

采用现代化 Dev+Ops 模型

现代化 Dev+Ops 通过弥合开发职能部门和运营职能部门之间的差距，使企业能够更快、更稳定地满足客户不断变化的需求。

DevOps 改变了世界。Amazon 和互联网改变了软件的交付方式，从商店购买的盒装磁盘转变为基于亚马逊云科技的数字服务。但正是 DevOps 将软件的更新周期从几个月甚至几年缩短至几天，并将 Dev+Ops 更紧密地结合在一起。

各种形式和规模的公司都抛弃了瀑布式产品管理周期，转而采用敏捷的 DevOps 来实现更快的创新、更好的安全性、性能和弹性，以及让开发人员和客户更快乐。但 DevOps 即使存在所有这些益处，它也不是完美的。在云端开发和托管应用程序成为标准实践之前，DevOps 就已经存在。尽管它已经存在很长时间，但各个企业的解释仍然千差万别。例如，许多企业仍然认为 DevOps 是一个专门的团队，而另一些企业则不这样认为，这可能意味着开发人员要负责所有运营工作。随着技术的发展，我们对 DevOps 的定义也在不断发展。我们将这种方法称为“现代化 Dev+Ops”，其核心是通过在开发过程的早期共享合规性、可观测性、弹性和基础设施等运营任务，并利用人工智能 / 机器学习 (AI/ML) 进行增强，将开发人员和运营团队更紧密地结合在一起。

DevOps 状况报告指出，当公司采用 DevOps 并将部署频率从每周 / 每月改为每小时 / 每天后，其交付时间从几个月缩短至几天，变更失败率从 46-60% 降至 0-15%。对于许多客户而言，云端的成功不仅取决于他们是否采用了按需、可扩展的基础设施，而且至少在一定程度上取决于他们如何转变自己的开发和运营实践。现在，[Coca-Cola Argentina](#)、[3M](#)、[Lululemon Athletica](#) 和 [The Washington Post](#) 等客户已经过渡到现代化 Dev+Ops 方法，亚马逊云科技的 Dev+Ops 服务使他们的开发团队与运营团队更紧密地结合，从而进一步扩展 DevOps 的益处。

“借助亚马逊云科技，我们的部署时间已经从六周缩短到一周，而且我们预计很快会每天进行多次部署。”¹

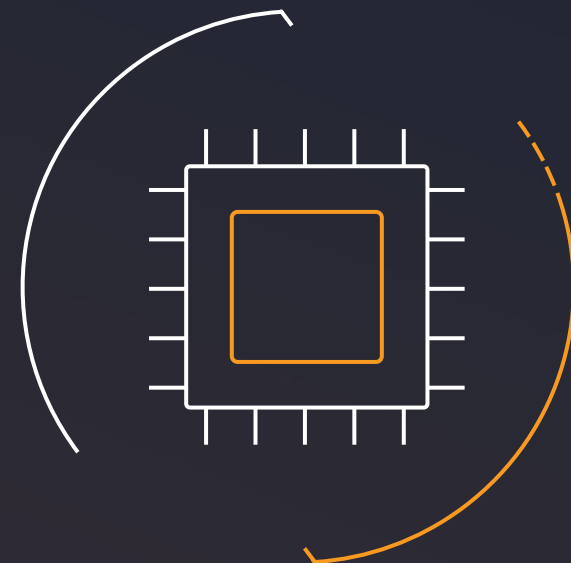
– Rick Austin，先进技术经理，
3M Health Information Systems

“借助 Amazon CloudFormation 模板和 Amazon CodePipeline，我们将构建新生产账户的时间从两天缩短到数分钟之短。这意味着我们可以启动成本极低且建立时间极短的小型项目。有了这种敏捷性，我们就可以进行试验并找到最佳解决方案，而不必止步于我们现有的资源。借助亚马逊云科技，我们就能够比以前更快地推出新功能和应用程序。”²

