请求、审查拉取请求和每个用户评论问题）有所增加<sup>9</sup>。可以说，尽管疫情，但该行业仍在继续加速发展，没有受到流感的影响，值得注意的是，在这段可怕的时期，我们没有看到SDO表现的下降趋势。

疫情改变了我们的工作方式，对许多人来说，它改变了我们的工作地点。出于这个原因，我们研究了疫情导致的远程工作的影响。我们发现89%的受访者因流感疫情在家工作。只有20%的人报告说在疫情之前曾在家工作。转向远程工作环境对我们如何开发软件、运维业务和共同工作具有重大影响。对许多人来说，在家工作使他们无法通过走廊上的即兴对话或亲自协作进行交流。

---

<sup>9</sup> <https://octoverse.github.com/>



## 什么能减缓工作倦怠？

尽管如此，我们还是发现了一个对团队是否因远程工作而疲惫不堪有很大影响的因素：文化。具有生成性团队文化的团队，由感觉自己属于团队的人组成，在疫情期间经历倦怠的可能性是前者的一半。这一发现强化了优先考虑团队和文化的重要性。表现更好的团队有能力度过更具挑战性的时期，这给团队和个人都带来了压力。

---

具有生成性团队文化的团队，由感觉自己属于团队的人组成，在疫情期间经历倦怠的可能性是其他人的一半。

# 文化

广义地说，文化由组织内的员工互动所产生。它是影响员工对组织和彼此的想法、感受和行为的任何东西。所有组织都有自己独特的文化，我们的研究结果一致表明，文化是组织和IT效能的首要驱动因素之一。具体而言，我们的分析表明，生成性文化——使用Westrum组织文化类型学以及人们在组织中的归属感和包容性来衡量——预测更高的软件交付和运维（SDO）效能。例如，我们发现，达到可靠性目标的精英效能者比低效能者更容易形成创造性的团队文化。同样，生成性文化可以预测更高的组织效能和更低的员工倦怠率。简言之，文化真的很重要。幸运的是，文化是流动的，多方面的，并且总是在不断变化，使它成为你可以改变的东西。

DevOps的成功执行需要你的组织拥有协作和跨职能工作的团队。2018年，我们发现在一个高效能的团队中，高效能和跨功能的团队交付软件的可能性是单一团队的两倍。这强化了协作与合作对任何组织的成功都至关重要的观点。一个关键问题是：哪些因素有助于创造一个鼓励和接纳跨职能协作的环境？

多年来，我们一直在努力使文化的构建成为有形的，并让DevOps社区更好地理解文化对组织和IT效能的影响。我们通过使用Westrum组织文化类型学对文化进行操作性定义开始了这段旅程。他确定了三种类型的组织：权力导向型、规则导向型和效能导向型。我们在自己的研究中使用了这个框架，发现以效能为导向的组织文化可以优化信息流、信任、创新和风险分担，从而预测高SDO效能。

随着我们对文化和DevOps的理解的发展，我们已经努力扩展我们对文化的初始定义，以包括其他心理社会因素，如**心理安全**。高效能的组织更有可能拥有一种文化，鼓励员工在不担心负面后果的情况下，有计划地、适度地承担风险。

---

文化是流动的，多方面的，并且总是在不断变化的，这使得它成为组织可以改变的东西。

## 归属与包容

鉴于文化对效能的影响一直很强，今年我们扩展了模型，探讨员工的归属感和包容性是否有助于文化对效能的有益影响。

心理学研究表明，人们天生就有与他人建立并保持牢固稳定关系的动机。<sup>10</sup>有动力与他人保持联系，并在我们所居住的不同群体中感到被接受。归属感会带来一系列良好的生理和心理结果。例如，研究表明归属感会积极影响动机，并导致学业成绩的提高。<sup>11</sup>

这种联系感的一个组成部分是这样一种理念，即人们应该乐于让自己全身心投入工作，他们独特的经历和背景受到重视和赞扬。<sup>12</sup>专注于在组织内创造包容性的归属文化有助于创建一支欣欣向荣、多元化、充满动力的员工队伍。

我们的研究表明，与文化不太积极的组织相比，注重归属感和包容性效能导向的组织更有可能较低比例的产生员工工作倦怠。

鉴于有证据表明心理社会因素如何影响SDO效能和员工的倦怠水平，我们建议，如果你正在寻求进行成功的DevOps转型，你可以将文化相关问题纳入到总体转型工作中。

---

10 Baumeister & Leary, 1995. 《The need to belong: Desire for interpersonal attachments as a fundamental human motivation》. Psychological Bulletin (心理学公报), 117[3], 497 – 529页。

