使用無 AWS 伺服器服務整合微服務

AWS 方案指引

AWS 方案指引: 使用無 AWS 伺服器服務整合微服務

Table of Contents

使用無 AWS 伺服器服務整合微服務

Tabby Ward、Abhishek Agawane 和 Amazon Web Services Matt Kahn

2025 年 9 月 (文件歷史記錄)

現代化組織軟體的一個重要方面是選擇正確的架構模式，以靈活敏捷地回應不斷變化的業務需求。在某些應用程式中，單體架構是常見的選擇。不過，對於許多組織而言，當使用案例符合微服務的優勢時，將單體重構為微服務可能是一種有效的現代化策略。

微服務和單體並不互斥，許多成功的組織會同時使用兩種模式，其中模組化單體提供一些網域，而微服務會處理其他網域。

當微服務是架構的一部分時，可能會呼叫數個服務來擷取一個商業交易的資料。實作這些整合需要仔細設計，以解決潛在的挑戰，例如資料一致性、延遲和操作複雜性。當微服務正確整合時，它們可以提供優勢，例如獨立擴展、改善開發速度和潛在的成本最佳化。

本指南是內容系列的一部分，涵蓋 建議的應用程式現代化方法 AWS。系列也包含：

- 中的應用程式現代化策略 AWS 雲端
- 中的應用程式現代化分階段方法 AWS 雲端
- 在 中評估應用程式的現代化準備程度 AWS 雲端
- 將整體分解為微服務

目標對象

本指南適用於應用程式擁有者、企業擁有者、架構師、技術主管和專案經理，他們判斷微服務適合其特定使用案例。本指南透過使用 和 Amazon API Gateway AWS Lambda AWS 服務 等無伺服器來實現自主性和可擴展性，引入了微型服務之間同步和非同步通訊的多種模式。

目標

透過使用本指南整合新的微服務，您可以有效率地將組織的架構轉換為微服務架構。這有助於透過高可擴展性、改善彈性、持續交付和故障隔離，快速調整以因應業務需求波動。微服務架構也會加速創新，因為每個微服務都可以個別部署和測試。

微服務架構也有助於為您的產品或服務提供更短的上市時間，因為每個微服務都有獨立的程式碼基底，可讓您更輕鬆、更快速地新增新功能並反覆使用這些功能。

安全

您必須正確保護微服務，以保護服務和資料的完整性，但確保安全性不會對應用程式的效能造成負面影響。

在微服務環境中，您必須考慮每個服務如何驗證和授權從外部用戶端或其他微服務收到的請求。同時考慮每個服務如何安全地存取其他服務 AWS 服務。

AWS 服務 應透過縮小範圍 AWS Identity and Access Management (IAM) 角色授予 的存取權。假設 IAM 角色以存取金鑰、存取秘密和工作階段字符的形式提供具有短期 IAM 憑證的微服務。這些由各種軟體開發套件 (SDKs) 使用 AWS 服務 AWS Signature 第 4 版 (SigV4) 對 簽署請求。

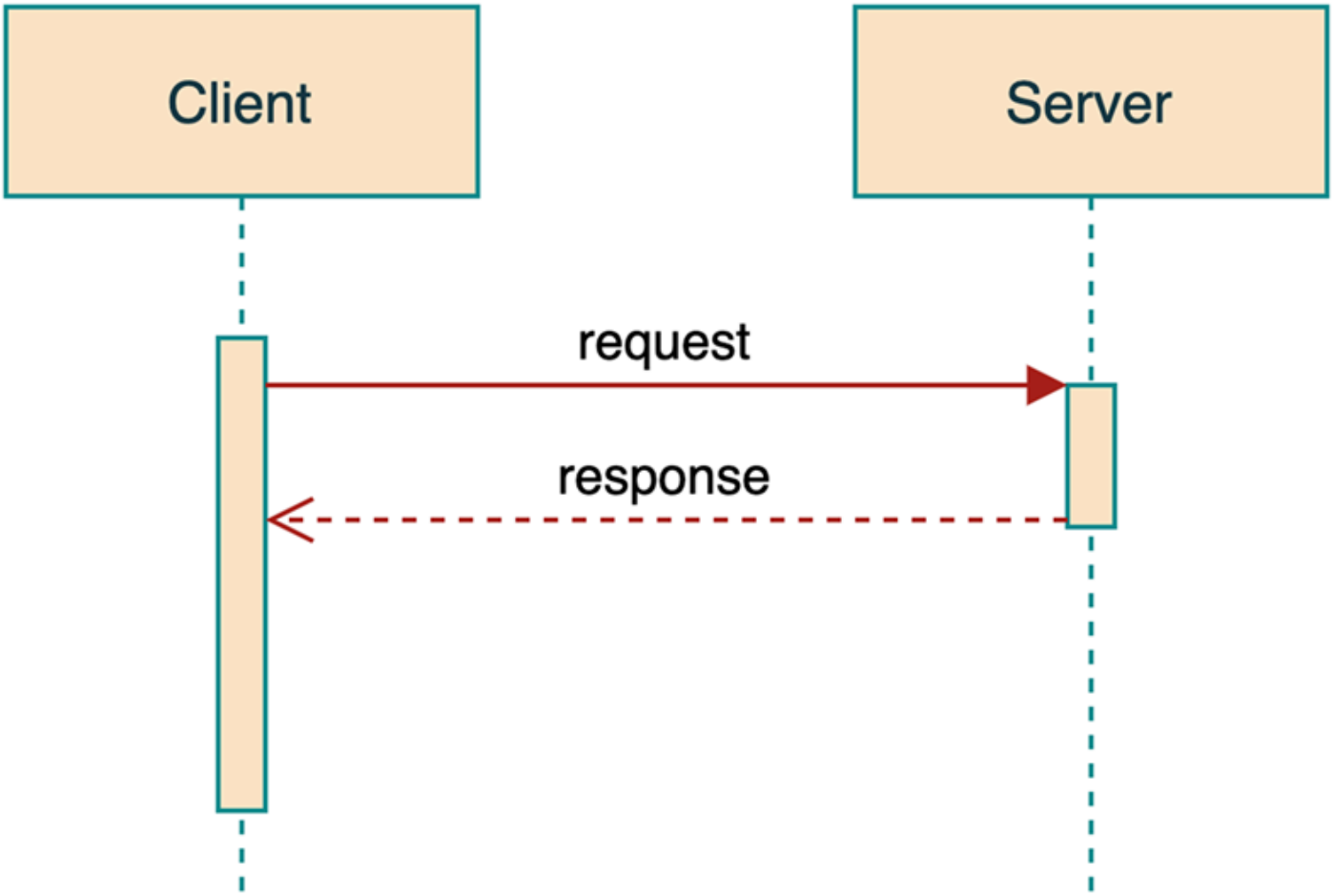
通訊模式

在微服務架構中，通訊會以兩種主要模式進行：同步和非同步。在同步通訊中，呼叫者會等待回應再繼續，類似於即時 HTTP REST API 呼叫。非同步通訊遵循以訊息為基礎的模式，呼叫者會繼續處理而不等待回應，例如使用訊息佇列時。下列各節會詳細檢查每個模式的實作、優點和使用案例。

主題

同步通訊

在同步通訊中，用戶端會啟動對服務的請求，如下圖所示。範例包括擷取資訊的請求，例如HTTP GET請求，或變更資料的請求，例如HTTP PUT請求。在任何一種情況下，用戶端都會等待伺服器回應，然後再繼續。大多數開發人員都熟悉同步呼叫，易於實作和疑難排解，而且在許多情況下， 是廣泛接受的通訊標準。



同步通訊的優點包括：

- 可預測的流程控制 – 確定性執行和清除請求回應週期，相較於非同步通訊，這些週期更容易理解。
- 高度一致性 – 立即確認資料變更和狀態更新。
- 簡單錯誤處理 – 直接傳播錯誤和例外狀況。
- 簡易偵錯 – 直接的請求追蹤和監控。
- 通訊協定支援 – 建立良好的通訊協定，例如 HTTP 和 REST，讓實作變得簡單。

同步通訊有一些缺點：

- 緊密耦合 – 服務對彼此可用性的直接相依性。
- 網路影響 – 由於持續開放連線而增加網路負載。
- 資源使用率 – 維護連線狀態的記憶體使用量較高。
- 串聯失敗 – 能夠讓一個服務中的問題快速傳播到系統中。

非同步通訊

相反地，在非同步通訊中，用戶端會向服務發出請求，但不會立即收到回應。在此情況下，用戶端通常只會收到接受請求的確認。

非同步通訊的優點包括：

- 事件驅動型架構支援 – 自然適用於事件來源和命令查詢責任隔離 (CQRS) 模式。
- 更好的資源管理 – 服務能夠根據其容量處理請求。
- 改善故障隔離 – 解耦 服務，防止層疊失敗。
- 峰值負載處理 – 透過訊息佇列更好地處理流量峰值。

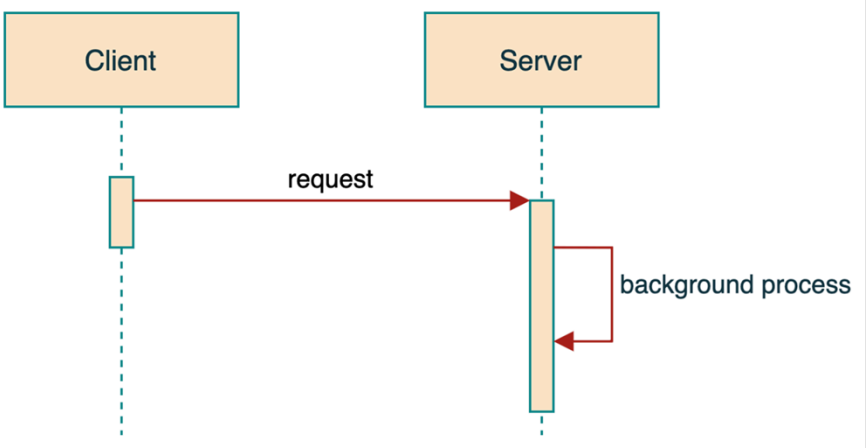
缺點包括複雜性。例如：

- 如果用戶端需要非同步操作的結果，則實作機制來擷取或接收該結果需要更多精力。
- 疑難排解非同步操作可能更困難，因為疑難排解需要跨多個系統檢查日誌。
- 測試非同步操作可能更困難，因為測試需要在多個系統和服務之間進行協調。

非同步通訊的方法包括火災和忘記、宣告檢查、回呼和雙向通訊。

Fire 和 忘記

在觸發和忘記模式中，用戶端會向伺服器發出請求，並同步接收確認，指出伺服器已收到訊息並進行處理。不過，實際處理尚未發生，而且用戶端無法查看何時或如何完成。下圖說明此模式。



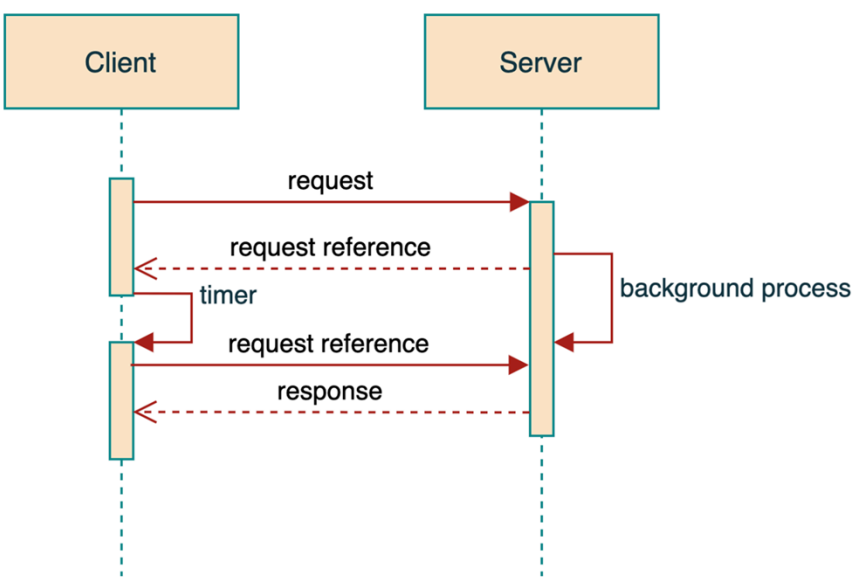
在此情況下，在物件持久保留之前，服務不應傳送確認。此持久性可以實作為資料庫寫入操作或將項目放入佇列中。

其他考量：

- 實作冪等性來處理重複的訊息。也就是說，每個訊息只應處理一次。
- 考慮無效字母佇列來處理失敗。
- 監控訊息處理成功率。

宣告檢查

如果用戶端需要服務呼叫的結果，您可以建置服務，在收到請求時發出宣告檢查。下圖說明此模式。宣告檢查會實作為服務在其確認中傳回的識別符。用戶端稍後可以使用此識別符來檢查請求的狀態，並在請求完成時擷取結果。