– Sam Keen，产品架构总监，
Lululemon Athletica

¹ <https://aws.amazon.com/solutions/case-studies/3m-health-information-systems/>

² <https://aws.amazon.com/solutions/case-studies/lululemon-athletica/>



DevOps 更技高一筹的团队

77%

在事件发生后不到一天的时间内
恢复服务

60%

在不到一天的时间内
完全修复安全漏洞

3 倍

提高变更管理效率

许多客户还没有采用 Dev+Ops，但正在转向这种方法，以寻找能够帮助他们更快地交付并减少错误的解决方案。其他客户已经开始了 Dev+Ops 之旅，但仍在努力达到他们预期的速度和成功水平。在很多情况下，客户面临的问题是他们觉得这太难了。

- 1 开始时太难了。它看起来令人望而生畏。要实现变革，有很多东西要学，也有很多事情要做。
- 2 现有的行业解决方案没有充分考虑企业需求：安全性、合规性、高可用性、访问控制等。因此，DevOps 团队花费了大量时间构建一种解决方案，接下来只需要替换为此解决方案即可。
- 3 许多行业解决方案不足以让 DevOps 团队轻松实现 Dev+Ops。例如，许多解决方案是被动的，而不是主动的。问题已经出现时，这些解决方案才告知客户，而不是在一开始就通过预先警告和建议最佳后续操作来帮助防止问题的发生。

改变文化是现代化 Dev+Ops 中最难的部分。要让所有人达成共识，需要领导力、时间和承诺。随着开发人员开始参与更多过去由运营团队管理的职责，就需要有责任共担意识。下面是一套文化和行为方面的工作方式，当采用这些方式时，能够建立一种机制来构建高性能的 Dev+Ops 企业。

