



加快取得可观测性成果

AWS 规范性指导



AWS 规范性指导: 加快取得可观测性成果

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon 的商标和商业外观不得用于任何非 Amazon 的商品或服务，也不得以任何可能引起客户混淆、贬低或诋毁 Amazon 的方式使用。所有非 Amazon 拥有的其他商标均为各自所有者的财产，这些所有者可能附属于 Amazon、与 Amazon 有关联或由 Amazon 赞助，也可能不是如此。

Table of Contents

简介	1
目标受众	1
目标	1
概述	2
为什么你需要重新考虑你的可观测性策略	2
可观测性工具和框架	2
第 1 阶段：定义你的北极星	4
在开发生命周期的早期集成可观察性（左移方法）	4
建立有效的组织和团队结构	5
跟踪成本分配	6
定义标准	7
建立上报流程	7
通过培训提高技能	7
第 2 阶段：实现可观察性	8
选择您的可观测性平台	8
功能集	8
许可和部署模式	9
定价维度	9
团队技能	7
操作和维护	9
对您的应用程序进行检测	9
收集遥测数据	10
实现可观测性组件	10
验证可观测性系统	10
第 3 阶段：检查、调整和迭代	11
实施定期审查	11
庆祝胜利	12
从事件中吸取教训	12
后续步骤和资源	13
资源	13
文档历史记录	14
.....	xv

加快取得可观测性成果

亚马逊 Web Services 的 Jyothi Madanlal、Abhaya Chauhan 和 Jose Augusto Ferronato

2025 年 7 月 ([文档历史记录](#))

亚马逊副总裁兼首席技术官沃纳·沃格尔斯博士曾经说过：“一切都会失败... 所以要做好失败的计划，一切都不会失败” (re [AWS : Invent 2024 主题演讲](#))。采用这种思维方式是有效而强大的可观察性实现的核心。

本策略文档探讨了如何提升可观测性态势，以创建更具弹性和更高效的系统。它还讨论了如何定义一个锚定目标（北极星），该目标可以在整个组织中提供有效的协调，从而提高客户信任度、品牌声誉和团队士气。该文件重点介绍了需要考虑的关键事项以及加快可观测性结果的推荐方法。这种方法包括开拓新的商机、提供更好的用户体验、优化性能和成本，以及提高整个组织的运营效率。无论您的可观测性实现是否成熟，本文档都提供了增量且可操作的见解，以实现您的可观测性目标。

目标受众

此信息面向以下受众：

- 负责定义可观测性策略的技术领导者和架构师
- 负责监督卓越运营的工程经理和 DevOps 领导者
- 负责构建可观测性基础的平台工程团队
- 负责实施可观测性实践的站点可靠性工程师 (SREs)

目标

本指南中的信息可帮助您提升可观察性实践，以创建更具弹性和更高效的系统，这些系统在整个组织中保持一致。它为您提供了一个框架，用于在人员、流程和技术这三个支柱上采用良好的可观察性实践。无论您选择使用何种工具，这些信息都具有广泛适用性。如果您对支持可观测性的 AWS 服务感兴趣，请参阅网站上的[监控和可观察性](#)。AWS

概述

为什么你需要重新考虑你的可观测性策略

可观察性从监控演变而来，监控侧重于收集遥测信号，例如日志、指标和跟踪，以帮助您调试应用程序。由于这种关联，可观察性通常是事后才想到的，导致仪器过多或太少，无法关联信号，可见性脱节，以及多个工具通常无法凝聚在一起。这导致人们认为缺乏价值和成本，这似乎超过了可观察性的好处。从业务角度来看，这些问题意味着更长的平均识别时间 (MTTI)，更长的平均恢复时间 (MTTR)，以及用户体验、信任、品牌声誉和收入的下降。如今，可观察性不仅涉及调试和诊断应用程序的能力，还涉及验证应用程序是否完全按照预期运行的能力。

想要为用户提供最佳体验的企业与可观测性工具和功能的演变融为一体，需要重新考虑和重新确定可观测性的优先级。

可观测性工具和框架

在 2019 年上市之前 OpenTelemetry，为应用程序性能监控 (APM) 和数字体验监控 (DEM) 提供可观测性解决方案的专业工具使遥测信号之间的断开连接更加明显，并凸显了糟糕的用户体验。