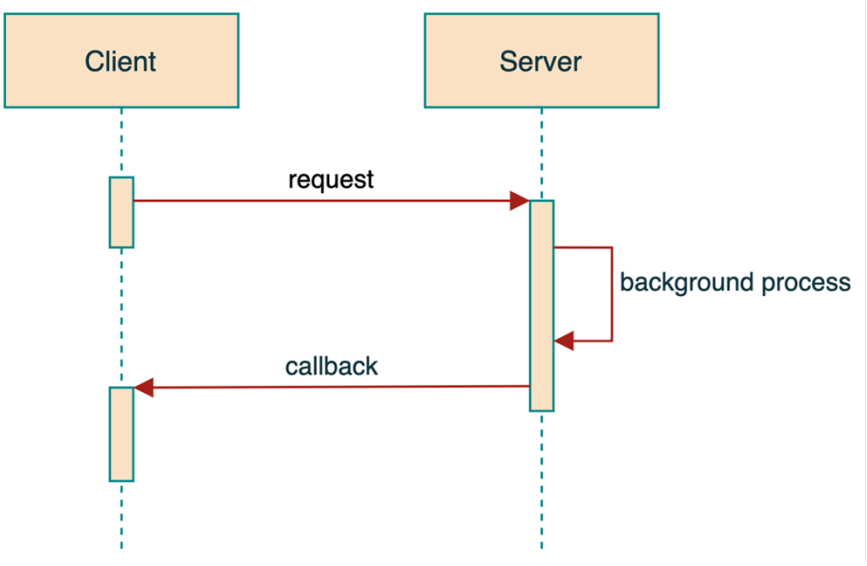
用戶端必須實作機制來輪詢結果。這可以是自動化（例如，每 n 分鐘可以執行一次檢查）或手動實作，其中執行檢查是為了回應另一個事件或使用者動作。實作宣告檢查模式的服務應明確宣告檢查有效的時間長度。

最佳實務：

- 實作指數退避輪詢。
- 為宣告檢查設定適當的存留時間 (TTL)。
- 提供狀態端點以進行進度追蹤。

回電

在回呼模式中，用戶端會向服務發出請求，並提供服務在處理完成時聯絡的位置。用戶端不會等待結果，而處理會繼續。服務負責在處理完成時聯絡 據點並提供結果。回應的常見位置類型為 REST APIs 或佇列。下圖說明回呼模式。

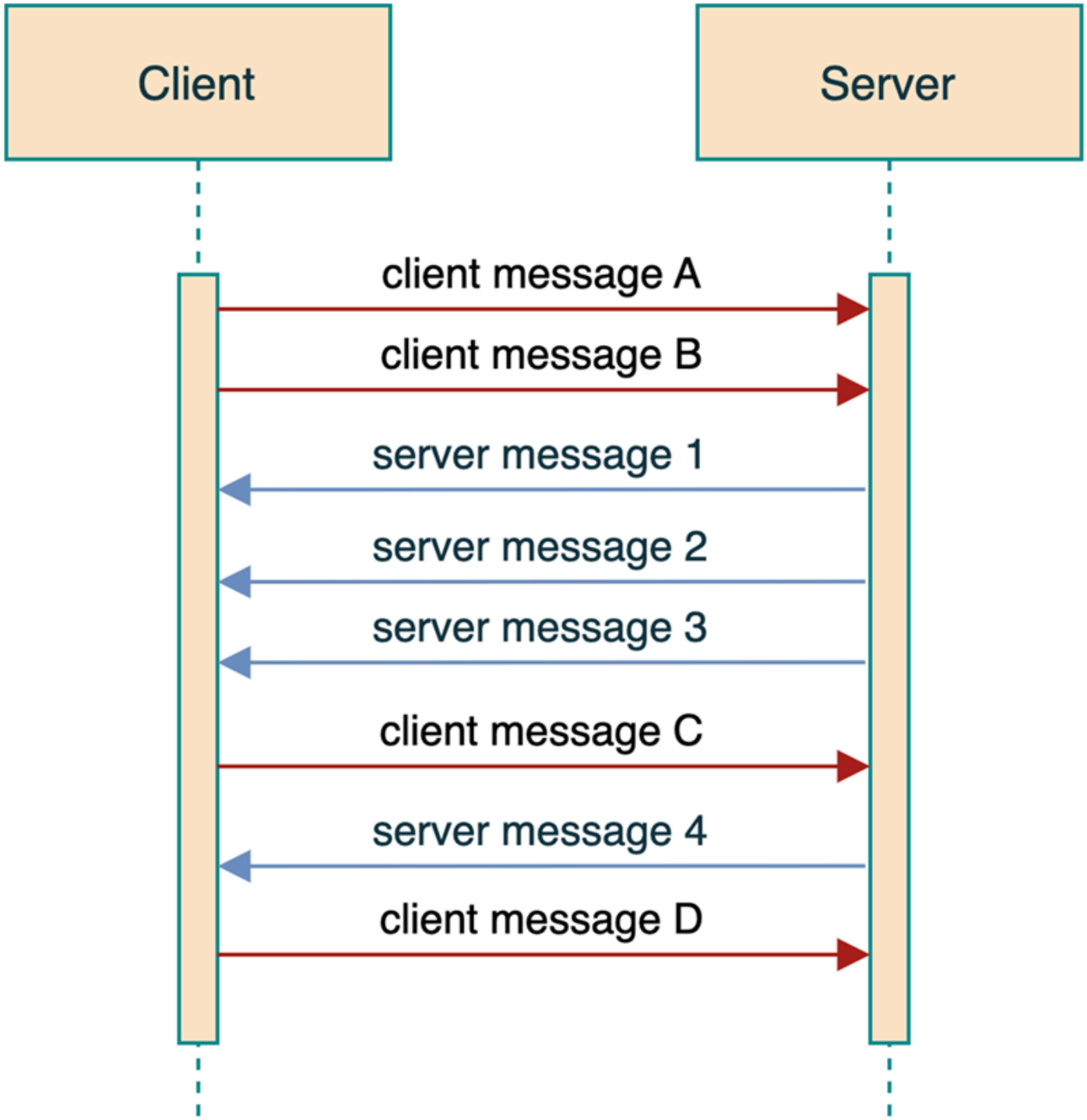


實作：

- 實作失敗回呼的重試機制。
- 像其他 服務一樣保護回呼位置。
- 處理回呼逾時。

雙向通訊

若要實作雙向通訊，您必須在用戶端和服務之間建立具狀態的連線，這可讓用戶端和服務傳送和處理訊息。這會在下列圖表中進行說明。雖然通訊是非同步的，但服務必須能夠支援每個用戶端的開放連線。


Client
Server
client message A
client message B
server message 1
server message 2
server message 3
client message C
server message 4
client message D

實作考量：

- 訊息排序
  - 序號

  - 分割區策略
  - 訊息排序
- 狀態管理
  - 事件來源模式
  - 狀態對帳
  - 一致性模式
- 錯誤處理
  - 無效字母佇列
  - 重試政策
  - 斷路器
  - 備用策略
- 監控與可觀測性
  - 關聯 IDs
  - 訊息追蹤
  - 效能指標
  - 系統運作狀態指標

協調選項

同步和非同步通訊非常適合呼叫單一服務或少量服務的用戶端。不過，在真實世界環境中，此通訊可能會快速變得複雜且難以擴展。完成工作單位可能需要數個微服務，這些服務可能有相互依存性。通常，這些互動會建模為工作流程。設計這些工作流程的方法有兩種：協同運作和編排。

主題

協調

在此方法中，單一協調器負責呼叫每個微服務、判斷是否依序或平行發出呼叫、沿途操作個別服務回應，以及編譯最終結果。協調器可以混合同步和非同步調用。

AWS Step Functions 和 Amazon Managed Workflows for Apache Airflow (Amazon MWAA) 是工作流程協調器的絕佳選擇。

當您的程序中有邏輯分支，而且您需要單一位置來封裝該邏輯時，協調是不錯的選擇。當您想要實作非同步宣告檢查模式時，這也很有用。例如，Step Functions 中的標準工作流程可以暫停工作流程，並等待來自其他服務的回呼。使用協調器也可改善程序的監控和可觀測性。

範例：Step Functions

您可以使用 Step Functions 來協調多個 Lambda 函數和其他函數 AWS 服務，以建置用於微服務整合的複雜工作流程。此選項對於涉及數個微服務的長期執行、多步驟程序特別有用。

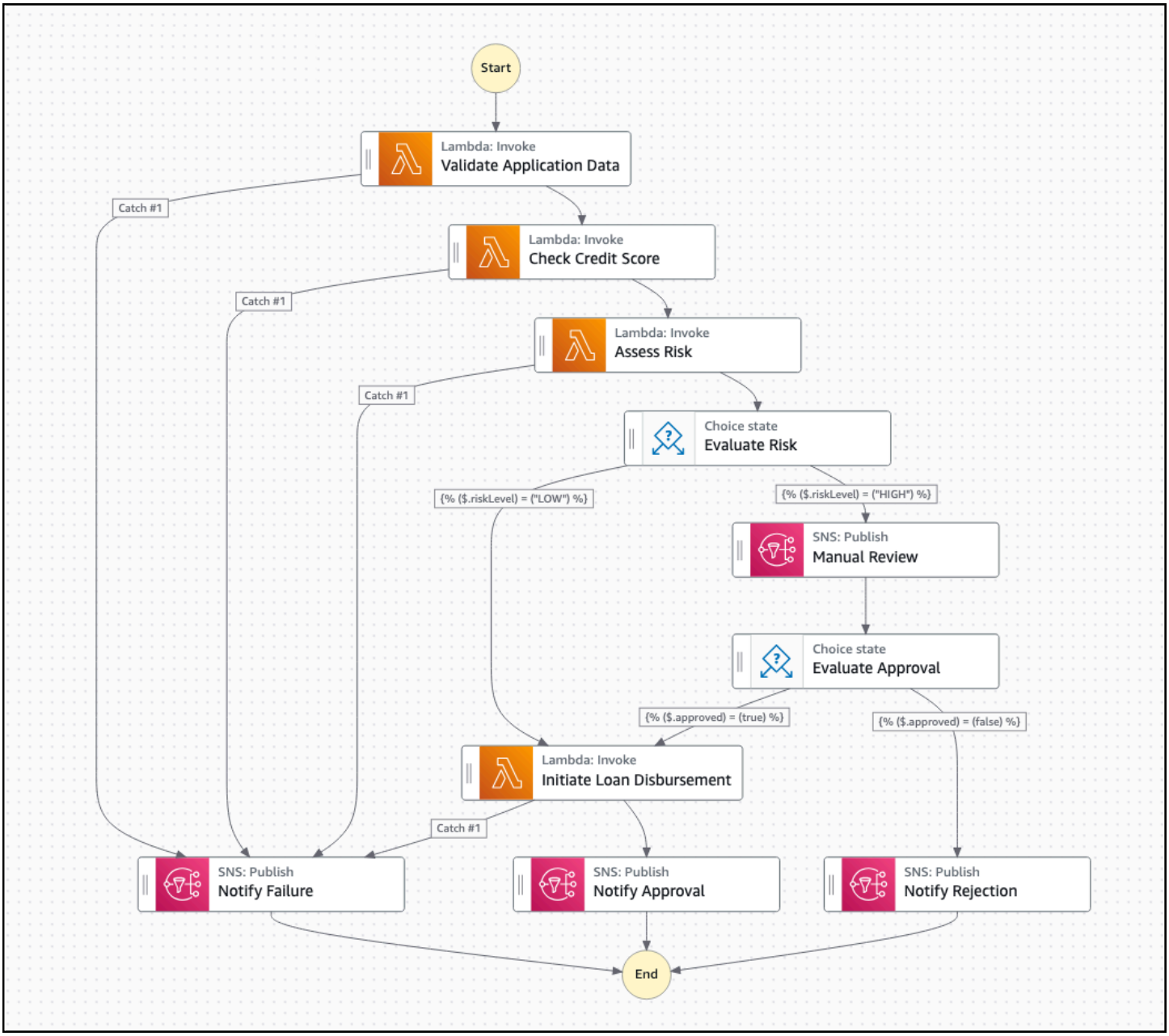
在以下情況下，您應該考慮使用 Step Functions：

- 您的微服務整合涉及複雜的多步驟程序。
- 您需要維持長時間執行操作的狀態。
- 您想要在工作流程層級實作錯誤處理和重試邏輯。
- 您需要同時協調同步和非同步操作。

Step Functions 提供用於設計複雜工作流程的視覺化編輯器，可簡化建立和管理狀態機器的程序。它提供內建的錯誤處理機制，包括重試邏輯和錯誤狀態管理，可增強應用程式的可靠性和耐用性。標準工作

流程支援長達一年的長時間執行程序，這適用於跨越較長期間的工作流程。此選項會將協同運作邏輯與應用程式程式碼分開，因此可大幅降低程式碼複雜性。這表示開發人員可以專注於核心商業邏輯，而 Step Functions 會處理分散式元件的流程控制和協調。

例如，考慮金融服務應用程式中的貸款核准程序，如下圖所示。程序會在提交貸款申請時開始。



在上圖所示的狀態機器中，Step Functions 會協調下列步驟：

- 驗證應用程式資料 (Lambda 函數)
- 檢查點數分數 （呼叫外部 API 的 Lambda 函數)

- 評估風險 (Lambda 函數）
- 如果為高風險，請路由至手動檢閱（人工核准任務）
- 如果核准，請啟動貸款支出 (Lambda 函數）
- 傳送通知給申請人 (Amazon SNS)