<https://doi.org/10.1037/0033-2909.117.3.497>

11 Walton等, 2012. 《Mere belonging: the power of social connections》. Personality and Social Psychology 期刊, 102[3]:513-32页。

<https://doi.org/10.1037/a0025731>

12 Mor Barak & Daya, 2014; 《Managing diversity: Toward a globally inclusive workplace. Sage》。

Shore, Cleveland, & Sanchez, 2018; 《Inclusive workplaces: A review and model》, Human Resources Review。  
<https://doi.org/10.1016/j.hrmr.2017.07.003>

## 第四章

# 哪些人参与了调研？

利用7年来的研究数据以及32,000多名专业人士的反馈，DevOps加速状态报告2021呈现了促使团队、企业获取最大成功的软件开发及DevOps的实践。

今年，1,200名来自全球各界的专业人士分享了他们的经验，帮助我们更好地理解获取更高效能的驱动因素。一言以蔽之，个人和企业统计数据中的关键指标与以往保持着高度的一致性。



# 个人和企业统计概况

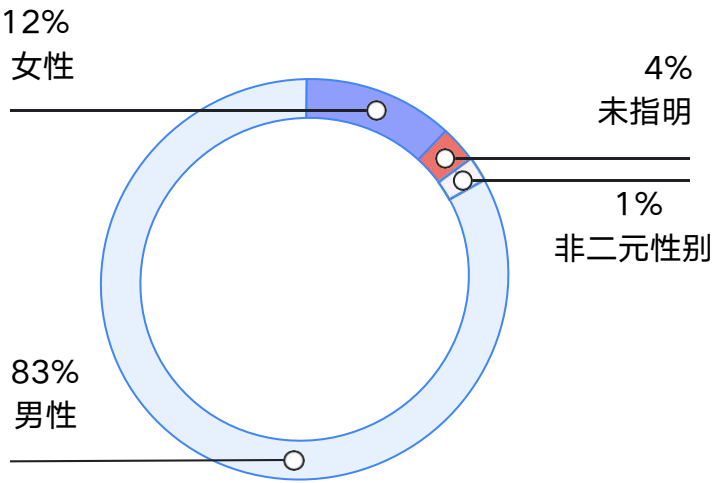
和往年一样，我们从每位受访者中获取个人统计数据，数据类型包括性别，残障情况，以及弱势群体从属情况。

而从今年的报告中可以看出企业统计数据中的关键指标与以往的报告保持着相当的一致性，包括公司规模、行业、地区。超过60%的受访者为企业工程师或经理，1/3的受访者工作于技术行业。同时我们还考察了来自金融服务业、零售业、工业生产企业的指标。

## 个人统计数据

### 性别

今年的样本中男性占83%、女性占12%、非二元性别占1%，与以往的调研基本一致。受访者表示所在团队中女性成员占25%，这一数值相比于2019年的16%有了大幅提高，与2018年的25%保持一致。

