



为亚马逊 Neptune AWS Analytics 应用 Well-Architected 框架

AWS 规范性指导



AWS 规范性指导: 为亚马逊 Neptune AWS Analytics 应用 Well-Architected 框架

Copyright © 2025 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
卓越运营支柱	3
使用 IaC 方法自动部署	3
运营设计	4
进行频繁的、微小的、可逆的更改	4
实现可观察性以获得可操作的见解	5
从所有操作失败中吸取教训	5
使用日志记录功能监控未经授权或异常的活动	5
安全支柱	7
实现数据安全	7
保护您的网络	8
实现身份验证和授权	8
可靠性支柱	9
了解 Neptune 服务配额	9
了解 Neptune 部署模式	10
管理和扩展 Neptune 集群	10
管理备份和故障转移事件	11
性能效率支柱	12
了解用于分析的图形建模	12
优化查询	14
优化写入	15
大小合适的图表	15
成本优化支柱	17
了解使用模式和所需的服务	17
选择资源时要注意成本	18
可持续发展支柱	20
考虑你的 AWS 区域 选择	20
优化消费	20
优化软件开发和架构模式	21
资源	22
参考信息	22
博客文章和视频	22

训练	22
文档历史记录	23
术语表	24
#	24
A	24
B	27
C	28
D	31
E	34
F	36
G	37
H	38
我	39
L	41
M	42
O	46
P	48
Q	50
R	51
S	53
T	56
U	57
V	58
W	58
Z	59
.....	lx

为亚马逊 Neptune AWS Analytics 应用 Well-Architected 框架

Michael Havey , 亚马逊 Web Services (AWS)

2024 年 12 月 ([文件历史记录](#))

[您可以使用 Amazon Neptune 在 Amazon Web Services \(AWS\) 上构建基于图形的解决方案。](#) Neptune 包括 [Neptune Analytics](#)，这是一款内存优化的图形分析引擎，可以快速分析大量图表数据以获得见解和发现趋势。它可以对现有 [Neptune 数据库](#) 集群中的数据进行分析，也可以加载和分析来自外部数据集的数据。本指南为在规划 Neptune Analytics [AWS 部署时应用 Well-Architected Framework](#) 原则提供了规范性指导。[为 AWS 亚马逊 Neptune 应用 Well-Architected 框架](#) 涵盖了海王星数据库的相同主题。

Well-Architected Framework 可帮助您为各种应用程序和工作负载构建安全、高性能、弹性和高效的基础架构。它还为您评估架构和实现可扩展设计提供了一种一致的方法。

Well-Architected 框架围绕以下六大支柱构建：

- 卓越运营
- 安全性
- 可靠性
- 性能效率
- 成本优化
- 可持续性

本指南提供了来自 Well-Architected Framework 设计支柱和最佳实践的信息，以及在部署 Neptune Analytics 时需要记住的注意事项。AWS

目标受众

本指南适用于设计和实施使用图表的解决方案的数据工程师、解决方案架构师和数据分析师 AWS。

目标

本指南可以帮助您和您的组织执行以下操作：

- 从支持的部署选项中进行选择。

- 遵循可帮助您提高 AWS 灵活性和安全性的 Well-Architected 设计模式。
- 设计您的查询以获得最佳性能并节省成本。
- 了解如何在生产环境中管理 Neptune Analytics 图表时提高运营效率。

卓越运营支柱

Well-Architect AWS ed Framework 的[卓越运营支柱](#)侧重于运行和监控系统，以及不断改进流程和程序。它包括能够有效地支持开发和运行工作负载，深入了解其运行情况，并不断改进支持流程和程序以实现业务价值。您可以通过自我修复工作负载降低运营复杂性，这些工作负载无需人工干预即可检测和修复大多数问题。您可以按照本节所述的最佳实践努力实现这一目标，并使用 Amazon Neptune Analytics 指标和机制在工作负载偏离预期行为时做出正确响应。 APIs

本次对卓越运营支柱的讨论侧重于以下关键领域：

- 基础设施即代码 (IaC)
- 变更管理
- 弹性策略
- 事件管理
- 合规审计报告
- 日志记录和监控

使用 IaC 方法自动部署

使用 IaC 在 Neptune 上自动部署的最佳实践包括以下内容：

- 应用 IaC 部署 Neptune Analytics 图表和相关资源。要实现一致的环境配置，请使用提供的对 [Neptune Analytics 的支持AWS CloudFormation](#)来配置图表和私有端点。
- 用于 CloudFormation 在 [Ama SageMaker zon AI 上配置 Neptune 笔记本实例](#)。您可以使用笔记本在 Neptune Analytics 图表中查询和可视化数据。
- [当您使用现有来源（例如海王星数据库集群或快照，或者在亚马逊简单存储服务 \(Amazon S3\) Simple Service Service 中暂存的数据文件）创建 Neptune Analytics 图表时，请监控批量导入任务。](#)
- 自动执行 Neptune Analytics 操作程序，例如[调整图表](#)大小、删除图表并生成图表快照、从快照中恢复图表以及重置和重新加载图表。使用 [Neptune Analytics API](#)，该API可通过 [AWS Command Line Interface \(AWS CLI\)](#) 或 [SDK](#) 获得。
- 评估图表所需的正常运行时间。分析通常是短暂的；只有在需要运行算法时才需要使用图表。如果是这种情况，请使用 AWS CLI 或 SDKs 对图表进行[快照，并在不再需要图表时将其删除](#)。然后，如有必要，您可以稍后[从快照中将其恢复](#)。

- 将来自客户端的连接字符串存储在外部的。您可以将连接字符串存储在[AWS Secrets Manager](#)、[Amazon DynamoDB](#) 或任何可以动态更改它们的位置。
- [使用标签将元数据添加到](#)您的 Neptune Analytics 资源中，并根据标签跟踪使用情况。标签有助于整理您的资源。例如，您可以将通用标签应用于特定环境或应用程序中的资源。您还可以使用标签来分析资源使用量的账单；有关更多信息，请参阅《AWS 账单用户指南》中的[使用 AWS 成本分配标签组织和跟踪成本](#)。此外，您还可以使用您的 AWS Identity and Access Management (IAM) 策略中的条件，根据该 AWS 资源上使用的标签来控制对资源的访问权限。您可以使用全局 `aws:ResourceTag/tag-key` 条件键完成此操作。有关更多信息，请参阅 IAM 用户指南中的[控制 AWS 资源访问权限](#)。

运营设计

采用各种方法来改进 Neptune Analytics 图表的操作方式：

- 维护单独的 Neptune Analytics 图表，以供开发、测试和生产使用。这些图表可能具有不同的数据集、用户和操作控件。
- 为不同的用途保留单独的 Neptune Analytics 图表。例如，如果两组分析用户需要不同的图表，这些图表具有不同的时间表、模型、性能和可用性以及使用模式 SLAs，则为每组用户保留单独的图表。
- 让用户和操作人员做好准备，了解 Neptune Analytics 的[维护更新](#)。

进行频繁的、微小的、可逆的更改

以下建议侧重于您可以进行的微小、可逆的更改，以最大限度地降低复杂性并降低工作负载中断的可能性：

- 将 IaC 模板和脚本存储在源代码控制服务（例如 GitHub 或 GitLab）中。

Important

不要在源代码管理中存储 AWS 凭据。

- 要求部署 IaC 才能使用持续集成和持续交付 (CI/CD) 服务，例如或。[AWS CodeDeploy](#)[AWS CodeBuild](#)在将代码提升到生产图表之前，在非生产版 Neptune Analytics 环境中编译、测试和部署代码。

实现可观察性以获得可操作的见解

全面了解工作负载行为、性能、可靠性、成本和运行状况。以下建议可帮助您在 Neptune Analytics 中获得这种程度的理解：

- 监控 Neptune Analytics 的亚马逊 CloudWatch 指标。根据这些指标，您可以确定图表的大小（节点、边和向量的数量，加上总字节大小）、CPU 利用率以及查询请求和错误率。
- 为关键指标（例如 NumQueuedRequestsPerSec、NumOpenCypherRequestsPerSecGraphStorageUsagePercent 和 CPUUtilization 以及应用程序日志中的 Neptune 客户端响应创建 CloudWatch 仪表板和警报。
- 设置通知以监控 Neptune Analytics 图表的运行状况，例如图表大小、请求速率或 CPU 利用率何时超过您的阈值。例如，如果您打算大幅增长的图表上 GraphStorageUsagePercent 已攀升至 90%，请决定是否增加内存优化的 Neptune 容量单位 (m-nCU) 容量。如果当前的 m-nCU 为 128，则将其增加到 256 将减少大约 45% 的存储空间。如果通常大于零，NumQueuedRequestsPerSec 则可以考虑增加 m-nCU 容量以提供更多的计算容量。或者，您可以减少客户端的并发性。

从所有操作失败中吸取教训

自我修复基础设施是一项长期的工作，在出现罕见问题或响应效果不如预期时，它会不断迭代发展。采用以下做法可以推动将注意力集中在实现该目标上：

- 通过从所有失败中吸取教训来推动改进。
- 跨团队和组织分享所学知识。如果您的组织中有多个团队使用 Neptune，请创建一个公共聊天室或用户群组来分享经验和最佳实践。

使用日志记录功能监控未经授权或异常的活动

使用日志记录来观察异常的性能和活动模式。考虑下面的最佳实践：

- Neptune Analytics 支持使用记录控制平面动作。AWS CloudTrail 有关更多信息，请参阅使用 [记录 Neptune Analytics API 调用](#)。AWS CloudTrail 通过此功能，您可以跟踪 Neptune Analytics 资源的创建、更新和删除。为了实现强大的监控和警报，您还可以将 CloudTrail 事件与 [Amazon CloudWatch 日志](#) 集成。为了增强您对 Neptune Analytics 服务活动的分析并识别活动的变化 AWS

账户，您可以[使用 Amazon Athena CloudTrail 查询日志](#)。例如，可以使用查询确定趋势，并根据属性（如源 IP 地址或用户）进一步隔离活动。

- 您还可以使用 CloudTrail 来[启用 Neptune Analytics 数据平面活动的日志记录](#)，例如查询执行。您可以查看正在运行哪些查询、其频率和来源。默认情况下，CloudTrail 不记录数据事件。记录数据事件将收取额外费用。有关更多信息，请参阅[AWS CloudTrail 定价](#)。
- 您还可以在控制平面或数据平面中记录对 Neptune Analytics 的应用程序调用。例如，如果您使用[适用于 Python \(Boto3\) 的 AWS SDK](#)进行查询，则可以[启用调试级别的日志记录](#)以获取对控制台或文件的查询跟踪。这在开发过程中很有用。我们还建议您捕获并记录应用程序中的异常。

安全支柱

云安全是重中之重 AWS。作为 AWS 客户，您可以受益于专为满足大多数安全敏感型组织的要求而构建的数据中心和网络架构。安全性是您和 AWS 的共同责任。[责任共担模式](#)将其描述为云的 安全性和云中 的安全性：

- 云安全 — AWS 负责保护在云 AWS 服务 中运行的基础架构 AWS Cloud。AWS 还为您提供可以安全使用的服务。作为[AWS 合规计划](#)的一部分，第三方审计师定期测试和验证 AWS 安全的有效性。要了解适用于 Neptune 的合规计划，请参阅[按合规计划划分的范围内的AWS 服务](#)。
- 云端安全 — 您的责任由您 AWS 服务 使用的内容决定。您还需要对其他因素负责，包括您的数据的敏感性、您公司的要求以及适用的法律法规。有关数据隐私的更多信息，请参阅[数据隐私 FAQs](#)。有关欧洲数据保护的信息，请参阅[责任AWS 共担模型和 GDPR](#) 博客文章。

Well-Architect AWS ed Framework [的安全支柱](#)可帮助你了解在使用 Neptune Analytics 时如何应用分担责任模型。以下主题说明了如何配置 Neptune Analytics 以满足您的安全和合规目标。您还将学习如何使用其他方法来帮助您监控和保护您 AWS 服务 的 Neptune Analytics 资源。安全支柱包括以下关键重点领域：

- 数据安全性
- 网络安全
- 身份验证和授权

实现数据安全

数据泄露和泄露会使您的客户处于危险之中，并可能对您的公司造成严重的负面影响。以下最佳实践有助于保护您的客户数据免遭无意和恶意泄露：

- 图表名称、标签、IAM 角色和其他元数据不应包含机密或敏感信息，因为这些数据可能会出现在账单或诊断日志中。
- URIs 或者指向作为数据存储在 Neptune 中的外部服务器的链接不应包含用于验证请求的凭据信息。
- Neptune Analytics 图表处于静态加密状态。您可以使用默认密钥或您选择的 AWS Key Management Service (AWS KMS) 密钥来加密图表。您还可以对批量导入期间导出到 Amazon S3 的快照和数据进行加密。导入完成后，您可以移除加密。

