AWS 多区域基础知识

AWS 规范性指导

AWS 规范性指导: AWS 多区域基础知识

Table of Contents

AWS 多区域基础知识

John Formento，Amazon Web Services ()AWS

2025 年 9 月（文档历史记录）

这份 300 级高级指南适用于云架构师和高级领导者，他们需要在多区域架构上构建工作负载， AWS 并有兴趣使用多区域架构来提高工作负载的弹性。本指南假设对 AWS 基础设施和服务有基本了解。它概述了常见的多区域用例，分享了围绕设计、开发和部署的基本多区域概念和含义，并提供了规范性指导，以帮助您更好地确定多区域架构是否适合您的工作负载。

在本指南中：

您使用 Well-Architected 了吗？

Wel AWS l-Architected Framework 可以帮助你了解在云端构建系统时所做决策的利弊。该框架的六大支柱为设计和运行可靠、安全、高效、具有成本效益和可持续的系统提供了架构最佳实践。您可以使用（上免费提供）AWS Well-Architected Tool，通过回答每个支柱的一组问题 AWS 管理控制台，根据这些最佳实践来审查您的工作负载。

有关云架构的其他专家指导和最佳实践，包括参考架构部署、图表和技术指南，请参阅AWS 架构中心。

简介

每个可用区都AWS 区域由一个地理区域内的多个独立且物理上独立的可用区组成。每个区域的软件服务之间保持了严格的逻辑分离。这种有针对性的设计可确保一个地区的基础设施或服务故障不会导致另一个地区的相关故障。

大多数 AWS 用户可以使用多个可用区或区域可用区来实现其针对单个区域工作负载的弹性目标 AWS 服务。但是，一部分用户选择多区域架构的原因有三个：

- 他们对最高级别的工作负载具有很高的可用性和操作连续性要求，并希望从影响单个区域资源的损伤中确定有限的恢复时间。
- 它们需要满足数据主权要求（例如遵守当地法律、法规和合规性），这些要求工作负载在特定的司法管辖区内运行。
- 他们需要在离最终用户最近的位置运行工作负载，从而改善工作负载的性能和客户体验。

本指南重点介绍操作的高可用性和连续性要求，并帮助您了解为工作负载采用多区域架构的注意事项。它描述了适用于多区域工作负载设计、开发和部署的基本概念，并提供了一个规范性框架来帮助您确定多区域架构是否是特定工作负载的正确选择。您需要确保多区域架构是您的工作负载的正确选择，因为这些架构具有挑战性，如果多区域架构构建不正确，则工作负载的整体可用性可能会降低。

设计和运营以提高单一地区的韧性

在深入探讨多区域概念之前，首先要确认您的工作负载已在单个区域中尽可能具有弹性。为实现这一目标，请根据Wel l-Architecte AWS d Framework的可靠性支柱和卓越运营支柱评估您的工作量，并根据权衡和风险评估进行任何必要的更改。Well-Architected Framework 中 AWS 涵盖了以下概念：

- 基于域边界的工作负载分段
- 定义明确的服务合同
- 依赖关系管理和耦合
- 处理失败、重试和退缩策略
- 等能操作以及有状态事务与无状态事务
- 运营准备和变更管理
- 了解工作负载运行状况
- 对事件做出响应

要进一步提高单区域弹性，请查看并应用论文《高级多可用区弹性模式：检测和缓解灰色故障》中讨论的概念。本 paper 提供了在每个可用区中使用副本来控制故障的最佳实践，并扩展了架构 AWS 完善的框架中引入的多可用区概念。尽管多区域架构可以缓解与可用区域绑定的故障模式，但您应该考虑多区域方法带来的权衡取舍。这就是为什么我们建议您从多可用区方法入手，然后根据多区域架构的基础知识评估特定的工作负载，以确定多区域方法能否提高工作负载的弹性。