这本电子书涵盖了亚马逊云科技 在高绩效的 Dev+Ops 企业中确定 的指导原则：

- › 实践问责制
- › 衡量自己
- › 逐步改进
- › 实现完全自动化
- › 尽可能编写编码
- › 实施鼓励和强制措施
- › 将工具标准化

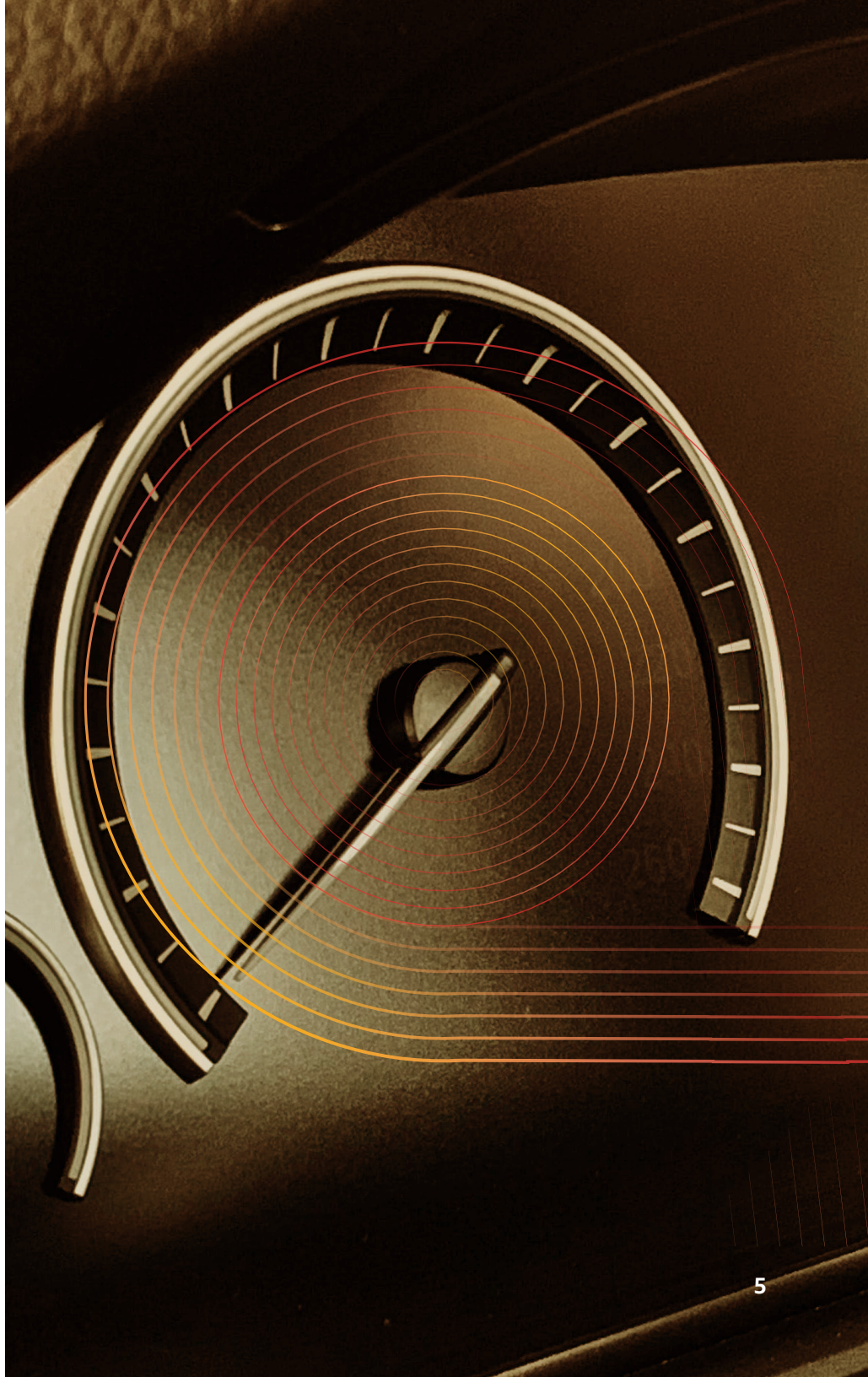
实践问责制

Dev+Ops 的目标可能是软件交付的速度，但最初目的是将开发和运营更紧密地结合在一起，使其协作解决问题。虽然企业结构随着时间的推移发生了变化，但注重问责制文化的解决方案不曾改变。团队和个人可以拥有专长和专业领域，但他们必须在处理和解决问题时共担责任。

衡量自己

Dev+Ops 的最终目标是更快地前进，但是团队经常不知道从哪里开始。您可以从衡量自己开始。这意味着衡量您当前的发布时间。软件交付过程的哪些部分耗时最长？为什么？我们可以改进吗？对您的成功来说，还有哪些重要的关键绩效指标？在交付过程中，在开发和运营之间是否有一些领域被孤立起来？从衡量这些指标的输入和结果开始，您能够找到可以着重处理并展现成果的地方，从而在您的团队中建立信心。

- › 发布时间
- › 管道中的瓶颈或障碍
- › 发布速度



逐步改进

高性能的 Dev+Ops 企业不是在几天或几周内就能从无到有建立的。重要的是要确定一些指导原则，然后从小处着手，进行稳步但渐进的改进。同样，如果您一直把自己束缚在大的变化和大的发布中，您将无法加快软件交付过程。对于人类来说，在需要理解的新事物数量有限的情况下，处理小的、渐进的变化比处理大的、复杂的变化要容易得多。

实现完全自动化

Dev+Ops 的目标是让一切保持连续运动。自动化减少了与人工干预相关的时间和错误。有很多方法可以为您的 Dev+Ops 工作流添加自动化，但最基本的一种方法是，添加具有自动化测试、部署和回滚功能的 Dev+Ops CI/CD 管道。

尽可能编写编码

自动化和 Dev+Ops 技巧的另一部分是，对基础设施和策略等内容进行编码，以便人类和机器都能理解它们。最基本的使用案例是基础设施即代码（IaC），它根据您的定义自动预置您的云基础设施，为您提供一个蓝图，帮助您跟踪更改并在整个企业内共享最佳实践。另一个例子是策略即代码，它使中心团队能够编写自动执行的规则。