- APM 实时跟踪和分析软件应用程序的行为。它可以衡量关键指标，例如响应时间、错误率和资源使用情况，同时监控跨应用程序组件的用户交易。APM 工具可帮助团队在性能问题、瓶颈和错误影响用户之前快速识别这些问题。他们的主要目标是保持最佳的应用程序性能和用户体验，同时缩短解决问题所需的时间。
- DEM 从用户的角度衡量和分析用户与数字服务的互动质量。它结合了真实用户监控 (RUM)、综合监控和端点监控，以提供完整的用户体验视图。DEM 可跨不同设备、浏览器和位置跟踪页面加载时间、应用程序响应速度和用户旅程完成情况等指标。这可以帮助组织了解用户如何体验其数字服务，识别影响用户满意度的性能问题，并优化数字接触点。这些见解使企业能够做出数据驱动的决策，以改善客户体验并保持竞争优势。

2019 [OpenTelemetry](#) 年的发布为生成、收集、管理和导出遥测数据提供了一个开源、统一的标准。该框架侧重于通过添加上下文、提供更好的信号相关性以及提供更好的衍生值来弥合遥测信号之间的差距。例如，使用带有附加上下文的结构化日志可以帮助您从摄取的日志中得出指标，并以不同的方式分析信息，从而更快地找到根本原因。以前 OpenTelemetry，信号是单独查看的。要添加功能，您必须修改代码以向现有指标添加新维度或创建新指标，等待代码完成开发生命周期，然后等待在合适的环境中观察到指标后才能进行扣除。此过程会延迟可见性，并影响您在必要时将数据与日志或跟踪关联的能力。

对 OpenTelemetry 这种支持的支持以及由此产生的工具改进可帮助您从可观测性平台中获得更高的价值，增强用户体验，并提高运营效率和团队士气。

如果你想改善和增强自己的可观察性姿势，你实际上是从哪里开始的？如何开始？我们推荐一种由三个步骤组成的方法，本指南将详细讨论这些步骤：

- [第 1 阶段：定义你的北极星](#)
- [第 2 阶段：实现可观察性](#)
- [第 3 阶段：检查、调整和迭代](#)

第 1 阶段：定义你的北极星

可观测性的成功实施不仅仅是运营和工具，而是要培养一种主人翁意识、持续改进和主动解决问题的文化。与任何成功的策略一样，您的可观察性策略需要全面考虑三个支柱：人员、流程和技术。

当你想建立或改善自己的可观察性态势时，我们建议你从定义重要内容开始，根据业务结果进行反思，随着业务、团队和产品的发展，不断审查、调整和调整战略。

在第一阶段，你定义并建立你的北极星，这是对你的组织来说什么样子的一个商定且众所周知的定义。我们建议您随着业务的发展、推出新产品、应用程序或服务或设计重大架构变更时，重新审视此阶段的部分或全部活动，以重新评估您的可观测性平台和组织需求。

在开发生命周期的早期集成可观察性（左移方法）

将可观察性作为工程、运营和产品团队每位成员的责任，并将其视为主要的功能要求，就像处理单元测试或安全性一样。这不会将责任从运营团队转移到开发团队，而是突显了跨多个团队所需的协作。对于团队来说，在开发生命周期的早期协作执行以下活动会很有帮助。您可能需要按票证、每项功能或每件产品执行这些操作。

- 确定利益相关者。谁是利益相关者？如果此功能或产品不能按预期运行，对他们有什么重要意义？在确定利益相关者时，应考虑功能、可用性、安全性、成本、销售和产品销售等方面。利益相关者可以包括您的团队、产品的客户、内部业务利益相关者、平台运营团队的成员和应用程序开发人员。视情况而定，您的安全和财务团队也可以成为利益相关者。
- 确定关键成果。确定关键结果及其对业务和每个利益相关者的影响。确定每项结果和利益相关者的成功与失败。结果通常被定义为服务级别目标 (SLO)，并且必须是可量化的。SLO 是衡量每个结果的指标。一个好的 SLO 有一个应该努力实现或保持的目标值作为目标。SLO 可以用来衡量用户满意度。服务级别指标 (SLI) 是用于确定您是否满足 SLO 的实际衡量标准：它是您根据目标跟踪的可量化数据点。例如，将 MTTR 降低 60%，将应用程序可用性保持在 99.99%，或将开发人员工作效率提高 30%。

