



为 InfluxDB 应用亚马逊 Timestream 的 AWS Well-Architected 框架

AWS 规范性指导



AWS 规范性指导: 为 InfluxDB 应用亚马逊 Timestream 的 AWS Well-Architected 框架

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	2
目标	2
卓越运营	3
使用 IaC 方法自动部署	3
频繁进行可逆的小规模更改	3
预测故障	4
从所有操作故障中吸取教训	4
使用日志记录功能来监控未经授权或异常的活动	5
安全性	6
实现数据安全	6
保护网络安全	6
实现身份验证和授权	7
可靠性	8
工作负载架构，包括服务配额和部署模式	8
部署模式	8
管理和扩展 InfluxDB 的时间流	9
性能效率	10
Influx 数据建模和查询优化	10
优化写入	12
成本优化	13
了解您的用例的要求和成本	13
在选择资源时要注意成本	13
在不超支的情况下进行扩展以满足业务需求	14
调整数据存储和传输的大小	14
可持续性	15
AWS 区域 选择	15
根据用户行为模式消耗资源	15
优化软件开发和架构模式	16
资源	17
参考信息	17
博客文章	17
文档历史记录	18
术语表	19

#	19
A	19
B	22
C	24
D	26
E	30
F	31
G	33
H	34
我	35
L	37
M	38
O	42
P	44
Q	46
R	46
S	49
T	52
U	53
V	54
W	54
Z	55
.....	lvi

在 InfluxDB 上应用适用于亚马逊 AWS Timestream 的 Well-Architected 框架

Balwanth Bobilli , Amazon Web Services

2025 年 1 月 ([文档历史记录](#))

您可以使用 Amazon [Timestream 在亚马逊](#)网络服务 (AWS) 上构建 time-series-based 解决方案。Amazon Timestream 为从低延迟查询到大规模数据摄取等工作负载提供完全托管、专门构建的时间序列数据库引擎。借助[适用于 InfluxDB 的 Amazon Timestream](#)，您可以在时间序列应用程序上运行开源 InfluxDB 数据库，例如实时警报和监控基础设施可靠性，响应时间 AWS 为毫秒。InfluxDB 的 Timestream 提供高达 99.9% 的可用性。

本指南为在规划 InfluxDB [AWS 部署的时间流时应用 Well-Architected Framework](#) 原则提供了规范性指导。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。它还为您提供评估架构和实施可扩展设计的一致方法。

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 设计支柱的信息。在上部署适用于 InfluxDB 的 Amazon Timestream 时，可以考虑使用这些最佳实践。AWS

Note

有些 AWS 服务并非全部可用 AWS 区域。有关区域可用性，请参阅 AWS 文档中的[服务终端节点和配额](#)页面，然后选择该服务的链接。

目标受众

本指南适用于为时序数据设计和实施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，执行优化的写入并实现精细的访问。
- 遵循有助于提高弹性、AWS 性和安全性的 Well-Architected 设计模式。
- 设计查询以获得最佳性能。
- 了解如何在生产环境中管理 InfluxDB 的 Timestream 实例时提高运营效率。

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以实现业务价值。卓越运营支柱包括支持开发和有效运行工作负载的能力，以及深入了解其运营的能力。

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。要努力实现这一目标，请遵循本节中描述的最佳实践。[使用适用于 InfluxDB 的 Amazon Timestream 的亚马逊 CloudWatch 指标、InfluxDB 原生指标终端节点 APIs，以及在工作负载偏离预期行为时做出响应的机制。](#)

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
- 韧性策略
- 事件管理
- 用于审计目的的日志和监控

使用 IaC 方法自动部署

使用 iaC 在 InfluxDB 的 Timestream 上自动部署的最佳实践包括以下内容：

- 尽可能使用 iaC 为 InfluxDB 部署 Timestream。为了实现一致的环境配置，请使用[AWS CloudFormation](#)模板或 [HashiCorp Terraform](#) 为您的实例创建所有必需的资源。[AWS Cloud Development Kit \(AWS CDK\)](#)
- 自动执行 InfluxDB 操作过程的时间流，例如调整实例大小。
- 使用标签向时间流中添加 InfluxDB 资源的元数据，并根据标签跟踪使用情况。有关更多信息，请参阅为 InfluxD [B 标记亚马逊 Timestream](#)。

频繁进行可逆的小规模更改

以下建议侧重于小的、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中，例如或。 GitHub GitLab不要在源代码管理中存储 AWS 凭据。

- 要求 IaC 部署才能使用持续集成和持续交付 (CI/CD) 服务，例如 [AWS CodeDeploy](#) 或 [AWS CodeBuild](#)。这些服务会在包含临时 InfluxDB 实例的非生产环境中编译、测试和部署代码，然后再影响您的生产 InfluxDB 实例。
- 在将基础设施和应用程序查询部署到生产环境之前，先在较低的环境中对其进行测试。这样可以最大限度地减少中断的可能性，并有助于确保它们在您的工作负载和规模下表现良好。

预测故障

自我修复的基础设施通过预测故障并尝试在没有干预的情况下解决任何问题来体现卓越运营。以下建议可帮助您使用适用于 InfluxDB 的 Timestream 实现这一成熟度：