多区域基础知识 1：了解需求

如前所述，高可用性和操作连续性是采用多区域架构的常见原因。可用性指标衡量的是工作负载在定义的时间段内可供使用的时间百分比，而操作连续性指标衡量的是大规模（通常是持续时间更长）事件的恢复时间。

衡量可用性几乎是一个持续的过程。具体的衡量标准可能有所不同，但通常围绕目标可用性指标汇总，通常被称为 9（例如 99.99% 的可用性）。就可用性目标而言，不能一刀切。您应该在工作负载级别制定可用性目标，将非关键组件与关键组件分开，而不是将单个目标应用于所有工作负载。

为了保证操作的连续性，通常使用以下 point-in-time测量：

- 恢复时间目标 (RTO) — RTO 是服务中断和恢复服务之间可接受的最大延迟。此值决定了服务受损的可接受持续时间。
- 恢复点目标 (RPO)-RPO 是自上一个数据恢复点以来的最大可接受时间量。这决定了在最新恢复点和服务中断之间哪些数据丢失被认为是可接受的。

与设置可用性目标类似，还应在工作负载级别定义 RTO 和 RPO。更积极的运营连续性或高可用性需要增加投资。也就是说，并非每个应用程序都能要求或需要相同级别的弹性。协调业务和 IT 所有者，根据业务影响评估应用程序的重要程度，然后相应地对它们进行分层，可以帮助提供一个起点。下表提供了分层示例。

下表显示了服务级别协议的弹性分层示例（ ）SLAs。

弹性等级	可用性 SLA	可接受的停机时间/年
铂	99.99%	52.60 分钟
黄金	99.90%	8.77 小时
银	99.5%	1.83 天

下表显示了 RTO 和 RPO 的弹性分层示例。

弹性等级	最大 RTO	最大 RPO	标准	成本
铂	15 分钟	5 分钟	任务关键型工作负载	$$$
黄金	15 分钟 — 6 小时	2 小时	重要但不是任务关键型工作负载	$$
银	6 小时 — 几天	24 小时	非关键工作负载	$

在设计具有弹性的工作负载时，请考虑高可用性与操作连续性之间的关系。例如，如果工作负载需要 99.99% 的可用性，则每年的停机时间不能超过 53 分钟。检测故障可能至少需要 5 分钟，操作员还需要 10 分钟才能接触、决定恢复步骤并执行这些步骤。花费 30 到 45 分钟才能从单个问题中恢复过来的情况并不少见。在这种情况下，采用多区域策略来提供一个可以消除相关影响的隔离实例是有益的。这允许您在独立对初始减值进行分类的同时，在有限的时间内进行故障切换，从而继续运营。这需要定义适当的限定恢复时间并确保保持一致。

多区域方法可能适用于具有极高的可用性需求（例如 99.99% 或更高的可用性）或严格的运营连续性要求（只有通过故障转移到另一个区域才能满足）的任务关键型工作负载。但是，这些要求通常仅适用于企业工作负载组合中的一小部分，这些工作负载的恢复时间限制在几分钟或几小时内。除非应用程序需要几分钟或几小时的恢复时间，否则在受影响地区内等待应用程序的区域中断得到补救可能是更好的方法。这种方法通常与较低级别的工作负载保持一致。

在实施多区域架构之前，业务决策者和技术团队应就成本影响进行调整，包括运营和基础设施成本驱动因素。典型的多区域架构可能产生的成本是单区域方法的两倍。尽管业务连续性有几种多区域模式，例如在热备用、热待机或指示灯下运行，但实现恢复目标风险最低的模式将涉及运行热备用，并且会使您的工作负载成本增加一倍。

关键指导

- 运营目标（例如 RTO 和 RPO）的可用性和连续性应根据工作负载确定，并与业务和 IT 利益相关者保持一致。
- 大多数可用性和运营连续性目标都可以在单个区域内实现。对于在单个区域内无法实现的目标，可以考虑多区域，同时要清楚地权衡成本、复杂性和收益。