让我们以将应用程序可用性保持在 99.99% 为例，并定义衡量和验证成功所需的 SLO、SLI 和指标。在此示例中，让我们考虑一个 RESTful 应用程序，并将应用程序可用性定义为所有传入请求的成功完成。这需要衡量向应用程序提出的请求总数以及每个请求的完成状态。当您将它们转换为 SLO 和 SLI 时，您需要一个用于捕获传入请求的指标和另一个用于捕获请求状态的指标。如果所有请求都成功完成，则认为该应用程序可用。如果一个或多个请求导致错误，则认为该应用程序不可用。因此，SLI 将是错误的请求完成次数之和除以 5 分钟间隔内传入请求的总和，实际上就是错误率。您

可以向此 SLI 添加一个目标，将其变成 SLO；例如：在连续 3 个 5 分钟间隔内，争取错误率低于 0.1%。

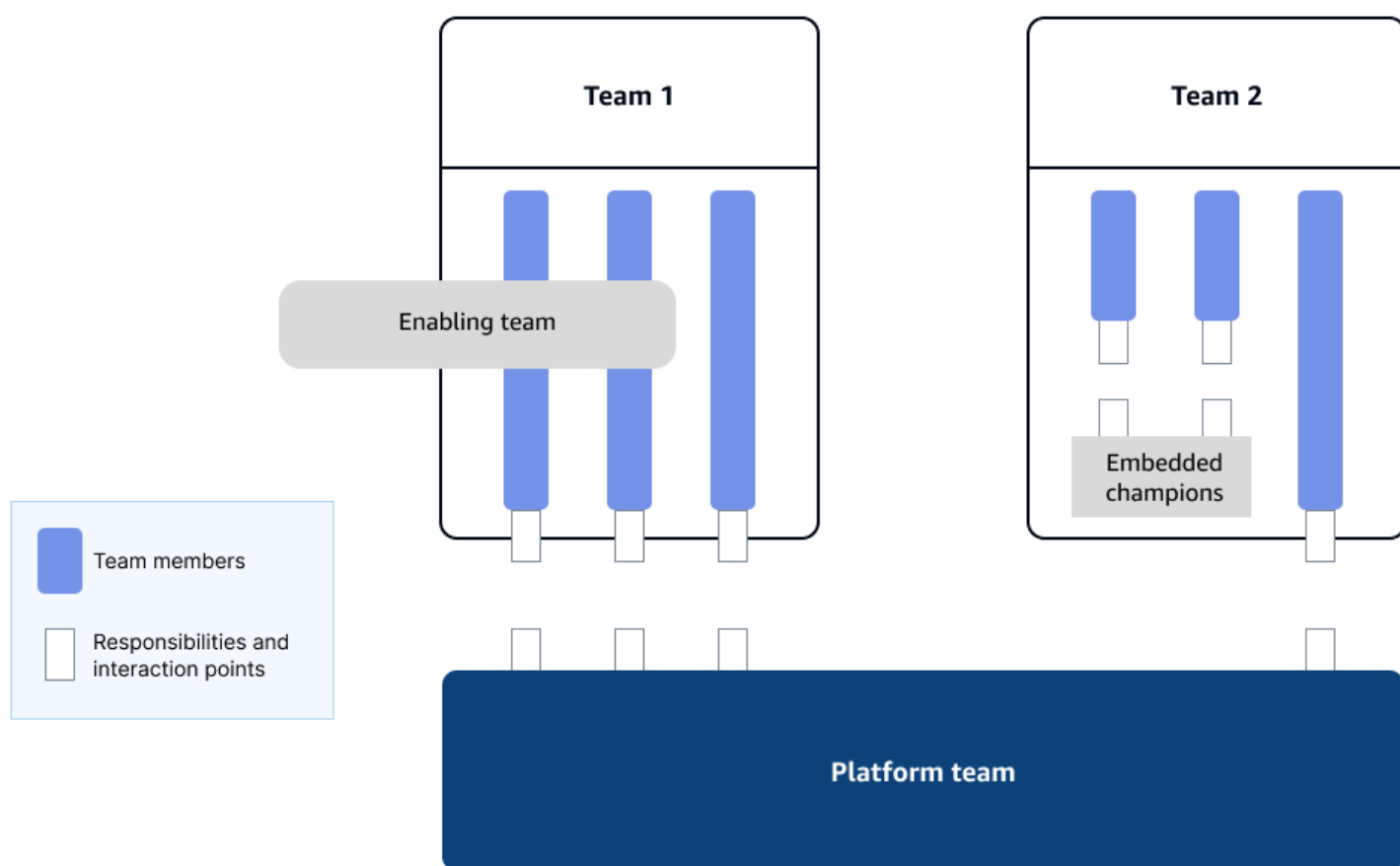
- 确定关键结果的优先顺序。根据你为每个结果设定的优先级，你可以选择首先关注影响最大的结果，而不是同时做所有事情。从小处着手，进行迭代，并以较小的增量改善您的可观察性姿势。可观察性是一个需要持续审查、审计、增强和改进的过程，以提高成熟度和收益。确定优先级还可以让你有机会定义实现已确定结果的渐进里程碑。
- 确定所需的仪器。架构或实现中有哪些组件和相关特征可以影响前面步骤中确定的重要结果？例如，当您在亚马逊弹性计算云 (Amazon EC2) 实例上运行应用程序时，内核数量和可用 RAM 会影响应用程序的响应能力和吞吐量。在此阶段，确定您使用的工具或库是否已经提供了其中一些工具也可能会有所帮助。进行一系列初步审查或在工单的[就绪 \(DoR\) 定义](#)中添加以下问题可以使该活动成为标准流程的一部分。