- 使用指标监控您的内存、CPU 和存储使用情况。可将 CloudWatch 设置为在使用模式发生更改或接近部署容量时向您发送通知。这样，您就可以保持系统的性能和可用性。
- 当您接近资源限制时，可以扩展数据库实例。存储和内存中应含有一些缓冲区，以适应应用程序的意外增大需求。
- 如果您的数据库工作负载需要 I/O 超出预配置的范围，则故障转移或数据库故障后的恢复速度会很慢。增加 I/O capacity of a DB instance, migrate to a different DB instance that has higher I/O 容量。
- 如果您的客户端应用程序正在缓存数据库实例的 DNS 数据，请将 time-to-live (TTL) 值设置为小于 30 秒。数据库实例的底层 IP 地址在故障转移后可能会发生变化。长时间缓存 DNS 数据可能会导致连接失败。您的应用程序可能会尝试连接到不再使用的 IP 地址。
- 如果您的应用程序需要在完全 AWS 区域 中断后存活下来，请考虑设置复制或写入其他区域，作为灾难恢复 (DR) 计划的一部分。了解设置复制时的限制。有关复制的更多信息，请参阅 [InfluxDB 文档](#)。

从所有操作故障中吸取教训

自我修复基础架构是一项长期的工作，当出现罕见问题或响应效果不如你想要的有效时，你需要反复开发这个基础架构。要专注于实现自我修复基础架构，请采用以下做法：

- 从所有故障中吸取教训，推动改进。
- 在团队和组织内部分享经验教训。如果组织内的多个团队使用 Timestream for InfluxDB，请创建一个公共聊天室或用户组来分享经验教训和最佳实践。

使用日志记录功能来监控未经授权或异常的活动

要观察异常的性能和活动模式，请考虑以下做法：

- 启用 [日志传输](#)，将 InfluxDB 日志存储在亚马逊 [简单存储服务 \(Amazon S3\) Service](#) 中。InfluxDB 日志记录的信息可以帮助检查以下内容：
 - [数据平面 API 事件](#)
 - 响应时间
 - 压实细节
 - 系统遇到的任何严重错误或警告

查看日志中是否存在未经授权的访问或异常情况。总体而言，日志记录为故障排除提供了诊断信息。

- InfluxDB 的 Timestream 支持使用记录控制平面操作。AWS CloudTrail 有关更多信息，请参阅使用 [记录 InfluxDB API 调用的时间流](#)。AWS CloudTrail
- 你可以在 `timeStream/InfluxDB > < Namespace >` 中监控 `CPUUtilizationMemoryUtilization`、和 `DiskUtilization` 指标。CloudWatch

有关更多信息，请参阅 [InfluxDB 的时间流](#) 文档。

安全支柱

[安全支柱](#)可以帮助你了解在使用 Timestream for InfluxDB 时如何应用分担责任模型。

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

以下各节介绍如何为 InfluxDB 配置 Timestream 以满足您的安全性和合规性目标。您还将学习如何使用其他 AWS 服务方法来监控和保护 InfluxDB 资源的时间流。

实现数据安全

数据泄露和违规行为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。AWS [分担责任模型适用于 InfluxDB 的 Timestream 中的数据保护](#)。以下做法有助于保护您的客户数据免遭无意和恶意泄露：

- 请勿在实例名称、标签、参数组、AWS Identity and Access Management (IAM) 角色和其他元数据中包含机密信息。这些数据可能会出现在账单或诊断日志中。
- 使用加密来增强部署在云中的应用程序的数据保护。加密实例可帮助保护您的数据免遭未经授权访问底层存储，从而提供额外的数据保护层。InfluxDB 加密的时间流默认处于启用状态。
- APIs 尽可能使用参数化。

有关数据保护和加密的更多信息，请参阅 [InfluxDB 的 Timestream](#) 文档。

保护网络安全

您只能在上的虚拟私有云 (VPC) 中为 InfluxDB 实例创建 Timestream。AWS 要进一步保护您的实例，请执行以下操作：

- 通过执行以下任一操作来限制对实例终端节点的公共访问：
 - 在“创建 InfluxDB 数据库”页面上选择“不可公开访问”（默认选中）。
 - 将该 [PubliclyAccessible](#) 属性设置为 false in AWS CloudFormation (AWS::Timestream::InfluxDBInstance)。

当您选择不可公开访问或设置PubliclyAccessible为false，实例的终端节点只能在 VPC 内访问。这些终端节点通常可从与 InfluxDB 实例的 Timestream 运行在同一 VPC 中的[亚马逊弹性计算云 \(Amazon EC2\)](#) 实例进行访问。

- 使用安全组进一步保护您在 VPC 内访问 Timestream for InfluxDB 的网络访问权限。
- 用于 SSL/TLS 与 AWS 资源通信。InfluxDB 的时间流需要 TLS 1.2，我们推荐 TLS 1.3。
- 按照[创建并连接到 InfluxDB 实例的 Timestream 中的说明安全地连接到您的实例](#)。
- [通过创建接口 VPC 终端节点，在您的 VPC 和 Amazon Timestream 之间为 InfluxDB 控制平面 API 终端节点建立私有连接。](#)

有关安全性的更多信息，请参阅 [InfluxDB 的 Timestream](#) 文档。

实现身份验证和授权

使用 IAM 证书控制 InfluxDB 实例在 Timestream 上执行的管理操作。当你使用 IAM 凭证连接到 Timestream for InfluxDB 时，你的 IAM 角色必须具有 IAM 策略，这些策略授予执行 Timestream 对 InfluxDB 管理操作所需的权限。确保遵循最低权限原则，仅授予完成任务所需的权限。有关更多信息，请参阅适用于 InfluxDB 的 [Amazon Timestream 的身份和访问管理](#)。

你可以在 IAM 策略中使用 [Amazon Timestream 进行 InfluxDB 操作](#)，以控制谁可以管理你的 InfluxDB 实例的时间流。

[要为存储在 Amazon Timestream for InfluxDB 实例中的数据提供精细的访问控制，请向用户提供 InfluxDB API 令牌。](#)