您可以使用此方法可靠地管理複雜、可能長時間執行的程序，具有內建錯誤處理和包含自動化和手動步驟的功能。

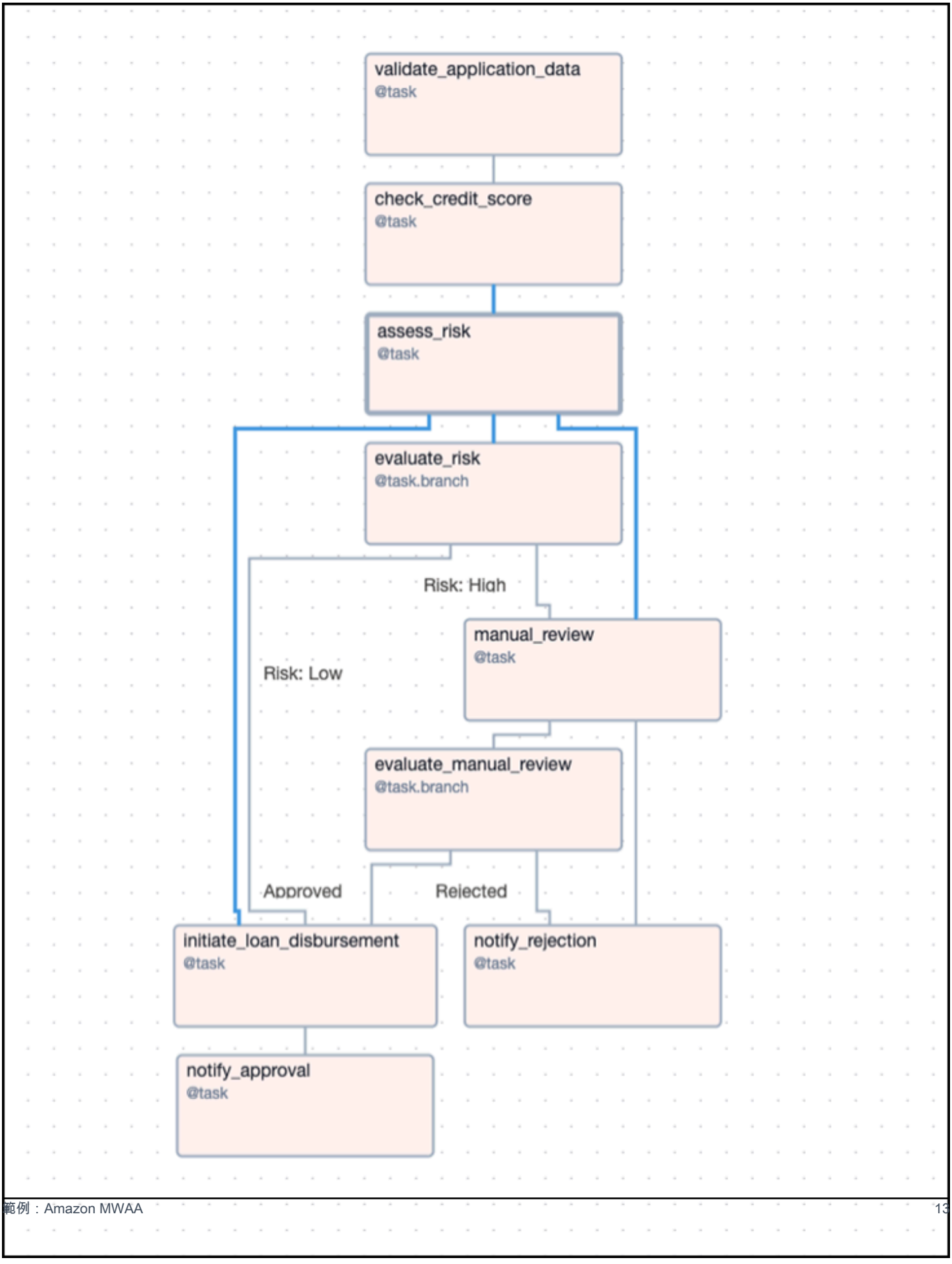
考量：

- 仔細設計您的狀態機器，以處理所有可能的情況。
- 盡可能平行執行步驟。
- 使用 Step Functions 中的內建錯誤處理和重試機制來處理永久和暫時故障。
- 考慮根據您的使用案例使用標準或快速工作流程。對於短期或大量工作流程，快速工作流程可能較適合。
- 監控執行指標以最佳化您的工作流程。
- 使用巢狀工作流程在多個狀態機器中封裝和重複使用功能。
- 對於複雜的工作流程，請考慮使用 Amazon Bedrock 代理程式作為 Step Functions 的替代方案。

如需詳細資訊，請參閱 Step Functions 文件。

範例：Amazon MWAA

如果您的組織已經使用 Apache Airflow，Amazon MWAA 是工作流程協調器的自然選擇。在 Apache Airflow 中，您可以使用 Python 依指示建立工作流程無環圖 (DAGs)。Step Functions 區段中說明的狀態機器的 DAG 表示可能如下所示：


validate_application_data
@task
check_credit_score
@task
assess_risk
@task
evaluate_risk
@task.branch
Risk: High
Risk: Low
manual_review
@task
evaluate_manual_review
@task.branch
Approved
Rejected
initiate_loan_disbursement
@task
notify_rejection
@task
notify_approval
@task

如需使用 DAGs的詳細資訊，請參閱 Amazon MWAA 文件。

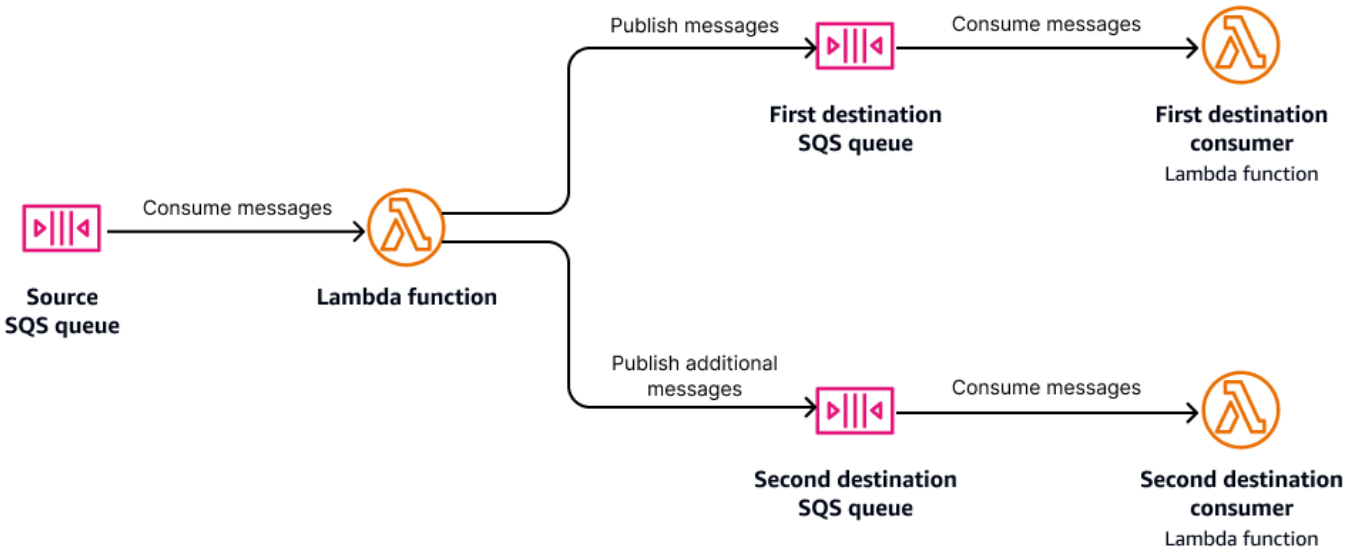
Step Functions 和 Amazon MWAA 之間的主要差異

- Step Functions 是全受管無伺服器服務，因此不需要預先佈建基礎設施，也不需要排程維護時段。Amazon MWAA 必須提前部署，而且您可以選擇叢集中的節點大小和數量。
- 在 Step Functions 中，您可以透過各種方式撰寫狀態機器，包括 Workflow Studio、直接做為 JSON 或使用 AWS Cloud Development Kit (AWS CDK)。Apache Airflow DAGs是以 Python 撰寫。
- 使用 Step Functions，當沒有執行中的工作流程時，您無須付費。使用 Amazon MWAA，即使沒有執行 DAGs也會產生成本。

編排

在編排的系統中，個別元件會收到任務、執行一些工作，並可能發出任務以進行後續工作。沒有中央協同運作機制。編目可讓您輕鬆獨立擴展服務，因為每個服務都會以相對隔離的方式運作。它會在接收工作時，以服務能夠達到的任何輸送量來執行工作。編排通常是事件驅動型架構 (EDA) 的核心部分。

在下圖中，Lambda 函數之間沒有協調。每個函數只會處理訂閱佇列中的訊息。每個函數都負責自己的錯誤處理，並且可以控制並行，例如，如果下游相依性具有每秒請求數 (RPS) 限制。



EDA 提供多種優點，例如服務鬆散耦合和可擴展性。EDA 原則的完整討論超出本指南的範圍。如需詳細資訊，請參閱：

- AWS Well-Architected Framework – 無伺服器應用程式鏡頭
- 事件驅動架構簡介 (Serverless Land)

- 轉換為事件驅動型架構（無伺服器開發人員指南)

選擇您的協調方法

編排和協同運作在整合微服務時都有其用途。在單一微服務的界限內選擇編排，您可以在其中完全控制相依性。當您跨微服務界限工作時，請選擇協同運作。例如，參與分散式交易的多個微服務將受益於協調，以考慮因故障而轉返。處理其他微服務可能感興趣的事件的微服務將受益於編排和事件驅動型架構。

當單一交易涉及多個系統時，實作轉返的常見模式是 saga 模式。

管理 APIs

適當的 API 管理可讓您的微型服務可供內部和外部消費者存取。 AWS 提供各種服務，您可以一起用來安全地公開您的微型服務 APIs。這些服務可讓您強制執行 APIs的安全性，並從中央位置實作監控和可觀測性。如果您的 APIs 使用者與託管服務的 地理位置相距，您也可以使用 Amazon CloudFront AWS 區域 來改善效能。

Amazon API Gateway

Amazon API Gateway 是一項全受管服務，可讓開發人員建立、發佈、維護、監控和保護任何規模的 REST 和 WebSocket APIs。您可以使用 API Gateway 實作本指南通訊模式一節所述的許多模式。

REST APIs 有兩種主要類型：REST 和 HTTP。這兩種類型都支援 RESTful APIs但提供不同的功能。若要判斷最符合您的需求，請參閱 APIs Gateway 文件中的選擇 REST APIs和 HTTP API。本指南的本節著重於 API Gateway REST APIs。

使用 API Gateway 做為 APIs的進入點提供單一位置來實作常見的考量，例如請求驗證和安全性。API Gateway REST APIs提供請求驗證，可讓您使用 JSON 結構描述定義請求的格式。API Gateway 會根據您定義的結構描述驗證傳入請求，並拒絕格式不正確的請求。

身分驗證和授權

API Gateway REST APIs支援下列身分驗證 (authN) 和授權 (authZ) 機制：

- IAM – 如果您使用 IAM，則必須使用 AWS Signature 第 4 版 (SigV4) 簽署對 API 的請求。
- Amazon Cognito – API Gateway 將驗證呈現的承載字符是否由 Amazon Cognito 使用者集區發出。如果您已經在使用第三方身分提供者 (IdP)，也可以設定 Amazon Cognito 使用者集區來整合。您也可以使用 Amazon Cognito 使用者集區進行machine-to-machine(M2M) 身分驗證。
- AWS Lambda 授權方 – API Gateway 會叫用您指定的 Lambda 函數，以執行您想要的任何檢查，以判斷是否應授權請求。

如需詳細資訊，請參閱 API Gateway 文件中的控制和管理對 REST APIs存取。

API 金鑰和速率限制

您可以使用 APIs金鑰和用量計劃，控制誰可以呼叫您的 API 和以什麼速率呼叫。API 金鑰不應用於身分驗證，但可與先前提到的方案搭配使用。使用者不一定需要提供自己的 API 金鑰，例如，Lambda

授權方可以傳回使用者的 API 金鑰。用量計劃可讓您指定輸送量、爆量限制和每月配額。如需詳細資訊，請參閱 API Gateway 文件中的 REST APIs 的使用計劃和 API 金鑰。

公有和私有 APIs

可透過網際網路存取的 API Gateway REST APIs 支援兩種端點類型：

- 邊緣最佳化，這表示發起人的請求會路由到附近的 CloudFront 存在點 (POP)。這可以改善不同地理位置的用戶端的效能。
- 區域，這表示請求路由到特定 內的資源 AWS 區域。當您的所有用戶端都靠近部署 API 的區域時，這是不錯的選擇。

API Gateway REST APIs也支援私有 API 端點，可透過使用界面 VPC 端點從虛擬私有雲端 (VPC) 存取。您也可以在其他 VPC 甚至其他 VPC 中建立介面 VPCs 端點，安全地共用私有 REST APIs AWS 帳戶。如需詳細資訊，請參閱 API Gateway 文件中的 REST APIs API 端點類型。

何時使用 API Gateway

API Gateway 是 RESTful Web 服務和即時 WebSocket 連線的理想選擇。當您在 API Gateway 中使用 WebSocket APIs 時，您可以新增連線和中斷連線事件的行為，例如將連線 IDs儲存在與用戶端屬性相關聯的外部資料存放區中。您也可以使用訊息屬性，將請求路由到自訂行為。

REST 和 WebSocket APIs 都可以直接與許多 AWS 服務 整合，而無需單獨的運算資源，例如 Lambda 函數。這可以提高效能並降低成本。

REST APIs同時支援路徑型和標頭型路由，您可以單獨或一起使用。常見的模式是提供 REST API 做為許多 APIs，以實作先前討論的共用問題，然後採取類似反向代理的行為，並將授權請求路由到正確的 API 端點。

簡訊

如通訊模式一節中所述，您可以使用簡訊在服務之間同步或非同步通訊。有許多無 AWS 伺服器服務可供選擇，您的選擇應該根據您的整合需求。例如，如果您需要按順序交付訊息，您應該選擇服務，例如 Amazon Simple Queue Service (Amazon SQS) 或 Amazon Simple Notification Service (Amazon SNS)。這兩個服務都支援先進先出 (FIFO) 交付，而不是 Amazon EventBridge，而 Amazon EventBridge 則不支援。

下列各節會更詳細地討論這些服務。

主題

- Amazon SQS
- Amazon SNS
- Amazon EventBridge
- AWS AppSync 事件和 API Gateway

Amazon SQS

Amazon SQS 支援標準佇列，其不保證排序，而 FIFO 佇列則保證在指定的訊息群組中排序。

佇列是編目微服務的一種常見方法，可為訊息提供長達 14 天的耐用儲存。佇列由生產者填入，並由取用者耗盡。當您以取用者 AWS Lambda 身分使用 時，您可以將 SQS 佇列設定為事件來源。在此情況下，Lambda 服務事件來源映射 (ESM) 會為您輪詢佇列，並在 Lambda 函數可用時傳送訊息給您的 Lambda 函數。在 Amazon Elastic Container Service (Amazon ECS) 或 Amazon Elastic Compute Cloud (Amazon EC2) 等其他類型的運算服務上執行的微服務，必須實作自己的輪詢機制，以便在新訊息可用時從佇列中擷取新訊息。