- 使用 OpenCypher 语言时，请练习正确的输入验证和[参数化](#)技术，以防止 SQL 注入和其他形式的攻击。避免构造使用字符串连接和用户提供的输入的查询。使用参数化查询或预处理语句将输入参数安全地传递到图形数据库。有关更多信息，请参阅 Neptune [文档中的 OpenCypher 参数化查询示例](#)。

保护您的网络

您可以为公共连接启用 Neptune Analytics 图表，以便可以从虚拟私有云 (VPC) 外部访问该图。默认情况下，此连接处于禁用状态。该图表需要 IAM 身份验证。调用者必须获得身份并拥有使用图表的权限。例如，要[运行 OpenCypher 查询](#)，调用者需要对特定图表具有读取、写入或删除权限。

您也可以为图表[创建私有终端节点](#)，以便从 VPC 内访问图表。创建终端节点时，您可以指定 VPC、子网和安全组来限制对图表的调用权限。

为了保护传输中的数据，Neptune Analytics 强制通过 HTTPS 与图表建立 SSL 连接。有关更多信息，请参阅 Neptune Analytics [文档中的 Neptune Analytics 中的数据保护](#)。

实现身份验证和授权

调用 Neptune Analytics 图表需要 IAM 身份验证。调用者必须获得身份并拥有足够的权限才能在图表上执行操作。有关 API 操作及其所需权限的描述，请参阅 [Neptune Analytics API 文档](#)。您可以[强制执行条件检查](#)以按标签限制访问权限。

IAM 身份验证使用[AWS 签名版本 4 \(Sigv4\) 协议](#)。为了简化应用程序的使用，我们建议您使用 S [AWS SDK](#)。例如，在 Python 中，使用 Neptune Graph 的 [Boto3 客户端](#)，它抽象了 Sigv4。

当您将数据加载到图表中时，[批量加载](#)会使用调用者的 IAM 证书。调用者必须有权从已设置信任关系的 Amazon S3 下载数据，这样 Neptune Analytics 才能担任将数据从 Amazon S3 文件加载到图表中的角色。

[批量导入](#)可以在图表创建期间执行（由基础设施团队执行），也可以在现有的空图表上执行（由有权启动导入任务的数据工程团队执行）。在这两种情况下，Neptune Analytics 都假设调用方作为输入提供的 IAM 角色。此角色允许其读取和列出存放输入数据的 Amazon S3 文件夹的内容。

可靠性支柱

在 AWS Well-Architected Framework 的 [可靠性](#) 支柱包括工作负载能够在预期时正确、一致地执行其预期功能。这包括在工作负载的整个生命周期中对其进行操作和测试的能力。

可靠的工作负载始于前期的软件和基础设施设计决策。您的架构选择会影响您在所有 Well-Architected 支柱上的工作负载行为。为了提高可靠性，必须遵循一些特定的模式，如本节所述。

可靠性支柱侧重于以下关键领域：

- 工作负载架构，包括服务配额和部署模式
- 变更管理
- 故障管理

了解 Neptune 服务配额

您的 AWS 账户对每个配额都有默认配额（以前称为限制）AWS 服务。除非另有说明，否则，每个配额是区域特定的。您可以申请提高部分（但不是全部）配额。

要查找 Neptune Analytics 的配额，请打开 [服务配额控制台](#)。在导航窗格中，选择 AWS 服务，然后选择亚马逊 Neptune Analytics。请注意图表和快照数量的配额、图表的最大预配置内存以及 API 请求速率。

如果最大预配置内存不足以容纳您的数据集，请评估哪些节点和边缘类型对于您的预期分析用途至关重要。加载数据子集，以便在允许的预配置容量内进行分析。许多分析工作负载，尤其是那些运行图形算法的工作负载，只需要具有有限属性的拓扑，而不是完整的交易图。（有关事务性工作负载和分析工作负载之间差异的讨论，请参阅 [性能效率支柱](#) 部分。）

如果图表的最大数量不足以满足您的预期用途：

- 考虑组合具有相似用途的图表。
- 评估在给定时间必须运行多少张图表。如果您有临时分析用例，请在不再需要图表时对其进行快照并删除。这减少了与配额对应的图表数量。
- 请考虑不同的配置图 AWS 账户。

了解 Neptune 部署模式

计划部署 Neptune Analytics 图表时，请了解以下决策点：

- **播种**：决定是创建空图，还是在创建时使用来自 Amazon S3、现有 Neptune 数据库集群或现有 Neptune 数据库快照的数据将数据加载到其中。

建议：如果源是 Neptune 集群或快照，则必须在创建图表时加载其数据。如果源是 Amazon S3，则如果加载数据的工作量很大，并且最好作为基础设施配置活动执行，则在创建时加载数据。如果您更喜欢将数据作为数据工程或应用程序活动加载，请创建一个空图表并稍后从 Amazon S3 加载数据。

- **容量**：根据数据大小和预期的应用程序使用情况，估计图表所需的预配置容量。

建议：在创建时，[指定最大预配置内存](#)以限制图形大小。此设置是强制性的。如有必要，您可以稍后更改容量。

- **可用性和容错能力**：决定是否需要副本才能获得可用性。副本充当热备用副本，以便在图形出现故障时进行恢复。具有副本的图的恢复速度比没有副本的图更快。还要考虑图表需要多长时间，它是否仅用于临时分析，如果是，何时将其删除。

建议：在创建图表之前，请确定可用性要求，例如图表不可用多长时间以及何时可以将其删除。

- **网络和安全**：确定您是否需要公共连接、私有连接或两者兼而有之，以及是否要加密数据。

建议：在创建图表之前，请先了解组织要求，例如是否允许公共连接以及图形客户端应用程序将在何处部署。

- **备份和恢复**：确定是否应创建快照，如果是，则确定何时或在何种条件下创建。考虑您的组织是否有灾难恢复 (DR) 要求。

建议：创建快照是一项手动活动。在创建图表之前，决定何时创建快照并考虑灾难恢复要求。

管理和扩展 Neptune 集群

Neptune Analytics 图表由单个内存优化实例组成。实例的容量 (m-nCU) 是在创建时设置的。通过[管理操作](#)增加预配置容量，可以纵向扩展实例；也可以减少预配置容量。副本是被动的故障转移目标，因此它们不会增加图表的规模。在这方面，图形副本不同于 [Neptune 数据库只读副本](#)，后者是 Neptune 集群中的活动实例，可以处理来自应用程序的读取操作。

副本会产生成本。该副本按图表的 m-nCU 价格定价。例如，如果一个图表预配置了 128 m-nCU 并具有单个副本，则成本是没有副本的同等图表的两倍。

在分析领域，扩大规模有两个主要原因：

- 为了为分析查询和算法提供更多的内存和 CPU，由于单个查询的成本很高，因此要运行的图形算法本质上很复杂，并且根据其输入需要更多的资源，或者并发请求速率很高。如果此类查询遇到 out-of-memory 错误，则扩大规模是一种合理的补救措施。
- 支持比计划更大的图表大小。例如，如果当前预配置的容量为 128 M-nCU 以支持 60 GB 的源数据，而您又需要增加 40 GB 的源数据，则有必要将容量增加到 256 M-nCU。

监控 Neptune Analytics 的 CloudWatch 指标，例

如 NumQueuedRequestsPerSecNumOpenCypherRequestsPerSecGraphStorageUsagePercent、Gra 和，以确定是否需要扩展。您可以通过控制台、或 AWS CLI，更新图表的配置 SDKs。（有关示例和最佳实践，请参阅[卓越运营支柱](#)部分。）

管理备份和故障转移事件

使用副本来确保图形在出现故障时仍然可用。图表使用基于日志的持久性在中跨可用区提交更改。AWS 区域副本充当热备用副本，可以访问这些数据。如果出现故障，图表将恢复对副本的操作。应用程序继续使用相同的端点连接到图表。故障期间的动态请求会生成错误，并出现服务不可用异常。考虑在应用程序代码中使用[带退避模式的重试](#)来捕获错误，然后在短暂的间隔后重试。在故障转移期间发出的新请求会被排队，延迟时间可能会更长。

如果未配置任何副本并且图表出现故障，Neptune Analytics 将从持久存储中恢复，但恢复需要更长的时间，因为 Neptune 必须重新初始化资源。

创建图表的快照。（Neptune Analytics 不自动拍摄快照。）如果图表在创建后定期修改，请经常拍摄快照以捕捉其当前状态。如果不需要恢复到较早的时间点，请删除较早的快照。

您可以与其他账户共享快照，也可以跨账号共享快照 AWS 区域。如果您有灾难恢复要求，请考虑从快照恢复不同区域的图表是否符合您的恢复时间目标 (RTO) 和恢复点目标 (RPO) 要求。

性能效率支柱

Well-Architect AWS ed Framework 的 [性能效率支柱](#) 侧重于如何在摄取或查询数据时优化性能。性能优化是一个循序渐进的持续过程，包括以下内容：

- 确认业务需求
- 衡量工作负载性能
- 识别性能不佳的组件
- 调整组件以满足您的业务需求

性能效率支柱提供了特定于用例的指南，可以帮助您确定要使用的正确图表数据模型和查询语言。它还包括在向 Neptune Analytics (海王星分析) 摄取数据和使用数据时应遵循的最佳实践。

绩效效率支柱侧重于以下关键领域：

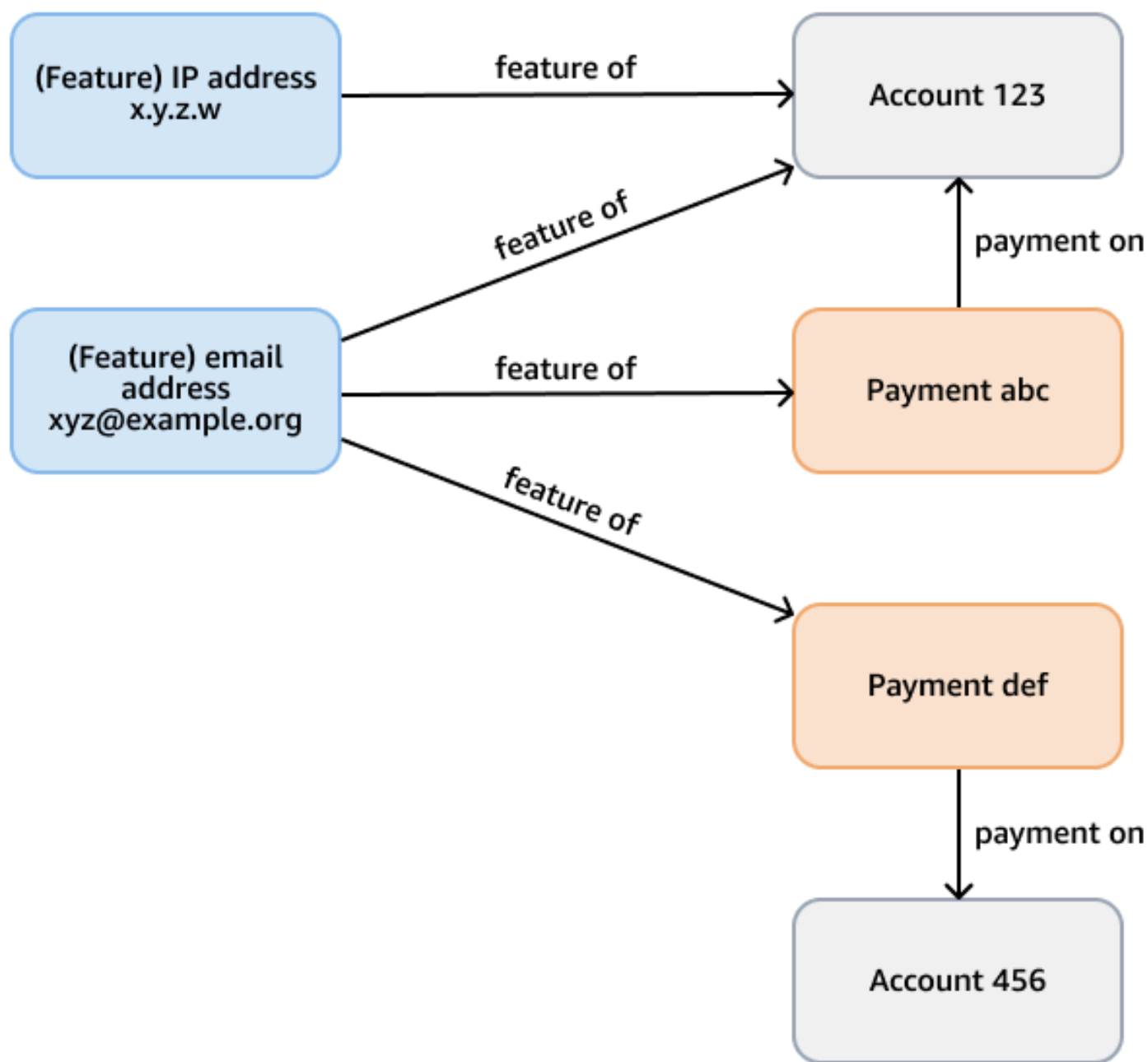
- 图形建模
- 查询优化
- 调整图表大小
- 写入优化

了解用于分析的图形建模

《在 Amazon Neptune 上应用 AWS Well-Architected Framework》指南 [讨论了如何](#) 提高性能效率的图形建模。影响性能的建模决策包括选择需要哪些节点和边 IDs、它们的标签和属性、边的方向、标签应该是通用标签还是特定标签，以及查询引擎在图表中导航以处理常见查询的效率。