基础设施即代码文件包括：

- › 计算
- › 存储
- › 身份、访问、安全
- › 应用程序资源

实施鼓励和强制措施

为确保遵循最佳实践，既要鼓励使用又要强制实施最佳实践。可以通过使用在企业内共享的最佳实践模板（通常使用基础设施即代码），来鼓励使用最佳实践。可以通过策略即代码等流程来强制实施最佳实践。但紧接着鼓励和强制措施的是，使用规则和人工智能在潜在错误影响用户之前发现它们，并就如何实施最佳实践为 Dev+Ops 团队提供指导。

将工具标准化

有了标准化的工具，企业就可以进行扩展。如果不能跨异构工具集强制实施策略和最佳实践，那么编写策略和确保最佳实践就会困难得多。使用个别供应商提供的旨在协同工作的工具会有所帮助，但最终更多地实现标准化才能节省时间并减少错误。



为什么选择亚马逊云科技？

亚马逊云科技处于独特的地位，通过为客户提供一套核心的现代化 Dev+Ops 实践、工具和培训，帮助客户更快地进行云转型。借助亚马逊云科技，客户可以更快地实现云的益处，并加速创新。

1

Amazon 是 Dev+Ops 和创新的先驱

Amazon 开创了许多在 Dev+Ops 中常见的最佳实践，在与高绩效客户合作以提供满足他们需求的解决方案方面，拥有十多年的经验。默认情况下，亚马逊云科技服务包含 Amazon 最佳实践。

2

亚马逊云科技拥有最广泛、最深入的现代化 Dev+Ops 集成服务组合

从核心监控和 CI/CD 服务到基础设施即代码，再到使用大数据和人工智能（AI）提供主动建议和推荐的最新服务，亚马逊云科技拥有您构建高绩效 Dev+Ops 团队或企业所需的所有服务。

3

安全性是工作重点；可用性是首要工作

默认情况下，亚马逊云科技 Dev+Ops 服务采用安全最佳实践构建。例如，默认情况下，每个项目的 CI/CD 管道都是隔离的，因此可以轻松控制访问权限，使之仅限于需要相应权限的用户。此外，亚马逊云科技服务都是高度可用的，因此团队可以高效工作。

4

为开发人员和企业构建

许多开发工具忽略了考虑企业需求。亚马逊云科技专注于开发人员和企业的需求，便于实施治理和合规性控制，从而使开发团队能够快速、安全地开展工作。

行动计划

那么，您如何在亚马逊云科技上实现现代化 Dev+Ops？
下面是各种形式和规模的企业用来制定上述指导原则的推荐最佳实践：

GitOps

GitOps 是一种方法，它将基础设施即代码托管在 Git 存储库中，并遵循与应用程序软件代码相同的合并请求流程。GitOps 旨在消除对应用程序基础设施的任何带外更改。基础设施即代码是一种以代码格式定义应用程序基础设施的实践，通常使用模板或文件，并自动预置基础设施以匹配该文件定义。使用 GitOps 时，基础设施定义文件托管在 Git 存储库中，每次提交都遵循合并请求流程，就像应用程序软件代码一样，并通过持续集成和持续交付（CI/CD）自动化来执行流程。使用 GitOps 时，基础设施文件会进行测试，基础设施预置实现自动化，并在出现错误时回滚到之前的状态。

亚马逊云科技提供了围绕 Amazon CloudFormation 构建的完整 GitOps 解决方案。借助 Amazon CloudFormation，开发人员和运营团队能够在控制台使用 YAML 或 JSON 等常用基础设施语言或者使用 Amazon Cloud Development Kit（CDK）轻松定义、预置和管理他们的基础设施即代码文件。借助 Amazon CDK，开发人员能够使用 Python、TypeScript、C#、Java 和 JavaScript 等常用编程语言定义 CloudFormation 模板。CloudFormation 与 Amazon CodePipeline 具有一流的集成，可实现 CI/CD 工作流自动化，并与 Amazon CodeCommit、GitLab、Bitbucket 和 GitHub 等流行的版本控制服务集成。创建新的基础设施定义文件后，该文件会托管在受版本控制的环境中。对 CloudFormation 模板进行更改时，会触发 CI/CD 自动化，开发人员能够轻松地部署或回滚 / 前滚他们所做的更改。