多区域基础知识 2：了解数据

当您采用多区域架构时，管理数据是一个不容忽视的问题。区域之间的地理距离会带来不可避免的延迟，这种延迟表现为跨区域复制数据所花费的时间。在可用性、数据一致性以及为使用多区域架构的工作负载引入更高的延迟之间进行权衡是必要的。无论您使用异步复制还是同步复制，都需要修改应用程序以应对复制技术带来的行为变化。数据一致性和延迟方面的挑战使得使用专为单一区域设计的现有应用程序并使其成为多区域变得非常困难。了解特定工作负载的数据一致性要求和数据访问模式对于权衡利弊至关重要。

2.a：了解数据一致性要求

CAP 定理为推理数据一致性、可用性和网络分区之间的权衡提供了参考。对于一个工作负载，只能同时满足其中两个要求。顾名思义，多区域架构包括区域之间的网络分区，因此您必须在可用性和一致性之间做出选择。

如果您选择跨区域的数据可用性，则在事务写入操作期间不会出现明显的延迟，因为依赖区域间提交的数据的异步复制会导致复制完成之前各区域之间的一致性降低。对于异步复制，当主区域出现故障时，写入操作很有可能等待从主区域进行复制。这会导致一种情况，即在恢复复制之前，最新数据不可用，并且需要对账流程来处理未从经历中断的地区复制的正在进行的交易。这种情况需要了解您的业务逻辑，并创建一个特定的流程来重播事务或比较各区域之间的数据存储。

对于偏爱异步复制的工作负载，您可以使用 Amazon Aurora 和 Amazon D ynamo DB 等服务进行异步跨区域复制。Amazon Aurora 全球数据库和亚马逊 DynamoDB 全局表都有默认的 CloudWatch亚马逊指标，以帮助监控复制延迟。Aurora 全球数据库由一个写入数据的主区域和最多五个只读辅助区域组成。DynamoDB 全局表由跨任意数量的区域的多活动副本表组成，您的数据将从中写入和读取。

设计工作负载以利用事件驱动架构对多区域策略来说是一个好处，因为这意味着工作负载可以包括数据的异步复制，并通过重播事件来实现状态重建。由于流媒体和消息服务将消息负载数据缓冲到单个区域，因此区域故障转移或故障恢复过程必须包括重定向客户端输入数据流的机制。该流程还必须核对存储在经历中断的地区的飞行中或未交付的有效载荷。

如果您选择 CAP 一致性要求并使用跨区域同步复制的数据库来支持在多个区域同时运行的应用程序，则可以消除数据丢失的风险并使数据在区域之间保持同步。但是，这会引入更高的延迟特征，因为写入需要提交到多个区域，而且这些区域彼此之间可能相隔数百或数千英里。您需要在应用程序设计中考虑这种延迟特性。此外，同步复制可能会导致相关故障，因为写入操作需要提交到多个区域才能成功。如果一个区域内存在损失，则需要形成法定人数才能成功写入。这通常涉及在三个区域中设置数据库，并建立三个区域中两个的法定人数。 诸如 Paxos 之类的技术可以帮助同步复制和提交数据，但需要大量的开发人员投资。

当写入涉及跨多个区域的同步复制以满足严格的一致性要求时，写入延迟会增加一个数量级。如果不进行重大更改（例如重新审视应用程序的超时和重试策略），通常无法将较高的写入延迟改装到应用程序中。理想情况下，在首次设计应用程序时必须将其考虑在内。对于优先考虑同步复制的多区域工作负载，AWS Partner 解决方案可以提供帮助。