为了与其他人互动 AWS 服务，适用于 InfluxDB 的 Amazon Timestream 使用 IAM 服务相关角色。服务相关角色是一种独特的 IAM 角色，直接链接到 InfluxDB 的 Timestream。服务相关角色由 Timestream 为 InfluxDB 预定义，它们包括该服务代表您调用其他 AWS 服务角色所需的所有权限。有关更多信息，请参阅[使用适用于 InfluxDB 的 Amazon Timestream 的服务相关角色](#)。

可靠性支柱

[可靠性支柱](#)包括工作负载在预期时正确、一致地执行其预期功能的能力。这包括能够在工作负载的整个生命周期中对其进行操作和测试。

配置可靠的工作负载始于软件和基础架构的前期设计决策。您的架构选择将影响所有 Well-Architected 支柱的工作负载行为。要实现可靠性，必须遵循特定的模式。

可靠性支柱重点关注以下关键领域：

- 工作负载架构，包括服务配额和部署模式
- 管理和扩展 InfluxDB 实例

工作负载架构，包括服务配额和部署模式

每个 AWS 账户 都有各自提供的资源配额 AWS 区域。例如，无论实例大小如何，每个区域都有 [InfluxDB实例的时间流配额](#)。达到某个区域中的最大实例数后，创建实例的其他调用会失败，但会出现异常。InfluxDB 实例存储卷的时间流可以增加至所有支持的最大大小为 16 太字节 (TiBs)。AWS 区域

部署模式

要为 InfluxDB 实例的 Timestream 提供[高可用性和](#)故障转移支持，您可以将多可用区部署与单个备用数据库实例一起使用。这种类型的部署称为多可用区数据库实例部署。适用于 InfluxDB 的 Amazon Timestream 使用亚马逊故障转移技术。在多可用区数据库实例部署中，Amazon Timestream 会自动在不同可用区中配置和维护一个同步备用副本。为了提供数据冗余，主数据库实例跨可用区同步复制到备用副本。

运行具有高可用性的数据库实例可以在数据库实例故障或可用区中断期间提供可用性。如果您的数据库实例因基础设施缺陷而导致计划外中断，则适用于 InfluxDB 的 Amazon Timestream 会自动切换到备用副本。完成失效转移所用的时间取决于在主数据库实例变为不可用时的数据库活动和其他条件。

故障转移时间通常为 60–120 秒。但是，具有高基数数据的大型事务或具有预热要求的漫长恢复过程可能会增加故障转移时间。故障转移完成后，可能需要更多时间才能让 Timestream 控制台反映新的可用区。

如果您的应用程序在完全 AWS 区域 中断期间必须保持可用，请考虑在灾难恢复 (DR) 计划中设置复制或写入其他区域。但是，在设置复制之前，请务必了解这些限制。有关更多信息，请参阅 [InfluxDB 文档](#)。

适用于 InfluxDB 的 Amazon Timestream 会定期进行内部备份并将其保留 24 小时，以支持可用性和持久性。快照在删除期间拍摄，并保留 30 天以支持还原操作。要访问或使用这些内容，请在上创建案例[AWS 支持](#)。

管理和扩展 InfluxDB 的时间流

InfluxDB 的 Timestream 支持非常适合在开源 InfluxDB 数据库中运行内存密集型工作负载的实例类。不同的 [db.influx 实例类](#)对 v CPUs、内存、存储和网络带宽都有限制。要选择符合应用程序写入和查询延迟要求的实例类，请在测试期间遵守 Amazon CloudWatch CPUUtilization 和DiskUtilization指标。MemoryUtilization您可以根据工作负载要求向[上和向下扩展实例](#)。InfluxDB 的 Timestream 提供了多个存储层，这些存储层已预先配置为不同类型的工作负载所需的最佳 IOPS 和吞吐量。根据您的要求选择最适合您的工作负载的内容。

如果您的扩展需求在可预测的时间发生变化，则可以使用[AWS Lambda 函数](#)或自定义调度程序并运行 API 或 SDK 以在缓冲时间内向上和向下扩展。

你可以使用参数组中的参数在 InfluxDB 的 Timestream 中管理你的 InfluxDB 配置。参数组充当应用于一个或多个数据库实例的 InfluxDB 配置选项的容器。修改参数组中的参数时，要了解静态参数和动态参数之间的区别，以及如何和何时应用它们。要查看当前应用的配置，请使用 [GetDbParameterGroup](#)API 操作。

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
- 识别表现不佳的组件
- 调整组件以满足您的业务需求

性能效率支柱提供了可以帮助您选择高性能数据模型的指导方针。性能效率支柱包括查询和写入优化最佳实践。

绩效效率支柱侧重于以下关键领域：

- Influx 数据建模和查询优化
- 写入优化

Influx 数据建模和查询优化

设计有效的架构对于优化 InfluxDB 中时间序列数据的性能和查询功能至关重要。首先选择正确的标签和字段。InfluxDB 为标签编制索引，因此查询引擎无需扫描测量中的每条记录即可找到标签值。这意味着查询标签比查询字段更高效。为了压缩和存储数据，存储引擎按系列键对字段值进行分组，然后按时间对这些字段值进行排序。系列密钥由度量、标签键和值以及字段键定义。有关数据设计的更多信息，请参阅 [InfluxDB 文档](#)。

