



Search. Observe. Protect.

APM 介绍

实施内容、原因和方法。

elastic.co/cn

目录

我们为何需要应用程序性能监测 (APM)?	3
应用程序为业务提供支持	3
性能不佳相当于收入损失	4
现代云原生应用非常复杂	4
究竟什么是 APM?	5
端到端可见性	5
呈现延迟和错误	6
生产环境之外的 APM	6
APM 的工作原理	7
APM 代理	7
检测和配置	8
分析	8
关键词	9
分布式跟踪	9
跨度	10
事务	10
轨迹	11
真实用户监测	11
选择 APM 工具	12
技术能力	12
易用性	13
部署选项	13
支持开放标准/开放数据	13
架构和可扩展性	14
安全性	14
APM 之外的能力	15
定价	15
关于 Elastic	16

我们为何需要应用程序性能监测 (APM)?

简要介绍: 在本指南中, 我们将讨论[应用程序性能监测 \(APM\)](#) 在[可观测性](#)方面所起到的支柱作用。APM 与日志和指标一起, 在构建可观测的系统方面发挥着关键作用。

应用程序为业务提供支持

应用程序是现代组织的公共形象。应用程序是我们使用产品和服务的方式, 无论是电子商务店面、共享单车应用, 还是我们每天使用的各种协作和生产力工具, 都不例外。当我们在网上购物车中加入一副耳机, 或为晚宴创建最契合的音乐列表时, 我们就是在与应用程序交互, 在这个过程中也形成了我们对应用程序的看法。品牌感知、忠诚度和评级都取决于这些体验。



性能不佳相当于收入损失

想一想您在工作中或在家中个人设备上使用的应用程序。您想到了什么？易用性如何？响应速度有多快？多久会遇到一次错误？以上问题的答案是否会影响您对这些产品背后公司的看法？

当应用程序不能按预期运行时，会对业务产生直接影响。这关系到收入和客户信心。

如果网上购物车体验较差，用户就会前往其他商店，导致销售损失。如果响应速度一直很慢，或错误频繁发生，那么即使是忠诚的用户，也可能决定完全不再使用相应产品或服务。每个组织都清楚服务中断对利润来说意味着什么。平均检测时间 (MTTD) 和平均响应时间 (MTTR) 都与收入挂钩。



现代云原生应用非常复杂

应用程序开发和交付方式的演变，以及由此带来的复杂性，使得全面监测性能的需求更加迫切。

微服务架构的采用有效地改变了应用程序的构建和维护方式，提供了前所未有的连续交付和扩展能力。但是这些架构的分布式和多语言（使用不同的语言/框架编写）特性也增加了复杂性。连续交付和自动化的原则支持频繁推送代码，但也使得跟踪性能影响的能力变得更为关键；只有具备这种能力，方可在部署未按计划进行时快速修补问题或还原更改。

遵循云原生开发原则，团队可以不断向用户提供出色的数字体验，对反馈做出快速反应，并根据需要进行调整。用户看不到的是，所有这些设备正在后台不停运行，为他们的每一个请求提供支持。那么团队该如何监测这些应用程序呢？

作为人类，我们无法以有意义的方式处理如此海量的数据。我们需要一种有条理的视图，让人能够清晰地洞察到有用的信息。让我们欢迎 APM。

究竟什么是 APM?