 - 如果此操作失败，您需要知道什么来解决失败？典型或有问题的操作会如何影响所涉及的组件？此操作应发送什么样的信号：日志、指标或跟踪？与其价值相比，这种仪器的成本是多少？在不违反 SLO 的情况下，哪种聚合是可以接受的？
 - 哪些组件和依赖关系可能导致此操作失败？您将如何确定哪个组件或依赖项导致了故障？这些组件和依赖关系有哪些不同的配置杠杆，它们对操作有何影响？
 - 确保精确测量 SLI 和 SLO 所需的指标粒度和采样率是多少？
- 定义成功标准。对于每个按优先顺序排列的结果，定义与达到或未达目标的相一致的影响阈值。成功标准为团队在响应警报时提供了额外的背景信息。它们还使您能够预测和权衡仪器的成本，以获得所需的可见性。

建立有效的组织和团队结构

根据架构的复杂性和业务规模，您可能需要组建一个专注于可观察性的专门团队。该团队将负责配置可观测性工具并为其他团队设置可观测性平台。如果您选择标准 OpenTelemetry 实施，我们还建议您成立一个专门的团队。在小型组织中，您可以将可观察性作为每个团队成员的额外责任，还可以任命可观察性倡导者，在团队中宣传和实施最佳实践。这些倡导者自愿抽出一天中的一部分时间来定义流程并为组织设定标准。他们要么作为一个自我规范的团队工作，要么可以由专门的可观测性专家领导。下图显示了您的投资如何决定您的组织方法。



冠军可以完全融入团队（如下图所示的第 2 组），也可以成为跨团队轮换以建立和推广最佳实践的支持团队的一员（插图中的第 1 组）。



跟踪成本分配

Organizations 应实施全面的成本跟踪和跨指标、日志和跟踪的可见性，同时建立针对特定团队的资源使用和成本问责制。要成功整合财务运营 (FinOps) 实践，就需要带有预算警报的自动监控系统，以及