“随着 Rivian 的快速发展，我们需要一个高度可扩展的系统。原本需要五天的更改现在只需几分钟即可完成。”³

– Surendra Balu, 3DExperience 技术负责人, Rivian

持续集成和持续交付

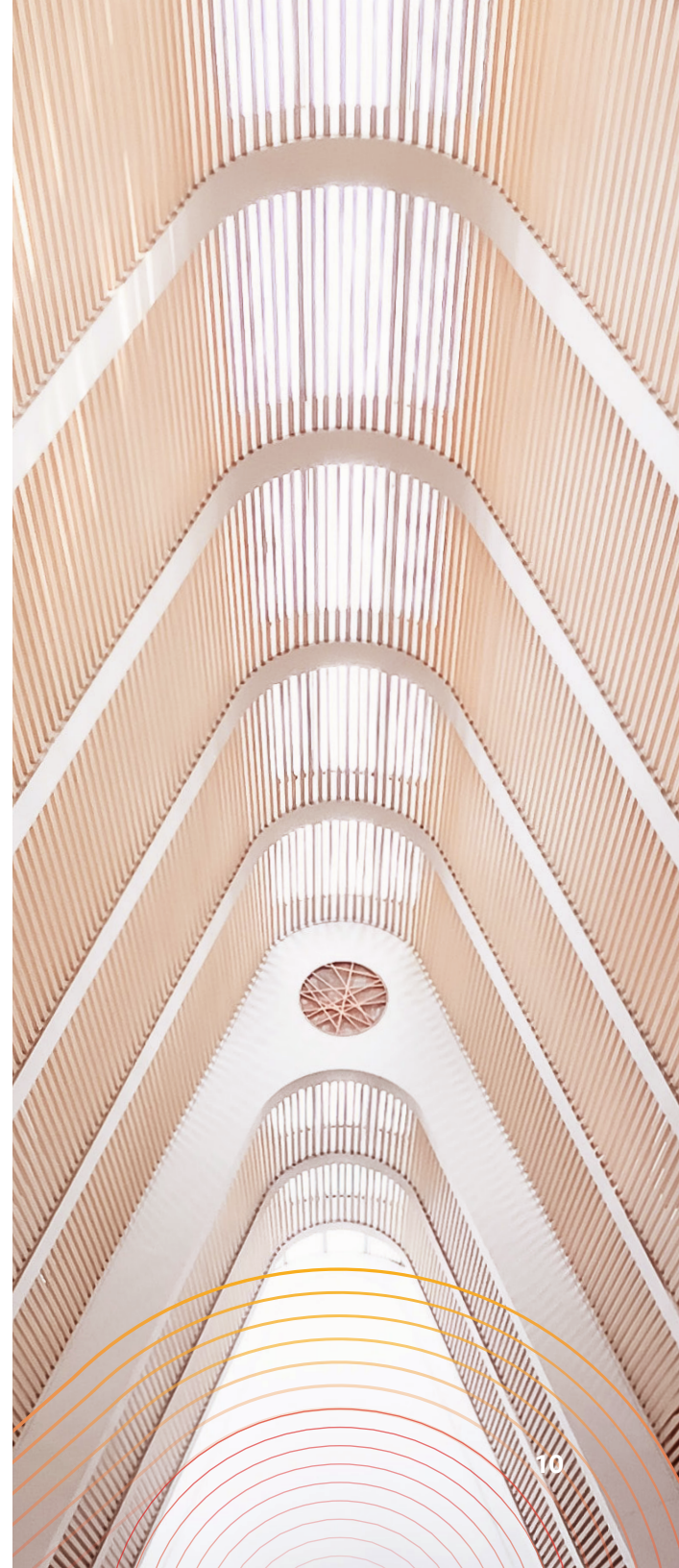
CI/CD 是使用包含构建、测试和部署的管道来实现软件发布准备工作自动化的基本最佳实践。CI/CD 通过消除容易出错的手动流程并消除照看软件发布的需要，帮助团队加快进度并减少错误。对于发布时间较长的客户，CI/CD 通过对系统进行持续更改来解决这个问题。工程师编写测试来检测代码中的潜在错误或违规，并将更新退回各团队。当一个错误被发布到生产环境时，各团队可以自动回滚到一个稳定的版本并维护正常运行时间。