2.b：了解数据访问模式

工作负载数据访问模式要么是读取密集型，要么是写密集型。了解特定工作负载的这一特性将有助于您选择合适的多区域架构。

对于读取密集型工作负载，例如完全只读的静态内容，您可以实现主动-主动多区域架构，与写入密集型工作负载相比，该架构的工程复杂性更低。使用内容分发网络 (CDN) 在边缘提供静态内容，通过缓存最接近最终用户的内容来确保可用性；在 Amazon 中使用诸如源故障转移之类的功能集 CloudFront 可以帮助实现这一目标。另一种选择是在多个区域部署无状态计算，并使用 DNS 将用户路由到最近的区域以读取内容。您可以使用带有地理定位路由策略的 Amazon Route 53 来实现这一目标。

对于读取流量比例高于写入流量百分比的读取密集型工作负载，您可以使用本地读取、写入全局策略。这意味着所有写入请求都将发送到特定区域的数据库，数据将异步复制到所有其他区域，并且可以在任何区域进行读取。这种方法需要工作负载才能实现最终一致性，因为跨区域写入复制的延迟增加可能会导致本地读取过时。

Aurora 全球数据库可以帮助在只能在本地处理所有读取流量的备用区域中配置只读副本，并在特定区域预置单个主数据存储以处理写入流量。数据从主数据库异步复制到备用数据库（只读副本），如果您需要将操作故障转移到备用区域，则可以将备用数据库提升为主数据库。您也可以在这种方法中使用 DynamoDB。D@@ ynamoDB 全局表可以跨区域预配置副本表，每个区域都可以扩展以支持任意数量的本地读取或写入流量。当应用程序将数据写入一个区域中的副本表时，DynamoDB 会自动将写操作传播到其他 区域的其他副本表。使用此配置，数据将从定义的主区域异步复制到备用区域中的副本表。任何区域中的副本表都可以随时接受写入，因此将备用区域提升为主区域是在应用程序级别管理的。同样，工作负载必须包含最终的一致性，如果不是从一开始就为此而设计的，则可能需要对其进行重写。

对于写入密集型工作负载，应选择主区域，并在工作负载中设计故障转移到备用区域的功能。与主动-主动方法相比，主备用方法还有额外的权衡取舍。这是因为对于主动-主动架构，必须重写工作负载，以处理到区域的智能路由、建立会话关联性、确保等效事务以及处理潜在的冲突。

大多数使用多区域方法实现弹性的工作负载不需要主动-主动方法。您可以使用分片策略通过限制减值对整个客户群的影响范围来提高弹性。如果您可以有效地对客户群进行分片，则可以为每个分片选择不同的主区域。例如，您可以对客户端进行分片，使一半的客户端与区域一对齐，一半的客户端与区域二

对齐。通过将区域视为单元，您可以创建多区域单元方法，从而缩小工作负载的影响范围。有关更多信息，请参阅有关此方法的 AWS re: Invent 演示文稿。

您可以将分片方法与主备用方法相结合，为分片提供故障转移功能。您需要为工作负载设计经过测试的故障转移过程和数据协调流程，以确保故障转移后数据存储的事务一致性。本指南稍后将详细介绍这些内容。

关键指导

- 出现故障时，待复制的写入操作很可能不会提交到备用区域。在恢复复制之前，数据将不可用（假设异步复制）。
- 作为故障转移的一部分，需要一个数据协调过程来确保使用异步复制的数据存储保持事务一致的状态。这需要特定的业务逻辑，而不是由数据存储本身处理的事情。
- 当需要强一致性时，需要修改工作负载，以容忍同步复制的数据存储所需的延迟。

多区域基础知识 3：了解您的工作负载依赖关系

特定工作负载在一个区域中可能有多个依赖关系，例如已 AWS 服务 使用的依赖关系、内部依赖关系、第三方依赖关系、网络依赖关系、证书、密钥、密钥和参数。为了确保工作负载在故障情况下运行，主区域和备用区域之间不应存在任何依赖关系；每个区域都应能够独立运行。为此，请仔细检查工作负载中的所有依赖关系，以确保它们在每个区域中都可用。这是必需的，因为主区域的故障不会影响备用区域。此外，您必须了解当依赖关系处于降级状态或完全不可用时工作负载是如何运行的，这样您才能设计出适当的解决方案来处理这个问题。