系统的数据保留和收集优化。工程和财务团队应通过共享仪表板和定期审查来调整其目标。组织受益于实施清晰的退款模式和成本分配策略，以推动所有权和问责制。

定义标准

识别和定义应用程序所需的基本信号和遥测数据，包括警报和仪表板策略。为每份申请创建清单或正式审查流程。[AWS Observability Best Practices](#) 网站提供了警报和仪表板创建指南，例如设置适当的警报阈值、最大限度地减少警报疲劳、为每个角色创建具有足够上下文的仪表板等。有关互联和精心策划的可观测性体验，请参阅 Amazon CloudWatch [文档中的应用程序信号](#)。

建立上报流程

建立和执行上报机制、警报所有权和响应程序非常重要。我们建议您提倡一种不容忍升级的文化。

通过培训提高技能

确定提高现有和新团队成员技能的最佳方法，强调可观察性的重要性，培养持续改进的文化。根据组织的需求，您可以选择预先录制的按需培训或由可观察性倡导者或专家提供的课堂培训。您的 AWS 账户团队可以提供深入的动手培训课程，例如 [One Observability Workshop](#) 或 [GameDays](#)，或者指导和提高可观察性技能和最佳实践。此外，还应采用各种机制来强化最佳实践，并推广贵组织所定义的标准。

第 2 阶段：实现可观察性

在这个阶段，你将开始让你的团队逐步走向北极星的过程。

选择您的可观测性平台

第一步是确定合适的工具来摄取、可视化和分析信号，以及发送警报。选择工具时，请考虑其功能集、许可模式、价格、技能要求和维护。

功能集

以下是一些需要考虑的问题：

- 可配置性和自定义。该工具提供了哪些功能来简化调查体验并帮助降低 MTTR？该工具是否提供警报关联、指标数学、灵活处理丢失的遥测数据或异常检测？
- 粒度。遥测信号摄取和可视化的支持粒度是多少？
- 人物角色。该工具是否支持您想要为开发人员、平台工程师和其他角色提供的体验？它对技术角色和商务角色都有用吗？
- 小工具。仪表板支持哪些类型的微件？该工具是否允许创建自定义控件？
- 预先构建的解决方案。该工具提供了哪些类型的预建可观测性解决方案来缩短价值实现时间？
- 自动化和生成式人工智能。该工具提供了哪些功能可以帮助您和您的团队自动化或减少工作量？例如，自动异常检测、预测分析和其他生成式 AI 功能可以帮助减轻假设和未知未知数（即你既不知道也不完全理解的事情）带来的压力。该工具是否支持使用生成 AI/ML 模型来增强对大规模数据的分析？它是否为你提供了自动化和实施的选项 AIOps？
- 安全。该工具支持哪些类型的身份验证和授权集成？用户体验和登录体验是否满足贵组织的需求？
- OpenTelemetry 支持。该工具和代理是否支持 OpenTelemetry？大多数可观测性平台都支持摄取 OpenTelemetry 兼容信号，但并非所有代理都提供将这些信号转发到可观测性平台的配置选项。
- 集成。该工具提供哪些集成？考虑是否需要向 Slack、页面团队成员发送提醒，还是需要自动解决问题。
- 可扩展性。该工具的可扩展性和性能如何？可观察性解决方案必须随着需求和使用量的增加而扩展，因此即使您的应用程序不可用，它也可以提供诊断。
- Support。提供什么支持模式？即使您的应用程序出现故障，您的可观察性工具也必须可用，这样才能满足 MTTR 和应用程序可用性目标或服务级别协议（SLA）。SLAs 开源解决方案可能提供的正式支持有限。

许可和部署模式

考虑解决方案的许可模式（开源或商用）和部署模式（自托管或基于云）。开源选项的前期成本通常较低，但在提供价值之前，可能需要更多时间进行部署、设置和配置、维护和团队技能提升。如果您正在考虑开源选项，则可能需要一支由可观测性专家组成的专门团队。商业软件通常以更高的前期成本提供更快的价值实现时间，随着采用率、复杂性和成熟度的提高，对专门的可观察性团队的需求会随着时间的推移而变化。与基于云的解决方案相比，自托管解决方案需要更多时间进行部署、设置和配置、维护和运营开销。