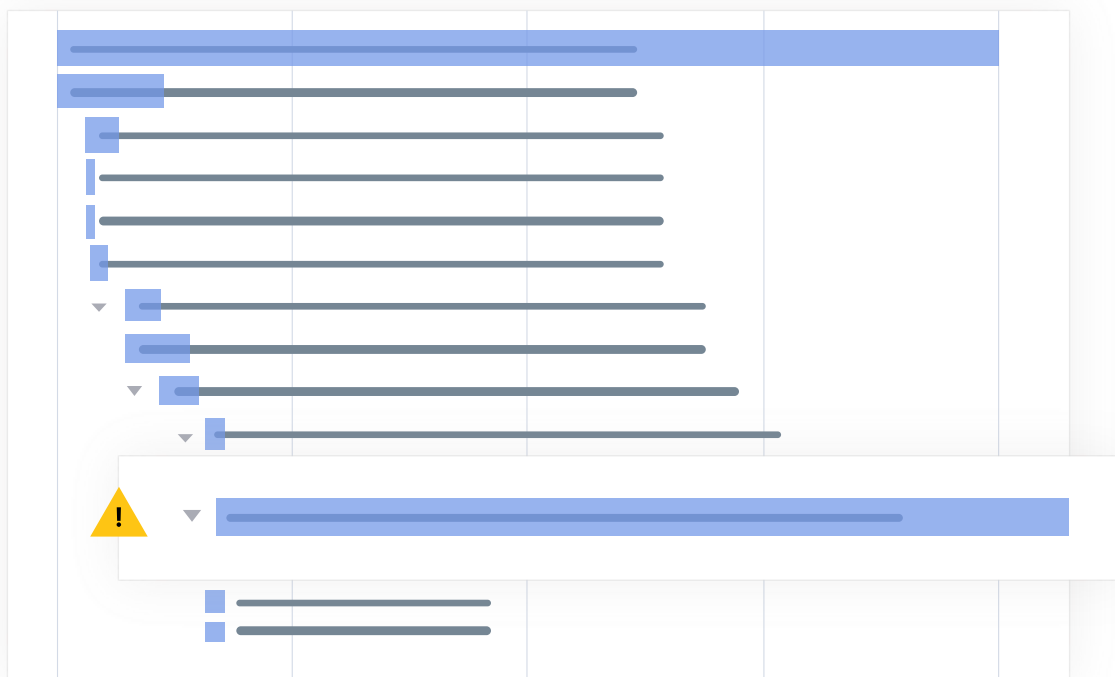
APM 是通过应用程序 (以及其所依赖的服务) 收集、监测和分析事务的端到端性能和行为的解决方案。从用户开始与应用程序交互的一刻起, 直到用户达到预期的结果为止, APM 都会跟踪用户体验。

由于 APM 与用户体验直接挂钩, 因此有助于将 IT 与业务目标联系起来。尽管这种关系可能不完全是 1:1, 但确实具有正相关性。通过不断提供有关应用程序运行情况的洞察, APM 使您能够主动监测并解决问题, 而不是在通过工单系统甚至社交媒体收到客户投诉后才对问题作出反应。

端到端可见性

只要对应用程序进行了插桩, 您就可以通过得到的数据准确地了解应用程序内部发生了什么。APM 可以全程跟踪事务, 因此无论您的架构有多复杂, 每个请求和响应都会得到记录和测量。

将所有这些跟踪结果关联在一起后, APM 就可以为您提供应用性能的完整图景, 从服务的鸟瞰图 (以及其如何交互) 一直到代码级别的洞察。



您可能会问，难道日志不能告诉我们应用程序内部发生了什么吗？日志确实能够通过错误类型或发生的其他事件等上下文信息为我们提供有价值的信息。但日志无法展现端到端的用户体验旅程，这正是 APM 跟踪（和分布式跟踪）所能填补的细节。将跟踪结果与日志和指标相匹配，将让您更统一地了解您的生态系统。

呈现延迟和错误

APM 可以监测与用户体验密切相关的两个属性：事务持续时间和错误。HTTP 请求通常需要多长时间获得响应？在一个小时内，我们通常会遇到多少个 500 错误？

当响应请求花费的时间比平时更长时，就会出现延迟问题。例如，当用户单击查看产品时，需要好几秒钟时间才能加载响应。

错误自然意味着遇到了意外的结果。当请求未能成功完成时，就会发生这种情况。

一切调查的最终目标都是解决问题。为了有效地解决问题，团队需要执行根因分析，以具体确定哪个组件导致了问题。我们能否快速确定哪个服务成为了性能的瓶颈？我们能否准确地找出导致此问题的具体函数调用或方法？借助 APM，两个问题都能得到肯定的答案。

生产环境之外的 APM

我们经常在生产环境中的应用程序用户体验背景下讨论 APM，但实际上 APM 在更早阶段就已开始发挥作用，并影响到多个团队和环境。例如，APM 可以为开发人员提供创建代码所需的反馈，从而实现更快速的开发-测试-部署循环。

企业跨阶段使用 APM 可以帮助团队更高效地构建、QA、部署和监测应用。如果组织跨多个团队使用 APM，就可以改进流程并制定更高效的工作流，从而留出更多的时间进行创新（并减少“救火”时间）。