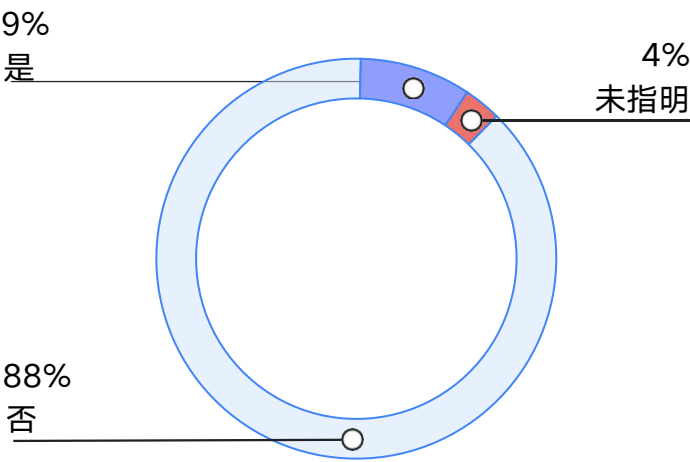


今年的受访者表示有25%的团队接纳了女性、中性成员，相比于2019年的低谷有了显著的恢复。



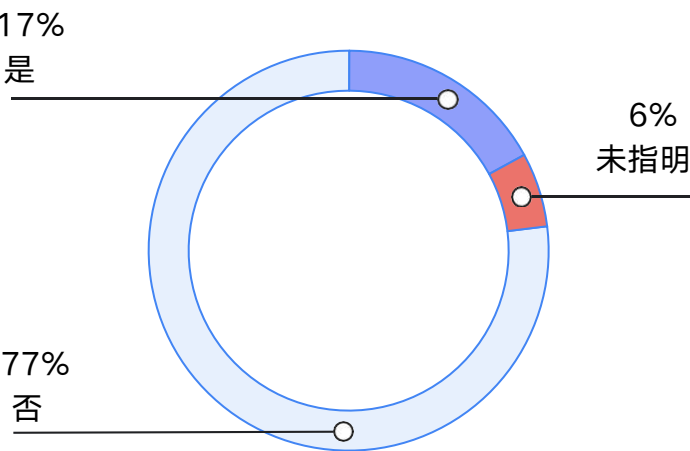
残障情况

残障情况依据联合国华盛顿小组残疾功能状况量表中的六个维度来确定。<sup>13</sup>今年是我们了解残障信息的第三年，残障人士的占比与2019年一样维持在9%。



弱势群体从属情况

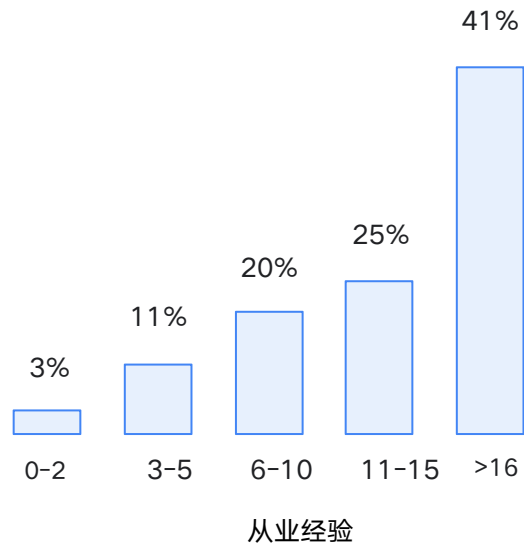
受访者是否属于弱势群体可以从种族、性别或其他特征来确定。今年是我们索取弱势群体信息的第四年，今年弱势群体人士的占比为17%，相比于2019年的13.7%有了小幅增长。



<sup>13</sup> <https://www.washingtongroup-disability.com/question-sets/wg-short-set-on-functioning-wg-ss/>

### 从业经验

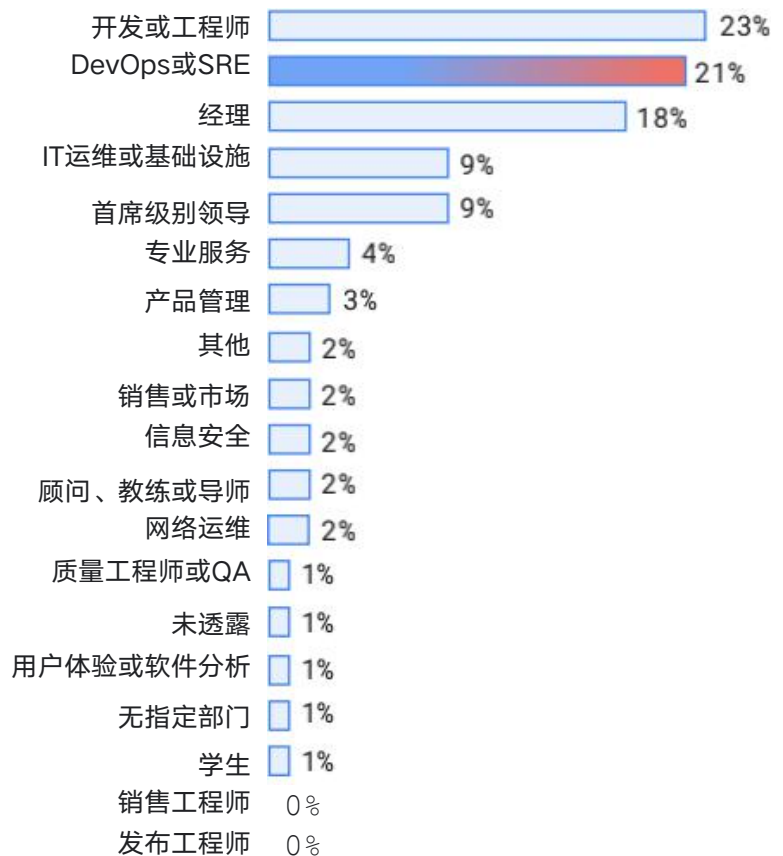
今年的受访者有非常丰富的从业经验，41%的受访者有16年或以上的从业经验，超过85%的受访者至少有6年的从业经验。