定价维度

随着您的应用程序获得新用户、更大的架构占用空间或新功能和应用程序，该工具的定价模式将如何影响您的总拥有成本 (TCO)？例如，一些典型的许可模式是永久许可模式或基于订阅、指定用户数量、消耗量或数量的许可模式。考虑一下您的应用程序和可观测性工具的使用量将如何扩展，以及许可模式如何影响工具的成本。

团队技能

根据团队当前的技能组合和成熟度，你需要确定需要多少技能提升。考虑供应商提供什么样的支持来培训您的团队。还要考虑您的组织结构是否支持您选择的工具的配置和管理。例如，如果你愿意 OpenTelemetry，你应该考虑成立一个专门研究可观察性的独立团队。

操作和维护

评估以下问题：

- 可观测性代理或收集器提供哪些部署选项？这些选项是否符合您的架构要求？例如，如果您为可观测性工具使用容器化部署，它是否支持守护程序集或边车？运营团队还需要采取或使用哪些其他步骤或工具来确保与安全和所有其他流程保持一致？
- 维护解决方案需要付出多少努力？更新代理或收集器的过程有多简单或自动化？尽管代理或收集器的管理保持不变，但与自行部署和托管的应用程序相比，完全托管和基于云的可观测性接口的操作开销通常较低。考虑您的团队结构，并考虑维持所选选项的人力成本。

对您的应用程序进行检测

上一节中问题的答案为您提供了检测应用程序所需的信息，也就是说，添加代码以捕获应用程序中的遥测信号，以及测量、观察和验证行为。你可以使用 SDKs 诸如应用程序语言的 OpenTelemetry SDK 来自动检测你的应用程序。您可能仍需要添加手动检测代码以弥补任何空白并确保 end-to-end 可见性。

请谨慎对待您添加的遥测数据，并确保可以将其连接到一个或多个遥测系统 SLOs，SLIs 并且是在上一阶段建立的。

收集遥测数据

[配置遥测收集器或代理，使其接收相关的遥测信号，使其与您在第 1 阶段优先考虑的结果保持一致。](#)

实现可观测性组件

当遥测数据流入可观测性平台时，创建仪表板、警报、行动手册和运行手册。

- 仪表板：创建包含相关信息的仪表板，包括与您的优先结果相关的当前和历史趋势的可视化表示。将这些仪表板提供给您在[第 1 阶段](#)中定义的利益相关者。有关更多信息，请参阅 Amazon Builders Library 网站上的[构建控制面板以实现运营可见性](#)。
- 警报：定义警报，以便在结果面临风险或被违反时通知您的团队。考虑[为安全和性能问题添加警报](#)。通过采用以下方法来优化[警报以减少警报疲劳](#)和成本：
 - 使用异常检测可以避免设置需要频繁调整的硬阈值，并减少未知未知数的出现。
 - 使用可同时查看多个相关指标的智能警报组合，而不是为每个指标设置单独的警报。例如，与其分别为 CPU、内存和响应时间设置警报，不如创建一个有意义的警报，只有当这些指标共同表明存在实际问题时才会触发。这种方法可以显著减少警报噪音，帮助您的团队专注于影响服务的实际问题，而不必对孤立的指标峰值做出响应。
 - 仅在用户体验或结果面临风险时才生成警报。例如，当您的应用程序没有活跃用户时，请避免收到有关自动升级导致的 CPU 峰值的警报。
- 剧本：剧本为响应事件或警报的人提供了心理模型和背景信息，并帮助他们更快地确定根本原因。考虑为高度耦合、复杂的应用程序以及缺少工具但直接影响您在第 1 [阶段](#)确定和优先考虑的结果的应用程序创建行动手册。
- 运行手册：使用运行手册定义解决事件或警报所需的步骤。

验证可观测性系统

在整个软件开发生命周期 (SDLC) 中，验证仪表板在系统测试期间是否提供了预期的行为和更新。实施[混沌工程](#)并验证行动手册和运行手册中记录的步骤，以确保它们准确且符合其目的。您还应验证警报所有权和升级路径。

