扩展 Amazon EKS 基础设施以优化计算、工作负载和网络性能

AWS 规范性指导

AWS 规范性指导: 扩展 Amazon EKS 基础设施以优化计算、工作负载和网络性能

Table of Contents

扩展 Amazon EKS 基础设施以优化计算、工作负载和网络性能

Aniket Dekate、Aniket Kurzadkar 和 Amazon Web Services 的 Ishwar Chauthaiwale ()AWS

2024 年 11 月（文件历史记录）

亚马逊 Elastic Kubernetes Service（亚马逊 EKS）是一项托管 Kubernetes 服务。借助 Amazon EKS，您可以在容器化的云环境中运行 Kubernetes pod，而无需安装和操作自己的控制平面。通过 AWS 管理控制平面，Amazon EKS 减少了组织运营管理。使用 Amazon EKS 的其他好处包括云环境中的扩展、可靠性和安全性。

本指南旨在帮助组织在以下领域优化其 Amazon EKS 基础设施：

- 在动态 Kubernetes 环境中，计算扩展是应用程序性能的关键组成部分：
  - 高效的资源分配-了解动态分配计算资源以满足不同需求的技术。
  - 自动化工具-概述可自动扩展计算的工具和服务，从而减少手动干预的需求。
- 工作负载扩展有助于确保应用程序能够在不降低性能的情况下处理不同的工作负载：
  - 横向 pod 自动扩缩器 — 深入了解 HPA 如何帮助根据实时指标扩展工作负载。
  - 集群比例自动扩缩器 — 了解 CPA 如何自动扩展和维护节点和副本之间的比例关系，随着集群规模的变化向上或向下扩展工作负载。
  - 事件驱动的扩展-查看针对特定事件或触发器扩展应用程序的策略。
- 网络扩展有助于在动态环境中保持服务之间的无缝通信和高效的数据流：
  - 亚马逊 VPC CNI 插件 — 了解 VPC CNI 插件如何在 Amazon EKS 集群中实现可扩展联网。
  - 自定义联网-查看 Amazon EKS 集群上的 IP 地址管理和网络流量隔离。
  - 前缀委托-概述如何在大型且可扩展的 Amazon EKS 集群中简化 IP 管理。
  - Amazon VPC Lattice — 概述 VPC 莱迪思如何管理跨虚拟私有云和 service-to-service网络以实现无缝扩展。
- 成本优化可帮助企业了解其资源的使用情况，并适当地将费用分配给部门或项目：
  - 适当@@ 调整资源规模-考虑针对工作负载适当调整云资源规模的技术。
  - 成本监控和控制 — 查看用于跟踪和优化云支出的工具和最佳实践。

每个部分都重点介绍创建可靠、有效且经济实惠的云环境所必需的特定目标。

目标

本指南可以帮助您和您的组织实现以下业务目标：

- 提高资源效率 — 根据实时需求动态扩展计算、工作负载和网络资源，实现最佳资源利用率。

  这一目标强调了根据实际使用模式向上和向下扩展资源的重要性。水平容器自动扩缩器和 Amazon VPC CNI 插件等工具可帮助组织仅使用所需的资源，从而最大限度地减少浪费并最大限度地提高性能。

- 提高应用程序性能-即使在工作负载和流量模式波动的情况下，也能保持应用程序的高性能和响应能力。

  该目标侧重于帮助确保应用程序能够在不影响性能的情况下处理峰值流量和繁重工作负载的策略。事件驱动的工作负载扩展、高效的计算分配和可扩展的网络架构等技术是实现这一目标的关键。

- 无缝可扩展性-支持基础设施组件的平稳扩展，从而可以轻松增长和适应不断变化的业务需求。

  对于预计增长或遇到不同流量水平的组织来说，无缝可扩展性至关重要。该目标探讨了跨计算、工作负载和网络资源实施可扩展解决方案的重要性，以便实现自动、高效和透明的扩展。