### 企业统计数据

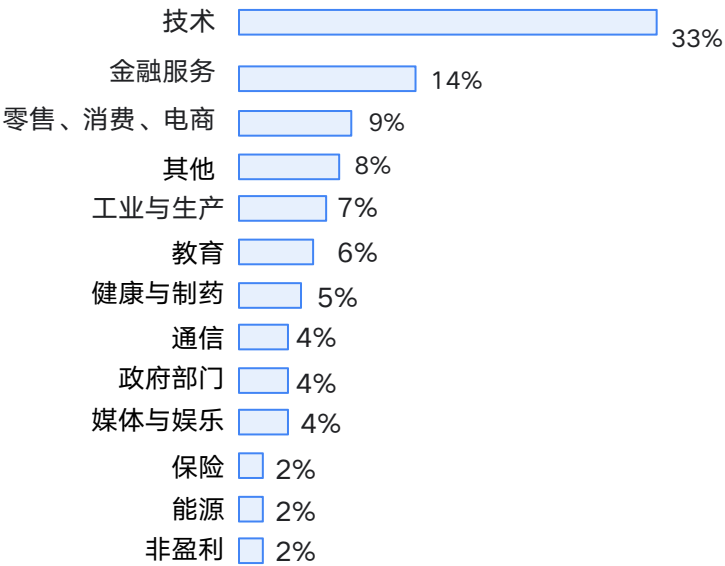
#### 部门

大部分受访者由开发或工程师团队（23%）、DevOps或SRE团队（21%）、经理（18%）及IT运维或基础设施团队（9%）组成。我们观察到顾问的占比由2019年的4%下滑至2%，首席级别领导的占比由2019年的4%上升至9%。



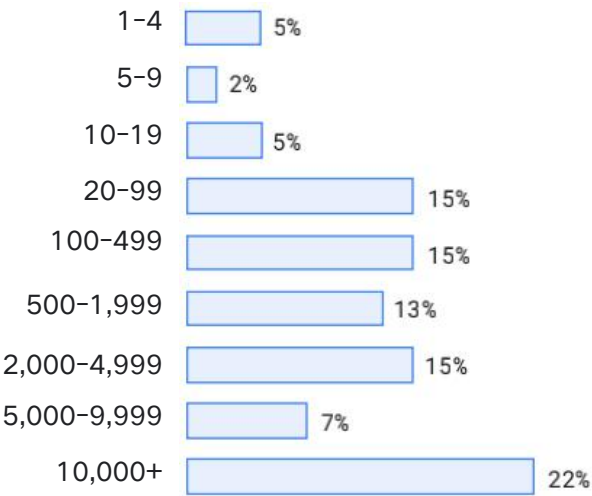
## 行业

和以往的报告一样，今年大多数的受访者工作于技术行业，紧随其后的是金融服务业、零售业以及其他。



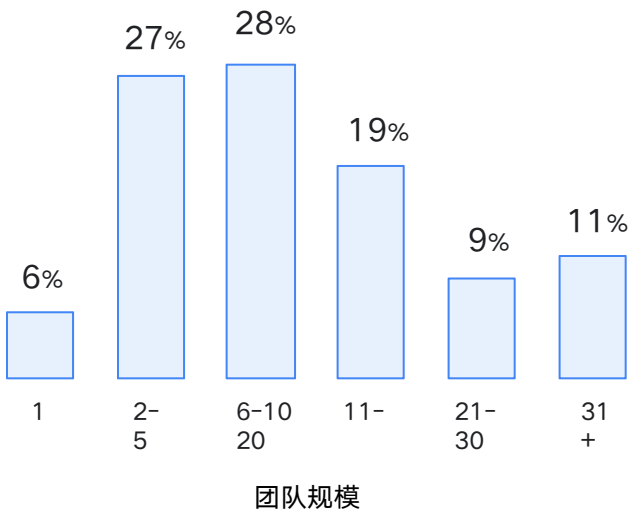
## 雇员规模

和以往的报告一样，受访者来自雇员规模各异的组织企业，其中来自规模超过10,000人的组织的受访者占22%，规模在5,000-9,999人的占7%，在2,000-4,999人的占15%，在500-1,999人的占13%，在100-499人的占15%，最后在20-99人的占15%。