適用於 Amazon SQS 的 Lambda ESM 也支援訊息篩選，這可讓您根據訊息內文的內容，僅處理佇列中的訊息子集。

輪詢

Amazon SQS 支援短輪詢和長輪詢訊息。短輪詢會查詢伺服器子集以尋找可用的訊息，並立即傳回這些訊息。不過，它可能不會傳回所有可用的訊息。當您的應用程式需要盡快取用訊息，或無法容忍等待更長的時間時，這會很有用。

長輪詢會等到已超過可設定的時間量或已接收可設定的訊息數量，再傳回訊息。這可能會減少空輪詢的數量，也就是未傳回訊息的輪詢數量，尤其是未接收許多訊息的佇列。減少空輪詢數量可以降低 Amazon SQS 成本，因為此服務會針對每個請求收取費用，而每個輪詢操作都是請求。

指引

在下列情況下，佇列是不錯的選擇：

- 您想要解耦元件，不需要在它們之間進行同步通訊。
- 您正在具有不同可用性服務層級協議 (SLAs) 或服務層級目標 (SLOs元件之間進行通訊。
- 您通常有一組訊息的單一取用者。

如果出現下列情況，請考慮使用替代選項：

- 您需要同步通訊。
- 您需要複雜的路由邏輯，才能傳送訊息給正確的消費者。

Amazon SNS

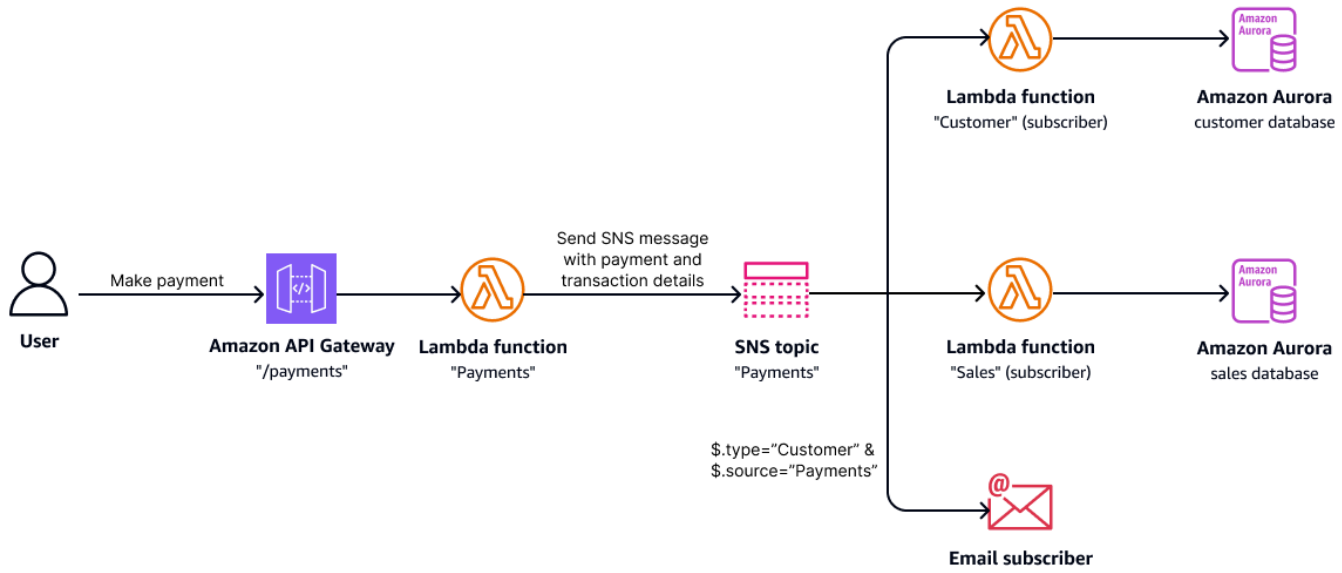
Amazon SNS 可讓您建立標準和 FIFO 主題。主題用於實作發佈/訂閱 (pub/sub) 架構。Amazon SNS 支援各種訂閱類型，包括電子郵件、SMS （假設您已設定起始身分，例如免付費電話號碼或 10 位數長碼）、HTTP(S) 端點和 SQS 佇列。訂閱者必須確認 SNS 主題的最終使用者訂閱，例如電子郵件訊息和簡訊。Amazon SNS 可讓服務廣泛散發，這表示單一訊息可以傳遞給可能大量的訂閱者。SNS 標準主題的預設限制為 1，250 萬個訂閱。

在微服務環境中，SNS 主題適用於從發佈者解耦訊息路由和交付邏輯。這可以透過使用主題篩選條件來實作。概念上，主題篩選條件與 Amazon EventBridge 規則略有相似，但它們是為每個訂閱者設定的，而不是從集中位置提供。例如，假設您有：

- Order 服務，可處理訂單。
- 履行服務，可處理訂單履行。
- 忠誠度服務，可授予會員訂單的忠誠度點數。

當訂單已準備好履行時，它會發佈訊息到主題。Fulfillment 服務會訂閱主題，但不會套用篩選條件，因為它想要了解所有訂單。假設您有一個忠誠度服務，該服務負責在成員下訂單時將點數授予他們。不過，並非所有訂單都是由成員下單。忠誠度服務會訂閱主題，但會實作訂閱篩選條件來檢查屬性，指出訂單是針對成員還是訪客。

假設系統收到最終使用者的付款請求，如下圖所示。在這種情況下，多個下游系統需要知道已提出請求，才能採取各種動作。當您使用 Amazon SNS 時，付款會發佈至 SNS 主題，而 Lambda 函數會訂閱主題以更新客戶和銷售資料庫。此外，電子郵件訂閱（必須由客戶確認）會使用訂閱篩選條件傳送電子郵件確認給客戶。



指引

本節中針對 Amazon SNS 所述的部分功能與事件匯流排所提供的功能重疊，例如 EventBridge。考慮在以下情況下使用 Amazon SNS：

- 您將有大量訂閱者加入主題。
- 您想要使用 EventBridge 原生不支援的訂閱類型（例如電子郵件或簡訊）。
- 訂閱者應該能夠判斷其訂閱篩選條件。
- 您需要按順序交付給訂閱者（每個訊息群組）。

如果您有許多主題，而且訂閱和篩選條件用於在微服務之間路由訊息，則 EventBridge 可能是更好的選擇。

Amazon EventBridge

Amazon EventBridge 是無伺服器事件匯流排服務，通常做為事件驅動型架構 (EDA) 的基礎。您也可以使用它在微服務之間以非同步方式路由和傳遞訊息。生產者使用 EventBridge 將事件發佈至匯流排。您可以根據事件的內容來設定符合事件的規則，並選擇一或多個符合該規則的事件要交付的目

標。EventBridge 支援各種規則目標。使用事件匯流排可讓您將生產者與消費者分離，並合併您的路由和交付邏輯。

在 EventBridge 中，您也可以建立排程規則，以便在特定時間採取動作。您可以使用以 Cron 為基礎的表達式和以速率為基礎的表達式來定義事件。

EventBridge 管道可讓您將訊息從來源輸送到目標，而無需使用 等運算服務 AWS Lambda。例如，假設您有一個接收訊息的 SQS 佇列，這應該會觸發 AWS Step Functions 狀態機器。您可以使用 EventBridge 管道來為您執行此操作，而不是建立具有事件來源映射來取用佇列訊息的 Lambda 函數，以及使用 AWS SDK 撰寫程式碼來叫用狀態機器，而無需撰寫任何自訂程式碼。

EventBridge 通常與其他簡訊服務搭配使用，例如 Amazon SQS 和 Amazon SNS。例如，將事件交付至 SQS 佇列，可讓接收服務彈性地在能夠使用訊息時，以獨立於事件產生速率的速率來取用訊息。同樣地，您也可以將應該散發的事件傳遞給 SNS 主題的大量訂閱者。

指引

在下列情況下使用 EventBridge：

- 您不需要服務之間的同步通訊。
- 您想要將訊息路由邏輯與微服務分離。微型服務只會產生事件並將其發佈到事件匯流排，而感興趣的服務會建立規則來比對和交付這些事件。
- 您需要將訊息從一個支援的服務傳遞到另一個服務。

在下列情況下，請考慮其他 服務：

- 您需要嚴格排序事件。在這些情況下，請考慮 Amazon SQS FIFO 佇列或 Amazon SNS FIFO 主題。或者，請考慮事件串流服務，例如 Amazon Kinesis Data Streams 或 Amazon Managed Streaming for Apache Kafka (Amazon MSK)。

AWS AppSync 事件和 API Gateway

AWS AppSync 事件和 Amazon API Gateway 都為您的微服務提供受管 WebSocket 體驗。

AWS AppSync 事件透過使用 WebSocket 提供簡化的即時傳訊體驗。 AWS AppSync Events 支援單點傳送和多點傳送傳訊，以及將頻道靈活分組到命名空間，並支援萬用字元。Microservices 可以使用 AWS AppSync Events 以各種方式互相通訊。例如，接收即時資料的服務可以轉換資料並將其發佈到適當的頻道，訂閱者將在此頻道中即時接收資料。

API Gateway 也支援 WebSocket APIs。您可以定義與 的整合 AWS 服務，例如 AWS Lambda 和 Amazon DynamoDB，並設定映射到這些整合的路由選擇表達式。API Gateway 具有特殊路由，可用來授權和管理 WebSocket 連線。根據您的需求，您可以將 WebSocket 連線資訊存放在 DynamoDB 等資料存放區。使用此資訊，訊息可以透過 REST API 發佈至特定的 WebSocket 連線，並指定特定的連線 ID。

指引

在以下情況下使用 AWS AppSync 事件：

- 您有多個訊息管道，這些管道會分組為命名空間，並想要使用萬用字元來發佈和訂閱頻道群組。
- 您的通訊主要在不同系統之間，而不是在系統之間 AWS 服務。

在下列情況下使用 API Gateway WebSocket APIs：

- 您想要讓用戶端擁有與 AWS 服務 整合的即時持久性連線。
- 您想要自行管理 WebSocket 連線。例如，您可能想要允許其他系統在查詢連線 ID 之後，將訊息傳送到特定用戶端。
- 您想要使用 API Gateway 功能，例如階段部署或代理整合，或您想要設定自己的子通訊協定。

常見問答集

如何結合不同的整合模式？

在大多數情況下，您會想要結合整合模式。例如，您可以使用 來 AWS Step Functions 協調使用宣告檢查模式呼叫遠端服務的程序。或者，您可能有一個協調程序，將訊息放入佇列，進而觸發編排的服務。

使用微服務架構的主要優點是什麼？

主要優點包括獨立擴展服務、改善故障隔離、透過平行團隊工作增強開發速度，以及持續交付和部署 (CI/CD) 的能力。

如何以這些模式實作錯誤處理？

您可以使用 中的內建機制來實作錯誤處理 AWS 服務。例如，可以使用重試邏輯設定 AWS Lambda 函數，Amazon SQS 支援無效字母佇列來處理持久性故障。此外，Step Functions 在工作流程層級提供錯誤處理和重試機制。

在非同步通訊中使用宣告檢查模式有何好處？

宣告檢查模式可讓用戶端在提交請求時接收識別符。稍後可以使用此識別符來檢查狀態並擷取結果。此模式透過提供機制輪詢結果而不會同步等待，從而使用戶端受益。如需詳細資訊，請參閱本指南前面的宣告檢查一節。

回呼模式如何改善微服務中的非同步通訊？

回呼模式透過允許用戶端在處理完成時提供服務聯絡的位置，來改善非同步通訊。這會將用戶端與等待回應分離，並使其能夠繼續執行其他任務。如需詳細資訊，請參閱本指南稍早的回呼一節。

我可以使用所描述的模式在微服務中實作雙向通訊嗎？

您可以透過在用戶端和服務之間建立具狀態的連線來實作雙向通訊，以便它們可以非同步地傳送和處理訊息。這需要 服務支援每個用戶端的開放連線。如需詳細資訊，請參閱本指南前面的雙向通訊一節。

如何以非同步通訊模式最佳化 Lambda 函數的使用？

您可以透過確保 Lambda 函數等同處理潛在的訊息重複、使用訊息群組等 Amazon SQS 功能進行排序，以及實作長輪詢來降低成本，來最佳化 Lambda 函數。此外，您可以監控執行指標，以識別最佳化機會。

針對 pub/sub 模式使用 Amazon SNS 和 EventBridge 有什麼主要差異？

Amazon SNS 會傳送單一訊息給所有訂閱者，其中可能包含某些訂閱者不必要的資料。Amazon EventBridge 可讓您擁有多個符合單一事件的規則，而每個規則都會觸發不同的下游服務或動作，藉此進行更精細的控制。如需詳細資訊，請參閱本指南稍早的 Amazon SNS 和 EventBridge 章節。

資源

AWS 服務 文件

- Amazon API Gateway
- AWS AppSync 事件
- Amazon EventBridge
- Amazon MWAA
- Amazon SNS
- Amazon SQS
- AWS Step Functions

其他讀取

- 中的應用程式現代化策略 AWS 雲端
- 中的應用程式現代化分階段方法 AWS 雲端
- 在 中評估應用程式的現代化準備程度 AWS 雲端
- 將整體分解為微服務
- 使用 AWS 簡訊服務實作企業整合模式：point-to-point管道
- Pub/sub 訊息：非同步事件通知