- 成本优化-最大限度地降低云成本，同时保持或提高性能和可扩展性。

  成本优化可以包括减少开支，例如合理调整资源规模、使用经济实惠的扩展解决方案以及监控支出。目标是在成本节省与对高性能和可扩展性的需求之间取得平衡。

计算缩放

计算扩展是动态 Kubernetes 环境中应用程序性能的关键组成部分。Kubernetes 通过根据实时需求动态调整计算资源（例如 CPU 和内存）来减少浪费。此功能有助于避免配置过度或配置不足，从而节省运营开支。Kubernetes 使基础设施能够在高峰时段自动向上扩展，在非高峰时段自动向下扩展，从而有效地消除了手动干预的需求。

Kubernetes 的整体计算扩展实现了扩展过程的自动化，从而提高了应用程序的灵活性和可扩展性，并增强了其容错行为。最终，Kubernetes 的功能可以提高卓越运营和生产力。

本节讨论以下类型的计算扩展：

- Cluster Autoscaler
- 带有预留空间的集群自动扩缩器
- Karpenter

集群 AutoScaler

根据容器的需求，Clu ster Autoscaler 工具会自动修改大小，方法是在必要时添加节点，或者在不需要节点且未充分利用时移除节点。

可以考虑将 Cluster Autoscaler 工具作为一种扩展解决方案，适用于需求逐渐增加且扩展延迟不是主要问题的工作负载。

集群自动扩缩器工具提供以下主要功能：

- 扩展-根据实际资源需求动态向上和向下扩展节点。
- Pod 调度 — 有助于确保每个 Pod 都在运行并拥有运行所需的资源，从而防止资源稀缺。
- 成本效益 — 通过消除未充分利用的节点来消除运营未充分利用的节点所产生的不必要开支。

带有预留空间的集群自动扩缩器

集群自动扩缩器具有超额配置功能，与集群自动扩缩器类似，因为它可以有效地部署节点，并通过在节点上运行低优先级的 pod 来节省时间。通过这种技术，流量会被重定向到这些容器中，以应对需求的突然激增，从而使应用程序能够不间断地继续运行。

具有预留空间的 Cluster Autoscaler 提供了虚拟 pod 的功能，当工作负载非常大、不需要延迟且需要快速扩展时，可以使用这些功能轻松部署和运行节点。

带预留空间的集群自动扩缩器提供以下主要功能：

- 更快的响应能力 — 通过持续访问多余的容量，可以缩短扩展集群以应对需求高峰所需的时间。
- 资源预留 — 有效管理意外流量高峰有助于在很少停机时间内进行正确的管理。
- 平滑扩展 — 最大限度地减少资源分配延迟有助于实现更加无缝的扩展过程。

Karpenter

Kubernetes 版 Karpenter 在开源、性能和可定制性方面的表现优于传统的集群自动扩缩工具。借助 Karpenter，您可以仅自动启动所需的计算资源来实时处理集群的需求。Karpenter 旨在提供更高效、响应更快的扩展。

使用Karpenter可以极大地受益于工作负载变化极大或复杂的应用程序，在这些应用程序中，快速做出扩展决策是必不可少的。它与集成 AWS，提供了改进的部署和节点选择优化。

Karpenter 包括以下主要功能：

- 动态配置 — Karpenter 为此目的提供正确的实例和大小，并根据 Pod 的特定要求动态配置新节点。
- 高级调度 — 通过巧妙的容器放置，Karpenter 可以安排节点，以便尽可能有效地使用 GPU、CPU、内存和存储等资源。
- 快速缩放 — Karpenter 可以快速缩放，经常在几秒钟内做出反应。这种响应能力对于突发流量模式或工作负载需要立即扩展时非常有用
- 成本效益 — 通过仔细选择最有效的实例，您可以降低运营成本并利用由提供的其他节省成本的替代方案 AWS，例如按需实例、竞价型实例和预留实例。

工作负载扩展

Kubernetes 中的工作负载扩展对于在动态环境中保持应用程序性能和资源效率至关重要。扩展有助于确保应用程序能够在不降低性能的情况下处理不同的工作负载。Kubernetes 提供了根据实时指标自动向上或向下扩展资源的功能，从而使组织能够快速响应流量变化。这种弹性不仅可以改善用户体验，还可以优化资源利用率，从而有助于最大限度地降低与未充分利用或过度配置资源相关的成本。