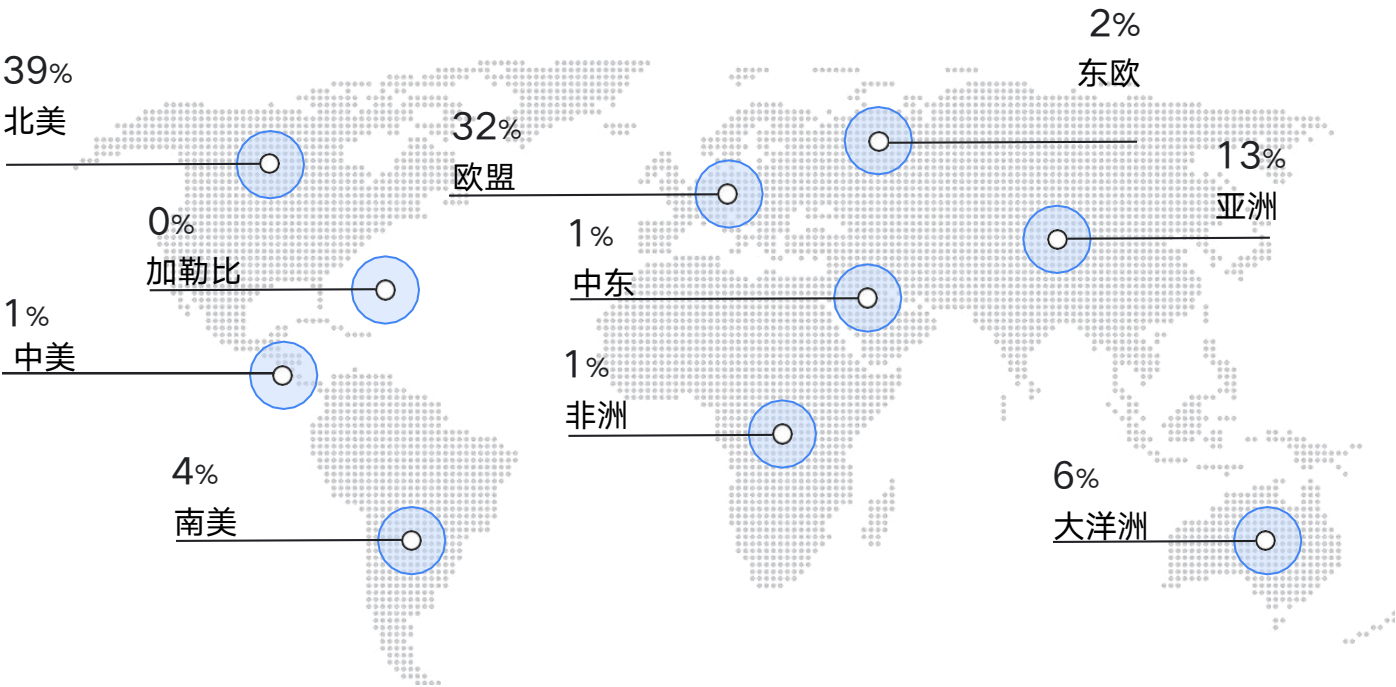
### 团队规模

超过一半的受访者（62%）工作于不超过10人的团队（6-10人的占28%，2-5人的占27%，还有6%的单人团队）。另有19%的受访者工作于11-20人的团队。



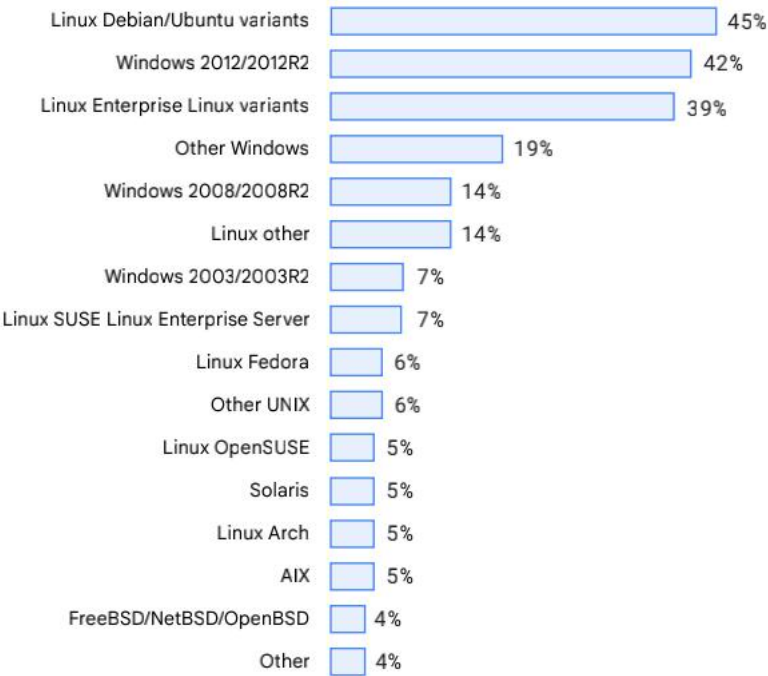
### 地区

在今天的报告中北美的占比有了下滑（2019年的50%到2021年的39%），此消彼长的是欧洲（2019年的29%到2021年的32%）、亚洲（2019年的9%到2021年的13%）、大洋洲（2019年的4%到2021年的6%）以及南美洲（2019年的2%到2021年的4%）占比的增长。