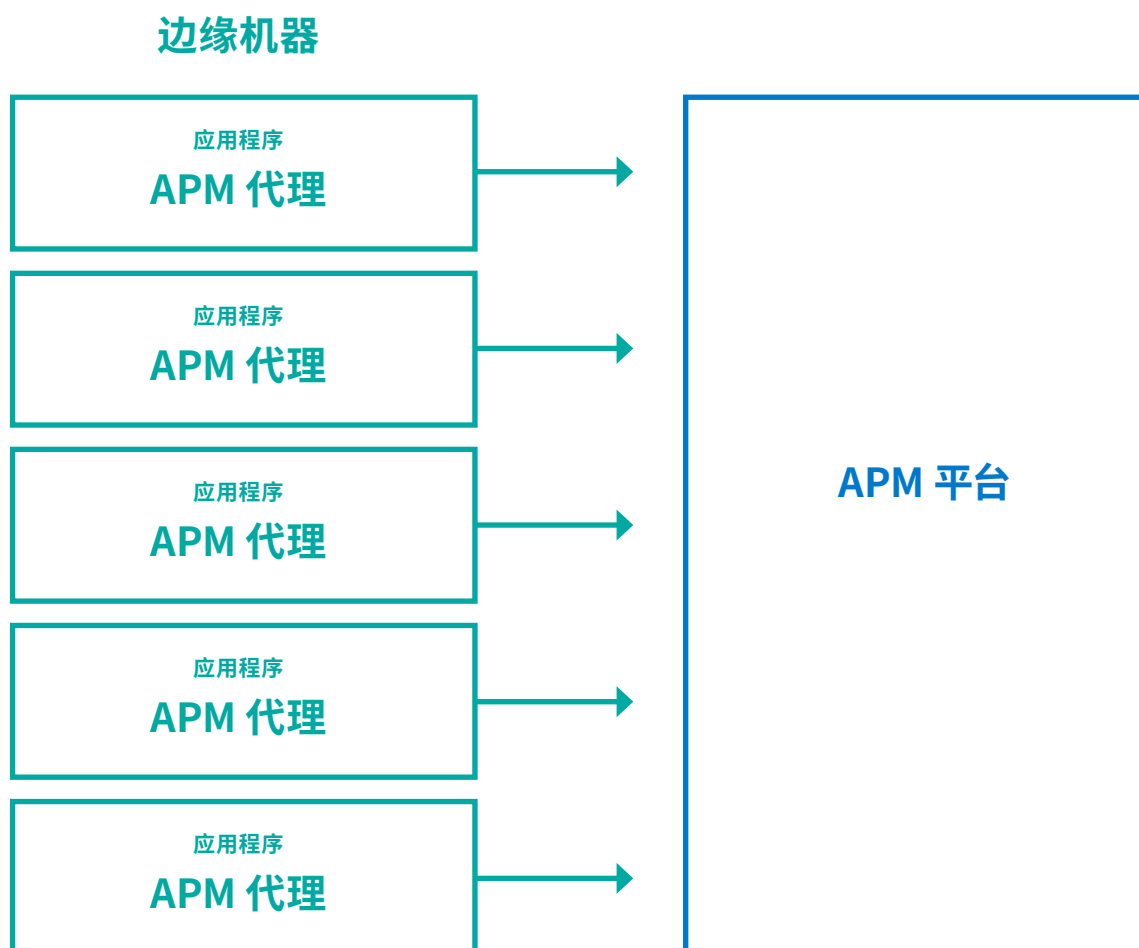
了解了好处之后，接下来我们深入了解一下 APM 的工作原理。

APM 的工作原理

APM 的工作原理是检测您希望监测的系统（主机、容器、应用程序等）的各个方面，然后将收集到的性能数据发送到您选择的位置进行分析。

APM 代理

APM 代理是一种库、插件或扩展，可以监测我们上文描述的性能指标。根据您需要监测的对象（以及所用的编程语言），您可能需要一个或多个代理。一旦确定了要监测的所有对象，就可以将 APM 代理部署到所有这些部件中。虽然代理因供应商而异，但大多数代理都会检测您的代码、收集性能数据，然后将数据发送到服务器或收集器中。