此外，有效的工作负载扩展支持高可用性，确保应用程序即使在需求高峰期也能保持响应能力。Kubernetes 中的工作负载扩展使组织能够通过动态调整容量来满足当前需求，从而更好地利用云资源。

本节讨论以下类型的工作负载扩展：

- 水平吊舱自动扩缩器
- 集群比例自动扩缩器
- 基于 Kubernetes 的事件驱动型自动扩缩器

Horizontal Pod Autoscaler

水平容器自动扩缩器 (HPA) 是一项 Kubernetes 功能，可根据观察到的 CPU 利用率或其他选定指标自动调整部署、复制控制器或状态集中的容器副本数量。HPA 确保应用程序无需手动干预即可管理波动的流量和工作负载级别。HPA 提供了一种在有效利用可用资源的同时保持最佳性能的方法。

在用户需求可能随着时间的推移而大幅波动的情况下，例如 Web 应用程序、微服务和 APIs HPA 特别有用。

水平吊舱自动扩缩器提供以下主要功能：

- 自动扩展 — HPA 会根据实时指标自动增加或减少 pod 副本的数量，从而确保应用程序可以扩展以满足用户需求。
- 基于指标的决策 — 默认情况下，HPA 基于 CPU 利用率进行扩展。但是，它也可以使用自定义指标，例如内存使用率或特定于应用程序的指标，从而可以制定更量身定制的扩展策略。
- 可配置的参数-您可以选择最小和最大副本数量以及所需的利用率百分比，从而可以控制扩展的严重程度。
- 与 Kubernetes 集成 — 为了监控和修改资源，HPA 与 Kubernetes 生态系统的其他元素协同工作，包括指标服务器、Kubernetes API 和自定义指标适配器。

- 提高资源利用率 — HPA 通过动态修改 pod 的数量，帮助确保资源得到有效利用，降低成本并提高性能。

集群比例自动扩缩器

集群比例自动扩缩器 (CPA) 是一个 Kubernetes 组件，旨在根据可用节点的数量自动调整集群中 Pod 副本的数量。与基于资源利用率指标（如 CPU 和内存）进行扩展的传统自动扩缩器不同，CPA 会根据集群本身的大小按比例扩展工作负载。

这种方法对于需要相对于集群规模保持一定冗余或可用性的应用程序（例如 CoreDNS 和其他基础设施服务）特别有用。CPA 的一些主要用例包括：

- 过度配置
- 横向扩展核心平台服务
- 扩展工作负载，因为 CPA 不需要指标服务器或 Prometheus 适配器

通过自动化扩展过程，CPA可以帮助企业保持平衡的工作负载分配，提高资源效率，并确保应用程序配置得当，以满足用户需求。

聚类比例自动扩缩器提供以下主要功能：

- 基于节点的扩展 — CPA 根据可以调度的集群节点数量来扩展副本，从而使应用程序能够与集群大小成比例地扩展或收缩。
- 按比例调整 — 为了确保应用程序可以根据集群大小的变化进行扩展，自动扩缩程序会在节点数量和副本数量之间建立成比例的关系。此关系用于计算工作负载所需的副本数量。
- 与 Kubernetes 组件集成 — CPA 可与标准 Kubernetes 组件（例如水平容器自动扩缩器 (HPA)）配合使用，但特别关注节点数量而不是资源利用率指标。这种集成允许更全面的扩展策略。
- Golang API 客户端 — 为了监控节点的数量及其可用内核，CPA 使用在 pod 中运行并与 Kubernetes API 服务器通信的 Golang API 客户端。
- 可配置参数 — 使用`ConfigMap`，用户可以设置阈值和缩放参数，CPA 使用这些参数来修改其行为，并确保其遵循预期的扩展计划。

基于 Kubernetes 的事件驱动型自动扩缩器

基于 Kubernetes 的事件驱动自动扩缩器 (KEDA) 是一个开源项目，它使得 Kubernetes 工作负载能够根据需要处理的事件数量进行扩展。KEDA 允许应用程序动态响应不同的工作负载，尤其是事件驱动的工作负载，从而增强了应用程序的可扩展性。