文件歷史紀錄

下表描述了本指南的重大變更。如果您想收到有關未來更新的通知，可以訂閱 RSS 摘要。

變更	描述	日期
主要更新	擴展、重組和更新指南以反映 AWS 服務 更新。	2025 年 9 月 10 日
初次出版	—	2021 年 1 月 11 日

AWS 規範性指引詞彙表

以下是 AWS Prescriptive Guidance 提供的策略、指南和模式中常用的術語。若要建議項目，請使用詞彙表末尾的提供意見回饋連結。

數字

7 R

將應用程式移至雲端的七種常見遷移策略。這些策略以 Gartner 在 2011 年確定的 5 R 為基礎，包括以下內容：

- 重構/重新架構 – 充分利用雲端原生功能來移動應用程式並修改其架構，以提高敏捷性、效能和可擴展性。這通常涉及移植作業系統和資料庫。範例：將您的內部部署 Oracle 資料庫 遷移至 Amazon Aurora PostgreSQL 相容版本。
- 平台轉換 (隨即重塑) – 將應用程式移至雲端，並引入一定程度的優化以利用雲端功能。範例：將您的內部部署 Oracle 資料庫 遷移至 中的 Amazon Relational Database Service (Amazon RDS) for Oracle AWS 雲端。
- 重新購買 (捨棄再購買) – 切換至不同的產品，通常從傳統授權移至 SaaS 模型。範例：將您的客戶關係管理 (CRM) 系統 遷移至 Salesforce.com。
- 主機轉換 (隨即轉移) – 將應用程式移至雲端，而不進行任何變更以利用雲端功能。範例：將您的現場部署 Oracle 資料庫 遷移至 中 EC2 執行個體上的 Oracle AWS 雲端。
- 重新放置 (虛擬機器監視器等級隨即轉移) – 將基礎設施移至雲端，無需購買新硬體、重寫應用程式或修改現有操作。您可以將伺服器從內部部署平台遷移到相同平台的雲端服務。範例：將 Microsoft Hyper-V應用程式 遷移至 AWS。
- 保留 (重新檢視) – 將應用程式保留在來源環境中。其中可能包括需要重要重構的應用程式，且您希望將該工作延遲到以後，以及您想要保留的舊版應用程式，因為沒有業務理由來進行遷移。
- 淘汰 – 解除委任或移除來源環境中不再需要的應用程式。

A

A2A Agent-to-Agent)

支援任務委派和狀態轉移的agent-to-agent協同合作的狀態通訊協定。

ABAC

請參閱屬性型存取控制。

抽象服務

請參閱 受管服務。

ACID

請參閱原子性、一致性、隔離性、耐久性。

主動-主動式遷移

一種資料庫遷移方法，其中來源和目標資料庫保持同步 (透過使用雙向複寫工具或雙重寫入操作)，且兩個資料庫都在遷移期間處理來自連接應用程式的交易。此方法支援小型、受控制批次的遷移，而不需要一次性切換。它更靈活，但比主動-被動遷移需要更多的工作。

主動-被動式遷移

一種資料庫遷移方法，其中來源和目標資料庫保持同步，但只有來源資料庫會在資料複寫至目標資料庫時處理來自連線應用程式的交易。目標資料庫在遷移期間不接受任何交易。

客服人員

一種 AI 系統，可使用工具自動推理、規劃和採取行動來實現目標。

客服人員操作

在生產環境中大規模建置、測試、部署和執行 AI 代理器的操作實務。

彙總函數

在一組資料列上運作的 SQL 函數，會計算群組的單一傳回值。彙總函數的範例包括 SUM和 MAX。

AI

請參閱人工智慧。

AIOps

請參閱人工智慧操作。

匿名化

永久刪除資料集中個人資訊的程序。匿名化有助於保護個人隱私權。匿名資料不再被視為個人資料。

反模式

經常用於重複性問題的解決方案，其中解決方案具有反效益、無效或比替代解決方案更有效。

應用程式控制

一種安全方法，僅允許使用核准的應用程式，以協助保護系統免受惡意軟體攻擊。

應用程式組合

有關組織使用的每個應用程式的詳細資訊的集合，包括建置和維護應用程式的成本及其商業價值。此資訊是產品組合探索和分析程序的關鍵，有助於識別要遷移、現代化和優化的應用程式並排定其優先順序。

人工智慧 (AI)

電腦科學領域，致力於使用運算技術來執行通常與人類相關的認知功能，例如學習、解決問題和識別模式。如需詳細資訊，請參閱什麼是人工智慧？

人工智慧操作 (AIOps)

使用機器學習技術解決操作問題、減少操作事件和人工干預以及提高服務品質的程序。如需有關如何在 AWS 遷移策略中使用 AIOps 的詳細資訊，請參閱操作整合指南。

非對稱加密

一種加密演算法，它使用一對金鑰：一個用於加密的公有金鑰和一個用於解密的私有金鑰。您可以共用公有金鑰，因為它不用於解密，但對私有金鑰存取應受到高度限制。

原子性、一致性、隔離性、耐久性 (ACID)

一組軟體屬性，即使在出現錯誤、電源故障或其他問題的情況下，也能確保資料庫的資料有效性和操作可靠性。

屬性型存取控制 (ABAC)

根據使用者屬性 (例如部門、工作職責和團隊名稱) 建立精細許可的實務。如需詳細資訊，請參閱《 AWS Identity and Access Management (IAM) 文件》中的 ABAC for AWS。

授權資料來源

存放主要版本資料的位置，被視為最可靠的資訊來源。您可以將授權資料來源中的資料複製到其他位置，以處理或修改資料，例如匿名、修訂或假名化資料。

可用區域

中的不同位置 AWS 區域 ，可隔離其他可用區域中的故障，並提供相同區域中其他可用區域的低成本、低延遲網路連線。

AWS 雲端採用架構 (AWS CAF)

的指導方針和最佳實務架構 AWS ，可協助組織制定高效且有效的計劃，以成功地移至雲端。 AWS CAF 將指導方針整理成六個重點領域：業務、人員、治理、平台、安全和營運。業務、人員和控管層面著重於業務技能和程序；平台、安全和操作層面著重於技術技能和程序。例如，人員層面針對處理人力資源 (HR)、人員配備功能和人員管理的利害關係人。因此， AWS CAF 為人員開發、訓練和通訊提供指引，協助組織做好成功採用雲端的準備。如需詳細資訊，請參閱 AWS CAF 網站和 AWS CAF 白皮書。

AWS 工作負載資格架構 (AWS WQF)

一種工具，可評估資料庫遷移工作負載、建議遷移策略，並提供工作預估值。 AWS WQF 隨附於 AWS Schema Conversion Tool (AWS SCT)。它會分析資料庫結構描述和程式碼物件、應用程式程式碼、相依性和效能特性，並提供評估報告。

B

錯誤的機器人

旨在中斷或傷害個人或組織的機器人。

BCP

請參閱業務持續性規劃。

行為圖

資源行為的統一互動式檢視，以及一段時間後的互動。您可以將行為圖與 Amazon Detective 搭配使用來檢查失敗的登入嘗試、可疑的 API 呼叫和類似動作。如需詳細資訊，請參閱偵測文件中的行為圖中的資料。

大端序系統

首先儲存最高有效位元組的系統。另請參閱 Endianness。

二進制分類

預測二進制結果的過程 (兩個可能的類別之一)。例如，ML 模型可能需要預測諸如「此電子郵件是否是垃圾郵件？」等問題 或「產品是書還是汽車？」

Bloom 篩選條件

一種機率性、記憶體高效的資料結構，用於測試元素是否為集的成員。

藍/綠部署

一種部署策略，您可以在其中建立兩個不同但相同的環境。您可以在一個環境（藍色）中執行目前的應用程式版本，並在另一個環境（綠色）中執行新的應用程式版本。此策略可協助您快速復原，並將影響降至最低。

機器人

透過網際網路執行自動化任務並模擬人類活動或互動的軟體應用程式。有些機器人有用或有益，例如在網際網路上編製資訊索引的 Web 爬蟲程式。某些其他機器人稱為惡意機器人，旨在中斷或傷害個人或組織。

殭屍網路

受到惡意軟體感染且受單一方控制之機器人的網路，稱為機器人繼承器或機器人運算子。殭屍網路是擴展機器人及其影響的最佳已知機制。

分支

程式碼儲存庫包含的區域。儲存庫中建立的第一個分支是主要分支。您可以從現有分支建立新分支，然後在新分支中開發功能或修正錯誤。您建立用來建立功能的分支通常稱為功能分支。當準備好發佈功能時，可以將功能分支合併回主要分支。如需詳細資訊，請參閱關於分支 (GitHub 文件)。

碎片存取

在特殊情況下，並透過核准的程序，讓使用者快速取得他們通常無權存取 AWS 帳戶 之 的存取權。如需詳細資訊，請參閱 Well-Architected 指南中的 AWS 實作打破玻璃程序指標。

棕地策略

環境中的現有基礎設施。對系統架構採用棕地策略時，可以根據目前系統和基礎設施的限制來設計架構。如果正在擴展現有基礎設施，則可能會混合棕地和綠地策略。

緩衝快取

儲存最常存取資料的記憶體區域。

業務能力

業務如何創造價值 (例如，銷售、客戶服務或營銷)。業務能力可驅動微服務架構和開發決策。如需詳細資訊，請參閱在 AWS上執行容器化微服務白皮書的圍繞業務能力進行組織部分。

業務連續性規劃 (BCP)

一種解決破壞性事件 (如大規模遷移) 對營運的潛在影響並使業務能夠快速恢復營運的計畫。

C

CAF

請參閱AWS 雲端採用架構。

Canary 部署

版本對最終使用者的緩慢和增量版本。當您有信心時，您可以部署新版本並完全取代目前的版本。

CCoE

請參閱 Cloud Center of Excellence。

CDC

請參閱變更資料擷取。

變更資料擷取 (CDC)

追蹤對資料來源 (例如資料庫表格) 的變更並記錄有關變更的中繼資料的程序。您可以將 CDC 用於各種用途，例如稽核或複寫目標系統中的變更以保持同步。

混沌工程

故意引入故障或破壞性事件，以測試系統的彈性。您可以使用 AWS Fault Injection Service (AWS FIS) 執行實驗，為您的 AWS 工作負載帶來壓力，並評估其回應。

CI/CD

請參閱持續整合和持續交付。

分類

有助於產生預測的分類程序。用於分類問題的 ML 模型可預測離散值。離散值永遠彼此不同。例如，模型可能需要評估影像中是否有汽車。

公民開發人員

在沒有專業技術技能的情況下，使用無程式碼/低程式碼平台建立 AI 應用程式的商業使用者。

用戶端加密

在目標 AWS 服務 接收資料之前，在本機加密資料。

雲端卓越中心 (CCoE)

一個多學科團隊，可推動整個組織的雲端採用工作，包括開發雲端最佳實務、調動資源、制定遷移時間表以及領導組織進行大規模轉型。如需詳細資訊，請參閱 AWS 雲端 企業策略部落格上的 CCoE 文章。

雲端運算

通常用於遠端資料儲存和 IoT 裝置管理的雲端技術。雲端運算通常連接到邊緣運算技術。

雲端操作模型

在 IT 組織中，用於建置、成熟和最佳化一或多個雲端環境的操作模型。如需詳細資訊，請參閱建置您的雲端操作模型。

採用雲端階段

組織在遷移至 時通常會經歷的四個階段 AWS 雲端：

- 專案 – 執行一些與雲端相關的專案以進行概念驗證和學習用途
- 基礎 – 進行基礎投資以擴展雲端採用 (例如，建立登陸區域、定義 CCoE、建立營運模型)
- 遷移 – 遷移個別應用程式
- 重塑 – 優化產品和服務，並在雲端中創新

這些階段由 Stephen Orban 在部落格文章 The Journey Toward Cloud-First 和 Enterprise Strategy 部落格上的採用階段中定義。 AWS 雲端 如需有關它們如何與 AWS 遷移策略關聯的資訊，請參閱遷移整備指南。

CMDB

請參閱組態管理資料庫。

程式碼儲存庫

透過版本控制程序來儲存及更新原始程式碼和其他資產 (例如文件、範例和指令碼) 的位置。常見的雲端儲存庫包括 GitHub或 Bitbucket Cloud。程式碼的每個版本都稱為分支。在微服務結構中，每個儲存庫都專用於單個功能。單一 CI/CD 管道可以使用多個儲存庫。

冷快取

一種緩衝快取，它是空的、未填充的，或者包含過時或不相關的資料。這會影響效能，因為資料庫執行個體必須從主記憶體或磁碟讀取，這比從緩衝快取讀取更慢。

冷資料

很少存取且通常是歷史資料的資料。查詢這類資料時，通常可接受慢查詢。將此資料移至效能較低且成本較低的儲存層或類別，可以降低成本。

電腦視覺 (CV)

AI 欄位???，使用機器學習從數位影像和影片等視覺化格式分析和擷取資訊。例如，Amazon SageMaker AI 提供 CV 的影像處理演算法。

組態偏離

對於工作負載，組態會從預期狀態變更。這可能會導致工作負載不合規，而且通常是漸進和無意的。

組態管理資料庫 (CMDB)

儲存和管理有關資料庫及其 IT 環境的資訊的儲存庫，同時包括硬體和軟體元件及其組態。您通常在遷移的產品組合探索和分析階段使用 CMDB 中的資料。

一致性套件

您可以組合的 AWS Config 規則和修補動作集合，以自訂您的合規和安全檢查。您可以使用 YAML 範本，將一致性套件部署為 AWS 帳戶 和 區域中或整個組織的單一實體。如需詳細資訊，請參閱 AWS Config 文件中的一致性套件。

持續整合和持續交付 (CI/CD)

自動化軟體發行程序的來源、建置、測試、暫存和生產階段的程序。CI/CD 通常被描述為管道。CI/CD 可協助您將程序自動化、提升生產力、改善程式碼品質以及加快交付速度。如需詳細資訊，請參閱持續交付的優點。CD 也可表示持續部署。如需詳細資訊，請參閱持續交付與持續部署。