这些注意事项也适用于 Neptune Analytics；但是，区分交易和分析使用模式很重要。在事务数据库（例如 Neptune 数据库）中高效查询的图形模型可能需要重塑才能进行分析。

例如，假设 Neptune 数据库中的欺诈图表，其目的是检查信用卡付款中的欺诈模式。此图表可能包含代表账户和付款的账户、付款和功能（例如电子邮件地址、IP 地址、电话号码）的节点。此关联图支持诸如遍历可变长度路径之类的查询，该路径从给定付款开始，需要几跳才能找到相关的功能和账户。下图显示了这样的图表。



分析要求可能更具体，例如查找通过功能关联的账户社区。为此，您可以使用[弱连接组件 \(WCC\)](#) 算法。在前面的示例中，针对模型运行它效率低下，因为它需要遍历几种不同类型的节点和边。下图中的模型效率更高。如果账户本身（或来自账户的付款）共享一项功能，则它会将account节点与shares feature优势联系起来。例如，Account 123具有电子邮件功能xyz@example.org，并Account 456使用相同的电子邮件进行付款（Payment def）。



WCC 的计算复杂度为 $O(|E| \log D)$ ，其中 $|E|$ 是图中的边数， D 是连接节点的直径（最长路径的长度）。由于事务模型省略了不必要的节点和边，因此它优化了边的数量和直径，并降低了 WCC 算法的复杂性。

使用 Neptune Analytics 时，请根据所需的算法和分析查询进行回溯。如有必要，请重塑模型以优化这些查询。在将数据加载到图表之前，您可以重塑模型，也可以编写修改图表中现有数据的查询。

优化查询

请按照以下建议优化 Neptune Analytics 查询：

- 使用[参数化查询和查询计划缓存](#)，默认情况下处于启用状态。使用计划缓存时，引擎会准备查询以供日后使用（前提是查询在 100 毫秒或更短的时间内完成），这样可以节省后续调用的时间。
- 对于慢速查询，请运行[解释计划](#)以发现瓶颈并相应地进行改进。
- 如果您使用[向量相似度搜索](#)，请确定较小的嵌入是否产生准确的相似度结果。您可以更高效地创建、存储和搜索较小的嵌入内容。
- 遵循[记录在案的最佳实践，在 Neptune Analytics 中使用 OpenCypher](#)。例如，在 UNWIND 子句中使用[扁平化地图](#)，并尽可能[指定边缘标签](#)。
- 使用[图形算法](#)时，要了解算法的输入和输出、其计算复杂性以及它的工作原理。
 - 在调用图形算法之前，请使用子MATCH句最小化输入节点集。例如，要限制节点进行[广度优先搜索 \(BFS\)](#)，请按照 Neptune Analytics 文档中提供的[示例](#)进行操作。
 - 如果可能，对节点和边缘标签进行筛选。例如，BFS 具有用于筛选对特定节点标签 (vertexLabel) 或特定边缘标签 (edgeLabels) 的遍历的输入参数。
 - 使用诸如之类的边界参数maxDepth来限制结果。
 - 用concurrency参数进行实验。尝试使用值为 0，它使用所有可用的算法线程来并行化处理。将其与单线程执行进行比较，方法是将参数设置为 1。算法可以在单个线程中更快地完成，尤其是在较小的输入中，例如浅广度优先搜索，在这种情况下，并行性不会显著缩短执行时间，并且可能会带来开销。

- 在相似类型的算法之间进行选择。例如，[Bellman-Ford](#) 和 [delta-stepping](#) 都是单源最短路径算法。使用自己的数据集进行测试时，请尝试两种算法并比较结果。增量步进通常比贝尔曼-福特快，因为它的计算复杂性较低。但是，性能取决于数据集和输入参数，尤其是delta参数。

优化写入

请按照以下做法优化 Neptune Analytics 中的写入操作：

- 寻找最有效的方法将数据加载到图表中。从 Amazon S3 中的数据加载时，如果数据大小超过 50 GB，则使用[批量导入](#)。对于较小的数据，请使用[批量加载](#)。如果您在运行批量加载时遇到 out-of-memory 错误，请考虑增加 m-ncu 值或将负载分成多个请求。实现此目的的一种方法是在 S3 存储桶中将文件拆分为多个前缀。在这种情况下，请分别为每个前缀调用 batch load。
- 使用批量导入或批处理加载器填充初始图表数据集。仅对较小的更改使用事务性 OpenCypher 创建、更新和删除操作。
- 使用批量导入或并发度为 1（单线程）的批处理加载器将嵌入数据提取到图表中。尝试使用以下方法之一预先加载嵌入内容。
- 评估矢量相似度搜索算法中精确相似度搜索所需的向量嵌入维度。如果可能，请使用较小的尺寸。这样可以加快嵌入的加载速度。
- 如果需要，使用变异算法来记住算法结果。例如，[度突变中心性算法](#)会找到每个输入节点的度数，并将该值作为节点的属性写入。如果这些节点周围的连接随后没有改变，则该属性将保持正确的结果。无需再次运行该算法。
- 如果需要重新开始，请使用[图表重置管理操作](#)来清除所有节点、边和嵌入内容。如果图形很大，则使用 OpenCypher 查询删除所有节点、边和嵌入是不可行的。对大型数据集进行一次删除查询可能会超时。随着大小的增加，数据集需要更长的时间才能移除，交易规模也会增加。相比之下，完成图表重置的时间大致保持不变，并且该操作提供了在运行快照之前创建快照的选项。

大小合适的图表

总体性能取决于 Neptune Analytics 图表的预配置容量。容量以称为内存优化的 Neptune 容量单位 (m-) 的单位来衡量。NCUs 确保您的图表大小足以支持您的图表大小和查询。请注意，增加容量并不一定能提高单个查询的性能。

如果可能，请通过从现有来源（例如 Amazon S3 或现有 Neptune 集群或快照）导入数据来创建图表。[您可以对最小容量和最大容量设置界限](#)。您也可以[更改现有图表上的预配置容量](#)。

监控诸

如NumQueuedRequestsPerSec、NumOpenCypherRequestsPerSecGraphStorageUsagePercentG和之类的 CloudWatch 指标，CPUUtilization以评估图表的大小是否合适。确定是否需要更多容量来支持您的图表大小和负载。有关如何解释其中一些指标的更多信息，请参阅[卓越运营支柱](#)部分。

成本优化支柱

Well-Architected AWS d Framework [的成本优化支柱](#)侧重于避免不必要的成本。以下建议可以帮助您满足 Neptune Analytics 的成本优化设计原则和架构最佳实践。

成本优化支柱侧重于以下关键领域：

- 了解一段时间内的支出并控制资金分配
- 选择类型和数量正确的资源
- 在不超支的情况下进行扩展以满足业务需求

了解使用模式和所需的服务

在采用 Neptune Analytics 之前，请评估您的用例是否适合进行图表分析。

- **图形数据库：**如果您的数据模型具有清晰的图形结构，并且您的查询需要探索关系并遍历多个跃点，则诸如 Neptune 之类的图形数据库非常适合您的工作负载。图形数据库不适合以下模式：
 - 主要是单跳查询。在此用例中，请考虑您的数据是否可以更好地表示为对象的属性。
 - 作为属性存储的 JSON 或二进制大对象 (blob) 数据。
- **图表分析：**Neptune Analytics 是一个图形分析数据库引擎，可以快速分析内存中的大量图形数据，以获得见解和发现趋势。您可以在 Neptune 数据库和 Neptune Analytics 图表中存储和查询图表数据。Neptune 数据库最适合可扩展的在线事务处理 (OLTP) 需求。Neptune Analytics 最适合临时分析工作负载。您可以将两者结合使用，方法是将数据从面向交易的 Neptune 数据库加载到 Neptune Analytics 图表中，对这些数据进行分析。分析完成后，您可以移除 Neptune Analytics 图表。有关更详细的比较，请参阅 Neptune Analytics 文档[中的何时使用 Neptune Analytics 以及何时使用 Neptune 数据库](#)。

在考虑成本的基础上，确定如何最好地填充您的 Neptune Analytics 图表。

- **[批量导入](#)**暂存在 S3 存储桶中的图表数据。如果您的数据之前已暂存以批量加载到 Neptune 数据库，或者您已经拥有或可以轻松生成要以 [CSV 或批量导入所需的其他支持格式](#)进行分析的数据，我们建议您使用此选项。您可以将批量导入作为图表创建过程的一部分运行。[您可以对最小和最大容量设置界限](#)。您还可以在[先前创建的空图表上运行导入](#)，并在[导入任务运行时对其进行监控](#)。

- [您可以创建一个空图，然后使用批量加载通过 OpenCypher 查询填充它。](#) 如果要加载的数据暂存在 Amazon S3 中且小于 50 GB，则此选项非常理想。
- 您可以[从 Neptune 数据库集群（在 Neptune 数据库版本 1.3.0 或更高版本中支持）中的数据填充图表](#)。这种模式的目的是对当前存在于图表数据库中的数据进行分析。即使数据库最初是通过批量加载填充的，此后也可能发生了重大变化。要从数据库导入，Neptune Analytics 会克隆您的数据库并将克隆中的数据导出到 S3 存储桶。此过程会产生成本：尤其是运行克隆的 Neptune 数据库成本，以及 Amazon S3 存储和使用导出数据成本。导出完成后，克隆将被删除。您可以在 Amazon S3 中删除导出的数据。
- 您可以[从 Neptune 数据库集群的快照中填充图形](#)。这与前一个选项类似，唯一的区别是源是数据库快照。要从快照导入，Neptune Analytics 首先将快照恢复到新的数据库集群，然后将数据导出到 S3 存储桶。此过程会产生成本：尤其是运行已恢复集群的 Neptune 数据库成本，以及 Amazon S3 存储和使用导出数据成本。
- 您还可以通过在图表上使用符合原子性、一致性、隔离性、耐久性 (ACID) 的事务来执行 OpenCypher 查询来创建、更新或删除数据。我们建议使用这种方法来进行小幅更新，但不能作为播种图表的一种方式。

如果分析所需的数据已在 Amazon S3 中暂存，我们建议批量导入或批量加载。它们比从 Neptune 数据库集群或快照填充图表更具成本效益。

选择资源时要注意成本

Nep@@@ [tune Analytics 的定价](#)使用一种称为内存优化型海王星容量单位 (m-nCU) 的单位。使用给定的 m-nCU 运行图表需要支付固定的每小时成本。图表可能有用于故障转移的副本，而且这些副本还会产生每小时 m-nCU 成本。

我们建议采用以下最佳做法来估算容量、限制成本并根据性能监控成本：

- 如果可能，请通过从现有来源导入数据来创建图表：在 Amazon S3 中暂存的数据或现有 Neptune 集群或快照。这可以节省您的精力，因为 Neptune Analytics 执行了为图表播种的繁重工作，并且[您可以指定最大容量](#)。
- 您可以[更改现有图表上的预配置容量](#)。
- 当不再需要该图表时，您可以[创建快照并删除该图表](#)。如果您需要再次使用它，可以从快照中恢复图表。
- 在创建图表时，您可以选择副本的数量。根据您的分析可用性要求设置该值。通过最小化此设置来节省成本。最大值为 2 允许在不同的可用区中使用两个副本实例。最小值为 0 表示 Neptune Analytics

不会运行副本。但是，当副本可用时，恢复速度会更快。有关图表失败和恢复的说明，请参阅“[可靠性](#)”支柱部分。

- 使用监控当前和过去账单周期的 Neptune Analytics 费用。[AWS 账单与成本管理](#)
- 监控 Neptune Analytics 的指标 `GraphSizeBytes`，尤其 `NumQueuedRequestsPerSec` 是 `CPUUtilization`、`NumOpenCypherRequestsPerSec` 和 `GraphStorageUsagePercent`，以评估配置的容量大小是否适合图表。CloudWatch `NumOpenCypherRequestsPerSec` `GraphStorageUsagePercent` 确定较小的容量能否容纳观察到的请求速率、CPU 使用率和图形大小。
- 如果您的图表需要私有终端节点，请注意弹性 IPs、虚拟私有云 (VPC) 终端节点、NAT 网关的成本或其他与 VPC 相关的费用。有关更多信息，请参阅 [Amazon VPC 定价](#) 和 [亚马逊 EC2 定价](#)。
- 您可能需要运行一个或多个 Neptune 笔记本实例，以提供客户端界面来帮助开发人员和分析师查询和可视化图表（参见 [Neptune 工作台定价](#)）。为了最大限度地降低成本，请在用户之间共享实例，并为每个用户创建单独的笔记本文件夹。不使用实例时将其关闭。有关自动关闭的方法，请参阅 AWS 博客文章 [使用资源标签自动停止和启动 Amazon Neptune 环境资源](#)。