3.a: AWS 服务

在设计多区域架构时，重要的是要了解将要使用的架构、这些服务的多区域功能，以及需要设计哪些解决方案才能实现多区域目标。 AWS 服务 例如，Amazon Aurora 和 Amazon DynamoDB 可以将数据异步复制到备用区域。所有 AWS 服务 依赖关系都需要在运行工作负载的所有区域中都可用。要确认您使用的服务在所需区域中可用，请查看AWS 服务 按区域列出的列表。

3.b: 内部依赖关系和第三方依赖关系

确保每个工作负载的内部依赖关系在其运行的区域中都可用。例如，如果工作负载由许多微服务组成，请识别构成业务功能的所有微服务，并验证所有这些微服务是否部署在工作负载运行的每个区域。或者，定义一种策略来优雅地处理不可用的微服务。

不建议在工作负载内的微服务之间进行跨区域调用，并且应保持区域隔离。这是因为创建跨区域依赖关系会增加相关故障的风险，这抵消了工作负载的孤立区域实施的好处。本地依赖关系也可能是工作负载的一部分，因此重要的是要了解如果主区域发生变化，这些集成的特征会如何变化。例如，如果备用区域距离本地环境更远，则延迟增加可能会产生负面影响。

了解软件即服务 (SaaS) 解决方案、软件开发套件 (SDKs) 和其他第三方产品依赖关系，并能够演练这些依赖关系降级或不可用的场景，将使人们更深入地了解系统链在不同故障模式下的运行和行为。这些依赖关系可能存在于您的应用程序代码中，例如使用在外部管理机密 AWS Secrets Manager，也可能涉及第三方保管库解决方案（例如 HashiCorp），或者依赖于AWS IAM Identity Center联合登录的身份验证系统。

在依赖关系方面拥有冗余可以提高弹性。如果 SaaS 解决方案或第三方依赖项使用与工作负载相同的主 AWS 区域 服务器，请与供应商合作，确定他们的弹性状况是否符合您对工作负载的要求。

此外，请注意工作负载及其依赖关系（例如第三方应用程序）之间的共同命运。如果故障转移后在辅助区域（或来自辅助区域）的依赖关系不可用，则工作负载可能无法完全恢复。

3.c: 故障转移机制

DNS 通常用作故障转移机制，用于将流量从主区域转移到备用区域。严格审查和仔细检查故障转移机制所需要的所有依赖关系。例如，如果您的工作负载使用 Amazon Route 53，则了解控制平面托管在中us-east-1意味着您依赖该特定区域的控制平面。如果主区域也是us-east-1因为会造成单点故障，则不建议将其作为故障转移机制的一部分。如果您使用其他故障转移机制，则应深入了解故障转移无法按预期运行的情况，然后根据需要做好应急计划或开发新的机制。Amazon 应用程序恢复控制器 (ARC) 区域交换机是一项完全托管的多区域恢复服务，您可以将其用作故障转移机制。

如上一节所述，作为业务能力一部分的所有微服务都需要在部署工作负载的每个区域中可用。作为故障转移策略的一部分，作为业务功能一部分的所有微服务都应一起进行故障转移，以消除跨区域调用的机会。或者，如果微服务独立进行故障转移，则可能会出现不良行为，例如微服务可能会进行跨区域调用。这会带来延迟，并可能导致工作负载在客户端超时期间变得不可用。

3.d: 配置依赖关系

证书、密钥、机密、Amazon 系统映像 (AMIs)、容器映像和参数是设计多区域架构时所需的依赖关系分析的一部分。只要有可能，最好在每个区域内对这些组件进行本地化，这样它们就不会因为这些依赖关系而在区域之间共享命运。例如，您应该更改证书的到期日期，以防止即将到期的证书（警报设置为“提前通知”）影响多个区域。