通过根据事件自动执行扩展过程，KEDA 可帮助组织优化资源利用率、提高应用程序性能并降低与过度配置相关的成本。这种方法对于遇到不同流量模式的应用程序（例如微服务、无服务器功能和实时数据处理系统）特别有价值。

KEDA 提供以下主要功能：

- 事件驱动的扩展 — KEDA 允许您根据外部事件源（例如消息队列、HTTP 请求或自定义指标）定义扩展规则。此功能有助于确保应用程序能够根据实时需求进行扩展。
- 轻量级组件 — KEDA 是一个单一用途的轻量级组件，无需大量设置或开销即可轻松集成到现有的 Kubernetes 集群中。
- 与 Kubernetes 集成 — KEDA 扩展了 Kubernetes 原生组件的功能，例如水平容器自动扩缩器 (HPA)。KEDA 为这些组件添加了事件驱动的扩展功能，增强而不是取代了它们。
- 支持多种事件源 — KEDA 与各种事件源兼容，包括 RabbitMQ、Apache Kafka 等流行消息平台。由于这种适应性，您可以自定义扩展以适应您独特的事件驱动架构。
- 自定义缩放器 — 使用自定义缩放器，您可以指定特定指标，KEDA 可以使用这些指标来启动扩展操作以响应特定的业务逻辑或要求。
- 声明式配置 — 根据 Kubernetes 原则，您可以使用 KEDA 声明性地描述扩展行为，方法是使用 Kubernetes 自定义资源来定义应如何进行扩展。

网络扩展

Kubernetes 中的网络扩展对于保持服务之间的无缝通信和支持动态环境中的高效数据流至关重要。扩展网络基础设施有助于确保集群能够处理不同级别的流量，而不会遇到瓶颈或延迟问题。Kubernetes 提供了扩展网络资源的工具和机制，使组织能够在流量模式变化时保持最佳性能。

网络扩展的这种弹性可确保快速可靠的连接，从而增强整体用户体验。网络扩展还可以优化网络资源的使用，有助于降低与未充分利用或负担过重的网络组件相关的成本。

此外，有效的网络扩展对于支持高可用性和弹性至关重要。通过动态调整网络容量和路由，组织可以确保即使在需求高峰期或意外流量高峰期，服务也能保持可访问性和响应能力。这种方法可以更好地利用云网络资源，确保基础架构始终与当前需求保持一致。

本节讨论以下类型的网络扩展：

- 适用于 Kubernetes 的 Amazon VPC CNI 插件
- 自定义网络
- 前缀委托
- Amazon VPC Lattice

适用于 Kubernetes 的 Amazon VPC CNI 插件

适用于 Kubernetes 的亚马逊 VPC 容器网络接口 (CNI) 插件是亚马逊 EKS 中的关键组件。VPC CNI 插件通过将 Kubernetes 容器与亚马逊 VPC 集成，提供高级联网功能。使用此插件，每个 Pod 都将获得来自虚拟私有云 (VPC) 的唯一 IP 地址，从而增强网络隔离和性能。随着集群的增长和网络需求的波动，Amazon VPC CNI 插件在确保高效和可扩展的网络运营方面发挥着关键作用。

该插件可自动管理 VPC 内 IP 地址的分配和路由，从而简化网络管理并降低 IP 冲突风险。它支持诸如前缀委派之类的功能，从而实现更灵活的 IP 管理。

VPC CNI 插件可帮助组织优化网络性能、增强安全性并降低 IP 耗尽风险。这些功能对于网络需求波动的大规模、动态环境尤其有价值，例如微服务架构、高密度工作负载和多租户应用程序。

Amazon VPC CNI 插件提供以下主要功能：

- 增强联网 — VPC CNI 插件允许每个 Pod 直接从 VPC 接收自己的 IP 地址，从而提供强大的隔离和网络性能。这种方法对于需要高网络吞吐量和低延迟的工作负载至关重要。