CV

請參閱電腦視覺。

D

靜態資料

網路中靜止的資料，例如儲存中的資料。

資料分類

根據重要性和敏感性來識別和分類網路資料的程序。它是所有網路安全風險管理策略的關鍵組成部分，因為它可以協助您確定適當的資料保護和保留控制。資料分類是 AWS Well-Architected Framework 中安全支柱的元件。如需詳細資訊，請參閱資料分類。

資料偏離

生產資料與用於訓練 ML 模型的資料之間有意義的變化，或輸入資料隨時間有意義的變更。資料偏離可以降低 ML 模型預測的整體品質、準確性和公平性。

傳輸中的資料

在您的網路中主動移動的資料，例如在網路資源之間移動。

資料網格

架構架構，提供分散式、分散式資料擁有權與集中式管理。

資料最小化

僅收集和處理嚴格必要資料的原則。在 中實作資料最小化 AWS 雲端 可以降低隱私權風險、成本和分析碳足跡。

資料周邊

AWS 環境中的一組預防性防護機制，可協助確保只有信任的身分才能從預期的網路存取信任的資源。如需詳細資訊，請參閱在 上建置資料周邊 AWS。

資料預先處理

將原始資料轉換成 ML 模型可輕鬆剖析的格式。預處理資料可能意味著移除某些欄或列，並解決遺失、不一致或重複的值。

資料來源

在整個資料生命週期中追蹤資料的來源和歷史記錄的程序，例如資料的產生、傳輸和儲存方式。

資料主體

正在收集和處理資料的個人。

資料倉儲

支援商業智慧的資料管理系統，例如 分析。資料倉儲通常包含大量歷史資料，通常用於查詢和分析。

資料庫定義語言 (DDL)

用於建立或修改資料庫中資料表和物件之結構的陳述式或命令。

資料庫處理語言 (DML)

用於修改 (插入、更新和刪除) 資料庫中資訊的陳述式或命令。

DDL

請參閱資料庫定義語言。

深度整體

結合多個深度學習模型進行預測。可以使用深度整體來獲得更準確的預測或估計預測中的不確定性。

深度學習

一個機器學習子領域，它使用多層人工神經網路來識別感興趣的輸入資料與目標變數之間的對應關係。

深度防禦

這是一種資訊安全方法，其中一系列的安全機制和控制項會在整個電腦網路中精心分層，以保護網路和其中資料的機密性、完整性和可用性。當您在 上採用此策略時 AWS，您可以在 AWS Organizations 結構的不同層新增多個控制項，以協助保護資源。例如，defense-in-depth方法可能會結合多重要素驗證、網路分割和加密。

委派的管理員

在 中 AWS Organizations，相容的服務可以註冊 AWS 成員帳戶，以管理組織的帳戶和管理該服務的許可。此帳戶稱為該服務的委派管理員。如需詳細資訊和相容服務清單，請參閱 AWS Organizations 文件中的可搭配 AWS Organizations運作的服務。

deployment

在目標環境中提供應用程式、新功能或程式碼修正的程序。部署涉及在程式碼庫中實作變更，然後在應用程式環境中建置和執行該程式碼庫。

開發環境

請參閱 環境。

偵測性控制

一種安全控制，用於在事件發生後偵測、記錄和提醒。這些控制是第二道防線，提醒您注意繞過現有預防性控制的安全事件。如需詳細資訊，請參閱在 AWS上實作安全控制中的偵測性控制。

開發值串流映射 (DVSM)

一種程序，用於識別對軟體開發生命週期中的速度和品質造成負面影響的限制並排定優先順序。DVSM 擴展了最初專為精簡製造實務設計的價值串流映射程序。它著重於透過軟體開發程序建立和移動價值所需的步驟和團隊。

數位分身

真實世界系統的虛擬呈現，例如建築物、工廠、工業設備或生產線。數位分身支援預測性維護、遠端監控和生產最佳化。

維度資料表

在星星結構描述中，較小的資料表包含有關事實資料表中量化資料的資料屬性。維度資料表屬性通常是文字欄位或離散數字，其行為類似於文字。這些屬性通常用於查詢限制、篩選和結果集標記。

災難

防止工作負載或系統在其主要部署位置中實現其業務目標的事件。這些事件可能是自然災難、技術故障或人為動作的結果，例如意外設定錯誤或惡意軟體攻擊。

災難復原 (DR)

您用來將災難造成的停機時間和資料遺失降至最低的策略和程序。如需詳細資訊，請參閱 AWS Well-Architected Framework 中的 上工作負載的災難復原 AWS：雲端中的復原。

DML

請參閱資料庫處理語言。

領域驅動的設計

一種開發複雜軟體系統的方法，它會將其元件與每個元件所服務的不斷發展的領域或核心業務目標相關聯。Eric Evans 在其著作 Domain-Driven Design: Tackling Complexity in the Heart of Software (Boston: Addison-Wesley Professional, 2003) 中介紹了這一概念。如需有關如何將領域驅動的設計與 strangler fig 模式搭配使用的資訊，請參閱使用容器和 Amazon API Gateway 逐步現代化舊版 Microsoft ASP.NET (ASMX) Web 服務。

DR

請參閱災難復原。

偏離偵測

追蹤與基準組態的偏差。例如，您可以使用 AWS CloudFormation 來偵測系統資源中的偏離，也可以使用 AWS Control Tower 來偵測登陸區域中可能影響控管要求合規性的變更。 https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/using-cfn-stack-drift.html

DVSM

請參閱開發值串流映射。

E

EDA

請參閱探索性資料分析。

EDI

請參閱電子資料交換。

邊緣運算

提升 IoT 網路邊緣智慧型裝置運算能力的技術。與雲端運算相比，邊緣運算可以減少通訊延遲並改善回應時間。

電子資料交換 (EDI)

在組織之間自動交換商業文件。如需詳細資訊，請參閱什麼是電子資料交換。

加密

一種運算程序，可將人類可讀取的純文字資料轉換為加密文字。

加密金鑰

由加密演算法產生的隨機位元的加密字串。金鑰長度可能有所不同，每個金鑰的設計都是不可預測且唯一的。

端序

位元組在電腦記憶體中的儲存順序。大端序系統首先儲存最高有效位元組。小端序系統首先儲存最低有效位元組。

端點

請參閱 服務端點。

端點服務

您可以在虛擬私有雲端 (VPC) 中託管以與其他使用者共用的服務。您可以使用 建立端點服務，AWS PrivateLink 並將許可授予其他 AWS 帳戶 或 AWS Identity and Access Management (IAM) 委託人。這些帳戶或主體可以透過建立介面 VPC 端點私下連接至您的端點服務。如需詳細資訊，請參閱 Amazon Virtual Private Cloud (Amazon VPC) 文件中的建立端點服務。

企業資源規劃 (ERP)

一種系統，可自動化和管理企業的關鍵業務流程（例如會計、MES 和專案管理）。

信封加密

使用另一個加密金鑰對某個加密金鑰進行加密的程序。如需詳細資訊，請參閱 AWS Key Management Service (AWS KMS) 文件中的信封加密。

環境

執行中應用程式的執行個體。以下是雲端運算中常見的環境類型：

- 開發環境 – 執行中應用程式的執行個體，只有負責維護應用程式的核心團隊才能使用。開發環境用來測試變更，然後再將開發環境提升到較高的環境。此類型的環境有時稱為測試環境。
- 較低的環境 – 應用程式的所有開發環境，例如用於初始建置和測試的開發環境。
- 生產環境 – 最終使用者可以存取的執行中應用程式的執行個體。在 CI/CD 管道中，生產環境是最後一個部署環境。
- 較高的環境 – 核心開發團隊以外的使用者可存取的所有環境。這可能包括生產環境、生產前環境以及用於使用者接受度測試的環境。

epic

在敏捷方法中，有助於組織工作並排定工作優先順序的功能類別。epic 提供要求和實作任務的高層級描述。例如，AWS CAF 安全概念包括身分和存取管理、偵測控制、基礎設施安全、資料保護和事件回應。如需有關 AWS 遷移策略中的 Epic 的詳細資訊，請參閱計畫實作指南。

ERP

請參閱企業資源規劃。

探索性資料分析 (EDA)

分析資料集以了解其主要特性的過程。您收集或彙總資料，然後執行初步調查以尋找模式、偵測異常並檢查假設。透過計算摘要統計並建立資料可視化來執行 EDA。

F

事實資料表

星狀結構描述中的中央資料表。它存放有關業務操作的量化資料。一般而言，事實資料表包含兩種類型的資料欄：包含度量的資料，以及包含維度資料表外部索引鍵的資料欄。

快速失敗

一種使用頻繁和增量測試來縮短開發生命週期的理念。這是敏捷方法的關鍵部分。

故障隔離界限

在 中 AWS 雲端，像是可用區域 AWS 區域、控制平面或資料平面等界限會限制故障的影響，並有助於改善工作負載的彈性。如需詳細資訊，請參閱AWS 故障隔離界限。

功能分支

請參閱分支。

特徵

用來進行預測的輸入資料。例如，在製造環境中，特徵可能是定期從製造生產線擷取的影像。

功能重要性

特徵對於模型的預測有多重要。這通常表示為可以透過各種技術來計算的數值得分，例如 Shapley Additive Explanations (SHAP) 和積分梯度。如需詳細資訊，請參閱 的機器學習模型可解譯性 AWS。

特徵轉換

優化 ML 程序的資料，包括使用其他來源豐富資料、調整值、或從單一資料欄位擷取多組資訊。這可讓 ML 模型從資料中受益。例如，如果將「2021-05-27 00:15:37」日期劃分為「2021」、「五月」、「週四」和「15」，則可以協助學習演算法學習與不同資料元件相關聯的細微模式。

少量擷取提示

在要求 LLM 執行類似的任務之前，提供少量示範任務和所需輸出的範例。此技術是內容內學習的應用程式，其中模型會從內嵌在提示中的範例 (快照) 中學習。少量的提示對於需要特定格式、推理或網域知識的任務來說非常有效。另請參閱零鏡頭提示。

FGAC

請參閱精細存取控制。

精細存取控制 (FGAC)

使用多個條件來允許或拒絕存取請求。

閃切遷移

一種資料庫遷移方法，透過變更資料擷取使用連續資料複寫，以盡可能在最短的時間內遷移資料，而不是使用分階段方法。目標是將停機時間降至最低。

FM

請參閱基礎模型。

基礎模型 (FM)

大型深度學習神經網路，已針對廣義和未標記資料的大量資料集進行訓練。FMs 能夠執行各種一般任務，例如了解語言、產生文字和影像，以及以自然語言交談。如需詳細資訊，請參閱什麼是基礎模型。

FM 閘道

集中式中介，可控制和標準化對基礎模型的存取。也稱為 LLM 閘道。

G

生成式 AI

已針對大量資料進行訓練的 AI 模型子集，可使用簡單的文字提示建立新的內容和成品，例如影像、影片、文字和音訊。如需詳細資訊，請參閱什麼是生成式 AI。

地理封鎖

請參閱地理限制。

地理限制 (地理封鎖)

Amazon CloudFront 中的選項，可防止特定國家/地區的使用者存取內容分發。您可以使用允許清單或封鎖清單來指定核准和禁止的國家/地區。如需詳細資訊，請參閱 CloudFront 文件中的限制內容的地理分佈。

Gitflow 工作流程

這是一種方法，其中較低和較高環境在原始碼儲存庫中使用不同分支。Gitflow 工作流程被視為舊版，而以幹線為基礎的工作流程是現代、偏好的方法。

黃金影像

系統或軟體的快照，做為部署該系統或軟體新執行個體的範本。例如，在製造中，黃金映像可用於在多個裝置上佈建軟體，並有助於提高裝置製造操作的速度、可擴展性和生產力。

綠地策略

新環境中缺乏現有基礎設施。對系統架構採用綠地策略時，可以選擇所有新技術，而不會限制與現有基礎設施的相容性，也稱為棕地。如果正在擴展現有基礎設施，則可能會混合棕地和綠地策略。

防護機制

有助於跨組織單位 (OU) 來管控資源、政策和合規的高層級規則。預防性防護機制會強制執行政策，以確保符合合規標準。透過使用服務控制政策和 IAM 許可界限來將其實作。偵測性防護機制可偵測政策違規和合規問題，並產生提醒以便修正。它們是透過使用 AWS Config AWS Security Hub CSPM、Amazon GuardDuty、、Amazon Inspector AWS Trusted Advisor和自訂 AWS Lambda 檢查來實作。

護欄 (AI)

可篩選、驗證和限制代理程式輸入和輸出的安全機制，以協助確保負責任且安全的 AI 行為。

H

HA

請參閱高可用性。

異質資料庫遷移

將來源資料庫遷移至使用不同資料庫引擎的目標資料庫 (例如，Oracle 至 Amazon Aurora)。異質遷移通常是重新架構工作的一部分，而轉換結構描述可能是一項複雜任務。AWS 提供有助於結構描述轉換的 AWS SCT。

高可用性 (HA)

工作負載在遇到挑戰或災難時持續運作的能力，無需介入。HA 系統的設計目的是自動容錯移轉、持續提供高品質的效能，並處理不同的負載和故障，並將效能影響降至最低。

歷史現代化

一種方法，用於現代化和升級操作技術 (OT) 系統，以更好地滿足製造業的需求。歷史資料是一種資料庫，用於從工廠中的各種來源收集和存放資料。

保留資料

從用於訓練機器學習模型的資料集中保留的部分歷史標記資料。您可以使用保留資料，透過比較模型預測與保留資料來評估模型效能。

human-in-the-loop (HitL)