检测和配置

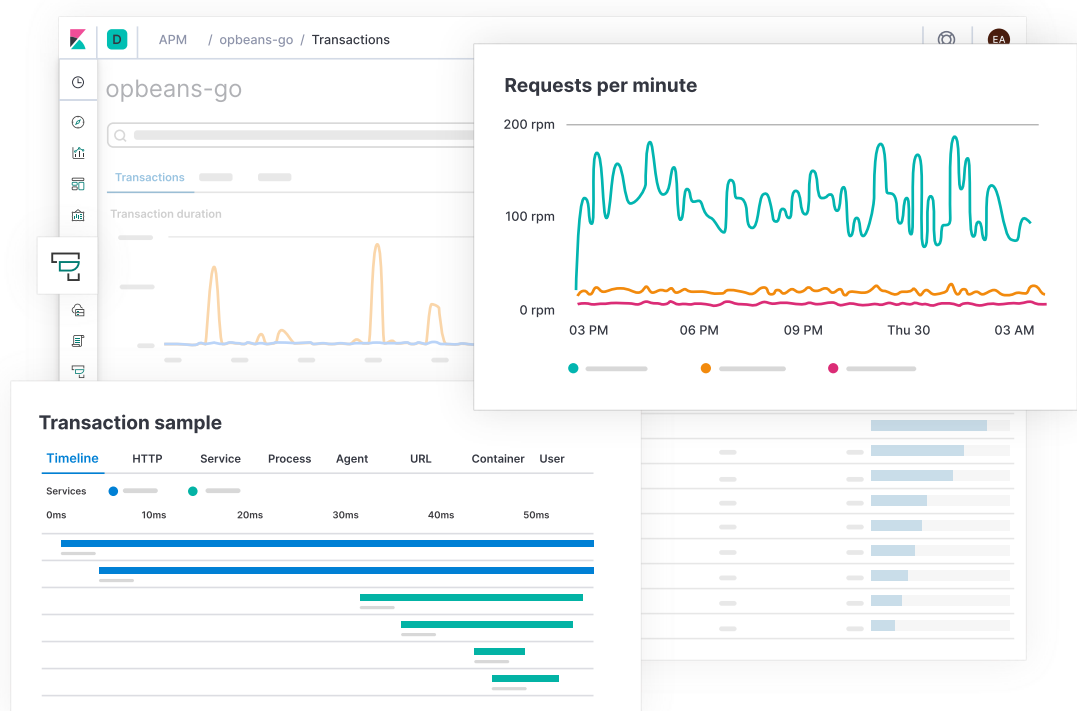
检测是指扩展应用程序代码以报告跟踪数据的过程。安装完必要的代理后，您可以检测要分析的应用程序的特定部分，以收集性能数据并将其发送到您选择的位置（通常是 APM 工具确定的终端）。

您可能想要配置环境名称、采样率、检测和指标等变量，以帮助您的团队轻松识别和分析流入 APM 工具的数据。通常情况下，这些变量可以使用工具的 UI 和 API 配置，也可以直接在环境变量中配置。

分析

性能数据发送到您选择的位置后，您就可以开始分析数据。大多数工具都提供了 UI 帮助您识别可能影响到用户的错误、延迟问题和其他异常。

这可以成为调查所报告问题很好的切入点，更值得一提的是，可以在用户受到问题影响之前捕捉到问题。准确识别出哪些服务（深入到代码级别）遇到了问题可以加快根因分析和 MTTR。