- 前缀委托 — 为了克服大型集群中的 IP 地址耗尽问题，前缀委派会动态地将较大的区块分配 IPs 给节点，然后对其进行细分以供 Pod 使用。这种方法可确保高效的 IP 利用率并简化网络扩展。
- 自定义网络 — 用户可以为 pod 配置自定义网络接口 (ENIs)，这有助于在多个接口之间分配 Pod 流量，从而减少网络拥塞并提高可扩展性。
- S@@ upp ort for IPv6 — 通过 IPv6 在 Amazon EKS 集群中启用，用户可以显著扩展可用的 IP 地址空间，从而在不受 IPv4 限制的情况下促进大型分布式应用程序的扩展。
- 与 Kubernetes 集成 — VPC CNI 插件可与 Kubernetes 网络组件无缝协作，确保 IPs 跨容器、服务和外部端点进行高效管理，并支持高级功能，例如容器的安全组。

自定义联网

Amazon EKS 中的自定义联网允许向 Pod 分配特定的网络接口，从而增强对 IP 地址管理和网络流量的控制。在 IP 地址耗尽问题或出于安全、合规性或性能原因需要隔离网络流量的情况下，这种方法特别有用。自定义网络可帮助组织高效管理 IP 地址空间、隔离流量并确保可扩展的网络性能。

通过自定义网络，管理员可以更有效地管理网络资源。管理员可以使用自定义网络来帮助确保 Pod 具有必要的网络隔离，并且集群可以在不遇到 IP 地址限制的情况下进行扩展。

自定义网络提供以下主要功能：

- 增强的 IP 管理 — 自定义网络允许将特定的网络接口 (ENIs) 分配给 pod，通过在多个 ENIs pod 之间分配 pod 流量来帮助管理 IP 地址耗尽。此功能在具有高密度工作负载的集群中尤为重要。
- 流量隔离 — 使用自定义网络接口，您可以根据特定标准（例如应用程序类型或安全要求）分离 Pod 流量。这种方法可以更好地控制集群内外的流量流向。
- S upport for IPv6 — Amazon EKS 中的自定义联网也支持 IPv6，为 IPv4地址限制提供了解决方案。即使在大规模部署中，网络也可以高效扩展，而不会发生 IP 地址冲突。
- 可扩展性和灵活性 — 随着集群的扩展，自定义网络可以实现对网络接口的动态管理。无需人工干预即可为新 Pod 分配适当的网络资源。这种方法有助于维护灵活且可扩展的网络环境，以适应不断变化的工作负载。

前缀委派

Kubernetes 中的前缀委托，尤其是 Amazon EKS 中的前缀委托，旨在随着集群规模的扩展，简化和优化 IP 地址管理。通过向节点动态分配更大的 IP 地址块（前缀），前缀委派降低了 IP 耗尽的风险并简化了 IP 空间的管理。

这种方法可以提高网络效率，最大限度地减少分段，并帮助集群平稳扩展，无需手动调整 IP 范围。前缀委派对于大规模部署、高密度工作负载以及灵活、动态 IP 管理对于维持网络性能和可扩展性至关重要的环境尤其重要。

前缀委派提供以下主要功能：

- 高效的 IP 地址管理 — 前缀委派允许动态分配 IP 范围，从而降低 IP 耗尽的风险并确保有效利用可用 IP 空间。
- 简化网络管理 — 通过允许节点处理自己的 IP 分配，前缀委派可最大限度地减少网络碎片并简化路由过程，从而更轻松地根据需要扩展集群。
- 支持大规模部署 — 在具有高密度工作负载的大型集群中，前缀委派允许新节点加入集群，无需手动调整 IP 范围，即可实现无缝扩展。

Amazon VPC Lattice

Amazon VPC Lattice 支持在内部和相互之间进行高效 VPCs、安全的 service-to-service通信，尤其是在微服务架构中。VPC Lattice 除了集成（IAM）外，还使用安全组和网络访问控制列表 AWS Identity and Access Management （网络 ACLs）等安全措施来实现精细的应用程序身份验证。莱迪思是VPC核心的第七层代理服务，提供连接、负载平衡、身份验证、授权、可观察性、流量管理和服务发现。

通过简化网络和安全配置，VPC Lattice帮助组织优化流量管理，增强应用性能，并在多个 VPCs 和 AWS 区域之间无缝扩展。这对于需要一致和可靠网络的分布式应用程序尤其重要，例如微服务、跨区域部署和复杂的云原生环境。