存储引擎使用时间结构化合并树 (TSM) 数据格式。有关 TSM 数据格式的更多信息，请参阅 [InfluxDB 文档](#)。

想象一下，您正在收集数据

(timestamp, host_id, region, cpu, memory, network_in_bytes, network_out_bytes, disk_io_bytes) 作为 DevOps 用例的一部分。标签 (包括记录时间戳) 提供了背景信息，以帮助识别记录的谁、内容、时间和地点。标签用于组织和分类数据，以及筛选作为查询一部分的数据。

host_id 和 region 标签是组织和分类 DevOps 用例的理想标签。这些列有助于筛选特定主机的数据或根据区域列运行分析。

度量为对数据进行数学计算（例如计算总数、平均值和变化率差异）和定量分析提供了基础。因此，cpu、memory、network_in_bytes、network_out_bytes、并 disk_io 捕获与随着时间的推移 DevOps 而变化的重要指标。您可以使用这些指标来执行各种分析，例如计算不同主机的 CPU 和内存。您可以使用这些指标值来做出数据驱动的决策，以帮助避免生产中断和执行基础设施规划。

基数是唯一标签值的组合。尽量降低基数。如果您的应用程序需要每个数据点的唯一标识符，请使用字段值而不是标签值。这将显著改善查询延迟。良好的架构设计可以防止高序列基数，从而提高查询的性能。如果您注意到数据读取和写入速度变慢，或者您想了解基数如何影响性能，请参阅 InfluxDB 的 [Timestream 文档](#)。

如果您的应用程序发出 JSON 对象，请将其转换为单个列（标签或字段），然后将这些列加载到 InfluxDB 中。InfluxDB 专为时间序列数据而设计，因此使用单独的列组织数据是充分利用该服务功能的最佳实践。

一个 InfluxDB v2.7 OSS 实例支持在所有组织中主动写入或查询大约 20 个 InfluxDB 存储桶。超过 20 个存储桶可能会对性能产生不利影响。有些 InfluxDB 配置选项有限制，有些选项可以根据自己的用例进行配置。在测试阶段，根据应用程序工作负载验证配置。数据保留是在存储桶级别配置的，因此具有不同数据保留要求的数据应存储在不同的存储桶中。有关配置选项的更多信息，请参阅 [InfluxDB 的 Timestream 文档](#)。

将数据存储存储在标签值或字段值中，而不是存储在标签键、字段键或测量值中。如果您将架构设计为将数据存储存储在标签和字段值中，则您的查询将更易于编写且效率更高。有关数据建模的更多最佳实践，[请参阅性能设计](#)。

使用 [InfluxDB 任务](#) 来预聚合数据，将数据加载到不同的测量值或存储桶中，并从中生成用于仪表板和可视化的数据。

InfluxDB OSS 公开了一个 [/metrics 端点](#)，该端点返回以 Prometheus 纯文本公开格式格式格式格式的性能、资源和使用情况指标。使用 InfluxDB 模板设置 [监控和警报](#)，以主动检测问题，例如高查询延迟、写入吞吐量下降或资源使用高峰。

InfluxDB 的时间流提供 Influx IO 内含存储。选择适当的 IOPS 大小可以显著加快查询执行速度。这对于需要扫描大量数据或处理大量请求的查询特别有用。在某些情况下，可能需要将扩展实例和增强 IOPS 结合起来才能实现所需的性能改进。

我们建议匹配开发环境和生产环境（实例类别、存储类型、配置）。在进入生产环境之前，在较低的环境中测试每个版本的更改。在 Influx IO Included 存储卷上，InfluxDB 的 Timestream 提供了三个存储层，这些存储层已预先配置了最佳 IOPS（3,000、12,000、16,000）和不同类型的工作负载所需的吞吐量。大多数用例需要的 IOPS 不到 3,000。仅当性能测试表明需要高 IOPS 时，才选择 12,000 或 16,000。有关更多信息，请参阅 InfluxDB 的 Timestream 文档中的 [设置部分](#)。

优化写入

为了优化对 InfluxDB 的写入，我们建议每个请求以 5,000 行线路协议为单位批量写入数据，以最大限度地减少网络开销。为了获得更好的性能，请在写入数据点之前按字典顺序对标签进行排序。对时间戳使用尽可能粗略的时间精度，而不是纳秒，也可以提高性能。启用 gzip 压缩是加快写入速度和减少网络带宽的另一种方法。在 `telegraf.conf` 文件的 `influxdb_v2` 输出插件配置中，将 `content_encoding` 选项设置为 `gzip`。实施这些优化可以显著提高向 InfluxDB 写入数据的性能和效率。如需了解更多 InfluxDB 写入最佳实践，请参阅[优化对 InfluxDB 的写入](#)。

InfluxDB 的写入性能通常与可用的 IOPS 密切相关。写入数据时，InfluxDB 需要执行大量的 I/O 操作来存储数据。当你提高 IOPS 时，InfluxDB 每秒可以处理更多的写入。

成本优化支柱

Well-Architected AWS Framework 的 [成本优化支柱](#) 侧重于避免不必要的成本，并以成本优化的方式构建架构。以下建议可以帮助您满足适用于 InfluxDB 的 Amazon Timestream 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：

- 了解您的用例的要求和成本
- 在选择资源时要注意成本
- 在不超支的情况下进行扩展以满足业务需求
- 调整数据存储和传输的大小

了解您的用例的要求和成本

在以下用例中，我们建议不要在 InfluxDB 上使用 Timestream：

- 如果你的数据模型包含关系数据，那么 InfluxDB 的 Timestream 不是正确的解决方案。
- 如果您无法在查询中使用时间过滤器，Influx 将扫描所有系列，这会降低效率。

在选择资源时要注意成本

[InfluxDB 实例](#) 的费用基于您的实例运行时间的小时费率。平均而言，实例占数据库运行总成本的 85% AWS，因此正确调整大小可能会带来显著的成本影响。调整实例大小的最佳方法是测试应用程序性能：