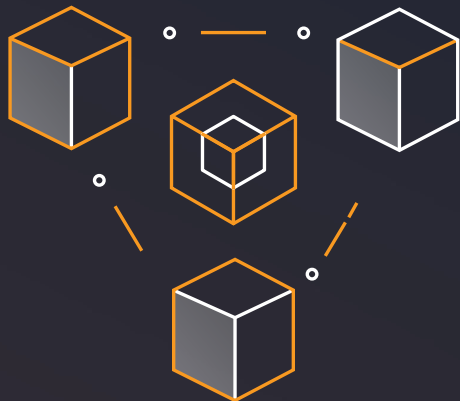
亚马逊云科技为在企业或初创公司中构建 CI/CD，提供了一种完全托管式解决方案。Amazon CodeBuild、Amazon CodeArtifact、Amazon CodePipeline 和 Amazon CodeDeploy 是一组互连的 Dev+Ops 服务，专门用于帮助客户实施 CI/CD。这些服务借鉴了 Amazon 的内部 CI/CD 工具，并在默认情况下鼓励使用最佳实践。例如，通过在单个 CodePipeline 管道中隔离每个项目，安全意识强的客户可以轻松地将权限锁定给那些应该有权访问该项目的用户，从而无需担心共享

编译服务器中的漏洞。此外，亚马逊云科技使 CI/CD 更轻松，需要的专业知识更少。在软件发布周期的拉取请求阶段，像 Amazon CodeGuru Reviewer 这样的服务能够提供帮助，方法是通过机器学习来增强代码审查，帮助识别开发过程中的关键问题、安全漏洞和难以发现的错误。开发人员能够获得指导，并且能够在新功能部署到生产环境之前解决问题，从而改进代码。此外，Amazon Proton 是一个服务平台，工程团队可以用来为他们的开发团队预配置基础设施预置、代码部署、监控和更新所需的所有不同工具。

“借助 Amazon CodeBuild，我们的应用程序构建过程现在大约需要 10 分钟；而过去使用 Jenkins 时，需要长达一个小时。在 Jenkins 上要获得同样的性能，我们需要花费四倍的成本，因为我们需要启动 50 个 Jenkins 实例才能快速完成构建。”

– Bryan Kane，高级工程师，Coursera





AIOps

AIOps 是向更自动化和更主动的机制的转变，使团队能够自信地更快地创新。AIOps 旨在通过在 Dev+Ops 工作流中利用 AI 来减少开发人员所需的知识量并增强他们的体验，能够在问题出现之前提供有用的洞察，帮助团队采取主动，默认执行最佳实践，并最终加快创新速度。例如，在与最佳实践的偏差、并发问题和其他常见编码错误有机会影响系统之前，开发人员就能够检测到它们，并获得可提高系统性能的可行建议。

Amazon CodeGuru Reviewer 是一种基于机器学习的开发工具服务，借助以二十年的 Amazon.com 和亚马逊云科技卓越运营为基础的 Amazon Web Services Machine Learning 模型，能够轻松分析代码，确保安全最佳实践无处不在。借助 Amazon CodeGuru Profiler，客户能够识别运行应用程序时成本高昂且效率低下的方法，并提供可操作的步骤来修复它们，从而节省资金。此外，借助 Amazon DevOps Guru，客户能够采用 AIOps 解决方案来提高应用程序的操作性能和可用性。DevOps Guru 能够识别异常的应用程序行为，发现可能导致应用程序或服务中断的关键问题，并提供纠正问题的建议。