Amazon VPC Lattice 提供以下主要功能：

- Service-to-service 联网 — VPC Lattice简化了微服务架构中服务之间的联网和安全配置。它为管理通信提供了一个统一的平台，因此服务可以在保持高性能和安全性的同时独立扩展。
- 跨VPC网络 — VPC Lattice对于管理跨多个 VPCs 或区域的流量至关重要。它提供了一个一致的网络框架，允许服务无缝通信，无论其物理位置如何。此功能对于跨越多个 VPCs 或地理区域的大型应用程序尤其重要。
- 增强安全管理 — 通过将安全策略直接集成到网络层，VPC Lattice支持既安全又高效的 service-to-service通信。此功能降低了在分布式环境中管理安全的复杂性，从而可以更轻松地扩展并减少运营开销。
- 简化流量管理 — VPC Lattice提供高级流量管理功能，包括路由、负载平衡和故障转移机制。借助这些功能，可以在服务之间高效地分配流量，从而优化网络性能并增强应用程序的可扩展性。

成本优化

为了支持有效的资源控制，Kubernetes 成本最小化对于使用这种容器编排技术的企业至关重要。由于Kubernetes设置的复杂性，包括多个组件，例如Pod和节点，因此很难正确跟踪其支出。通过应用成本优化技术，企业可以看到其资源的使用情况，并适当地将费用分配给部门或项目。

尽管动态扩展具有优势，但如果管理不当，可能会导致不可预见的开支。高效的成本管理有助于仅在真正需要时才分配资源，从而避免意想不到的支出激增。

本节讨论以下成本优化方法：

- Kubecost
- Goldilocks
- AWS Fargate
- 竞价型实例
- 预留实例
- AWS Graviton 实例

Kubecost

Kubecost 是一种成本管理解决方案，可帮助企业跟踪、控制和最大限度地提高其在云基础设施上的支出。它是专门为 Kubernetes 集群制作的。Kubecost 可让您深入了解资源利用率和实时成本意识，使您能够更好地了解云资源的使用位置和使用量。借助这些见解，您可以优化基础架构支出，提高资源效率，并就云投资做出更明智的决策。

Kubecost 提供了以下主要功能：

- 成本分配 — Kubecost 为 Kubernetes 资源（包括工作负载、服务、命名空间和标签）提供全面的成本分配。此功能可帮助团队按环境、项目或团队监控成本。
- 实时成本监控 — 它提供对云成本的实时监控，使组织可以立即了解支出模式，并有助于防止意外的成本超支。
- 优化建议 — Kubecost 为最大限度地减少资源利用率提供了实用建议，包括减少闲置资源、调整工作负载规模和最大限度地提高存储开支。
- 预算和提醒 — Kubecost用户可以创建预算，并在支出接近或超过预定标准时收到提醒。此功能可帮助团队遵守财务限制。

金发姑娘

Goldilocks 是一个 Kubernetes 实用程序，旨在帮助用户优化 Kubernetes 工作负载的资源请求和限制。它提供了有关如何为 Kubernetes 集群中运行的容器配置 CPU 和内存资源的建议。这些建议可帮助您确保应用程序拥有适当数量的资源，以便在不浪费的情况下高效运行。这种优化可以节省成本、提高性能并更有效地使用 Kubernetes 集群。

Goldilocks 提供以下主要功能：

- 资源建议 — Goldilocks 通过分析 Kubernetes 工作负载过去的 CPU 和内存消耗统计数据来确定资源请求和限制的理想设置。通过这样做，可以更轻松地避免配置不足或过度配置，这可能会导致性能问题和资源浪费。
- VPA 集成 — Goldilocks 利用 Kubernetes Vertical Pod Autoscaler (VPA) 来收集数据并提供建议。它在 “推荐模式” 下运行，这意味着它实际上不会更改资源设置，但会提供有关这些设置的指导。
- 基于命名空间的分析 — Goldilocks 允许您定位特定的命名空间进行分析，从而使您能够精细调节优化和监控哪些工作负载。
- 可视化仪表板 — 基于 Web 的仪表板直观地显示建议的资源请求和限制，这使您可以直接了解数据并对数据采取行动。
- 非侵入性操作 — Goldilocks 不会更改集群的设置，因为它在推荐模式下运行。如果需要，可以在查看建议后手动应用推荐的资源设置。