- CPU Utilization 和 Memory Utilization 一直处于高位还是低位？
- 价格和性能之间的平衡是什么？

实例成本呈线性扩展。db.influx.2xlarge 实例的每小时成本是 db.influx.xlarge 实例的两倍，尽管它的资源分配也是实例的两倍。该 db.influx.16xlarge 实例的每小时成本是实例每小时成本的 db.influx.xlarge 16 倍。

估算工作负载在特定时间段（秒、分钟、小时或天）内的写入和读取次数。InfluxDB 实例的时间流支持每秒 5 万至超过 500,000 次写入和每秒 10-100 次查询 (QPS)，具体取决于实例类型。例

如，`db.influx.2xlarge`通常每秒最多支持 150,000 次写入和大约 25 个 QPS。借助高效的数据模型和高效的查询，它可以超越该性能。如果您的要求因一天、一周或一个月的时间而异，则可以通过执行以下操作来安排向上和向下扩展：

- [创建和安排 AWS Lambda 函数](#)。
- 使用自定义调度程序并运行 API 或 SDK，在缓冲时间内向[上和向下扩展](#)。

在不超支的情况下进行扩展以满足业务需求

要使用适用于 InfluxDB 的 Timestream 进行入门级实验，你可以使用和 `db.influx.medium` `db.influx.large` 这些实例足够大，足以让您在投资更大的实例之前获得使用 Timestream for InfluxDB 的经验。

`db.influx.medium`和`db.influx.large`实例适用于低成本的开发环境。但是，它们具有较小的内存（8 GiB 和 16 GiB），更少的 v（CPU 1 vCPU 和 2 vCPUs），网络性能最高仅为 10 GB。并非所有工作负载都适用于这些实例类别。监控 CPU Utilization 和 Memory Utilization，并根据需要向上或向下扩展。内存与 vCPU 之间通常具有一致的比率。`db.influx` 实例类的 memory-to-vCPU 比率类似于 Amazon EC2 r7g 实例类。我们强烈建议在投入生产之前运行 end-to-end 性能或负载测试。

高效的数据建模、批量写入和优化的查询需要更少的内存和计算使用量。当需要较少的资源时，您可以使用较小的实例。

调整数据存储和传输的大小

要存储数据，请使用以下最佳实践：

- 在 InfluxDB 的时间流中仅存储时间序列数据。
- 在 InfluxDB 存储桶上设置适当的保留期，以便删除早于保留期的数据，并定期自动压缩分片。有关更多信息，请参阅 [InfluxDB 文档](#)。
- 优化磁盘使用率，以备将来的写入。
- 删除您的工作负载不需要的所有 InfluxDB 存储桶。InfluxDB 支持删除。如果符合您的用例，则可以按计划进行清理。

对于数据传输，我们建议将您的应用程序部署在与 InfluxDB 数据库实例的 Timestream AWS 区域相同，以避免跨区域网络开销。可能还会收取数据传输费用。有关数据传输的更多信息，请参阅[定价页面](#)。

可持续发展支柱

[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。可持续性支柱包含以下关键重点领域：

- 了解你的影响
- 可持续性目标
- 最大限度地利用以最大限度减少资源
- 预测和采用更高效的新硬件和软件产品
- 使用托管服务
- 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-Architected [AWS d Framework](#) ork。

您的选择和要求会对环境产生影响。要提高工作量的可持续性，请执行以下操作：

- 选择 AWS 区域 碳强度较低的产品。
- 调整资源大小以反映实际的工作负载需求，而不是最大限度地提高正常运行时间和耐用性。
- 优化您的数据模型并最大限度地利用计算资源。

下一节将讨论在工作负载设计和持续运营中可以采用的减少环境影响的实践。

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电网的地方。根据您的[可持续发展目标](#)和工作量要求评估区域。然后将你的可行区域列表与可用 [InfluxDB Timestream 的区域进行](#)交叉引用。

根据用户行为模式消耗资源

根据用户的流量和行为调整资源消耗，有助于 AWS 最大限度地减少服务对环境的影响。在设计解决方案时，请考虑以下最佳实践：

- 监控 Amazon CloudWatch 指标，例如 CPUUtilization 和 MemoryUtilization 以确定您的需求何时为最高和最低。在这段时间内，请确保您的实例资源大小合适。
- 除了业务连续性目标外，还可以考虑使您的服务水平协议与可持续性目标保持一致。放宽多区域灾难恢复、高可用性或长期备份保留等要求可以减少实现这些目标所需的资源量。非生产环境和非关键任务工作负载为降低需求提供了机会。

优化软件开发和架构模式

为防止浪费，请优化您的数据模型和查询。共享计算资源，以便使用 InfluxDB 实例的 Timestream 中可用的所有资源。我们建议实施以下最佳实践：

- 鼓励开发团队共享 InfluxDB 的 Timestream 堆栈，以便尽可能更好地利用资源。
- 实施能够最大限度地利用资源并最大限度地减少空闲时间的模式。模式示例包括使用并行线程加载数据，以及将记录批处理到更大的事务中。
- 优化您的查询和 InfluxDB 数据模型，以最大限度地减少计算结果所需的资源。
- 使用 [InfluxDB 任务](#) 来预聚合数据，减少不同用户对相同的原始数据进行可视化或仪表板管理的扫描。
- 让 InfluxDB 环境的时间流保持最新。最新版本的 InfluxDB Timestream 支持效率更高的最新 EC2 实例，例如 Graviton。最新的数据库版本还包括查询优化改进和错误修复，可减少计算查询所需的资源量。