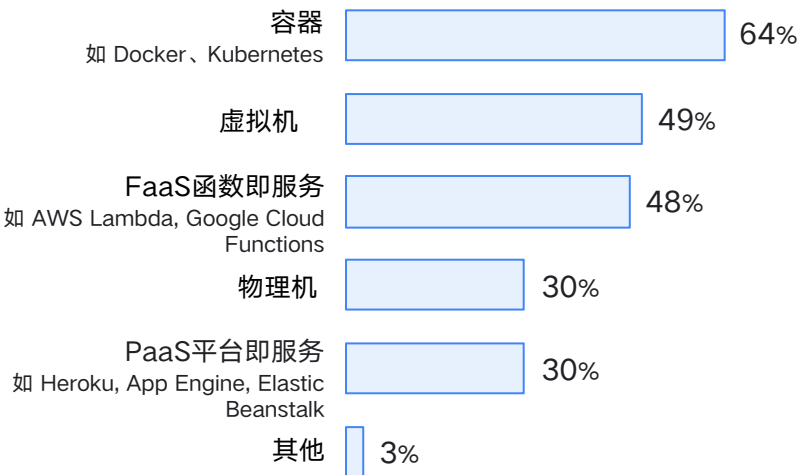
## 操作系统

操作系统的发行版本与以往的报告也保持一致。有受访者指出，这份操作系统列表中部分系统已经有更新的版本支持，我们特此表示感谢。



## 部署目标

今年我们考察了受访者将他们的服务或应用部署于何处。令人诧异的是大部分受访者使用了容器（64%），还有49%使用了虚拟机。也许这反映了行业逐步向现代部署技术转型。经核查，我们发现在部署目标上不同规模的企业并没有显著区别。