AWS Fargate

在 Amazon EKS 的背景下，https://docs.aws.amazon.com/eks/latest/userguide/fargate.htmlAWS Fargate 允许您在不管理底层亚马逊实例的情况下运行 Kubernetes pod。 EC2 它是一个无服务器计算引擎，可让您专注于部署和扩展容器化应用程序，而不必担心基础架构。

AWS Fargate 提供以下主要功能：

- 无需基础设施管理 — Fargate 无需预置、管理或扩展 Amazon EC2 实例或 Kubernetes 节点。 AWS 负责所有基础架构管理，包括修补和扩展。
- Pod 级隔离 — 与基于 Amazon 的工作节点不同， EC2Fargate 提供任务或舱级隔离。每个 Pod 都在自己的隔离计算环境中运行，从而增强了安全性和性能。
- 自动扩展 — Fargate 根据需求自动扩展 Kubernetes 容器。您无需管理扩展策略或节点池。
- 按秒计费 — 您只需为每个 Pod 在运行的确切时间内消耗的 vCPU 和内存资源付费，对于某些工作负载来说，这是一种经济实惠的选择。

- 减少开销 — Fargate 无需管理 EC2 实例，使您可以专注于构建和管理应用程序，而不是基础设施运营。

竞价型实例

与按需@@ 实例定价相比，竞价型实例可节省大量费用，并且是在 Amazon EKS 集群中运行 Amazon EC2 工作节点的经济实惠的选择。但是，在需要按需实例容量时，AWS 可能会中断竞价型实例。AWS 当需要容量时，可以在 2 分钟通知的情况下回收竞价型实例，从而降低它们对关键的有状态工作负载的可靠性。

对于对成本敏感且能够承受中断的工作负载，Amazon EKS 中的竞价型实例是一个不错的选择。在 Kubernetes 集群中结合使用竞价型实例和按需实例可以帮助您在不牺牲重要工作负载可用性的情况下节省资金。

竞价型实例提供以下主要功能：

- 节省成本 — 竞价型实例可能比按需实例定价便宜，因此非常适合成本敏感型工作负载。
- 容错工作负载的理想之选 — 非常适合无状态的容错工作负载，例如批处理、CI/CD 作业、机器学习或大规模数据处理，在这些工作负载中，可以在不发生重大中断的情况下更换实例。
- 自动扩展组集成 — Amazon EKS 将竞价型实例与 Kubernetes 集群自动扩缩器集成，后者可以自动将中断的竞价型实例节点替换为其他可用的竞价型实例或按需实例。

预留实例

在 Amazon EKS 中，预留实例是运行 Kubernetes EC2 工作负载的亚马逊工作节点的定价模型。使用预留实例即表示您承诺在 1 年或 3 年期内使用特定的实例类型，以换取与按需实例定价相比节省的成本。在 Amazon EKS 中预留实例是一种经济实惠的方式，可以在 Amazon EC2 工作节点上执行一致的长期工作负载。

预留实例通常用于 Amazon EC2。但是，您的 Amazon EKS 集群中的工作节点（即 EC2 实例）也可以从这种节省成本的模式中受益，前提是工作负载需要长期、可预测的使用量。

生产服务、数据库和其他需要高可用性和一致性能的有状态应用程序就是非常适合预留实例的稳定工作负载的示例。

预留实例提供以下主要功能：

- 节省成本 — 与按需实例相比，预留实例可节省开支，具体取决于期限（1 或 3 年）和付款计划（全额预付、部分预付或无预付）。
- 长期承诺 — 对于特定的实例类型、大小和，您承诺期限为 1 年或 3 年。 AWS 区域对于稳定且随时间推移持续运行的工作负载而言，这是理想的选择。
- 可预测的定价 — 由于您承诺使用特定的期限，因此预留实例可提供可预测的月度或前期成本，从而更轻松地为长期工作负载制定预算。
- 实例灵活性-使用可转换预留实例，您可以在预留期内更改实例类型、系列或大小。与不允许更改的标准预留实例相比，可转换预留实例具有更大的灵活性。
- 有保障的容量 — 预留实例可确保进行预留的可用区内有可用容量，这对于需要稳定计算能力的关键工作负载至关重要。
- 无中断风险 — 与竞价型实例不同，预留实例不会受到以下因素的中断 AWS。这使得它们非常适合运行需要保证正常运行时间的关键任务工作负载。