资源

参考信息

- [AWS Well-Architected](#)
- [AWS WellArchited Framework 文档](#)
- [适用于 InfluxDB 的亚马逊 Timestream 文档](#)
- [InfluxDB OSS v2 文档](#)

博客文章

- [使用 AWS InfluxDB 迁移脚本将你的 InfluxDB OSS 2.x 数据迁移到适用于 InfluxDB 的 Amazon Timestream](#)
- [使用 Amazon Timestream 运行和管理开源 InfluxDB 数据库](#)

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 1 月 28 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：

- Refactor/re-architect — 充分利用云原生功能来提高敏捷性、性能和可扩展性，从而移动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 数据库迁移到亚马逊 Aurora PostgreSQL-Compatible 版。
- 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：将本地 Oracle 数据库迁移到 AWS Cloud 中的 Amazon Relational Database Service (Amazon RDS) for Oracle。
- 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。
- 重新托管 (直接迁移)：将应用程序迁移到云，无需进行任何更改即可利用云功能。示例：将本地 Oracle 数据库迁移到 AWS Cloud 中 EC2 实例上的 Oracle。
- 重新放置 (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。您将服务器从本地平台迁移到同一平台的云服务中。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

A2A () Agent-to-Agent

一种支持任务委托和状态转移的代理到代理协作的状态协议。

ABAC

请参阅[基于属性的访问控制](#)。

抽象服务

请参阅[托管服务](#)。

ACID

请参阅[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。它比[主动-被动迁移](#)更灵活，但工作量更大。

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。

座席

一种能够使用工具自主推理、计划和采取行动来实现目标的人工智能系统。

特工行动

在生产环境中大规模构建、测试、部署和运行 AI 代理的操作实践。

聚合函数

一种 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

请参阅[人工智能](#)。

AIOps

请参阅[人工智能运营](#)。

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？](#)

人工智能运营 (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 AWS 迁移策略中使用 AIOps 的更多信息，请参阅[运营集成指南](#)。

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 [IAM](#)) 文档 [AWS 中的 AB AC](#)。

权威数据来源

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来源复制到其他位置，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

恶意机器人

一种旨在扰乱或伤害个人或组织的[机器人](#)。

BCP

请参阅[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。另请参阅[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。

blue/green 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应用程序版本（蓝色），在另一个环境中运行新应用程序版本（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交互的软件应用程序。有些机器人是有用或有益的，例如在互联网上索引信息的 Web 爬网程序。还有一些被称为恶意机器人的机器人，其目的是扰乱或伤害个人或组织。

僵尸网络

被**恶意软件**感染并受单方（称为僵尸网络控制者或僵尸网络操作者）控制的**僵尸网络**。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）。

紧急（break-glass）访问

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，请参阅指南中的[“实施破碎玻璃程序”](#) AWS Well-Architected 指示器。

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

请参阅 [AWS 云采用框架](#)。

金丝雀部署

缓慢而渐进地向最终用户发布版本。当您确信无误后，即可部署新版本，并完全替换当前版本。

CCoE

请参阅 [云卓越中心](#)。

CDC

请参阅 [更改数据捕获](#)。

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韧性。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

请参阅 [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。

公民开发者

使用无code/low代码平台创建 AI 应用程序但没有专业技术技能的企业用户。

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。云计算通常连接到[边缘计算](#)技术。

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 AWS Cloud 中时通常会经历四个阶段：

- 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，创建登录区、定义 CCoE、建立运营模型）
- 迁移 - 迁移单个应用程序
- Re-invention — 优化产品和服务，在云端进行创新

Stephen Orban 在 AWS Cloud 企业战略博客的博客文章 [《走向之旅 Cloud-First 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

请参阅[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。这会影晌性能，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)

一种 [AI](#) 领域，它使用机器学习来分析和提取数字图像和视频等视觉格式中的信息。例如，Amazon SageMaker AI 为 CV 提供了图像处理算法。

配置偏移

对于工作负载而言，一种偏离预期状态的配置更改。这可能会导致工作负载变得不合规，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 通常被描述为管道。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

请参阅[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《AWS Well-Architected 框架》中安全支柱的组成部分。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个人。

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

请参阅[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。

深度防御

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，深度防御方法可能将多因素身份验证、网络分段和加密结合起来。

委派管理员

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。此帐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](#)。

部署

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

请参阅[环境](#)。

侦测性控制

一种安全控制，在事件发生后进行检测、记录日志和发出提醒。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

[星型架构](#)中的一种较小的表，其中包含事实表中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)

您用来最大程度地减少由[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[《工作负载灾难恢复 AWS：AWS Well-Architected 框架中的云端恢复》](#)。

DML

请参阅[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。埃里克·埃文斯 (Eric Evans) 在他的《Domain-Driven 设计：解决软件核心的复杂性》(波士顿：Addison-Wesley 专业版，2003年)一书中介绍了这个概念。有关如何使用带有 strangler fig 模式的域驱动设计的信息，请参阅[使用容器和 Amazon API Gateway 逐步实现传统微软 ASP.NET \(ASMX\) 网络服务的现代化](#)。

DR

请参阅[灾难恢复](#)。

偏差检测

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

请参阅[开发价值流映射](#)。

E

EDA

请参阅[探索性数据分析](#)。

EDI

请参阅[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](#)比较时，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)

组织之间业务文件的自动交换。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加密文字的计算流程。

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。Big-endian 系统首先存储最重要的字节。Little-endian 系统首先存储最低有效字节。

端点

请参阅[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

请参阅 [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

[星型架构](#) 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。

快速失效机制

一种使用频繁且增量式的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

请参阅[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[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。