一種工作流程模式，其中代理程式執行會在關鍵決策點暫停進行人工審核和核准。

異質資料庫遷移

將您的來源資料庫遷移至共用相同資料庫引擎的目標資料庫 (例如，Microsoft SQL Server 至 Amazon RDS for SQL Server)。同質遷移通常是主機轉換或平台轉換工作的一部分。您可以使用原生資料庫公用程式來遷移結構描述。

熱資料

經常存取的資料，例如即時資料或最近的轉譯資料。此資料通常需要高效能儲存層或類別，才能提供快速的查詢回應。

修補程序

緊急修正生產環境中的關鍵問題。由於其緊迫性，通常會在典型 DevOps 發行工作流程之外執行修補程式。

超級護理期間

在切換後，遷移團隊在雲端管理和監控遷移的應用程式以解決任何問題的時段。通常，此期間的長度為 1-4 天。在超級護理期間結束時，遷移團隊通常會將應用程式的責任轉移給雲端營運團隊。

I

IaC

請參閱基礎設施即程式碼。

身分型政策

連接至一或多個 IAM 主體的政策，可定義其在 AWS 雲端 環境中的許可。

閒置應用程式

90 天期間 CPU 和記憶體平均使用率在 5% 至 20% 之間的應用程式。在遷移專案中，通常會淘汰這些應用程式或將其保留在內部部署。

IIoT

請參閱工業物聯網。

不可變的基礎設施

為生產工作負載部署新基礎設施的模型，而不是更新、修補或修改現有的基礎設施。不可變基礎設施本質上比可變基礎設施更一致、可靠且可預測。如需詳細資訊，請參閱 AWS Well-Architected Framework 中的使用不可變基礎設施部署最佳實務。

傳入 (輸入) VPC

在 AWS 多帳戶架構中，接受、檢查和路由來自應用程式外部之網路連線的 VPC。AWS 安全參考架構建議您使用傳入、傳出和檢查 VPC 來設定網路帳戶，以保護應用程式與更廣泛的網際網路之間的雙向介面。

增量遷移

一種切換策略，您可以在其中將應用程式分成小部分遷移，而不是執行單一、完整的切換。例如，您最初可能只將一些微服務或使用者移至新系統。確認所有項目都正常運作之後，您可以逐步移動其他微服務或使用者，直到可以解除委任舊式系統。此策略可降低與大型遷移關聯的風險。

工業 4.0

2016 年 Klaus Schwab 推出的術語，透過連線能力、即時資料、自動化、分析和 AI/ML 的進展，指製造程序的現代化。

基礎設施

應用程式環境中包含的所有資源和資產。

基礎設施即程式碼 (IaC)

透過一組組態檔案來佈建和管理應用程式基礎設施的程序。IaC 旨在協助您集中管理基礎設施，標準化資源並快速擴展，以便新環境可重複、可靠且一致。

工業物聯網 (IIoT)

在製造業、能源、汽車、醫療保健、生命科學和農業等產業領域使用網際網路連線的感測器和裝置。如需詳細資訊，請參閱建立工業物聯網 (IIoT) 數位轉型策略。

檢查 VPC

在 AWS 多帳戶架構中，集中式 VPC，可管理 VPCs 之間 （在相同或不同的 中 AWS 區域)、網際網路和內部部署網路之間的網路流量檢查。AWS 安全參考架構建議您使用傳入、傳出和檢查 VPC 來設定網路帳戶，以保護應用程式與更廣泛的網際網路之間的雙向介面。

物聯網 (IoT)

具有內嵌式感測器或處理器的相連實體物體網路，其透過網際網路或本地通訊網路與其他裝置和系統進行通訊。如需詳細資訊，請參閱什麼是 IoT？

可解釋性

機器學習模型的一個特徵，描述了人類能夠理解模型的預測如何依賴於其輸入的程度。如需詳細資訊，請參閱 的機器學習模型可解譯性 AWS。

IoT

請參閱物聯網。

IT 資訊庫 (ITIL)

一組用於交付 IT 服務並使這些服務與業務需求保持一致的最佳實務。ITIL 為 ITSM 提供了基礎。

IT 服務管理 (ITSM)

與組織的設計、實作、管理和支援 IT 服務關聯的活動。如需有關將雲端操作與 ITSM 工具整合的資訊，請參閱操作整合指南。

ITIL

請參閱 IT 資訊庫。

ITSM

請參閱 IT 服務管理。

L

標籤型存取控制 (LBAC)

強制存取控制 (MAC) 的實作，其中使用者和資料本身都會獲得明確指派的安全標籤值。使用者安全標籤和資料安全標籤之間的交集會決定使用者可以看到哪些資料列和資料欄。

登陸區域

登陸區域是架構良好的多帳戶 AWS 環境，可擴展且安全。這是一個起點，您的組織可以從此起點快速啟動和部署工作負載與應用程式，並對其安全和基礎設施環境充滿信心。如需有關登陸區域的詳細資訊，請參閱設定安全且可擴展的多帳戶 AWS 環境。

大型語言模型 (LLM)

預先訓練大量資料的深度學習 AI 模型。LLM 可以執行多個任務，例如回答問題、摘要文件、將文字翻譯成其他語言，以及完成句子。如需詳細資訊，請參閱什麼是 LLMs。

大型遷移

遷移 300 部或更多伺服器。

LBAC

請參閱標籤型存取控制。

最低權限

授予執行任務所需之最低許可的安全最佳實務。如需詳細資訊，請參閱 IAM 文件中的套用最低權限許可。

隨即轉移

請參閱 7 Rs。

小端序系統

首先儲存最低有效位元組的系統。另請參閱 Endianness。

LLM

請參閱大型語言模型。

較低的環境

請參閱 環境。

M

機器學習 (ML)

一種使用演算法和技術進行模式識別和學習的人工智慧。機器學習會進行分析並從記錄的資料 (例如物聯網 (IoT) 資料) 中學習，以根據模式產生統計模型。如需詳細資訊，請參閱機器學習。

主要分支

請參閱分支。

惡意軟體

旨在危及電腦安全或隱私權的軟體。惡意軟體可能會中斷電腦系統、洩露敏感資訊，或取得未經授權的存取。惡意軟體的範例包括病毒、蠕蟲、勒索軟體、特洛伊木馬、間諜軟體和鍵盤記錄器。

受管服務

AWS 服務 會 AWS 操作基礎設施層、作業系統和平台，而您會存取端點來存放和擷取資料。Amazon Simple Storage Service (Amazon S3) 和 Amazon DynamoDB 是受管服務的範例。這些也稱為抽象服務。

製造執行系統 (MES)

一種軟體系統，用於追蹤、監控、記錄和控制生產程序，將原物料轉換為現場成品。

MAP

請參閱遷移加速計劃。

MCP

請參閱模型內容通訊協定。

模型內容通訊協定 (MCP)

用於代理程式對工具通訊的無狀態通訊協定。

MCP 伺服器

透過模型內容通訊協定公開一或多個工具的服務。

機制

建立工具、推動工具採用，然後檢查結果以進行調整的完整程序。機制是在操作時強化和改善自身的循環。如需詳細資訊，請參閱 AWS Well-Architected Framework 中的建置機制。

成員帳戶

屬於組織一部分的管理帳戶 AWS 帳戶 以外的所有 AWS Organizations。帳戶一次只能是一個組織的成員。

製造執行系統

請參閱製造執行系統。

訊息佇列遙測傳輸 (MQTT)

根據發佈/訂閱模式的輕量型machine-to-machine(M2M) 通訊協定，適用於資源受限的 IoT 裝置。

微服務

一種小型的獨立服務，它可透過定義明確的 API 進行通訊，通常由小型獨立團隊擁有。例如，保險系統可能包含對應至業務能力 (例如銷售或行銷) 或子領域 (例如購買、索賠或分析) 的微服務。微服務的優點包括靈活性、彈性擴展、輕鬆部署、可重複使用的程式碼和適應力。如需詳細資訊，請參閱使用無 AWS 伺服器服務整合微服務。

微服務架構

一種使用獨立元件來建置應用程式的方法，這些元件會以微服務形式執行每個應用程式程序。這些微服務會使用輕量型 API，透過明確定義的介面進行通訊。此架構中的每個微服務都可以進行更新、部署和擴展，以滿足應用程式特定功能的需求。如需詳細資訊，請參閱在 上實作微服務 AWS。

Migration Acceleration Program (MAP)

此 AWS 計畫提供諮詢支援、訓練和服務，以協助組織建立強大的營運基礎，以移至雲端，並協助抵銷遷移的初始成本。MAP 包括用於有條不紊地執行舊式遷移的遷移方法以及一組用於自動化和加速常見遷移案例的工具。

大規模遷移

將大部分應用程式組合依波次移至雲端的程序，在每個波次中，都會以更快的速度移動更多應用程式。此階段使用從早期階段學到的最佳實務和經驗教訓來實作團隊、工具和流程的遷移工廠，以透過自動化和敏捷交付簡化工作負載的遷移。這是 AWS 遷移策略的第三階段。

遷移工廠

可透過自動化、敏捷的方法簡化工作負載遷移的跨職能團隊。遷移工廠團隊通常包括營運、業務分析師和擁有者、遷移工程師、開發人員以及從事 Sprint 工作的 DevOps 專業人員。20% 至 50% 之間的企業應用程式組合包含可透過工廠方法優化的重複模式。如需詳細資訊，請參閱此內容集中的遷移工廠的討論和雲端遷移工廠指南。

遷移中繼資料

有關完成遷移所需的應用程式和伺服器的資訊。每種遷移模式都需要一組不同的遷移中繼資料。遷移中繼資料的範例包括目標子網路、安全群組和 AWS 帳戶。

遷移模式

可重複的遷移任務，詳細描述遷移策略、遷移目的地以及所使用的遷移應用程式或服務。範例：使用 AWS Application Migration Service 重新託管遷移至 Amazon EC2。

遷移組合評定 (MPA)

線上工具，提供驗證商業案例以遷移至 的資訊 AWS 雲端。MPA 提供詳細的組合評定 (伺服器適當規模、定價、總體擁有成本比較、遷移成本分析) 以及遷移規劃 (應用程式資料分析和資料收集、應用程式分組、遷移優先順序，以及波次規劃)。MPA 工具 （需要登入） 可供所有 AWS 顧問和 APN 合作夥伴顧問免費使用。

遷移準備程度評定 (MRA)

使用 AWS CAF 取得組織雲端整備狀態的洞見、識別優缺點，以及建立行動計劃以消除已識別差距的程序。如需詳細資訊，請參閱遷移準備程度指南。MRA 是 AWS 遷移策略的第一階段。

遷移策略

用來將工作負載遷移至 的方法 AWS 雲端。如需詳細資訊，請參閱本詞彙表中的 7 個 Rs 項目，並請參閱動員您的組織以加速大規模遷移。

機器學習 (ML)

請參閱機器學習。

現代化

將過時的 (舊版或單一) 應用程式及其基礎架構轉換為雲端中靈活、富有彈性且高度可用的系統，以降低成本、提高效率並充分利用創新。如需詳細資訊，請參閱《》中的現代化應用程式的策略 AWS 雲端。

現代化準備程度評定

這項評估可協助判斷組織應用程式的現代化準備程度；識別優點、風險和相依性；並確定組織能夠在多大程度上支援這些應用程式的未來狀態。評定的結果就是目標架構的藍圖、詳細說明現代化程序的開發階段和里程碑的路線圖、以及解決已發現的差距之行動計畫。如需詳細資訊，請參閱《》中的評估應用程式的現代化準備 AWS 雲端程度。

單一應用程式 (單一)

透過緊密結合的程序作為單一服務執行的應用程式。單一應用程式有幾個缺點。如果一個應用程式功能遇到需求激增，則必須擴展整個架構。當程式碼庫增長時，新增或改進單一應用程式的功能也會變得更加複雜。若要解決這些問題，可以使用微服務架構。如需詳細資訊，請參閱將單一體系分解為微服務。

MPA

請參閱遷移產品組合評估。

MQTT

請參閱訊息佇列遙測傳輸。

多類別分類

一個有助於產生多類別預測的過程 (預測兩個以上的結果之一)。例如，機器學習模型可能會詢問「此產品是書籍、汽車還是電話？」 或者「這個客戶對哪種產品類別最感興趣？」

可變基礎設施

更新和修改生產工作負載現有基礎設施的模型。為了提高一致性、可靠性和可預測性， AWS Well-Architected Framework 建議使用不可變基礎設施做為最佳實務。

O

OAC

請參閱原始存取控制。

OAI

請參閱原始存取身分。

OCM

請參閱組織變更管理。

離線遷移

一種遷移方法，可在遷移過程中刪除來源工作負載。此方法涉及延長停機時間，通常用於小型非關鍵工作負載。

OI

請參閱 操作整合。

OLA

請參閱操作層級協議。

線上遷移

一種遷移方法，無需離線即可將來源工作負載複製到目標系統。連接至工作負載的應用程式可在遷移期間繼續運作。此方法涉及零至最短停機時間，通常用於關鍵的生產工作負載。

OPC-UA

請參閱開放程序通訊 - 統一架構。

開放程序通訊 - 統一架構 (OPC-UA)

用於工業自動化的machine-to-machine(M2M) 通訊協定。OPC-UA 提供資料加密、身分驗證和授權機制的互通性標準。

操作水準協議 (OLA)

一份協議，闡明 IT 職能群組承諾向彼此提供的內容，以支援服務水準協議 (SLA)。

操作整備審查 (ORR)

問題和相關最佳實務的檢查清單，可協助您了解、評估、預防或減少事件和可能失敗的範圍。如需詳細資訊，請參閱 AWS Well-Architected Framework 中的操作準備審查 (ORR)。

操作技術 (OT)

使用實體環境控制工業操作、設備和基礎設施的硬體和軟體系統。在製造中，整合 OT 和資訊技術 (IT) 系統是工業 4.0 轉型的關鍵重點。

操作整合 (OI)