可持续发展支柱

Well-Architect AWS ed Framework 的 [可持续发展支柱](#) 侧重于最大限度地减少运行云工作负载对环境的影响。关键主题包括可持续发展责任共担模式、了解影响以及最大限度地利用以最大限度地减少所需资源并减少对下游的影响。

可持续发展支柱包含以下关键重点领域：

- 你的影响力
- 可持续发展目标
- 最大限度地提高使用率
- 预见并采用新的、更高效的软件产品
- 托管服务的使用
- 减少对下游的影响

本指南侧重于了解您的影响。有关其他可持续发展设计原则的更多信息，请参阅 Well-Architect [AWS ed 框架](#)。

您的选择和要求会对环境产生影响。如果您可以选择 AWS 区域 碳强度较低的产品，并且如果您的要求反映了实际的工作量需求，而不仅仅是最大限度地提高正常运行时间和耐用性，那么工作负载的可持续性就会提高。下一节将讨论最佳实践和注意事项，如果在工作负载设计和持续运营中采用这些最佳做法和注意事项，这些做法和注意事项将对环境产生积极影响

考虑你的 AWS 区域 选择

AWS 区域 有些位于亚马逊可再生能源项目附近，或者位于电网公布的碳强度低于其他项目的地方。考虑可能适合您的工作负载的区域 [对可持续发展的影响](#)，并将您的列表与提供 Neptune Analytics 的 [区域](#) 进行交叉引用。

优化消费

通过练习以下几点，最大限度地减少 Neptune Analytics 的消耗：

- 分析通常是短暂的。只有在运行算法和记录结果时才需要图表。如果是这种情况，请在不再需要图表时对其进行 [快照和删除](#)。如有必要，您可以稍后 [从快照中将其恢复](#)。

- 如果工作负载是短暂的，并且您可以灵活地决定何时运行分析，请考虑功耗 day-to-day 趋势。在某些时期，对电力的需求会更高。如果你在美国，可以在美国能源信息管理局 (EIA) 网站上查看[每日用电量指标](#)。如果可能，请在您所在地区的非高峰时段运行工作负载。
- 如果工作负载不是临时性的，而只需要在有限的时间内可用，请删除图表，并在需要时将其从快照中恢复。如果其可用性按计划进行，请通过脚本自动执行恢复过程，以便图表在预定时间准备就绪。
- 如果数据是只读的，或者自上次快照以来没有发生变化，则在删除之前不要再次对其进行快照。
- 在 Neptune 笔记本电脑不使用时将其停用。

- 监

控 NumQueuedRequestsPerSec、NumOpenCypherRequestsPerSec GraphStorageUsagePercent 和等 CloudWatch 指标 CPUUtilization 以评估图表是否过大。确定较小的实例容量能否适应观察到的请求速率、CPU 使用率和图形大小。

优化软件开发和架构模式

为防止浪费，请优化模型和查询，共享计算资源，以便使用 Neptune 实例和集群中的所有可用资源。具体的最佳做法包括：

- 优化查询和图形算法调用。使用参数化查询并[使用查询计划缓存](#)，该缓存默认处于启用状态。对于慢速查询，请运行[解释计划](#)以进行改进。如果您使用[向量相似度搜索](#)，请确定较小的嵌入是否产生准确的相似度结果，因为可以更有效地创建、存储和搜索较小的嵌入。在调用[图形算法](#)之前，请使用子MATCH句最小化输入节点集。如果可能，对节点和边缘标签进行筛选。
- 寻找最有效的方法将数据加载到图表中。如果您从 Amazon S3 中的数据加载，则在数据大小超过 50 GB 时使用[批量导入](#)。对较小的数据使用[批量加载](#)。
- 要求开发者共享 Neptune 笔记本实例，而不是每个人创建自己的实例。在单个 Jupyter 实例上为每位开发者创建单独的笔记本文件夹。不使用实例时将其关闭。

资源

参考信息

- [AWS Well-Architected](#)
- [AWS WellArchited Framework 文档](#)
- [为亚马逊 Neptune 应用 AWS Well-Architected 框架](#)
- [Neptune Analytics 最佳实践](#)

博客文章和视频

- [Nep@@ tune 博客文章](#) (AWS 数据库博客)
- [使用资源标签自动停止和启动 Amazon Neptune 环境资源](#) (AWS 数据库博客)
- [亚马逊 Neptune 零食 \(短视频 \)](#) YouTube

训练

- [亚马逊 Neptune 入门](#)
- [使用亚马逊 Neptune 进行构建](#)
- [亚马逊 Neptune 的数据建模](#)
- [亚马逊 Neptune 分析研讨会](#)

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	—	2024 年 12 月 20 日

AWS 规范性指导词汇表

以下是 AWS 规范性指导提供的策略、指南和模式中的常用术语。若要推荐词条，请使用术语表末尾的提供反馈链接。

数字

7 R

将应用程序迁移到云中的 7 种常见迁移策略。这些策略以 Gartner 于 2011 年确定的 5 R 为基础，包括以下内容：

- 重构/重新架构 - 充分利用云原生功能来提高敏捷性、性能和可扩展性，以迁移应用程序并修改其架构。这通常涉及到移植操作系统和数据库。示例：将您的本地 Oracle 数据库迁移到兼容 Amazon Aurora PostgreSQL 的版本。
- 更换平台 - 将应用程序迁移到云中，并进行一定程度的优化，以利用云功能。示例：在中将您的本地 Oracle 数据库迁移到适用于 Oracle 的亚马逊关系数据库服务 (Amazon RDS) AWS Cloud。
- 重新购买 - 转换到其他产品，通常是从传统许可转向 SaaS 模式。示例：将您的客户关系管理 (CRM) 系统迁移到 Salesforce.com。
- 更换主机 (直接迁移) - 将应用程序迁移到云中，无需进行任何更改即可利用云功能。示例：在中的 EC2 实例上将您的本地 Oracle 数据库迁移到 Oracle AWS Cloud。
- 重新定位 (虚拟机监控器级直接迁移)：将基础设施迁移到云中，无需购买新硬件、重写应用程序或修改现有操作。您可以将服务器从本地平台迁移到同一平台的云服务。示例：将 Microsoft Hyper-V 应用程序迁移到 AWS。
- 保留 (重访) - 将应用程序保留在源环境中。其中可能包括需要进行重大重构的应用程序，并且您希望将工作推迟到以后，以及您希望保留的遗留应用程序，因为迁移它们没有商业上的理由。
- 停用 - 停用或删除源环境中不再需要的应用程序。

A

ABAC

请参阅[基于属性的访问控制](#)。

抽象服务

参见[托管服务](#)。

ACID

参见[原子性、一致性、隔离性、持久性](#)。

主动-主动迁移

一种数据库迁移方法，在这种方法中，源数据库和目标数据库保持同步（通过使用双向复制工具或双写操作），两个数据库都在迁移期间处理来自连接应用程序的事务。这种方法支持小批量、可控的迁移，而不需要一次性割接。与[主动-被动迁移](#)相比，它更灵活，但需要更多的工作。

主动-被动迁移

一种数据库迁移方法，在这种方法中，源数据库和目标数据库保持同步，但在将数据复制到目标数据库时，只有源数据库处理来自连接应用程序的事务。目标数据库在迁移期间不接受任何事务。

聚合函数

一个 SQL 函数，它对一组行进行操作并计算该组的单个返回值。聚合函数的示例包括SUM和MAX。

AI

参见[人工智能](#)。

AI Ops

参见[人工智能操作](#)。

匿名化

永久删除数据集中个人信息的过程。匿名化可以帮助保护个人隐私。匿名化数据不再被视为个人数据。

反模式

一种用于解决反复出现的问题的常用解决方案，而在这类问题中，此解决方案适得其反、无效或不如替代方案有效。

应用程序控制

一种安全方法，仅允许使用经批准的应用程序，以帮助保护系统免受恶意软件的侵害。

应用程序组合

有关组织使用的每个应用程序的详细信息的集合，包括构建和维护该应用程序的成本及其业务价值。这些信息是[产品组合发现和分析过程](#)的关键，有助于识别需要进行迁移、现代化和优化的应用程序并确定其优先级。

人工智能 (AI)

计算机科学领域致力于使用计算技术执行通常与人类相关的认知功能，例如学习、解决问题和识别模式。有关更多信息，请参阅[什么是人工智能？](#)

人工智能操作 (AIOps)

使用机器学习技术解决运营问题、减少运营事故和人为干预以及提高服务质量的过程。有关如何在 AIOps AWS 迁移策略中使用的更多信息，请参阅[操作集成指南](#)。

非对称加密

一种加密算法，使用一对密钥，一个公钥用于加密，一个私钥用于解密。您可以共享公钥，因为它不用于解密，但对私钥的访问应受到严格限制。

原子性、一致性、隔离性、持久性 (ACID)

一组软件属性，即使在出现错误、电源故障或其他问题的情况下，也能保证数据库的数据有效性和操作可靠性。

基于属性的访问权限控制 (ABAC)

根据用户属性 (如部门、工作角色和团队名称) 创建精细访问权限的做法。有关更多信息，请参阅 AWS Identity and Access Management (I [AM](#)) 文档 [AWS 中的 AB AC](#)。

权威数据源

存储主要数据版本的位置，被认为是最可靠的信息源。您可以将数据从权威数据源复制到其他位置，以便处理或修改数据，例如对数据进行匿名化、编辑或假名化。

可用区

中的一个不同位置 AWS 区域，不受其他可用区域故障的影响，并向同一区域中的其他可用区提供低成本、低延迟的网络连接。

AWS 云采用框架 (AWS CAF)

该框架包含指导方针和最佳实践 AWS，可帮助组织制定高效且有效的计划，以成功迁移到云端。AWS CAF 将指导分为六个重点领域，称为视角：业务、人员、治理、平台、安全和运营。业务、人员和治理角度侧重于业务技能和流程；平台、安全和运营角度侧重于技术技能和流程。例如，人员角度针对的是负责人力资源 (HR)、人员配置职能和人员管理的利益相关者。从这个角度来看，AWS CAF 为人员发展、培训和沟通提供了指导，以帮助组织为成功采用云做好准备。有关更多信息，请参阅 [AWS CAF 网站](#) 和 [AWS CAF 白皮书](#)。

AWS 工作负载资格框架 (AWS WQF)

一种评估数据库迁移工作负载、推荐迁移策略和提供工作估算的工具。AWS WQF 包含在 AWS Schema Conversion Tool (AWS SCT) 中。它用来分析数据库架构和代码对象、应用程序代码、依赖关系和性能特征，并提供评测报告。

B

坏机器人

旨在破坏个人或组织或对其造成伤害的[机器人](#)。

BCP

参见[业务连续性计划](#)。

行为图

一段时间内资源行为和交互的统一交互式视图。您可以使用 Amazon Detective 的行为图来检查失败的登录尝试、可疑的 API 调用和类似的操作。有关更多信息，请参阅 Detective 文档中的[行为图中的数据](#)。

大端序系统

一个先存储最高有效字节的系统。另请参见[字节顺序](#)。

二进制分类

一种预测二进制结果（两个可能的类别之一）的过程。例如，您的 ML 模型可能需要预测诸如“该电子邮件是否为垃圾邮件？”或“这个产品是书还是汽车？”之类的问题

bloom 筛选条件

一种概率性、内存高效的数据结构，用于测试元素是否为集合的成员。

蓝/绿部署

一种部署策略，您可以创建两个独立但完全相同的环境。在一个环境中运行当前的应用程序版本（蓝色），在另一个环境中运行新的应用程序版本（绿色）。此策略可帮助您在影响最小的情况下快速回滚。

自动程序

一种通过互联网运行自动任务并模拟人类活动或互动的软件应用程序。有些机器人是有用或有益的，例如在互联网上索引信息的网络爬虫。其他一些被称为恶意机器人的机器人旨在破坏个人或组织或对其造成伤害。

僵尸网络

被[恶意软件](#)感染并受单方（称为[机器人](#)牧民或机器人操作员）控制的机器人网络。僵尸网络是最著名的扩展机器人及其影响力的机制。

分支

代码存储库的一个包含区域。在存储库中创建的第一个分支是主分支。您可以从现有分支创建新分支，然后在新分支中开发功能或修复错误。为构建功能而创建的分支通常称为功能分支。当功能可以发布时，将功能分支合并回主分支。有关更多信息，请参阅[关于分支](#)（GitHub 文档）。