少样本提示

在要求 [LLM](#) 执行类似任务之前，先向其提供少量示例，以演示任务和预期输出。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。Few-shot 对于需要特定格式、推理或领域知识的任务，提示可能非常有效。另请参阅[零样本提示](#)。

FGAC

请参阅[精细访问控制](#)。

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，通过[更改数据捕获](#)使用连续数据复制，在极短的时间内迁移数据，而非使用分阶段方法。目标是将停机时间降至最低。

FM

请参阅[基础模型](#)。

基础模型 (FM)

一个大型深度学习神经网络，它已使用海量的通用和未标注数据集进行训练。FM 能够执行各种常规任务，例如理解语言、生成文本和图像以及使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](#)。

FM 网关

一种集中式中介，用于控制和规范对[基础模型](#)的访问。也称为 LLM 网关。

G

生成式人工智能

[AI](#) 模型的一个子集，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构件，例如图像、视频、文本和音频。有关更多信息，请参阅[什么是生成式人工智能](#)。

地理阻止

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工作流程，而[基于中继的工作流程](#)则是现代的、首选的方法。

黄金映像

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预调配软件，并有助于提高设备制造操作的速度、可扩展性和生产效率。

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

一种高级规则，用于跨组织单位 (OU) 管理资源、策略和合规性。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。侦测性护栏会检测策略违规和合规性问题，并生成提醒以进行修复。它们通过使用 AWS Config、Amazon、AWS Security Hub CSPM GuardDuty AWS Trusted Advisor、Amazon Inspector 和自定义 AWS Lambda 支票来实现。

护栏 (AI)

用于过滤、验证和限制[代理](#)输入和输出的安全机制，有助于确保负责任和安全的 AI 行为。

H

HA

请参阅[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。

保留数据

从用于训练[机器学习](#)模型的数据集中保留的一部分标注的历史数据。通过将模型预测与保留数据进行比较，您可以使用保留数据来评估模型性能。

人机在圈 (HitL)

一种工作流程模式，其中[代理](#)执行在关键决策点暂停以供人工审查和批准。

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。

hypercare 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 hypercare 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

laC

请参阅[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

请参阅[工业物联网](#)。

不可变基础设施

一种模型，可为生产工作负载部署新的基础设施，而不是更新、修补或修改现有基础设施。不可变基础设施本质上比[可变基础设施](#)更一致、更可靠、更可预测。有关更多信息，请参阅框架中的[使用不可变基础架构部署](#)最佳实践。AWS Well-Architected

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](#)建议使用入站、出站和检查 VPC 设置网络账户，保护应用程序与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 \(Klaus Schwab \)](#)在2016年推出，指的是通过连接性、实时数据、自动化、分析和的进步实现制造流程的现代化。AI/ML

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调配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 \(IIoT \) 数字化转型策略](#)。

检查 VPC

在 AWS 多账户架构中，一种集中式 VPC，用于管理 VPC (相同或不同 AWS 区域)、互联网和本地网络之间的网络流量检查。[AWS 安全参考架构](#)建议使用入站、出站和检查 VPC 设置网络账户，保护应用程序与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[机器学习模型的可解释性 AWS](#)。

物联网

请参阅[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。LLM 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 LLM](#)。

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

请参阅[7 R](#)。

小端序系统

一个先存储最低有效字节的系统。另请参阅[字节顺序](#)。

LLM

请参阅[大型语言模型](#)。

下层环境

请参阅[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

请参阅[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权限。恶意软件的示例包括病毒、蠕虫、勒索软件、木马、间谍软件和键盘记录器。

托管式服务

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。Amazon Simple Storage Service (Amazon S3) 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

请参阅[迁移加速计划](#)。

MCP

参见[模型上下文协议](#)。

模型上下文协议 (MCP)

一种用于[代理](#)与[工具](#)通信的无状态协议。

MCP 服务器

一种通过[模型上下文协议](#)公开一个或多个[工具](#)的服务。

机制

一个完整的过程，您可以在其中创建工具，推动工具的采用，然后检查结果以进行调整。机制是一种在运作过程中自我强化和改善的循环。有关更多信息，请参阅在 AWS Well-Architected 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

请参阅[制造执行系统](#)。

消息队列遥测传输 (MQTT)

[一种基于publish/subscribe模式的轻量级机器对机器 \(M2M\) 通信协议，适用于资源受限的物联网设备。](#)

微服务

一种小型独立服务，通过明确定义的 API 进行通信，通常由小型独立团队拥有。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 API 通过明确定义的接口进行通信。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

Cross-functional 通过自动化、敏捷的方法简化工作负载迁移的团队。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器的信息。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：使用 AWS 应用程序迁移服务重新托管向 Amazon EC2 的迁移。

迁移组合评测 (MPA)

一种在线工具，提供了用于验证迁移到 AWS Cloud 的业务案例的信息。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

将工作负载迁移到 AWS Cloud 的方法。有关更多信息，请参见术语表中的 [7 R](#) 词条，以及[动员您的组织以加快大规模迁移](#)。

ML

请参阅[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[在 AWS Cloud 中实现应用程序现代化的策略](#)。

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[在 AWS Cloud 中评估应用程序的现代化准备情况](#)。

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

请参阅[迁移组合评测](#)。

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？”

可变基础设施

一种用于更新和修改生产工作负载的现有基础设施的模型。为了提高一致性、可靠性和可预测性，该 AWS Well-Architected 框架建议使用[不可变基础设施](#)作为最佳实践。

O

OAC

请参阅[来源访问控制](#)。

OAI

请参阅[来源访问身份](#)。

OCM

请参阅[组织变革管理](#)。

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

请参阅[运营集成](#)。

OLA

请参阅[运营级别协议](#)。

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

请参阅[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机器对机器 (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)