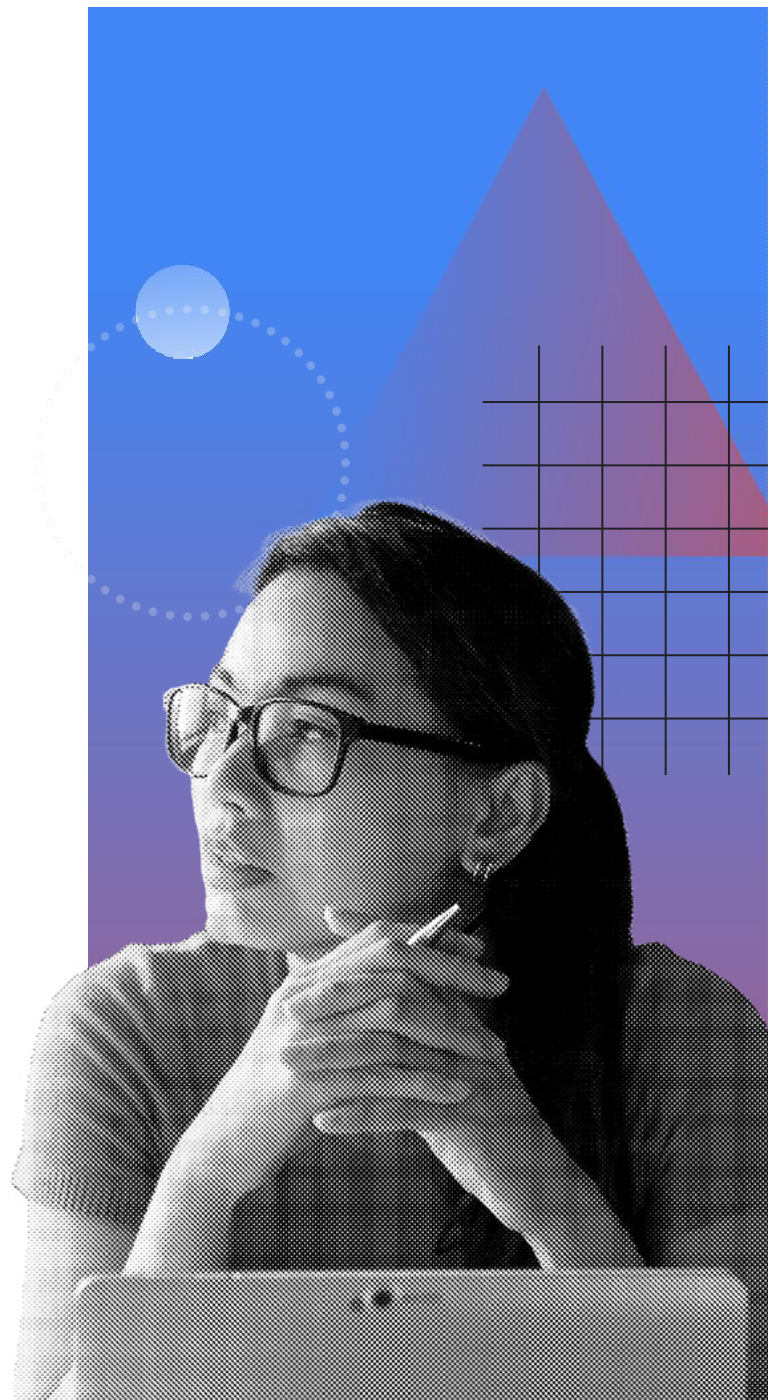
## 第五章

# 总结

经过7年的研究，我们继续感受到DevOps给组织带来的好处。组织年复一年的继续加速和改进。

一些团队拥抱其原则和能力，因而可以快速可靠地交付软件，这也直接为业务带来了价值。今年我们调查了SRE实践的影响、安全的软件供应链、质量文档，还重新回顾了对云计算的探索。每个领域都使人们和团队更加高效。我们关注于如何构建解决方案以适合人的能力，而不是让人去适合解决方案。

感谢所有参与今年调查的人，希望我们的调查能帮助您和您的组织建立更好的团队和更好的软件，同时保持工作和生活的平衡。



## 第六章

# 致谢

今年的报告是由一批充满激情的贡献者编写的。我们的同事付出了巨大的努力，设计调查问题、分析、写作、编辑和设计报告等只是这些努力中的一小部分。作者们想要感谢所有这些人的投入以及对今年报告的指导。所有的感谢人按字母顺序列出：

Pali Bhat	David Huh	Claire Peters
Maria Bledsoe	Vic Iglesias	Garrett Plasky
James Brookbank	Harish Jayakumar	John Ryan
Jan Bultmann	Nikhil Kaul	Vinay Srinivasan
Lolly Chessie	Lital Levy	Christina Storm
John Day	Amanda Lewis	Oren Teich
Rakesh Dhoopar	Ríona MacNamara	Finn Toner
Siobhán Doyle	Andrew Macvean	Marcin Treder
Alex Eldemir	Steve McGhee	Seth Vargo
Nicole Forsgren	Erin McKean	Salim Virji
Aaron Gillies	Jacinda Mein	Brenna Washington
Kelsey Hightower	Eric Maxwell	Michael Winser
Jez Humble	Raghu Nandan	Julia Yager-Reisen

## 第七章

# 作者



### Dustin Smith

Dustin Smith是谷歌的人类心理学家和员工用户体验研究经理，他在DORA项目上工作了3年。在过去的7年里，他研究了人们如何在各种情境下受到系统和环境的影响，包括软件工程、免费游戏、医疗保健和军事等。通过在谷歌的研究，他找到了让软件开发人员在开发过程中感到更快乐和更有效的领域。威奇托州立大学人因工程学博士学位。



### Daniella Villalba

Daniella Villalba是专属于DORA项目的用户体验研究员。她专注于寻找使开发人员快乐和高效的因素。在谷歌之前，丹妮拉研究了冥想训练的好处、影响大学生经历的心理社会因素、目击者记忆和对欺骗的忏悔等。她在佛罗里达国际大学获得了实验心理学博士学位。



## Michelle Irvine

Michelle Irvine是谷歌的一名技术作家，她的工作是在开发人员和他们所使用的工具之间架起桥梁。在谷歌之前，她从事教育出版工作，并且是一款物理模拟软件的作者。Michelle在滑铁卢大学获得了物理学学士学位，以及修辞和沟通设计硕士学位。



## Dave Stanke

Dave Stanke是谷歌的一名开发关系工程师，为客户提供采用DevOps和SRE的实践建议。在职业生涯中，他担任过各种职务，包括初创公司首席技术官、产品经理、客户支持、软件开发、系统管理员和平面设计师。他拥有哥伦比亚大学技术管理硕士学位。