破碎的玻璃通道

在特殊情况下，通过批准的流程，用户 AWS 账户 可以快速访问他们通常没有访问权限的内容。有关更多信息，请参阅 Well [-Architected 指南](#) 中的“[实施破碎玻璃程序](#)”指示 AWS 器。

棕地策略

您环境中的现有基础设施。在为系统架构采用棕地策略时，您需要围绕当前系统和基础设施的限制来设计架构。如果您正在扩展现有基础设施，则可以将棕地策略和[全新](#)策略混合。

缓冲区缓存

存储最常访问的数据的内存区域。

业务能力

企业如何创造价值（例如，销售、客户服务或营销）。微服务架构和开发决策可以由业务能力驱动。有关更多信息，请参阅[在 AWS 上运行容器化微服务](#)白皮书中的[围绕业务能力进行组织](#)部分。

业务连续性计划（BCP）

一项计划，旨在应对大规模迁移等破坏性事件对运营的潜在影响，并使企业能够快速恢复运营。

C

CAF

参见[AWS 云采用框架](#)。

金丝雀部署

向最终用户缓慢而渐进地发布版本。当你有信心时，你可以部署新版本并全部替换当前版本。

CCoE

参见[云卓越中心](#)。

CDC

请参阅[变更数据捕获](#)。

更改数据捕获 (CDC)

跟踪数据来源 (如数据库表) 的更改并记录有关更改的元数据的过程。您可以将 CDC 用于各种目的, 例如审计或复制目标系统中的更改以保持同步。

混沌工程

故意引入故障或破坏性事件来测试系统的弹性。您可以使用 [AWS Fault Injection Service \(AWS FIS\)](#) 来执行实验, 对您的 AWS 工作负载施加压力并评估其响应。

CI/CD

查看[持续集成和持续交付](#)。

分类

一种有助于生成预测的分类流程。分类问题的 ML 模型预测离散值。离散值始终彼此不同。例如, 一个模型可能需要评估图像中是否有汽车。

客户端加密

在目标 AWS 服务 收到数据之前, 对数据进行本地加密。

云卓越中心 (CCoE)

一个多学科团队, 负责推动整个组织的云采用工作, 包括开发云最佳实践、调动资源、制定迁移时间表、领导组织完成大规模转型。有关更多信息, 请参阅 AWS Cloud 企业战略博客上的 [CCoE 帖子](#)。

云计算

通常用于远程数据存储和 IoT 设备管理的云技术。云计算通常与[边缘计算](#)技术相关。

云运营模型

在 IT 组织中, 一种用于构建、完善和优化一个或多个云环境的运营模型。有关更多信息, 请参阅[构建您的云运营模型](#)。

云采用阶段

组织迁移到以下阶段时通常会经历四个阶段 AWS Cloud：

- 项目 - 出于概念验证和学习目的，开展一些与云相关的项目
- 基础 — 进行基础投资以扩大云采用率（例如，创建着陆区、定义 CCo E、建立运营模型）
- 迁移 - 迁移单个应用程序
- 重塑 - 优化产品和服务，在云中创新

Stephen Orban在 AWS Cloud 企业战略博客的博客文章[《云优先之旅和采用阶段》](#)中定义了这些阶段。有关它们与 AWS 迁移策略的关系的信息，请参阅[迁移准备指南](#)。

CMDB

参见[配置管理数据库](#)。

代码存储库

通过版本控制过程存储和更新源代码和其他资产（如文档、示例和脚本）的位置。常见的云存储库包括GitHub或Bitbucket Cloud。每个版本的代码都称为一个分支。在微服务结构中，每个存储库都专门用于一个功能。单个 CI/CD 管道可以使用多个存储库。

冷缓存

一种空的、填充不足或包含过时或不相关数据的缓冲区缓存。这会影响性能，因为数据库实例必须从主内存或磁盘读取，这比从缓冲区缓存读取要慢。

冷数据

很少访问的数据，且通常是历史数据。查询此类数据时，通常可以接受慢速查询。将这些数据转移到性能较低且成本更低的存储层或类别可以降低成本。

计算机视觉 (CV)

[人工智能](#)领域，使用机器学习来分析和提取数字图像和视频等视觉格式的信息。例如，Amazon SageMaker AI 为 CV 提供了图像处理算法。

配置偏差

对于工作负载，配置会从预期状态发生变化。这可能会导致工作负载变得不合规，而且通常是渐进的，不是故意的。

配置管理数据库 (CMDB)

一种存储库，用于存储和管理有关数据库及其 IT 环境的信息，包括硬件和软件组件及其配置。您通常在迁移的产品组合发现和分析阶段使用来自 CMDB 的数据。

合规性包

一系列 AWS Config 规则和补救措施，您可以汇编这些规则和补救措施，以自定义合规性和安全性检查。您可以使用 YAML 模板将一致性包作为单个实体部署在 AWS 账户 和区域或整个组织中。有关更多信息，请参阅 AWS Config 文档中的[一致性包](#)。

持续集成和持续交付 (CI/CD)

自动执行软件发布过程的源代码、构建、测试、暂存和生产阶段的过程。CI/CD is commonly described as a pipeline. CI/CD可以帮助您实现流程自动化、提高生产力、提高代码质量和更快地交付。有关更多信息，请参阅[持续交付的优势](#)。CD 也可以表示持续部署。有关更多信息，请参阅[持续交付与持续部署](#)。

CV

参见[计算机视觉](#)。

D

静态数据

网络中静止的数据，例如存储中的数据。

数据分类

根据网络中数据的关键性和敏感性对其进行识别和分类的过程。它是任何网络安全风险管理策略的关键组成部分，因为它可以帮助您确定对数据的适当保护和保留控制。数据分类是 Well-Architected AWS d Framework 中安全支柱的一个组成部分。有关详细信息，请参阅[数据分类](#)。

数据漂移

生产数据与用来训练机器学习模型的数据之间的有意义差异，或者输入数据随时间推移的有意义变化。数据漂移可能降低机器学习模型预测的整体质量、准确性和公平性。

传输中数据

在网络中主动移动的数据，例如在网络资源之间移动的数据。

数据网格

一种架构框架，可提供分布式、去中心化的数据所有权以及集中式管理和治理。

数据最少化

仅收集并处理绝对必要数据的原则。在中进行数据最小化 AWS Cloud 可以降低隐私风险、成本和分析碳足迹。

数据边界

AWS 环境中的一组预防性防护措施，可帮助确保只有可信身份才能访问来自预期网络的可信资源。有关更多信息，请参阅在[上构建数据边界](#)。AWS

数据预处理

将原始数据转换为 ML 模型易于解析的格式。预处理数据可能意味着删除某些列或行，并处理缺失、不一致或重复的值。

数据溯源

在数据的整个生命周期跟踪其来源和历史的过程，例如数据如何生成、传输和存储。

数据主体

正在收集和处理其数据的人。

数据仓库

一种支持商业智能（例如分析）的数据管理系统。数据仓库通常包含大量历史数据，通常用于查询和分析。

数据库定义语言（DDL）

在数据库中创建或修改表和对对象结构的语句或命令。

数据库操作语言（DML）

在数据库中修改（插入、更新和删除）信息的语句或命令。

DDL

参见[数据库定义语言](#)。

深度融合

组合多个深度学习模型进行预测。您可以使用深度融合来获得更准确的预测或估算预测中的不确定性。

深度学习

一个 ML 子字段使用多层人工神经网络来识别输入数据和感兴趣的目标变量之间的映射。

defense-in-depth

一种信息安全方法，经过深思熟虑，在整个计算机网络中分层实施一系列安全机制和控制措施，以保护网络及其中数据的机密性、完整性和可用性。当你采用这种策略时 AWS，你会在 AWS

Organizations 结构的不同层面添加多个控件来帮助保护资源。例如，一种 defense-in-depth 方法可以结合多因素身份验证、网络分段和加密。

委托管理员

在中 AWS Organizations，兼容的服务可以注册 AWS 成员帐户来管理组织的帐户并管理该服务的权限。此账户被称为该服务的委托管理员。有关更多信息和兼容服务列表，请参阅 AWS Organizations 文档中[使用 AWS Organizations 的服务](#)。

后

使应用程序、新功能或代码修复在目标环境中可用的过程。部署涉及在代码库中实现更改，然后在应用程序的环境中构建和运行该代码库。

开发环境

参见[环境](#)。

侦测性控制

一种安全控制，在事件发生后进行检测、记录日志和发出警报。这些控制是第二道防线，提醒您注意绕过现有预防性控制的安全事件。有关更多信息，请参阅在 AWS 上实施安全控制中的[侦测性控制](#)。

开发价值流映射 (DVSM)

用于识别对软件开发生命周期中的速度和质量产生不利影响的限制因素并确定其优先级的流程。DVSM 扩展了最初为精益生产实践设计的价值流映射流程。其重点关注在软件开发过程中创造和转移价值所需的步骤和团队。

数字孪生

真实世界系统的虚拟再现，如建筑物、工厂、工业设备或生产线。数字孪生支持预测性维护、远程监控和生产优化。

维度表

在[星型架构](#)中，一种较小的表，其中包含事实表中有关定量数据的数据属性。维度表属性通常是文本字段或行为类似于文本的离散数字。这些属性通常用于查询约束、筛选和结果集标注。

灾难

阻止工作负载或系统在其主要部署位置实现其业务目标的事件。这些事件可能是自然灾害、技术故障或人为操作的结果，例如无意的配置错误或恶意软件攻击。

灾难恢复 (DR)

您用来最大限度地减少[灾难](#)造成的停机时间和数据丢失的策略和流程。有关更多信息，请参阅 Well-Architected Framework AWS work 中的“[工作负载灾难恢复：云端 AWS 恢复](#)”。

DML

参见[数据库操作语言](#)。

领域驱动设计

一种开发复杂软件系统的方法，通过将其组件连接到每个组件所服务的不断发展的领域或核心业务目标。Eric Evans 在其著作领域驱动设计：软件核心复杂性应对之道 (Boston: Addison-Wesley Professional, 2003) 中介绍了这一概念。有关如何将领域驱动设计与 strangler fig 模式结合使用的信息，请参阅[使用容器和 Amazon API Gateway 逐步将原有的 Microsoft ASP.NET \(ASMX \) Web 服务现代化](#)。

DR

参见[灾难恢复](#)。

漂移检测

跟踪与基准配置的偏差。例如，您可以使用 AWS CloudFormation 来[检测系统资源中的偏差](#)，也可以使用 AWS Control Tower 来[检测着陆区中可能影响监管要求合规性的变化](#)。

DVSM

参见[开发价值流映射](#)。

E

EDA

参见[探索性数据分析](#)。

EDI

参见[电子数据交换](#)。

边缘计算

该技术可提高位于 IoT 网络边缘的智能设备的计算能力。与[云计算](#)相比，边缘计算可以减少通信延迟并缩短响应时间。

电子数据交换 (EDI)

组织之间自动交换业务文档。有关更多信息，请参阅[什么是电子数据交换](#)。

加密

一种将人类可读的纯文本数据转换为密文的计算过程。

加密密钥

由加密算法生成的随机位的加密字符串。密钥的长度可能有所不同，而且每个密钥都设计为不可预测且唯一。

字节顺序

字节在计算机内存中的存储顺序。大端序系统先存储最高有效字节。小端序系统先存储最低有效字节。

端点

参见[服务端点](#)。

端点服务

一种可以在虚拟私有云 (VPC) 中托管，与其他用户共享的服务。您可以使用其他 AWS 账户 或 AWS Identity and Access Management (IAM) 委托人创建终端节点服务，AWS PrivateLink 并向其授予权限。这些账户或主体可通过创建接口 VPC 端点来私密地连接到您的端点服务。有关更多信息，请参阅 Amazon Virtual Private Cloud (Amazon VPC) 文档中的[创建端点服务](#)。

企业资源规划 (ERP)

一种自动化和管理企业关键业务流程（例如会计、[MES](#) 和项目管理）的系统。

信封加密

用另一个加密密钥对加密密钥进行加密的过程。有关更多信息，请参阅 AWS Key Management Service (AWS KMS) 文档中的[信封加密](#)。

环境

正在运行的应用程序的实例。以下是云计算中常见的环境类型：

- 开发环境 — 正在运行的应用程序的实例，只有负责维护应用程序的核心团队才能使用。开发环境用于测试更改，然后再将其提升到上层环境。这类环境有时称为测试环境。
- 下层环境 — 应用程序的所有开发环境，比如用于初始构建和测试的环境。