AWS Graviton 实例

AWS Graviton 是一系列基于 ARM 的处理器，旨在为云工作 AWS 负载提供更高的性能和成本效益。在 Amazon EKS 的背景下，您可以使用 Graviton 实例作为工作节点来运行 Kubernetes 工作负载，从而显著提高性能并节省成本。

Graviton 实例是云原生和计算密集型应用程序的绝佳选择，因为它们的性价比高于 x86 实例。但是，在考虑采用 Graviton 实例时，请考虑 ARM 的兼容性。

AWS Graviton 实例提供以下主要功能：

- 基于 ARM 的架构 — AWS Graviton 处理器基于 ARM 架构构建，这与传统的 x86 架构不同，但对于许多工作负载来说效率很高。
- 成本效益 — 与基于 x86 的 EC2 实例相比，基于 Graviton 的 Amazon 实例通常具有更好的性价比。EC2对于运行 Amazon EKS 的 Kubernetes 集群来说，这使得它们成为一个有吸引力的选择。
- 性能 — 第二代 Gravit AWS on Graviton2 处理器在计算性能、内存吞吐量和能效方面提供了显著改进。它们非常适合 CPU 密集型和内存密集型工作负载。
- 不同的实例类型 — Graviton 实例有不同的系列，例如 t4g、m7g、c7g 和 r7g，涵盖了从通用到计算优化、内存优化和可突发性工作负载的一系列用例。
- Amazon EKS 节点组 — 您可以将由 Amazon EKS 管理的节点组或自行管理的节点组配置为包括基于 Graviton 的实例。通过这种方法，你可以在同一 Kubernetes 集群上运行针对 ARM 架构进行优化的工作负载以及基于 x86 的实例。

后续步骤

本指南提供的信息可帮助您在计算扩展、工作负载扩展、网络扩展和成本优化方面优化 Amazon EKS。通过理解和应用这些概念，组织可以实现高效、可扩展且经济实惠的云环境，以满足其动态需求。

有效实施计算和工作负载扩展有助于确保资源得到有效利用，即使在高峰时段，应用程序也能保持高性能。采用网络扩展技术（例如自定义联网和前缀委派）支持网络资源管理和无缝可扩展性。强调成本优化有助于组织在绩效与财务效率之间取得平衡。

将本指南整合到您的云战略中，可以帮助您增强基础架构的性能和可扩展性，并节省成本。这种全面的方法使您能够构建一个强大的云环境，以支持组织的发展并适应不断变化的业务需求。

资源

AWS 博客

- 使用 Spot 实例为 EKS 进行成本优化和弹性构建
- 使用 Amaz AWS on EKS 将 Graviton 与 x86 混合使用 CPUs 以优化成本和弹性

AWS 文档

- Amazon VPC CNI
- 亚马逊 Elastic Kubernetes Ser vice AWS （白皮书：部署选项概述） AWS
- 亚马逊 EKS 最佳实践指南
- Karpenter
- 了解有关 Kubecost 的更多信息
- 通过以下方式简化计算管理 AWS Fargate

其他资源

- 集群自动扩展（Kubernetes 文档）
- Goldilocks：用于推荐资源请求的开源工具（Fairwinds 博客）
- 容器水平自动缩放（Kubernetes 文档）
- Kubecost（Kubecost 文档）
- Kubernetes 事件驱动的自动扩展（KEDA 文档）

文档历史记录

下表描述了本指南的重大更改，即扩展 Amazon EKS 基础设施以优化计算、工作负载和网络性能。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 11 月 11 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、耐久性。

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。

AIOps

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-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

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 Fr ame 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。

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

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、记录和控制在车间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： 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-Architecte 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 Secret s 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。你可以使用 Amazon S CloudWatch ynthetics 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。