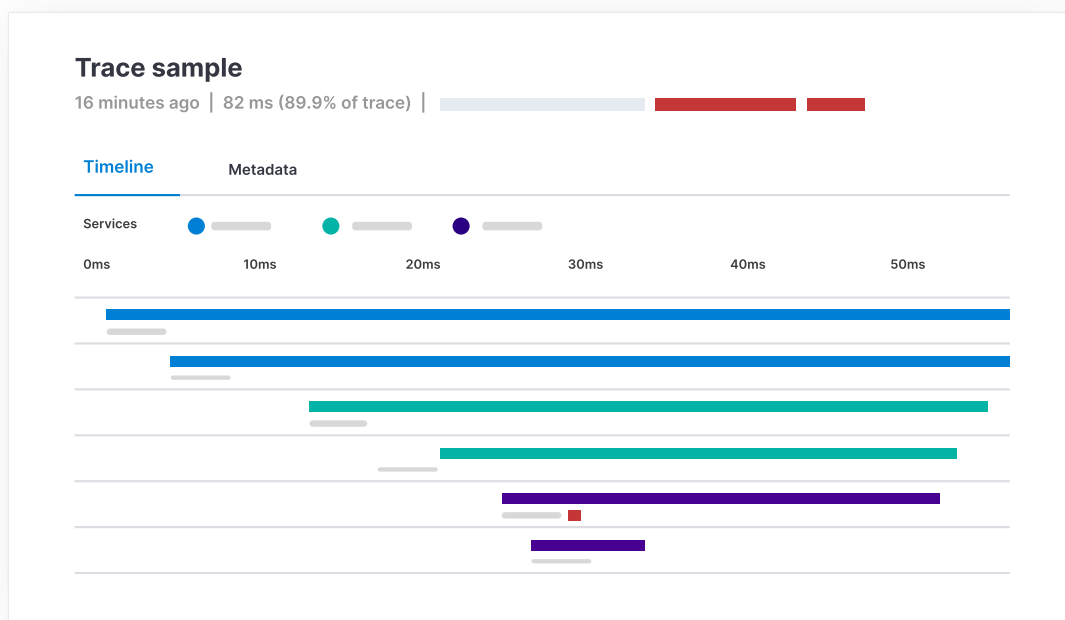
关键词

每个 APM 工具对这些字词的定义可能略有不同，但通常情况下，这些字词承载了提供相同的信息，以便为用户提供应用程序性能的洞察。

分布式跟踪

分布式跟踪通过跟踪从最初的 Web 请求到前端服务，再到对后端服务查询的所有请求，使您能够在一个视图中分析整个微服务架构的性能。

分布式跟踪能够按服务，再按该服务中的每个请求显示完整事件，从而很容易发现瓶颈。在代码级别以可操作的方式呈现错误和其他问题可以加快调查和 MTTR。



跨度

每个单元或工作流的每个部分都称为一个跨度。跨度是您在 APM 分析工具的瀑布视图中最常看到的元素，通常用水平条表示。这些部分是分布式跟踪的核心。跨度衡量了从活动开始到结束的全过程，并包含有关特定代码路径的执行信息。

跨度的常用属性包括：

- 开始时间
- 完成时间
- 名称
- 类型

事务

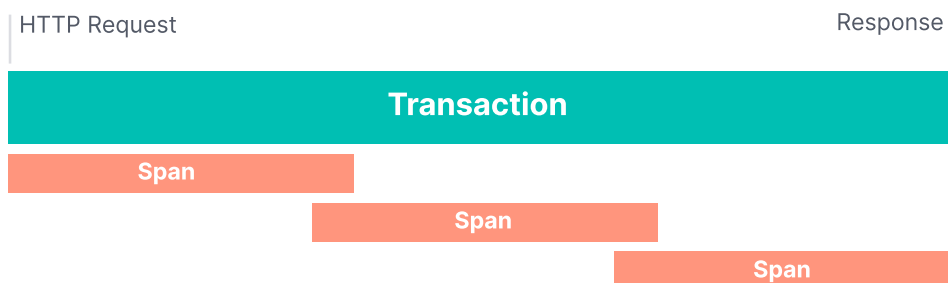
事务是用于描述事件的一种跨度类型，也可以包括多个跨度。

事务的一些示例包括：

- 对服务器的请求
- 批量作业
- 后台作业

事务还具有与之关联的其他属性，如有关所记录事件环境的数据：

- 服务：环境、框架、语言等
- 主机：架构、主机名、IP 等
- 进程：args、PID、PPID 等
- URL：完整、域、端口、查询等
- 用户：（如果提供）电子邮件、ID、用户名等