- 生产环境 — 最终用户可以访问的正在运行的应用程序的实例。在 CI/CD 管道中，生产环境是最后一个部署环境。
- 上层环境 — 除核心开发团队以外的用户可以访问的所有环境。这可能包括生产环境、预生产环境和用户验收测试环境。

epic

在敏捷方法学中，有助于组织工作和确定优先级的功能类别。epics 提供了对需求和实施任务的总体描述。例如，AWS CAF 安全史诗包括身份和访问管理、侦探控制、基础设施安全、数据保护和事件响应。有关 AWS 迁移策略中 epics 的更多信息，请参阅[计划实施指南](#)。

ERP

参见[企业资源规划](#)。

探索性数据分析 (EDA)

分析数据集以了解其主要特征的过程。您收集或汇总数据，并进行初步调查，以发现模式、检测异常并检查假定情况。EDA 通过计算汇总统计数据和创建数据可视化得以执行。

F

事实表

[星形架构](#)中的中心表。它存储有关业务运营的定量数据。通常，事实表包含两种类型的列：包含度量的列和包含维度表外键的列。

失败得很快

一种使用频繁和增量测试来缩短开发生命周期的理念。这是敏捷方法的关键部分。

故障隔离边界

在中 AWS Cloud，诸如可用区 AWS 区域、控制平面或数据平面之类的边界，它限制了故障的影响并有助于提高工作负载的弹性。有关更多信息，请参阅[AWS 故障隔离边界](#)。

功能分支

参见[分支](#)。

特征

您用来进行预测的输入数据。例如，在制造环境中，特征可能是定期从生产线捕获的图像。

特征重要性

特征对于模型预测的重要性。这通常表示为数值分数，可以通过各种技术进行计算，例如 Shapley 加法解释 (SHAP) 和积分梯度。有关更多信息，请参阅使用[机器学习模型的可解释性 AWS](#)。

功能转换

为 ML 流程优化数据，包括使用其他来源丰富数据、扩展值或从单个数据字段中提取多组信息。这使得 ML 模型能从数据中获益。例如，如果您将“2021-05-27 00:15:37”日期分解为“2021”、“五月”、“星期四”和“15”，则可以帮助学习与不同数据成分相关的算法学习精细模式。

少量提示

在要求[法学硕士](#)执行类似任务之前，向其提供少量示例，以演示该任务和所需的输出。这种技术是情境学习的应用，模型可以从提示中嵌入的示例 (镜头) 中学习。对于需要特定格式、推理或领域知识的任务，Few-shot 提示可能非常有效。另请参见[零镜头提示](#)。

FGAC

请参阅[精细的访问控制](#)。

精细访问控制 (FGAC)

使用多个条件允许或拒绝访问请求。

快闪迁移

一种数据库迁移方法，它使用连续的数据复制，通过[更改数据捕获](#)在尽可能短的时间内迁移数据，而不是使用分阶段的方法。目标是将停机时间降至最低。

FM

参见[基础模型](#)。

基础模型 (FM)

一个大型深度学习神经网络，一直在广义和未标记数据的大量数据集上进行训练。FMs 能够执行各种各样的一般任务，例如理解语言、生成文本和图像以及用自然语言进行对话。有关更多信息，请参阅[什么是基础模型](#)。

G

生成式人工智能

[人工智能](#)模型的子集，这些模型已经过大量数据训练，可以使用简单的文本提示来创建新的内容和工件，例如图像、视频、文本和音频。有关更多信息，请参阅[什么是生成式 AI](#)。

地理封锁

请参阅[地理限制](#)。

地理限制 (地理阻止)

在 Amazon 中 CloudFront，一种阻止特定国家/地区的用户访问内容分发的选项。您可以使用允许列表或阻止列表来指定已批准和已禁止的国家/地区。有关更多信息，请参阅 CloudFront 文档[中的限制内容的地理分布](#)。

GitFlow 工作流程

一种方法，在这种方法中，下层和上层环境在源代码存储库中使用不同的分支。Gitflow 工作流程被认为是传统的，而[基于主干的工作流程](#)是现代的首选方法。

金色影像

系统或软件的快照，用作部署该系统或软件的新实例的模板。例如，在制造业中，黄金映像可用于在多个设备上配置软件，并有助于提高设备制造运营的速度、可扩展性和生产力。

全新策略

在新环境中缺少现有基础设施。在对系统架构采用全新策略时，您可以选择所有新技术，而不受对现有基础设施 (也称为[棕地](#)) 兼容性的限制。如果您正在扩展现有基础设施，则可以将棕地策略和全新策略混合。

防护机制

一项高级规则，可帮助管理各组织单位的资源、策略和合规性 (OUs)。预防性防护机制会执行策略以确保符合合规性标准。它们是使用服务控制策略和 IAM 权限边界实现的。侦测性防护机制会检测策略违规和合规性问题，并生成警报以进行修复。它们通过使用 AWS Config、Amazon、AWS Security Hub GuardDuty AWS Trusted Advisor、Amazon Inspector 和自定义 AWS Lambda 支票来实现。

H

HA

参见[高可用性](#)。

异构数据库迁移

将源数据库迁移到使用不同数据库引擎的目标数据库 (例如，从 Oracle 迁移到 Amazon Aurora)。异构迁移通常是重新架构工作的一部分，而转换架构可能是一项复杂的任务。[AWS 提供了 AWS SCT](#) 来帮助实现架构转换。

高可用性 (HA)

在遇到挑战或灾难时，工作负载无需干预即可连续运行的能力。HA 系统旨在自动进行故障转移、持续提供良好性能，并以最小的性能影响处理不同负载和故障。

历史数据库现代化

一种用于实现运营技术 (OT) 系统现代化和升级以更好满足制造业需求的方法。历史数据库是一种用于收集和存储工厂中各种来源数据的数据库。

抵制数据

从用于训练[机器学习](#)模型的数据集中扣留的一部分带有标签的历史数据。通过将模型预测与抵制数据进行比较，您可以使用抵制数据来评估模型性能。

同构数据库迁移

将源数据库迁移到共享同一数据库引擎的目标数据库（例如，从 Microsoft SQL Server 迁移到 Amazon RDS for SQL Server）。同构迁移通常是更换主机或更换平台工作的一部分。您可以使用本机数据库实用程序来迁移架构。

热数据

经常访问的数据，例如实时数据或近期的转化数据。这些数据通常需要高性能存储层或存储类别才能提供快速的查询响应。

修补程序

针对生产环境中关键问题的紧急修复。由于其紧迫性，修补程序通常是在典型的 DevOps 发布工作流程之外进行的。

hypercure 周期

割接之后，迁移团队立即管理和监控云中迁移的应用程序以解决任何问题的时间段。通常，这个周期持续 1-4 天。在 hypercure 周期结束时，迁移团队通常会将应用程序的责任移交给云运营团队。

我

IaC

参见[基础设施即代码](#)。

基于身份的策略

附加到一个或多个 IAM 委托人的策略，用于定义他们在 AWS Cloud 环境中的权限。

空闲应用程序

90 天内平均 CPU 和内存使用率在 5% 到 20% 之间的应用程序。在迁移项目中，通常会停用这些应用程序或将其保留在本地。

IIoT

参见[工业物联网](#)。

不可变的基础架构

一种为生产工作负载部署新基础架构，而不是更新、修补或修改现有基础架构的模型。[不可变基础架构本质上比可变基础架构更一致、更可靠、更可预测](#)。有关更多信息，请参阅 Well-Architected Framework 中的[使用不可变基础架构 AWS 部署](#)最佳实践。

入站 (入口) VPC

在 AWS 多账户架构中，一种接受、检查和路由来自应用程序外部的网络连接的 VPC。[AWS 安全参考架构](#)建议设置您的网络帐户，包括入站、出站和检查，VPCs 以保护您的应用程序与更广泛的互联网之间的双向接口。

增量迁移

一种割接策略，在这种策略中，您可以将应用程序分成小部分进行迁移，而不是一次性完整割接。例如，您最初可能只将几个微服务或用户迁移到新系统。在确认一切正常后，您可以逐步迁移其他微服务或用户，直到停用遗留系统。这种策略降低了大规模迁移带来的风险。

工业 4.0

该术语由[克劳斯·施瓦布 \(Klaus Schwab \)](#)于2016年推出，指的是通过连接、实时数据、自动化、分析和人工智能/机器学习的进步实现制造流程的现代化。

基础设施

应用程序环境中包含的所有资源和资产。

基础设施即代码 (IaC)

通过一组配置文件预置和管理应用程序基础设施的过程。IaC 旨在帮助您集中管理基础设施、实现资源标准化和快速扩展，使新环境具有可重复性、可靠性和一致性。

工业物联网 (IIoT)

在工业领域使用联网的传感器和设备，例如制造业、能源、汽车、医疗保健、生命科学和农业。有关更多信息，请参阅[制定工业物联网 \(IIoT\) 数字化转型战略](#)。

检查 VPC

在 AWS 多账户架构中，一种集中式 VPC，用于管理对 VPCs（相同或不同 AWS 区域）、互联网和本地网络之间的网络流量的检查。[AWS 安全参考架构](#)建议设置您的网络帐户，包括入站、出站和检查，VPCs 以保护您的应用程序与更广泛的互联网之间的双向接口。

物联网 (IoT)

由带有嵌入式传感器或处理器的连接物理对象组成的网络，这些传感器或处理器通过互联网或本地通信网络与其他设备和系统进行通信。有关更多信息，请参阅[什么是 IoT ?](#)

可解释性

它是机器学习模型的一种特征，描述了人类可以理解模型的预测如何取决于其输入的程度。有关更多信息，请参阅使用[机器学习模型的可解释性 AWS](#)。

IoT

参见[物联网](#)。

IT 信息库 (ITIL)

提供 IT 服务并使这些服务符合业务要求的一套最佳实践。ITIL 是 ITSM 的基础。

IT 服务管理 (ITSM)

为组织设计、实施、管理和支持 IT 服务的相关活动。有关将云运营与 ITSM 工具集成的信息，请参阅[运营集成指南](#)。

ITIL

请参阅[IT 信息库](#)。

ITSM

请参阅[IT 服务管理](#)。

L

基于标签的访问控制 (LBAC)

强制访问控制 (MAC) 的一种实施方式，其中明确为用户和数据本身分配了安全标签值。用户安全标签和数据安全标签之间的交集决定了用户可以看到哪些行和列。

登录区

landing zone 是一个架构精良的多账户 AWS 环境，具有可扩展性和安全性。这是一个起点，您的组织可以从这里放心地在安全和基础设施环境中快速启动和部署工作负载和应用程序。有关登录区的更多信息，请参阅[设置安全且可扩展的多账户 AWS 环境](#)。

大型语言模型 (LLM)

一种基于大量数据进行预训练的深度学习 [AI](#) 模型。法学硕士可以执行多项任务，例如回答问题、总结文档、将文本翻译成其他语言以及完成句子。有关更多信息，请参阅[什么是 LLMs](#)。

大规模迁移

迁移 300 台或更多服务器。

LBAC

请参阅[基于标签的访问控制](#)。

最低权限

授予执行任务所需的最低权限的最佳安全实践。有关更多信息，请参阅 IAM 文档中的[应用最低权限许可](#)。

直接迁移

见 [7 R](#)。

小端序系统

一个先存储最低有效字节的系统。另请参见字[节顺序](#)。

LLM

参见[大型语言模型](#)。

下层环境

参见[环境](#)。

M

机器学习 (ML)

一种使用算法和技术进行模式识别和学习的人工智能。ML 对记录的数据 (例如物联网 (IoT) 数据) 进行分析和学习，以生成基于模式的统计模型。有关更多信息，请参阅[机器学习](#)。

主分支

参见[分支](#)。

恶意软件

旨在危害计算机安全或隐私的软件。恶意软件可能会破坏计算机系统、泄露敏感信息或获得未经授权的访问。恶意软件的示例包括病毒、蠕虫、勒索软件、特洛伊木马、间谍软件和键盘记录器。

托管服务

AWS 服务 它 AWS 运行基础设施层、操作系统和平台，您可以访问端点来存储和检索数据。亚马逊简单存储服务 (Amazon S3) Service 和 Amazon DynamoDB 就是托管服务的示例。这些服务也称为抽象服务。

制造执行系统 (MES)

一种软件系统，用于跟踪、监控、记录和控制将原材料转化为成品的生产过程。

MAP

参见[迁移加速计划](#)。

机制

一个完整的过程，在此过程中，您可以创建工具，推动工具的采用，然后检查结果以进行调整。机制是一种在运行过程中自我增强和改进的循环。有关更多信息，请参阅在 Well-Architect AWS ed 框架中[构建机制](#)。

成员账户

AWS 账户 除属于组织中的管理账户之外的所有账户 AWS Organizations。一个账户一次只能是一个组织的成员。

MES

参见[制造执行系统](#)。

消息队列遥测传输 (MQTT)

[一种基于发布/订阅模式的轻量级 machine-to-machine \(M2M\) 通信协议，适用于资源受限的物联网设备。](#)

微服务

一种小型的独立服务，通过明确的定义进行通信 APIs，通常由小型的独立团队拥有。例如，保险系统可能包括映射到业务能力（如销售或营销）或子域（如购买、理赔或分析）的微服务。微服务