加密密钥和机密也应特定于区域。这样，如果密钥或密钥的轮换出现错误，则影响仅限于特定区域。

最后，所有工作负载参数都应存储在本地，以便在特定区域中检索工作负载。

关键指导

- 多区域架构受益于区域间的物理和逻辑分离。在应用层引入跨区域依赖关系会破坏这一好处。避免此类依赖。
- 故障转移控制应在不依赖于主区域的情况下运行。
- 应在用户旅程中协调故障转移，以消除延迟增加和跨区域呼叫依赖性的可能性。

多区域基础知识 4：运营准备就绪

操作多区域工作负载是一项复杂的任务，会带来多区域架构特有的运营挑战。其中包括 AWS 账户 管理、重新调整部署流程、创建多区域可观测性策略、创建和测试恢复流程，然后管理成本。运营准备情况评估 (ORR) 可以帮助团队为生产工作负载做好准备，无论是在单个区域还是在多个地区运行。

4.a: 管理 AWS 账户

要跨区域部署工作负载 AWS 区域，请确保不同区域的账户内的所有AWS 服务 配额保持平等。首先，确定架构中的所有 AWS 服务 内容，查看备用区域的计划使用情况，然后将计划使用量与当前使用量进行比较。在某些情况下，如果以前未使用过备用区域，则可以参考默认服务配额来了解起点。然后，在将要使用的所有服务中，使用 S ervice Quotas 控制台（需要登录）申请增加配额，或者APIs。

在每个区域中配置 AWS Identity and Access Management (IAM) 角色，为操作员、自动化工具和 AWS 服务 备用区域内的资源提供相应权限。要实现多区域架构的区域隔离，请按区域隔离角色。在备用区域上线之前，请确保权限已到位。

4.b：部署实践

多区域功能可能会使将工作负载部署到多个区域变得复杂。您需要确保一次部署到一个区域。例如，如果您使用主动-被动方法，则应先部署到主区域，然后再部署到备用区域。 AWS CloudFormation可帮助您将基础设施部署到单个或多个区域，并且可以根据您的需求进行定制。 AWS CodePipeline帮助您构建持续 integration/continuous 交付 (CI/CD) 管道，该管道具有跨区域操作，允许部署到与管道所在区域不同的区域。再加上蓝/绿等强大的部署策略，可以实现最少至零的停机时间部署。

但是，当应用程序或数据的状态未外部化到持久存储时，有状态功能的部署可能会变得更加复杂。在这些情况下，请仔细调整部署流程以满足您的需求。将部署管道和流程设计为一次部署到一个区域，而不是同时部署到多个区域。这减少了区域之间出现相关故障的机会。要了解 Amazon 用于自动部署软件的技术，请参阅 AWS Builders Library 文章自动化安全、无需动手的部署。

4.c: 可观察性

在设计多区域时，请考虑如何监控每个区域中所有组件的运行状况，以全面了解区域运行状况。这可能包括复制延迟的监控指标，这不是单区域工作负载的考虑因素。

在构建多区域架构时，也要考虑从备用区域观察工作负载的性能。这包括在备用区域运行运行状况检查和加那利群岛（综合测试），以提供主区域健康状况的外部视图。此外，您还可以使用 Amazon

CloudWatch Internet Monit or 从最终用户的角度了解外部网络的状态和工作负载的性能。主区域应具有相同的可观察性，以监控备用区域。

备用区域的加那利群岛应监控客户体验指标，以确定工作负载的整体运行状况。这是必需的，因为如果主区域出现问题，则主区域的可观察性可能会受到损害，从而影响您评估工作负载运行状况的能力。

在这种情况下，在该区域之外进行观察可以提供见解。这些指标应汇总到每个区域可用的仪表板和在每个区域创建的警报中。由于CloudWatch是一项区域服务，因此需要在两个区域都设置警报。这些监控数据将用于调用从主区域故障转移到备用区域。