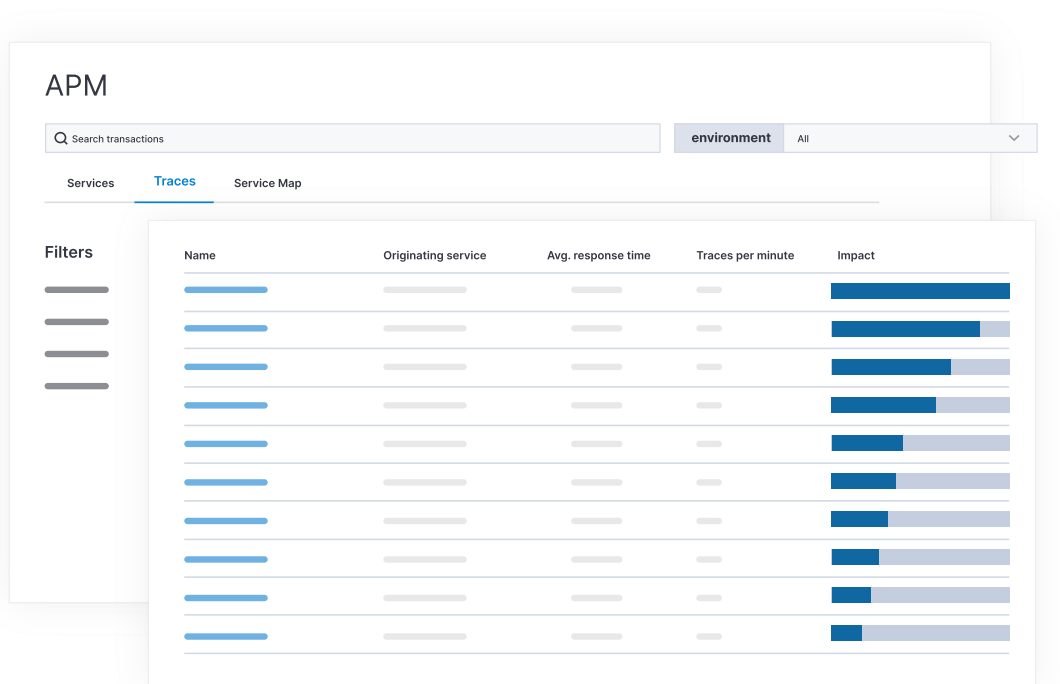


轨迹

轨迹衡量了通过应用程序发出的请求的状态和持续时间。跨度和事务共同构成了轨迹，连接了从用户请求一直到响应的每一个环节。

您可以看到完整操作中的各个轨迹，如处理付款、处理已完成订单和更新发货状态。

与来自应用程序（以及基础架构的其他方面）的日志和指标相匹配后，轨迹就可以提供对整个生态系统的完整可见性。



真实用户监测

真实用户监测 (RUM) 可以捕获生产环境中用户与客户端的直接交互。例如，RUM 可以向您准确显示用户如何与您的网站进行交互，以及响应时间和发生的错误。这些操作均由用户浏览器上的代理执行，能够收集性能数据并将其发送到 APM 工具进行分析。

选择 APM 工具

任何工具的选择过程都需要从明确定义需求开始。在您列出需求时, 请充分考虑您的发展和未来的需求。规划好概念验证 (POC), 以确保工具真正符合您的需求。供应商演示和营销的功能在您的环境中是否能按预期工作? POC 有助于避免过程后期的意外和隐藏成本。

技术能力

根据您的具体技术需求, 创建所需功能的清单。对您的需求进行全面细化。尽管不需要制定详尽的清单, 但以下这些方面应充分考虑:



语言和框架支持

- APM 工具是否支持在您的环境中广泛部署的语言和框架?



可见性的广度和深度

- 能否提供代码级别的性能可见性?
- 能否为您提供端到端的技术堆栈可见性?



搜索、分析和可视化探索功能

- 是否包括精心设计的 UI, 可以支持主动和被动调查 workflow?
- 能否提供瀑布图和依赖关系图等可视化工具来查看分布式应用的性能?
- 能否提供灵活 (快速) 的查询语言来支持临时调查?
- 能否在供应商提供的标准仪表板之外提供灵活的可视化框架?



融入更广泛的生态系统

- 能否与您环境中的其他标准工具无缝集成? 例如, 用于事件响应的 PagerDuty、用于案例管理的 ServiceNow 或用于通知的 Slack? 或者您的 CI/CD 框架能否帮助您自动跟踪代码推送和新部署对性能的影响?