的好处包括敏捷、灵活扩展、易于部署、可重复使用的代码和恢复能力。有关更多信息，请参阅[使用 AWS 无服务器服务集成微服务](#)。

微服务架构

一种使用独立组件构建应用程序的方法，这些组件将每个应用程序进程作为微服务运行。这些微服务使用轻量级通过定义明确的接口进行通信。APIs 该架构中的每个微服务都可以更新、部署和扩展，以满足对应用程序特定功能的需求。有关更多信息，请参阅[在上实现微服务。AWS](#)

迁移加速计划 (MAP)

AWS 该计划提供咨询支持、培训和服务，以帮助组织为迁移到云奠定坚实的运营基础，并帮助抵消迁移的初始成本。MAP 提供了一种以系统的方式执行遗留迁移的迁移方法，以及一套用于自动执行和加速常见迁移场景的工具。

大规模迁移

将大部分应用程序组合分波迁移到云中的过程，在每一波中以更快的速度迁移更多应用程序。本阶段使用从早期阶段获得的最佳实践和经验教训，实施由团队、工具和流程组成的迁移工厂，通过自动化和敏捷交付简化工作负载的迁移。这是 [AWS 迁移策略](#) 的第三阶段。

迁移工厂

跨职能团队，通过自动化、敏捷的方法简化工作负载迁移。迁移工厂团队通常包括运营、业务分析师和所有者、迁移工程师、开发 DevOps 人员和冲刺专业人员。20% 到 50% 的企业应用程序组合由可通过工厂方法优化的重复模式组成。有关更多信息，请参阅本内容集中[有关迁移工厂的讨论](#)和[云迁移工厂](#)指南。

迁移元数据

有关完成迁移所需的应用程序和服务器信息。每种迁移模式都需要一套不同的迁移元数据。迁移元数据的示例包括目标子网、安全组和 AWS 账户。

迁移模式

一种可重复的迁移任务，详细列出了迁移策略、迁移目标以及所使用的迁移应用程序或服务。示例：EC2 使用 AWS 应用程序迁移服务重新托管向 Amazon 的迁移。

迁移组合评测 (MPA)

一种在线工具，可提供信息，用于验证迁移到的业务案例。AWS Cloud MPA 提供了详细的组合评测（服务器规模调整、定价、TCO 比较、迁移成本分析）以及迁移计划（应用程序数据分析和数据收集、应用程序分组、迁移优先级排序和波次规划）。所有 AWS 顾问和 APN 合作伙伴顾问均可免费使用 [MPA 工具](#)（需要登录）。

迁移准备情况评测 (MRA)

使用 AWS CAF 深入了解组织的云就绪状态、确定优势和劣势以及制定行动计划以缩小已发现差距的过程。有关更多信息，请参阅[迁移准备指南](#)。MRA 是 [AWS 迁移策略](#) 的第一阶段。

迁移策略

用于将工作负载迁移到的方法 AWS Cloud。有关更多信息，请参阅此词汇表中的 [7 R](#) 条目和[动员组织以加快大规模迁移](#)。

ML

参见[机器学习](#)。

现代化

将过时的（原有的或单体）应用程序及其基础设施转变为云中敏捷、弹性和高度可用的系统，以降低成本、提高效率和利用创新。有关更多信息，请参阅[中的应用程序现代化策略](#)。AWS Cloud

现代化准备情况评估

一种评估方式，有助于确定组织应用程序的现代化准备情况；确定收益、风险和依赖关系；确定组织能够在多大程度上支持这些应用程序的未来状态。评估结果是目标架构的蓝图、详细说明现代化进程发展阶段和里程碑的路线图以及解决已发现差距的行动计划。有关更多信息，请参阅[中的评估应用程序的现代化准备情况](#) AWS Cloud。

单体应用程序 (单体式)

作为具有紧密耦合进程的单个服务运行的应用程序。单体应用程序有几个缺点。如果某个应用程序功能的需求激增，则必须扩展整个架构。随着代码库的增长，添加或改进单体应用程序的功能也会变得更加复杂。若要解决这些问题，可以使用微服务架构。有关更多信息，请参阅[将单体分解为微服务](#)。

MPA

参见[迁移组合评估](#)。

MQTT

请参阅[消息队列遥测传输](#)。

多分类器

一种帮助为多个类别生成预测（预测两个以上结果之一）的过程。例如，ML 模型可能会询问“这个产品是书、汽车还是手机？”或“此客户最感兴趣什么类别的产品？”

可变基础架构

一种用于更新和修改现有生产工作负载基础架构的模型。为了提高一致性、可靠性和可预测性，Well-Architect AWS ed Framework 建议使用[不可变基础设施](#)作为最佳实践。

O

OAC

请参阅[源站访问控制](#)。

OAI

参见[源访问身份](#)。

OCM

参见[组织变更管理](#)。

离线迁移

一种迁移方法，在这种方法中，源工作负载会在迁移过程中停止运行。这种方法会延长停机时间，通常用于小型非关键工作负载。

OI

参见[运营集成](#)。

OLA

参见[运营层协议](#)。

在线迁移

一种迁移方法，在这种方法中，源工作负载无需离线即可复制到目标系统。在迁移过程中，连接工作负载的应用程序可以继续运行。这种方法的停机时间为零或最短，通常用于关键生产工作负载。

OPC-UA

参见[开放流程通信-统一架构](#)。

开放流程通信-统一架构 (OPC-UA)

一种用于工业自动化的 machine-to-machine (M2M) 通信协议。OPC-UA 提供了数据加密、身份验证和授权方案的互操作性标准。

运营级别协议 (OLA)

一项协议，阐明了 IT 职能部门承诺相互交付的内容，以支持服务水平协议 (SLA)。

运营准备情况审查 (ORR)

一份问题清单和相关的最佳实践，可帮助您理解、评估、预防或缩小事件和可能的故障的范围。有关更多信息，请参阅 Well-Architected AWS d Frame [work 中的运营准备情况评估 \(ORR\)](#)。

操作技术 (OT)

与物理环境配合使用以控制工业运营、设备和基础设施的硬件和软件系统。在制造业中，OT 和信息技术 (IT) 系统的集成是[工业 4.0](#) 转型的重点。

运营整合 (OI)

在云中实现运营现代化的过程，包括就绪计划、自动化和集成。有关更多信息，请参阅[运营整合指南](#)。

组织跟踪

由此创建的跟踪 AWS CloudTrail，用于记录组织 AWS 账户 中所有人的所有事件 AWS Organizations。该跟踪是在每个 AWS 账户 中创建的，属于组织的一部分，并跟踪每个账户的活动。有关更多信息，请参阅 CloudTrail文档中的[为组织创建跟踪](#)。

组织变革管理 (OCM)

一个从人员、文化和领导力角度管理重大、颠覆性业务转型的框架。OCM 通过加快变革采用、解决过渡问题以及推动文化和组织变革，帮助组织为新系统和战略做好准备和过渡。在 AWS 迁移策略中，该框架被称为人员加速，因为云采用项目需要变更的速度。有关更多信息，请参阅[OCM 指南](#)。

来源访问控制 (OAC)

在中 CloudFront，一个增强的选项，用于限制访问以保护您的亚马逊简单存储服务 (Amazon S3) 内容。OAC 全部支持所有 S3 存储桶 AWS 区域、使用 AWS KMS (SSE-KMS) 进行服务器端加密，以及对 S3 存储桶的动态PUT和DELETE请求。

来源访问身份 (OAI)

在中 CloudFront，一个用于限制访问权限以保护您的 Amazon S3 内容的选项。当您使用 OAI 时，CloudFront 会创建一个 Amazon S3 可以对其进行身份验证的委托人。经过身份验证的委托人只能通过特定 CloudFront 分配访问 S3 存储桶中的内容。另请参阅[OAC](#)，其中提供了更精细和增强的访问控制。

ORR

参见[运营准备情况审查](#)。

OT

参见[运营技术](#)。

出站 (出口) VPC

在 AWS 多账户架构中，一种处理从应用程序内部启动的网络连接的 VPC。[AWS 安全参考架构](#)建议设置您的网络帐户，包括入站、出站和检查，VPCs 以保护您的应用程序与更广泛的互联网之间的双向接口。

P

权限边界

附加到 IAM 主体的 IAM 管理策略，用于设置用户或角色可以拥有的最大权限。有关更多信息，请参阅 IAM 文档中的[权限边界](#)。

个人身份信息 (PII)

直接查看其他相关数据或与之配对时可用于合理推断个人身份的信息。PII 的示例包括姓名、地址和联系信息。

PII

查看[个人身份信息](#)。

playbook

一套预定义的步骤，用于捕获与迁移相关的工作，例如在云中交付核心运营功能。playbook 可以采用脚本、自动化运行手册的形式，也可以是操作现代化环境所需的流程或步骤的摘要。

PLC

参见[可编程逻辑控制器](#)。

PLM

参见[产品生命周期管理](#)。

policy

一个对象，可以在中定义权限（参见[基于身份的策略](#)）、指定访问条件（参见[基于资源的策略](#)）或定义组织中所有账户的最大权限 AWS Organizations（参见[服务控制策略](#)）。

多语言持久性

根据数据访问模式和其他要求，独立选择微服务的数据存储技术。如果您的微服务采用相同的数据存储技术，它们可能会遇到实现难题或性能不佳。如果微服务使用最适合其需求的数据存储，则可以更轻松地实现微服务，并获得更好的性能和可扩展性。有关更多信息，请参阅[在微服务中实现数据持久性](#)。

组合评测

一个发现、分析和确定应用程序组合优先级以规划迁移的过程。有关更多信息，请参阅[评估迁移准备情况](#)。

谓词

返回true或的查询条件false，通常位于子WHERE句中。

谓词下推

一种数据库查询优化技术，可在传输前筛选查询中的数据。这减少了必须从关系数据库检索和处理的数据量，并提高了查询性能。

预防性控制

一种安全控制，旨在防止事件发生。这些控制是第一道防线，帮助防止未经授权的访问或对网络的意外更改。有关更多信息，请参阅在 AWS 上实施安全控制中的[预防性控制](#)。

主体

中 AWS 可以执行操作和访问资源的实体。此实体通常是 IAM 角色的根用户或用户。AWS 账户有关更多信息，请参阅 IAM 文档中[角色术语和概念](#)中的主体。

通过设计保护隐私

一种在整个开发过程中考虑隐私的系统工程方法。

私有托管区

一个容器，其中包含有关您希望 Amazon Route 53 如何响应针对一个或多个 VPCs 域名及其子域名的 DNS 查询的信息。有关更多信息，请参阅 Route 53 文档中的[私有托管区的使用](#)。

主动控制

一种[安全控制](#)措施，旨在防止部署不合规的资源。这些控件会在资源配置之前对其进行扫描。如果资源与控件不兼容，则不会对其进行配置。有关更多信息，请参阅 AWS Control Tower 文档中的[控制参考指南](#)，并参见在上实施安全[控制中的主动](#)控制 AWS。

产品生命周期管理 (PLM)

在产品的整个生命周期中，从设计、开发和上市，到成长和成熟，再到衰落和移除，对产品进行数据和流程的管理。

生产环境

参见[环境](#)。

可编程逻辑控制器 (PLC)

在制造业中，一种高度可靠、适应性强的计算机，用于监控机器并实现制造过程自动化。

提示链接

使用一个 [LLM](#) 提示的输出作为下一个提示的输入，以生成更好的响应。该技术用于将复杂的任务分解为子任务，或者迭代地完善或扩展初步响应。它有助于提高模型响应的准确性和相关性，并允许获得更精细的个性化结果。

假名化

用占位符值替换数据集中个人标识符的过程。假名化可以帮助保护个人隐私。假名化数据仍被视为个人数据。

publish/subscribe (pub/sub)

一种支持微服务间异步通信的模式，以提高可扩展性和响应能力。例如，在基于微服务的 [MES](#) 中，微服务可以将事件消息发布到其他微服务可以订阅的频道。系统可以在不更改发布服务的情况下添加新的微服务。

Q

查询计划

一系列步骤，例如指令，用于访问 SQL 关系数据库系统中的数据。

查询计划回归

当数据库服务优化程序选择的最佳计划不如数据库环境发生特定变化之前时。这可能是由统计数据、约束、环境设置、查询参数绑定更改和数据库引擎更新造成的。

R

RACI 矩阵

参见 [“负责任、负责、咨询、知情” \(RACI \)](#)。

RAG

请参见[检索增强生成](#)。

勒索软件

一种恶意软件，旨在阻止对计算机系统或数据的访问，直到付款为止。

RASCI 矩阵

参见 [“负责任、负责、咨询、知情” \(RACI \)](#)。

RCAC

请参阅[行和列访问控制](#)。

只读副本

用于只读目的的数据库副本。您可以将查询路由到只读副本，以减轻主数据库的负载。

重新架构师

见 [7 R](#)。

恢复点目标 (RPO)