一份问题核对清单和关联的最佳实践，可帮助您了解、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 AWS Well-Architected 框架中的[运营准备情况审查 \(ORR\)](#)。

运营技术 (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](#) 转型的关键重点。

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 AWS CloudTrail 此创建的跟踪记录组织 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

请参阅[运营准备情况审查](#)。

OT

请参阅[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](#) 建议使用入站、出站和检查 VPC 设置网络账户，保护应用程序与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

请参阅[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

请参阅[可编程逻辑控制器](#)。

PLM

请参阅[产品生命周期管理](#)。

policy

一个对象，可以定义权限（请参阅[基于身份的策略](#)）、指定访问条件（请参阅[基于资源的策略](#)）或定义 AWS Organizations 的组织中所有账户的最大权限（请参阅[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。

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 false 的查询条件，通常位于 WHERE 子句中。

谓词下推

一种数据库查询优化技术，可在传输之前筛选查询中的数据。这将减少从关系数据库检索和处理的数据量，并提高查询性能。

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。

隐私设计

一种在整个开发过程中都考虑隐私的系统工程方法。

私有托管区

私有托管区就是一个容器，其中包含的信息说明您希望 Amazon Route 53 如何响应一个或多个 VPC 中的某个域及其子域的 DNS 查询。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](#)，旨在防止部署不合规资源。这些控制会在资源预置之前对其进行扫描。如果资源与控制不兼容，则不会预置它。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)

对产品在其整个生命周期内的数据和流程的管理，从设计、开发和发布，到增长和成熟，再到衰退和淘汰。

生产环境

请参阅[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。

提示串接

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，可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

一系列用于访问 SQL 关系数据库系统中的数据的步骤，类似于指令。

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

请参阅[责任、问责、咨询和知情 \(RACI \)](#)。

RAG

请参阅[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

请参阅[责任、问责、咨询和知情 \(RACI \)](#)。

RCAC

请参阅[行列访问控制](#)。

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

请参阅 [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

请参阅 [7 R](#)。

Region

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[指定您的账户可以使用的 AWS 区域](#)。

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

请参阅 [7 R](#)。

版本

在部署过程中，推动生产环境变更的行为。

重新放置

请参阅 [7 R](#)。

更换平台

请参阅 [7 R](#)。

重新购买

请参阅 [7 R](#)。

韧性

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

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

请参阅 [7 R](#)。

停用

请参阅 [7 R](#)。

检索增强生成 (RAG)

一种[生成式人工智能](#)技术，其中 [LLM](#) 在生成响应之前引用其训练数据来源之外的权威数据来源。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证的过程。

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

请参阅[恢复点目标](#)。

RTO

请参阅[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 管理控制台 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

请参阅[监督控制和数据采集](#)。

SCP

请参阅[服务控制策略](#)。

机密密钥

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 文档中的[什么是 Amazon Secrets Manager 密钥？](#)。

安全设计

一种在整个开发过程中都考虑安全的系统工程方法。

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。安全控制有以下四种类型：[预防性](#)、[检测性](#)、[响应性](#)和[主动性](#)。

安全固化

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

一种预定义的程序化操作，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 实例或轮换凭证。

服务器端加密

由接收数据的人在目的地对数据 AWS 服务 进行加密。

服务控制策略 (SCP)

一种策略，用于集中控制 AWS Organizations 的组织中所有账户的权限。SCP 为管理员可以委托给用户或角色的操作定义了防护机制或设定了限制。您可以将 SCP 用作允许列表或拒绝列表，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。

服务水平指示器 (SLI)

对服务性能方面的衡量，例如错误率、可用性或吞吐量。

服务水平目标 (SLO)

代表服务运行状况的目标指标，由[服务水平指示器](#)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](#)。

暗影人工智能

在组织内受管控渠道之外构建或使用的未经授权的 [AI](#) 应用程序。

SIEM

请参阅[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

请参阅[服务水平协议](#)。

SLI

请参阅[服务水平指示器](#)。

SLO

请参阅[服务水平目标](#)。

split-and-seed 模式

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[在 AWS Cloud 中实现应用程序现代化的分阶段方法](#)。

SPOF

请参阅[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事务数据或测量数据，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 逐步实现传统微软 ASP.NET \(ASMX\) 网络服务的现代化](#)。

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。

监督控制和数据采集 (SCADA)

在制造业中，一种使用硬件和软件来监控实物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

一种为 [LLM](#) 提供上下文、说明或准则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

标签

Key-value 对充当用于组织 AWS 资源的元数据。标签有助于您管理、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

请参阅[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。

工具

[代理](#)可以调用以在外部系统中执行操作的函数或 API。

中转网关

中转网关是网络中转中心，您可用它来互连 VPC 和本地网络。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，该服务可以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。

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

请参阅[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

两个 VPC 之间的连接，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

一种对与当前记录有某种关联的一组行执行计算的 SQL 函数。窗口函数对于处理任务很有用，例如计算移动平均值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。

WORM

请参阅[一次写入多次读取](#)。

WQF

请参阅[AWS 工作负载资格鉴定框架](#)。

一次写入多次读取 (WORM)

一种存储模型，可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，但无法对其进行更改。此数据存储基础设施被认为[不可变](#)。

Z

零日漏洞利用

一种利用[零日漏洞](#)的攻击，通常为恶意软件。

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。

零样本提示

为[LLM](#)提供执行任务的说明，但没有可以帮助指导的示例（样本）。LLM 必须使用预先训练的知识来处理任务。零样本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[少样本提示](#)。

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。