“我们一直在寻找各种方法来减少我们的团队花在解决操作问题上的时间，我们现在正在使用 Amazon DevOps Guru 并利用其基于机器学习（ML）的洞察，来帮助我们快速识别、关联和纠正操作问题。借助 Amazon DevOps Guru 提供的洞察，我们的团队现在可以快速找到问题，而不必从头开始尝试找出问题的根源。我们的 IT 团队已经大大缩短了平均恢复时间（MTTR），并且节省了解决问题的数小时时间 – 同时能够确保客户尽可能获得最佳的最终用户体验。”

– HCL Technologies

持续可观测性

持续可观测性是能够在任何给定时间点了解系统状态的实践。可观测性使您能够检测、调查和修复问题。持续可观测性使工程团队能够深入洞察他们的应用程序运行状况和性能，以便他们能够对需要关注的问题进行故障诊断和解决，最终改善他们的终端用户体验。云应用程序和资源在永无止境的数据流中生成数十亿个指标、日志和跟踪，因此跨分布式应用程序分析性能会具有挑战性。通过识别来自任何来源的用户影响或者快速找到损坏或昂贵的代码路径，借助可观测性提高开发人员的工作效率。

做到持续可观测性的第一步是，使用 Amazon CloudWatch（或 Amazon Managed Service for Prometheus、Amazon Managed Service for Grafana 或 Amazon Distro for OpenTelemetry，根据偏好设置）构建可观测性控制面板。可观测性控制面板解决了在我们的云服务中掌握活动情况的挑战。它是面向用户的系统视图，通过显示时间序列指标、日志、跟踪和告警数据，提供系统行为方式的简明摘要。构建该控制面板后，客户应设置 CloudWatch 告警，以持续提醒他们云环境中的潜在问题。

“我们使用 CloudWatch 为 Amazon Simple Queue Service 创建提示，并为 EC2 Auto Scaling 组扩缩操作。这些洞察为我们的扩缩决策提供了依据，使我们能够在不影响客户体验的情况下，最有效地利用亚马逊云科技计算资源。借助 CloudFormation，我们能够为面向客户的应用程序所使用的亚马逊云科技资源部署逻辑堆栈，这样就跨多个亚马逊云科技账户实现了可重复且一致的部署。因此，我们能够使用配置即代码这种简单的方法来组合应用程序堆栈，而不是依赖于可能不一致且容易出错的手动配置。”

– Martin Costello，高级工程师，Just Eat

事实：

亚马逊云科技每月平均监控
1000 万亿个指标观测值

构建控制面板以实现操作可见性

在这篇 Amazon Builders' Library 文章中，亚马逊云科技首席工程师 John O'Shea 介绍了 Amazon DevOps 团队如何考虑构建控制面板以了解系统状态的情况。

[了解详情 >](#)

持续合规性

云端的合规性是一种全新的范例。云资源的速度和规模使得无法使用传统的合规性方法。电子表格和静态数据库无法处理资源流失情况。成功的关键是尽可能实现自动化和简化。亚马逊云科技提供广泛的服务来帮助客户简化云合规性。要了解为实现合规性而使用哪些服务，一种方法是使用内部审计师协会（国际公认的内部审计实践权威机构）的三线模型来构建它们。该模型建立了一种机制：企业通过特定 IT 角色和职责沿三道防线管理风险。第一道防线管理风险，第二道防线监督风险，第三道防线提供风险保证。

亚马逊云科技提供了非常适用于这三道防线中的每一道防线的广泛服务。例如，对于在第一道防线中管理风险的服务，我们有 Amazon Config、Amazon CloudTrail 和 Amazon Systems Manager。使用这些服务能够定义和部署单独的控制来管理风险。在第二道防线中，Amazon Security Hub 和 Amazon CloudWatch 可帮助您监督整个企业的风险。Amazon Audit Manager 是提供风险保证的完美工具，这是第三道防线，客户能够用来自动收集合规性证据以进行审计。