4.d：流程和程序

现在是回答“我什么时候应该进行故障切换？”这个问题的最佳时机 早在你需要之前。在问题出现之前，尽早制定包含人员、流程和技术的恢复计划，并定期对其进行测试。确定恢复决策框架。如果有经过良好实践的恢复过程并且对恢复时间了如指掌，则可以选择使用满足 RTO 目标的故障转移来启动恢复过程。该时间点可能是在发现主区域中的应用程序存在问题之后立即出现的，也可能是在该区域应用程序中的恢复选项已用尽之后。

故障转移操作本身应百分之百自动化，但激活故障切换的决定应由人来做出，通常是组织中少数预先确定的个人。这些人应考虑数据丢失和有关事件的信息。此外，还需要在组织内部明确定义和全面了解故障转移的标准。要定义和完成这些流程，您可以使用 Amazon Application Recovery Controller (ARC) 区域切换，它可以 end-to-end实现完全自动化，并确保测试和故障转移期间运行的流程的一致性。

当您创建区域切换计划时，它会自动在主区域和备用区域中复制您的计划，以确保不依赖于单个区域。当这种自动化到位后，请定义并遵循定期的测试节奏。这样可以确保在发生实际事件时，响应遵循组织有信心的明确、实践的流程。同样重要的是要考虑数据协调过程的既定容差。确认建议的流程符合既定 RPO/RTO 要求。

4.e: 测试

采用未经测试的恢复方法等于没有恢复方法。基本的测试级别是运行恢复过程来切换应用程序的操作区域。有时，这被称为应用程序轮换方法。我们建议您建立将区域切换到正常操作状态的能力；但是，仅此测试是不够的。

弹性测试对于验证应用程序的恢复方法也至关重要。这包括注入特定的故障场景，了解您的应用程序和恢复过程的反应，然后在测试未按计划进行时实施所需的任何缓解措施。在没有错误的情况下测试恢复过程并不能告诉您在出现故障时应用程序的整体行为方式。您必须制定计划，根据预期的故障情况测试恢复情况。 AWS Fault Injection Service提供了越来越多的场景供您入门。

这对于高可用性应用程序尤其重要，在这些应用程序中，需要进行严格的测试以确保实现业务连续性目标。主动测试恢复功能可以降低生产中出现故障的风险，从而增强人们对应用程序能够实现所需的有限恢复时间的信心。定期测试还可以培养运营专业知识，使团队能够在停机时快速可靠地从中断中恢复。运用康复方法中的人为因素或过程与技术方面同样重要。

4.f：成本和复杂性

多区域架构的成本影响是由更高的基础设施使用率、运营开销和资源时间造成的。如前所述，预配置时，备用区域的基础设施成本与主区域的基础设施成本相似，因此总成本会翻一番。配置容量，使其足以满足日常运营需求，但仍保留足够的缓冲容量来容纳需求激增。然后在每个区域配置相同的限制。

此外，如果您采用主动-主动架构，则可能需要进行应用程序级别的更改才能在多区域架构中成功运行应用程序。这些变更的设计和操作可能既耗时又耗费资源。组织至少需要花时间了解每个地区的技术和业务依赖关系，并设计故障转移和故障恢复流程。

团队还应进行正常的故障转移和故障恢复练习，以便对活动期间使用的运行手册感到满意。尽管这些活动对于从多区域投资中获得预期结果至关重要，但它们代表着机会成本，会占用其他活动的时间和资源。

4.g：组织多区域故障转移策略

AWS 区域 提供故障隔离边界，防止相关故障，并控制 AWS 服务 损伤发生时对单个区域的影响。您可以使用这些故障边界来构建由每个区域中独立的故障隔离副本组成的多区域应用程序，以限制共享命运场景。这使您可以构建多区域应用程序，并使用一系列方法（从备份和恢复，到指示灯，再到主动-主动）来实现您的多区域架构。但是，应用程序通常不会孤立运行，因此请考虑将要使用的组件及其依赖关系作为故障转移策略的一部分。通常，多个应用程序协同工作以支持用户故事，这是为最终用户提供的一项特定功能，例如在社交媒体应用程序上发布图片和标题或在电子商务网站上结账。因此，您应该制定组织多区域故障转移策略，以提供必要的协调和一致性，使您的方法取得成功。