易用性

在处理技术功能要求时，请评估这些功能的易用性和可访问性。考虑一下需要与该工具交互的各种角色，并在回答易用性问题时考虑他们的体验、期望和工作流。将您的评估维度从入门体验扩展到持续的运营开销。

- 您能以多快的速度检测应用程序并从零开始洞察？
- UI 是否直观且易于使用？
- 部署解决方案需要付出多少工作量？
- 升级平台有多容易？升级代理呢？
- 代理能否从一个地方集中管理？
- 扩大（或缩小）需要哪些工作？

部署选项

请务必确保所选的 APM 工具支持您的软件使用偏好。

- 您是否更喜欢使用 SaaS 选项来降低运营和管理成本？
- 您是否希望有一个选项可以帮助您减少所选云服务提供商（AWS、Azure、Google Cloud 等）的年度承诺支出？
- 您是否需要自管型产品，因为成本或合规性方面的原因而无法使用云服务？
- 您是否具有多云或混合策略，并希望 APM 解决方案的运行更靠近您的工作负载，以减少数据传输成本或延迟？

支持开放标准/开放数据

可观测性领域已开始发展并采用开放标准，如 OpenTelemetry（通过合并另外两个开放标准 OpenTracing 和 OpenCensus 形成），作为标准的、供应商中立的检测框架。这些计划的目标在于帮助开发人员轻松地将其应用程序移植到不同的 APM 解决方案，而无需重新检测。

如果您已有使用开放标准（如 Jaeger）检测的应用，那么切换到支持该标准的工具可以加快和简化迁移。开放标准也有助于让您的投资适应未来需求。

架构和可扩展性

评估您选择的 APM 工具是否具有强大的基础和架构来应对当前和未来的规模。能否快速有效地调查和解决应用程序性能问题取决于 APM 工具的性能。请务必在评估过程中对 APM 工具进行负载测试（如果不可行，请查阅基准数据），以确保能够处理环境中预期的采集和查询量，而不会被负载压垮。

- 是否以简单架构为基础构建？还是潜藏的基础架构杂乱无章，最终会崩溃？
- 是否支持高可用性？
- 能否轻松扩展以处理监测数据量的激增？
- 可以处理的量（应用、指标、查询等）是否有任何限制？

安全性

安全性审查过程应该是工具评估过程的核心部分。请务必考虑以下两个角度：

1. APM 供应商对工具构建和交付方式安全性的承诺
 - 应用程序中部署的 APM 代理是否需要过多的权限？
 - 内部组件之间的通信是否得到适当加密和保护？
 - 工具是否具有必要的认证（尤其是在 SaaS 中）？
 - 工具是否使用第三方扩展？这些第三方扩展是否安全？
2. 保护和控制对 APM 工具访问权限的能力
 - 工具能否与您的企业身份访问管理系统集成？
 - 工具是否支持基于角色的访问控制以及精细的权限模型？

APM 之外的能力

在评估工具时，要超越 APM 的传统界限思考。对工具整合和统一可见性的需求促使许多组织将 APM 与日志和指标（即另两个“可观测性支柱”）统一，以帮助简化调查并降低 MTTR。通过在应用程序跟踪数据的上下文中呈现日志和指标，评估 APM 工具是否支持这种统一的设想。如何采集传统 IT 环境之外的数据，如社会情绪或客户支持请求？

定价

最后，请务必仔细权衡定价选项，以确保您选择的工具不会迫使您在可见性或监测目标上妥协。对于许多其他标准，您需要在流程中同时考虑当前和未来的使用情况（以及架构）。APM 解决方案的定价方式存在很大差异（例如按代理数量、主机数量、硬件资源等进行定价），当您超过某些阈值（例如，容器或指标的数量）时，一些供应商也会增加额外的成本。以下是有关定价模型要思考的几个问题：

- 定价模型是否符合您的业务需求和架构选择？
- 随着计划中的增长和架构的演进（例如，从整体服务到微服务），成本将如何而增长？
- 是否有免费套餐？免费套餐包括什么？是否有任何使用限制？
- 包括什么级别的支持？



关于 Elastic

作为最常用的开源日志平台，Elastic 为 APM 带来了速度、规模和相关性。

在调查影响用户的问题时，每一秒都非常宝贵。借助 Elastic APM，您的性能数据将保存为 Elasticsearch 中的索引，使团队能够实时搜索和查找瓶颈。Elastic APM 还具有机器学习支持下的服务地图、自定义警报选项等功能，因此您可以为用户创造更好的数字体验。访问 elastic.co/cn/apm 了解详情。

APM 是拼图的一部分。在单个平台上统一日志、指标和 APM 跟踪，以实现整个生态系统真正的可观测性。访问 elastic.co/cn/observability 了解详情。

即日起开始免费试用