第 3 阶段：检查、调整和迭代

在实施可观测性系统后，我们建议您不断审查、评估、学习、调整和改进您的实现。您可以将可[AWS 观测性成熟度模型](#)用作评估实施成熟度的工具，并确定需要改进的领域并确定其优先顺序。

实施定期审查

可观察性是一个迭代过程。它需要定期对现有组件进行审计和评估，并进行更改和增强，以推动持续改进。我们建议您定期进行审查，以重新评估 SLO、警报阈值、仪表板、指标粒度、保留策略、抽样策略等，以确保这些策略能够为您的团队和业务带来价值。通过将可观测性成本与特定团队和服务联系起来，您可以根据数据做出覆盖范围和资源分配的决策。

在 Amazon，我们每周进行[运营准备情况评估 \(ORR\)](#)，根据最佳实践审计团队的流程和可观察性态势。这是一项非屏蔽活动，与 Amazon 的服务数量和发布频率保持一致。

根据组织的规模，您还可以设置一个“一切照旧”(BAU) 名单，其中每个团队中都有一名成员负责报告异常和趋势、发现未知数、删除不需要的工具和警报、改进仪表板，并确保可观察性解决方案继续为团队起作用并与团队的目标和成功指标保持一致。这也可能是重新评估警报策略的机会，以提高响应能力、主动性并更接近用户。[这些评论的目标是创建一个良性循环，如下图所示，并提高可观察性姿势成熟度的成熟度，如可观测性成熟度模型中所AWS 述。](#)



确定访问频率最高的行动手册，并考虑改进您的应用程序或添加更多工具。确定最常执行的运行手册，并考虑自动执行这些运行手册。

还与可观测性小组和专家分享了从这些评论中吸取的教训，以突出中央计划和可观测性平台的改进。例如，根据部署触发的事件的频率，您可能会决定优先考虑部署管道的改进，而不是其他组件。如果由于监控差距而导致MTTR更高，则可以优先改进可观测性平台及其配置。

庆祝胜利

分享使用可观测性工具的团队的成功案例。例如，重点介绍一个使用可观测性指标来实现更高效、更低延迟或成本的替代解决方案的团队所取得的成功。传达这一成功凸显了可观察性的重要性，并激励其他团队改善其可观察性态势并努力取得类似的成功。

从事件中吸取教训

进行无可指责的事后练习，类似于 Amazon [的错误更正 \(COE\)](#) 流程，以确定需要改进的领域并防止将来出现问题。与获胜一样，可以与其他团队广泛分享从本练习中吸取的教训，以增强可观察性和最佳实践的价值。

后续步骤和资源

通过从用户和业务成果入手，关注影响力，您可以从可观测性解决方案中获得更好的价值，同时控制成本并避免工具蔓延。请记住，可观测性是一个迭代过程，而不是目的地。设置您的团队和流程，以不断迭代和改善您的可观察性态势，整合最佳实践，并随着架构、业务需求和团队的发展缩小差距，并以明确定义的目标为指导。通过为失败做好计划、融入持续改进的文化以及采用本指南中讨论的最佳实践，你应该能够将沃纳·沃格尔斯博士的指导付诸实践：“一切都会失败... 所以要为失败做好计划，任何事情都不会失败。”

有关帮助您定义可观测性策略的实践研讨会，请联系您的 AWS 代表举办[可观测性策略](#)研讨会。

资源

有关在上实施有效的可观测性策略的更多信息和指导 AWS，请参阅以下资源：

- [监控和可观测性](#) ([可AWS 观测性服务](#))
- [AWS 可观测性最佳实践](#)
- [AWS 发行版适用于 OpenTelemetry](#)
- [卓越运营支柱](#) (WellArchitect AWS ed Framework)
- [构建控制面板以提高运营可见性](#) (Amazon Builders Library)
- [可观察性](#) (在DevOps指导中，Well-Ar AWS chitected Framework)

文档历史记录

下表介绍了本指南的一些重要更改。如果您希望收到有关未来更新的通知，可以订阅 [RSS 源](#)。

变更	说明	日期
初次发布	—	2025 年 7 月 16 日

本文属于机器翻译版本。若本译文内容与英语原文存在差异，则一律以英文原文为准。