组织可以从四种高级策略中进行选择，以指导多区域方法。从最精细的方法到最广泛的方法列出了这些方法：

- 组件级故障转移
- 单个应用程序故障转移
- 依赖关系图故障转移
- 整个应用程序组合故障转移

每种策略都有权衡取舍，可以应对不同的挑战，包括故障转移决策的灵活性、测试故障转移组合的能力、模式行为的存在以及组织在规划和实施方面的投资。要更详细地了解每种策略，请参阅 AWS 博客文章《创建组织多区域故障转移策略》。

关键指导

- 审查所有 AWS 服务 配额，确保这些配额在工作负载将运行的所有区域中保持平衡。
- 部署过程应一次针对一个区域，而不是同时涉及多个区域。
- 诸如复制延迟之类的其他指标特定于多区域场景，应予以监控。
- 将工作负载的监控范围扩展到主区域之外。监控每个区域的客户体验指标，并衡量每个运行工作负载的区域以外的这些数据。
- 定期测试故障转移和故障恢复。为故障转移和故障恢复流程实施单一操作手册，并将其用于测试和实时活动。测试和直播活动的运行手册应该没有区别。
- 了解故障转移策略的权衡取舍。实施依赖关系图或整个应用程序组合策略。

结论和资源

本指南涵盖了多区域架构的常见用例、实现这些架构的基础知识以及这种方法的含义。您可以将这些基础知识应用于任何工作负载，并将这些信息用作框架，以帮助确定多区域架构是否适合您的业务。

有关更多信息，请参阅以下资源：

- AWS 建筑中心
- AWS 架构完善的框架
- AWS Well-Architected Tool
- 创建组织多区域故障转移策略（AWS 博客文章）
- AWS 多区域功能（AWS Re: Post 文章）

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
ARC 区域切换的更新	在第 3.c 和 4.d 节中，添加了有关用于处理恢复任务的 Amazon 应用程序恢复控制器 (ARC) 区域切换的信息。	2025年9月29日
更新	整个指南都进行了更新。	2024 年 12 月 27 日
初次发布	—	2022 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：将本地 Oracle 数据库迁移到 Amazon Aurora PostgreSQL 兼容版。
- 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中，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

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。

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。

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 AWS Well-Architected Guidance 中的 Implement break-glass procedures 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。

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

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 管线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）

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，以自定义您的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 Well-Architecte 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。

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 Well-Architected Fr ame AWS work 中的 “工作负载灾难恢复：云端 AWS 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。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

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。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

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。此技术是上下文内学习的一种应用，其中模型可以从提示中嵌入的示例（样本）中学习。对于需要特定格式、推理或领域知识的任务，少样本提示可能非常有效。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，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

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

帮助管理各组织单位的资源、策略和合规性的高级规则 (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。

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。

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

IaC

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 AWS Well-Architected Framework 中的使用不可变基础设施进行部署最佳实践。

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 Klaus Schwab 在 2016 年提出，指的是通过连接、实时数据、自动化、分析和 AI/ML 的进步来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。

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 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

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、记录和控制在车间将原材料转化为成品的生产过程。

MAP

请参阅迁移加速计划。

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 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

请参阅制造执行系统。

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、开发人员和冲刺 DevOps 领域的专业人员。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，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 machine-to-machine（M2M）通信协议。OPC-UA 提供了一个包含数据加密、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 Framework 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

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

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，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，相互独立，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。

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。你可以使用 Amazon S CloudWatch ynthetics 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

键值对，用作组织资源的元数据。 AWS 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。

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

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。