自上一个数据恢复点以来可接受的最长时间。这决定了从上一个恢复点到服务中断之间可接受的数据丢失情况。

恢复时间目标 (RTO)

服务中断和服务恢复之间可接受的最大延迟。

重构

见 [7 R](#)。

区域

地理区域内的 AWS 资源集合。每一个 AWS 区域 都相互隔离，彼此独立，以提供容错、稳定性和弹性。有关更多信息，请参阅[指定 AWS 区域 您的账户可以使用的账户](#)。

回归

一种预测数值的 ML 技术。例如，要解决“这套房子的售价是多少？”的问题 ML 模型可以使用线性回归模型，根据房屋的已知事实（如建筑面积）来预测房屋的销售价格。

重新托管

见 [7 R](#)。

版本

在部署过程中，推动生产环境变更的行为。

搬迁

见 [7 R](#)。

更换平台

见 [7 R](#)。

回购

见 [7 R](#)。

故障恢复能力

应用程序抵御中断或从中断中恢复的能力。在中规划弹性时，[高可用性](#)和[灾难恢复](#)是常见的考虑因素。AWS Cloud有关更多信息，请参阅[AWS Cloud 弹性](#)。

基于资源的策略

一种附加到资源的策略，例如 AmazonS3 存储桶、端点或加密密钥。此类策略指定了允许哪些主体访问、支持的操作以及必须满足的任何其他条件。

责任、问责、咨询和知情 (RACI) 矩阵

定义参与迁移活动和云运营的所有各方的角色和责任的矩阵。矩阵名称源自矩阵中定义的责任类型：负责 (R)、问责 (A)、咨询 (C) 和知情 (I)。支持 (S) 类型是可选的。如果包括支持，则该矩阵称为 RASCI 矩阵，如果将其排除在外，则称为 RACI 矩阵。

响应性控制

一种安全控制，旨在推动对不良事件或偏离安全基线的情况进行修复。有关更多信息，请参阅在 AWS 上实施安全控制中的[响应性控制](#)。

保留

见 [7 R](#)。

退休

见 [7 R](#)。

检索增强生成 (RAG)

一种[生成式人工智能](#)技术，其中[法学硕士](#)在生成响应之前引用其训练数据源之外的权威数据源。

例如，RAG 模型可以对组织的知识库或自定义数据执行语义搜索。有关更多信息，请参阅[什么是 RAG](#)。

轮换

定期更新[密钥](#)以使攻击者更难访问凭据的过程。

行列访问控制 (RCAC)

使用已定义访问规则的基本、灵活的 SQL 表达式。RCAC 由行权限和列掩码组成。

RPO

参见[恢复点目标](#)。

RTO

参见[恢复时间目标](#)。

运行手册

执行特定任务所需的一套手动或自动程序。它们通常是为了简化重复性操作或高错误率的程序而设计的。

S

SAML 2.0

许多身份提供商 (IdPs) 使用的开放标准。此功能支持联合单点登录 (SSO)，因此用户无需在 IAM 中为组织中的所有人创建用户即可登录 AWS Management Console 或调用 AWS API 操作。有关基于 SAML 2.0 的联合身份验证的更多信息，请参阅 IAM 文档中的[关于基于 SAML 2.0 的联合身份验证](#)。

SCADA

参见[监督控制和数据采集](#)。

SCP

参见[服务控制政策](#)。

secret

在中 AWS Secrets Manager，您以加密形式存储的机密或受限信息，例如密码或用户凭证。它由密钥值及其元数据组成。密钥值可以是二进制、单个字符串或多个字符串。有关更多信息，请参阅 [Secrets Manager 密钥中有什么？](#) 在 Secrets Manager 文档中。

安全性源于设计

一种在整个开发过程中考虑安全性的系统工程方法。

安全控制

一种技术或管理防护机制，可防止、检测或降低威胁行为体利用安全漏洞的能力。安全控制主要有四种类型：[预防性](#)、[侦测](#)、[响应式](#)和[主动式](#)。

安全加固

缩小攻击面，使其更能抵御攻击的过程。这可能包括删除不再需要的资源、实施授予最低权限的最佳安全实践或停用配置文件中不必要的功能等操作。

安全信息和事件管理 (SIEM) 系统

结合了安全信息管理 (SIM) 和安全事件管理 (SEM) 系统的工具和服务。SIEM 系统会收集、监控和分析来自服务器、网络、设备和其他来源的数据，以检测威胁和安全漏洞，并生成警报。

安全响应自动化

一种预定义和编程的操作，旨在自动响应或修复安全事件。这些自动化可作为[侦探](#)或[响应式](#)安全控制措施，帮助您实施 AWS 安全最佳实践。自动响应操作的示例包括修改 VPC 安全组、修补 Amazon EC2 实例或轮换证书。

服务器端加密

在目的地对数据进行加密，由接收方 AWS 服务 进行加密。

服务控制策略 (SCP)

一种策略，用于集中控制组织中所有账户的权限 AWS Organizations。SCPs 定义防护措施或限制管理员可以委托给用户或角色的操作。您可以使用 SCPs 允许列表或拒绝列表来指定允许或禁止哪些服务或操作。有关更多信息，请参阅 AWS Organizations 文档中的[服务控制策略](#)。

服务端点

的入口点的 URL AWS 服务。您可以使用端点，通过编程方式连接到目标服务。有关更多信息，请参阅 AWS 一般参考 中的 [AWS 服务 端点](#)。

服务水平协议 (SLA)

一份协议，阐明了 IT 团队承诺向客户交付的内容，比如服务正常运行时间和性能。

服务级别指示器 (SLI)

对服务性能方面的衡量，例如其错误率、可用性或吞吐量。

服务级别目标 (SLO)

代表服务运行状况的目标指标，由服务[级别指标](#)衡量。

责任共担模式

描述您在云安全与合规方面共同承担 AWS 的责任的模型。AWS 负责云的安全，而您则负责云中的安全。有关更多信息，请参阅[责任共担模式](#)。

SIEM

参见[安全信息和事件管理系统](#)。

单点故障 (SPOF)

应用程序的单个关键组件出现故障，可能会中断系统。

SLA

参见[服务级别协议](#)。

SLI

参见[服务级别指标](#)。

SLO

参见[服务级别目标](#)。

split-and-seed 模型

一种扩展和加速现代化项目的模式。随着新功能和产品发布的定义，核心团队会拆分以创建新的产品团队。这有助于扩展组织的能力和服务，提高开发人员的工作效率，支持快速创新。有关更多信息，请参阅[中的分阶段实现应用程序现代化的方法。AWS Cloud](#)

恶作剧

参见[单点故障](#)。

星型架构

一种数据库组织结构，它使用一个大型事实表来存储交易数据或测量数据，并使用一个或多个较小的维度表来存储数据属性。此结构专为在[数据仓库](#)中使用或用于商业智能目的而设计。

strangler fig 模式

一种通过逐步重写和替换系统功能直至可以停用原有的系统来实现单体系统现代化的方法。这种模式用无花果藤作为类比，这种藤蔓成长为一棵树，最终战胜并取代了宿主。该模式是由 [Martin Fowler](#) 提出的，作为重写单体系统时管理风险的一种方法。有关如何应用此模式的示例，请参阅[使用容器和 Amazon API Gateway 逐步将原有的 Microsoft ASP.NET \(ASMX \) Web 服务现代化](#)。

子网

您的 VPC 内的一个 IP 地址范围。子网必须位于单个可用区中。

监控和数据采集 (SCADA)

在制造业中，一种使用硬件和软件来监控有形资产和生产操作的系统。

对称加密

一种加密算法，它使用相同的密钥来加密和解密数据。

综合测试

以模拟用户交互的方式测试系统，以检测潜在问题或监控性能。您可以使用 [Amazon S CloudWatch ynthetic](#) 来创建这些测试。

系统提示符

一种向[法学硕士提供上下文、说明或指导方针](#)以指导其行为的技术。系统提示有助于设置上下文并制定与用户交互的规则。

T

tags

键值对，充当用于组织资源的元数据。AWS 标签可帮助您管理、识别、组织、搜索和筛选资源。有关更多信息，请参阅[标记您的 AWS 资源](#)。

目标变量

您在监督式 ML 中尝试预测的值。这也被称为结果变量。例如，在制造环境中，目标变量可能是产品缺陷。

任务列表

一种通过运行手册用于跟踪进度的工具。任务列表包含运行手册的概述和要完成的常规任务列表。对于每项常规任务，它包括预计所需时间、所有者和进度。

测试环境

参见[环境](#)。

训练

为您的 ML 模型提供学习数据。训练数据必须包含正确答案。学习算法在训练数据中查找将输入数据属性映射到目标（您希望预测的答案）的模式。然后输出捕获这些模式的 ML 模型。然后，您可以使用 ML 模型对不知道目标的新数据进行预测。

中转网关

一个网络传输中心，可用于将您的网络 VPCs 和本地网络互连。有关更多信息，请参阅 AWS Transit Gateway 文档中的[什么是公交网关](#)。

基于中继的工作流程

一种方法，开发人员在功能分支中本地构建和测试功能，然后将这些更改合并到主分支中。然后，按顺序将主分支构建到开发、预生产和生产环境。

可信访问权限

向您指定的服务授予权限，该服务可代表您在其账户中执行任务。AWS Organizations 当需要服务相关的角色时，受信任的服务会在每个账户中创建一个角色，为您执行管理任务。有关更多信息，请参阅 AWS Organizations 文档中的[AWS Organizations 与其他 AWS 服务一起使用](#)。

优化

更改训练过程的各个方面，以提高 ML 模型的准确性。例如，您可以通过生成标签集、添加标签，并在不同的设置下多次重复这些步骤来优化模型，从而训练 ML 模型。

双披萨团队

一个小 DevOps 团队，你可以用两个披萨来喂食。双披萨团队的规模可确保在软件开发过程中充分协作。

U

不确定性

这一概念指的是不精确、不完整或未知的信息，这些信息可能会破坏预测式 ML 模型的可靠性。不确定性有两种类型：认知不确定性是由有限的、不完整的数据造成的，而偶然不确定性是由数据中固有的噪声和随机性导致的。有关更多信息，请参阅[量化深度学习系统中的不确定性](#)指南。

无差别任务

也称为繁重工作，即创建和运行应用程序所必需的工作，但不能为最终用户提供直接价值或竞争优势。无差别任务的示例包括采购、维护和容量规划。

上层环境

参见[环境](#)。

V

vacuum 操作

一种数据库维护操作，包括在增量更新后进行清理，以回收存储空间并提高性能。

版本控制

跟踪更改的过程和工具，例如存储库中源代码的更改。

VPC 对等连接

两者之间的连接 VPCs，允许您使用私有 IP 地址路由流量。有关更多信息，请参阅 Amazon VPC 文档中的[什么是 VPC 对等连接](#)。

漏洞

损害系统安全的软件缺陷或硬件缺陷。

W

热缓存

一种包含经常访问的当前相关数据的缓冲区缓存。数据库实例可以从缓冲区缓存读取，这比从主内存或磁盘读取要快。

暖数据

不常访问的数据。查询此类数据时，通常可以接受中速查询。

窗口函数

一个 SQL 函数，用于对一组以某种方式与当前记录相关的行进行计算。窗口函数对于处理任务很有用，例如计算移动平均线或根据当前行的相对位置访问行的值。

工作负载

一系列资源和代码，它们可以提供商业价值，如面向客户的应用程序或后端过程。

工作流

迁移项目中负责一组特定任务的职能小组。每个工作流都是独立的，但支持项目中的其他工作流。例如，组合工作流负责确定应用程序的优先级、波次规划和收集迁移元数据。组合工作流将这些资产交付给迁移工作流，然后迁移服务器和应用程序。

蠕虫

参见[一次写入，多读](#)。

WQF

参见[AWS 工作负载资格框架](#)。

一次写入，多次读取 (WORM)

一种存储模型，它可以一次写入数据并防止数据被删除或修改。授权用户可以根据需要多次读取数据，但他们无法对其进行更改。这种数据存储基础架构被认为是[不可变的](#)。

Z

零日漏洞利用

一种利用未修补[漏洞](#)的攻击，通常是恶意软件。

零日漏洞

生产系统中不可避免的缺陷或漏洞。威胁主体可能利用这种类型的漏洞攻击系统。开发人员经常因攻击而意识到该漏洞。

零镜头提示

向[法学硕士](#)提供执行任务的说明，但没有示例（镜头）可以帮助指导任务。法学硕士必须使用其预先训练的知识来处理任务。零镜头提示的有效性取决于任务的复杂性和提示的质量。另请参阅[few-shot 提示](#)。

僵尸应用程序

平均 CPU 和内存使用率低于 5% 的应用程序。在迁移项目中，通常会停用这些应用程序。

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。