## Nathen Harvey

Nathen Harvey是谷歌的一名开发关系工程师，他的职业生涯是帮助团队实现他们的潜力，同时将技术与业务成果结合起来。Nathen有幸与一些最好的团队和开源社区合作，帮助他们应用DevOps和SRE的原则和实践。Nathen参与编辑并贡献了《云工程师应该知道的97件事》。



## 第八章

# 研究方法

### 研究设计

本研究采用基于横断面研究理论的设计。这种设计被称为推理预测，是当今商业和技术研究中最常见的类型之一。推理设计应用于无法进行单纯的实验设计，同时倾向于站点实验的情况。

### 目标群体和样本

本次调查的目标人群是那些从事技术工作和正在转型的相关从业者和领导者，尤其是那些熟悉DevOps的人。我们通过电子邮件列表、在线推广、在线小组、社交媒体进行调查，并要求人们与他们的网络分享调查(即雪球取样)。

### 创建潜在的构想

我们尽可能使用先前验证过的构想来表述我们的假设和构想。我们在理论、定义和专家输入的基础上发展了新的构想。然后，我们采取了额外的步骤来澄清意图，以确保从调查中收集的数据具高度的可靠性和有效性。<sup>14</sup>

### 统计分析方法

**聚类分析。**我们根据部署频率、交付时间、服务恢复时长和变更失败率，使用聚类分析来确定我们的软件交付效能剖析。我们使用潜在类别分析<sup>15</sup>因为我们没有任何行业或理论原因来确定预定的集群数量，我们使用贝叶斯信息准则<sup>16</sup>来确定最优的集群数量。

**测量模型。**在进行分析之前，我们使用探索性因子分析和方差旋转主成分分析确定结构。<sup>17</sup>我们使用平均方差提取(AVE)、相关性、克隆巴赫系数法和组合信度对收敛和发散的有效性和可靠性进行了统计检验。<sup>18</sup>

**结构方程建模。**我们使用偏最小二乘法(PLS)分析对结构方程模型(SEM)进行了检验，这是一种基于相关性的SEM。<sup>19</sup>

---

14 Churchill Jr, G. A. 《A paradigm for developing better measures of marketing constructs》. Marketing Research期刊 16:1, (1979), 64 – 73页。

15 Hagenars, J. A.和 McCutcheon, A. L. (编著). (2002). 《Applied latent class analysis》. 剑桥大学出版社.

16 Vrieze, S. I. (2012). 《Model selection and psychological theory: a discussion of the differences between the Akaike information criterion (AIC) and the Bayesian information criterion (BIC)》(模型选择与心理学理论: 赤池信息准则(AIC)与贝叶斯信息准则(BIC)的区别探讨)。Psychological methods杂志, 17[2], 228页。

17 Straub, D., Boudreau, M. C.和 Gefen, D. (2004). 《Validation guidelines for IS positivist research》. 信息系统协会通讯, 13[1], 24页。

18 Nunnally, J.C. 《Psychometric Theory》. 纽约: McGraw-Hill (出版社), 1978

19 Hair Jr, J. F., Hult, G. T. M., Ringle, C. M.和Sarstedt, M.(2021). 《A primer on partial least squares structural equation modeling(PLS-SEM)》. Sage publications (出版社)。



## 第九章

# 扩展阅读

更多关于DevOps能力的信息：

<https://cloud.google.com/devops/capabilities>

查看更多关于站点工程师(SRE)的资源：

<https://sre.google>

DevOps快速检查：

<https://www.devops-research.com/quickcheck.html>

探索DevOps研究项目：

<https://www.devops-research.com/research.html>

了解关于谷歌云应用现代化项目：

<https://cloud.google.com/camp>

阅读“DevOps转换的ROI:如何量化现代化计划对你的影响”白皮书：

<https://cloud.google.com/resources/roi-of-devops-transformation-whitepaper>

参考之前的DevOps状态报告：

2014 DevOps状态报告: <https://services.google.com/fh/files/misc/state-of-devops-2014.pdf>

2015 DevOps状态报告: <https://services.google.com/fh/files/misc/state-of-devops-2015.pdf>

2016 DevOps状态报告: <https://services.google.com/fh/files/misc/state-of-devops-2016.pdf>

2017 DevOps状态报告: <https://services.google.com/fh/files/misc/state-of-devops-2017.pdf>

2018 DevOps状态报告: <https://services.google.com/fh/files/misc/state-of-devops-2018.pdf>

2019 DevOps状态报告: <https://services.google.com/fh/files/misc/state-of-devops-2019.pdf>