但使用这些工具只是本案例的一部分。客户需要通过持续合规性使流程自动化。为了采用持续合规性，我们建议利用合规性即代码的概念。通过合规性即代码方式，您可以使用像 Amazon CloudFormation 这样的工具定义三线模型的元素。然后，您可以使用自动化管道（如 Amazon CodePipeline）测试和部署这些元素。由于需要新的控制和证据，因此无需在每个账户中手动实施它们。代码在控制定义中得到更新，并通过管道推送。最终结果是实现了一种新的风险管理控制，这种控制立即被添加为审计证据。持续合规性还有一个额外的益处，即提供切实的保证，确保控制措施到位，并确保收集的证据没有被篡改。利用三线模型并通过持续合规性实现流程自动化，是大规模处理合规性的最佳方法。

“借助亚马逊云科技，我们彻底改变了工作方式，现在能够通过实现合规性流程自动化，在不增加员工人数的情况下进行扩展。对于我们这些对合规性没有深入了解的人来说，合规性管理起来会很复杂，而 Amazon Config 能够帮助我们使用开箱即用的模板实现合规性。控制面板为用户提供一些信息（如不合规资源），用户能够使用这些信息改善安保状况，还能够以此为起点，与企业其他人员讨论我们需要关注和修复的问题。作为通过云技术为内部团队和外部团队提供技术领导力的数字团队，能够集中查看和控制我们整个企业的安保状况是非常重要的。预先打包的 Config 一致性包模板简化了这一过程，是加快我们向云端的整体迁移速度的关键因素。”

– Andrew Clark, 高级解决方案架构师,
Baker Tilly Digital

构建共享服务平台

每个团队和企业实施 Dev+Ops 的方式略有不同。许多企业有一个中央平台团队，通过实现安全性、软件交付、监控和联网的标准化，帮助支持开发团队并减轻运营负担。共享服务平台是供开发人员使用的定制的自助服务界面，可简化代码部署。中央团队能够控制安全性、软件交付、监控和联网标准的制定，这些标准必须在部署的所有应用程序中采用。这使得开发人员的工作效率更高，平台团队的控制能力也更强。

亚马逊云科技为您提供了在其上构建共享服务平台所需的所有工具。对于基本的生产使用案例，Amazon Copilot CLI 能够轻松创建一个“功能齐备”的开发环境，其中提供开箱即用的多账户 CI/CD、安全组和监控功能。对于更复杂的情况，亚马逊云科技提供了一项名为 Amazon Proton 的完全托管式服务。Amazon Proton 是第一个面向容器和无服务器应用程序的完全托管式交付服务。平台团队能够使用 Amazon Proton 连接和协调基础设施预置、代码部署、监控和更新所需的所有不同

工具。Amazon Proton 通过为平台团队提供管理这种复杂性和实施一致标准所需的工具来解决这个问题，同时让开发人员能够轻松地使用容器和无服务器技术来部署代码。如果您的企业需要更多控制权来构建共享服务平台，可供借鉴的是，许多企业已成功使用 Amazon CloudFormation 和 Amazon Service Catalog 创建和管理获准在亚马逊云科技上使用的 IT 服务目录。同样，其平台团队熟悉动态编程语言的客户已使用 Amazon CDK 为其开发团队构建支持平台。

“我们的技术人员能够使用 Amazon Service Catalog 自助服务门户轻松地将应用程序构建部署到生产环境中，而无需开立 IT。因此，他们可以在五分钟内启动一个新的应用程序堆栈，而不是通常需要的数周时间。凭借 Amazon Service Catalog，我们正在 Wiley 实现 DevOps 和自动化。”⁴

– Meltem Dincer，平台能力副总裁，Wiley

⁴<https://aws.amazon.com/solutions/case-studies/wiley/>

无论您的企业处于 Dev+Ops 之旅的哪个阶段，亚马逊云科技都提供广泛的解决方案，旨在弥合开发人员和运营团队之间的差距。

亚马逊云科技开发工具易于上手，涵盖安全性、合规性、高可用性和访问控制等主要企业需求，并为客户提供主动解决方案，根据最佳实践提前提供警告和建议，在问题发生之前识别和预防问题。借助亚马逊云科技，客户能够通过我们的核心现代化 Dev+Ops 实践、工具和培训，更快地实现云科技的众多益处并加速创新。

了解有关现代化 DevOps 的更多信息，