在雲端中將操作現代化的程序，其中包括準備程度規劃、自動化和整合。如需詳細資訊，請參閱操作整合指南。

組織追蹤

由 建立的線索 AWS CloudTrail 會記錄 AWS 帳戶 組織中所有 的所有事件 AWS Organizations。在屬於組織的每個 AWS 帳戶 中建立此追蹤，它會跟蹤每個帳戶中的活動。如需詳細資訊，請參閱 CloudTrail 文件中的建立組織追蹤。

組織變更管理 (OCM)

用於從人員、文化和領導力層面管理重大、顛覆性業務轉型的架構。OCM 透過加速變更採用、解決過渡問題，以及推動文化和組織變更，協助組織為新系統和策略做好準備，並轉移至新系統和策略。在 AWS 遷移策略中，此架構稱為人員加速，因為雲端採用專案所需的變更速度。如需詳細資訊，請參閱 OCM 指南。

原始存取控制 (OAC)

CloudFront 中的增強型選項，用於限制存取以保護 Amazon Simple Storage Service (Amazon S3) 內容。OAC 支援所有 S3 儲存貯體中的所有伺服器端加密 AWS KMS (SSE-KMS) AWS 區域，以及對 S3 儲存貯體的動態PUT和DELETE請求。

原始存取身分 (OAI)

CloudFront 中的一個選項，用於限制存取以保護 Amazon S3 內容。當您使用 OAI 時，CloudFront 會建立一個可供 Amazon S3 進行驗證的主體。經驗證的主體只能透過特定 CloudFront 分發來存取 S3 儲存貯體中的內容。另請參閱 OAC，它可提供更精細且增強的存取控制。

ORR

請參閱操作整備審核。

OT

請參閱操作技術。

傳出 (輸出) VPC

在 AWS 多帳戶架構中，處理從應用程式內啟動之網路連線的 VPC。AWS 安全參考架構建議您使用傳入、傳出和檢查 VPC 來設定網路帳戶，以保護應用程式與更廣泛的網際網路之間的雙向介面。

P

許可界限

附接至 IAM 主體的 IAM 管理政策，可設定使用者或角色擁有的最大許可。如需詳細資訊，請參閱 IAM 文件中的許可界限。

個人身分識別資訊 (PII)

直接檢視或與其他相關資料配對時，可用來合理推斷個人身分的資訊。PII 的範例包括名稱、地址和聯絡資訊。

PII

請參閱個人身分識別資訊。

手冊

一組預先定義的步驟，可擷取與遷移關聯的工作，例如在雲端中提供核心操作功能。手冊可以採用指令碼、自動化執行手冊或操作現代化環境所需的程序或步驟摘要的形式。

PLC

請參閱可程式設計邏輯控制器。

PLM

請參閱產品生命週期管理。

政策

可定義許可的物件（請參閱身分型政策）、指定存取條件（請參閱資源型政策），或定義組織中所有帳戶的最大許可 AWS Organizations（請參閱服務控制政策）。

混合持久性

根據資料存取模式和其他需求，獨立選擇微服務的資料儲存技術。如果您的微服務具有相同的資料儲存技術，則其可能會遇到實作挑戰或效能不佳。如果微服務使用最適合其需求的資料儲存，則可以更輕鬆地實作並達到更好的效能和可擴展性。

組合評定

探索、分析應用程式組合並排定其優先順序以規劃遷移的程序。如需詳細資訊，請參閱評估遷移準備程度。

述詞

傳回 `true`或 的查詢條件`false`，通常位於 `WHERE` 子句中。

述詞下推

一種資料庫查詢最佳化技術，可在傳輸前篩選查詢中的資料。這可減少必須從關聯式資料庫擷取和處理的資料量，並改善查詢效能。

預防性控制

旨在防止事件發生的安全控制。這些控制是第一道防線，可協助防止對網路的未經授權存取或不必要變更。如需詳細資訊，請參閱在 AWS上實作安全控制中的預防性控制。

委託人

中可執行動作和存取資源 AWS 的實體。此實體通常是 AWS 帳戶、IAM 角色或使用者的根使用者。如需詳細資訊，請參閱 IAM 文件中角色術語和概念中的主體。

設計隱私權

透過整個開發程序將隱私權納入考量的系統工程方法。

私有託管區域

一種容器，它包含有關您希望 Amazon Route 53 如何回應一個或多個 VPC 內的域及其子域之 DNS 查詢的資訊。如需詳細資訊，請參閱 Route 53 文件中的使用私有託管區域。

主動控制

旨在防止部署不合規資源的安全控制。這些控制項會在佈建資源之前對其進行掃描。如果資源不符合控制項，則不會佈建。如需詳細資訊，請參閱 AWS Control Tower 文件中的控制項參考指南，並參閱實作安全控制項中的主動控制項。 AWS

產品生命週期管理 (PLM)

管理產品整個生命週期的資料和程序，從設計、開發和啟動，到成長和成熟，再到拒絕和移除。

生產環境

請參閱 環境。

可程式設計邏輯控制器 (PLC)

在製造中，高度可靠、可調整的電腦，可監控機器並自動化製造程序。

提示鏈結

使用一個 LLM 提示的輸出做為下一個提示的輸入，以產生更好的回應。此技術用於將複雜任務分解為子任務，或反覆精簡或展開初步回應。它有助於提高模型回應的準確性和相關性，並允許更精細、個人化的結果。

擬匿名化

以預留位置值取代資料集中個人識別符的程序。假名化有助於保護個人隱私權。假名化資料仍被視為個人資料。

發佈/訂閱 (pub/sub)

一種模式，可啟用微服務之間的非同步通訊，以提高可擴展性和回應能力。例如，在微服務型 MES 中，微服務可以將事件訊息發佈到其他微服務可訂閱的頻道。系統可以新增新的微服務，而無需變更發佈服務。

Q

查詢計劃

一系列步驟，如指示，用於存取 SQL 關聯式資料庫系統中的資料。

查詢計劃迴歸

在資料庫服務優化工具選擇的計畫比對資料庫環境進行指定的變更之前的計畫不太理想時。這可能因為對統計資料、限制條件、環境設定、查詢參數繫結的變更以及資料庫引擎的更新所導致。

R

RACI 矩陣

請參閱負責、負責、諮詢、告知 (RACI)。

RAG

請參閱擷取增強生成。

勒索軟體

一種惡意軟體，旨在阻止對計算機系統或資料的存取，直到付款為止。

RASCI 矩陣

請參閱負責、負責、諮詢、告知 (RACI)。

RCAC

請參閱資料列和資料欄存取控制。

僅供讀取複本

用於唯讀用途的資料庫複本。您可以將查詢路由至僅供讀取複本以減少主資料庫的負載。

重新架構師

請參閱 7 個 R。

復原點目標 (RPO)

自上次資料復原點以來可接受的時間上限。這會決定最後一個復原點與服務中斷之間可接受的資料遺失。

復原時間目標 (RTO)

服務中斷與服務還原之間的可接受延遲上限。

重構

請參閱 7 個 R。

區域

地理區域中的 AWS 資源集合。每個 AWS 區域 都獨立於其他 ，以提供容錯能力、穩定性和彈性。如需詳細資訊，請參閱指定 AWS 區域 您的帳戶可以使用哪些 。

迴歸

預測數值的 ML 技術。例如，為了解決「這房子會賣什麼價格？」的問題 ML 模型可以使用線性迴歸模型，根據已知的房屋事實 (例如，平方英尺) 來預測房屋的銷售價格。

重新託管

請參閱 7 個 R。

版本

在部署程序中，它是將變更提升至生產環境的動作。

重新放置

請參閱 7 個 R。

Replatform

請參閱 7 個 R。

回購

請參閱 7 個 R。

彈性

應用程式抵禦中斷或從中斷中復原的能力。在 中規劃彈性時，高可用性和災難復原是常見的考量 AWS 雲端。如需詳細資訊，請參閱AWS 雲端 彈性。

資源型政策

附接至資源的政策，例如 Amazon S3 儲存貯體、端點或加密金鑰。這種類型的政策會指定允許存取哪些主體、支援的動作以及必須滿足的任何其他條件。

負責者、當責者、事先諮詢者和事後告知者 (RACI) 矩陣

矩陣，定義所有參與遷移活動和雲端操作之各方的角色和責任。矩陣名稱衍生自矩陣中定義的責任類型：負責人 (R)、責任 (A)、諮詢 (C) 和知情 (I)。支援 (S) 類型為選用。如果您包含支援，則矩陣稱為 RASCI 矩陣，如果您排除它，則稱為 RACI 矩陣。

回應性控制

一種安全控制，旨在驅動不良事件或偏離安全基準的補救措施。如需詳細資訊，請參閱在 AWS上實作安全控制中的回應性控制。

保留

請參閱 7 個 R。

淘汰

請參閱 7 個 R。

檢索增強生成 (RAG)

一種生成式 AI 技術，其中 LLM 會在產生回應之前參考訓練資料來源以外的授權資料來源。例如，RAG 模型可能會對組織的知識庫或自訂資料執行語意搜尋。如需詳細資訊，請參閱什麼是 RAG。

輪換

定期更新秘密的程序，讓攻擊者更難存取登入資料。

資料列和資料欄存取控制 (RCAC)

使用已定義存取規則的基本、彈性 SQL 表達式。RCAC 包含資料列許可和資料欄遮罩。

RPO

請參閱復原點目標。

RTO

請參閱復原時間目標。

執行手冊

執行特定任務所需的一組手動或自動程序。這些通常是為了簡化重複性操作或錯誤率較高的程序而建置。

S

SAML 2.0

許多身分提供者 (IdP) 使用的開放標準。此功能會啟用聯合單一登入 (SSO)，讓使用者可以登入 AWS 管理主控台 或呼叫 AWS API 操作，而不必為您組織中的每個人在 IAM 中建立使用者。如需有關以 SAML 2.0 為基礎的聯合詳細資訊，請參閱 IAM 文件中的關於以 SAML 2.0 為基礎的聯合。

斯卡達

請參閱監督控制和資料擷取。

SCP

請參閱服務控制政策。

秘密

您以加密形式存放的 AWS Secrets Manager機密或限制資訊，例如密碼或使用者登入資料。它由秘密值及其中繼資料組成。秘密值可以是二進位、單一字串或多個字串。如需詳細資訊，請參閱 Secrets Manager 秘密中的內容？ Secrets Manager 文件中的 。

設計安全性

透過整個開發程序將安全性納入考量的系統工程方法。

安全控制

一種技術或管理防護機制，它可預防、偵測或降低威脅行為者利用安全漏洞的能力。安全控制有四種主要類型：預防性、偵測性、回應性和主動性。

安全強化

減少受攻擊面以使其更能抵抗攻擊的過程。這可能包括一些動作，例如移除不再需要的資源、實作授予最低權限的安全最佳實務、或停用組態檔案中不必要的功能。

安全資訊與事件管理 (SIEM) 系統

結合安全資訊管理 (SIM) 和安全事件管理 (SEM) 系統的工具與服務。SIEM 系統會收集、監控和分析來自伺服器、網路、裝置和其他來源的資料，以偵測威脅和安全漏洞，並產生提醒。

安全回應自動化

預先定義和程式設計的動作，旨在自動回應或修復安全事件。這些自動化可做為偵測或回應式安全控制，協助您實作 AWS 安全最佳實務。自動化回應動作的範例包括修改 VPC 安全群組、修補 Amazon EC2 執行個體或輪換登入資料。

伺服器端加密

由 AWS 服務 接收資料的 在其目的地加密資料。

服務控制政策 (SCP)

為 AWS Organizations中的組織的所有帳戶提供集中控制許可的政策。SCP 會定義防護機制或設定管理員可委派給使用者或角色的動作限制。您可以使用 SCP 作為允許清單或拒絕清單，以指定允許或禁止哪些服務或動作。如需詳細資訊，請參閱 AWS Organizations 文件中的服務控制政策。

服務端點

的進入點 URL AWS 服務。您可以使用端點，透過程式設計方式連接至目標服務。如需詳細資訊，請參閱 AWS 一般參考 中的 AWS 服務 端點。

服務水準協議 (SLA)

一份協議，闡明 IT 團隊承諾向客戶提供的服務，例如服務正常執行時間和效能。

服務層級指標 (SLI)

服務效能方面的測量，例如其錯誤率、可用性或輸送量。

服務層級目標 (SLO)

代表服務運作狀態的目標指標，由服務層級指標測量。

共同責任模式

描述您與 共同 AWS 承擔雲端安全與合規責任的模型。 AWS 負責雲端的安全，而 負責雲端的安全。如需詳細資訊，請參閱共同責任模式。

陰影 AI

在組織內受管頻道之外建置或使用的未授權 AI 應用程式。

SIEM

請參閱安全資訊和事件管理系統。

單一故障點 (SPOF)

應用程式的單一關鍵元件故障，可能會中斷系統。

SLA

請參閱服務層級協議。

SLI

請參閱服務層級指標。

SLO

請參閱服務層級目標。

先拆分後播種模型

擴展和加速現代化專案的模式。定義新功能和產品版本時，核心團隊會進行拆分以建立新的產品團隊。這有助於擴展組織的能力和服務，提高開發人員生產力，並支援快速創新。如需詳細資訊，請參閱 中的階段式應用程式現代化方法 AWS 雲端。

SPOF

請參閱單一故障點。

星狀結構描述

使用一個大型事實資料表來存放交易或測量資料的資料庫組織結構，並使用一或多個較小的維度資料表來存放資料屬性。此結構旨在用於資料倉儲或商業智慧用途。

Strangler Fig 模式

一種現代化單一系統的方法，它會逐步重寫和取代系統功能，直到舊式系統停止使用為止。此模式源自無花果藤，它長成一棵馴化樹並最終戰勝且取代了其宿主。該模式由 Martin Fowler 引入，作為重寫單一系統時管理風險的方式。如需有關如何套用此模式的範例，請參閱使用容器和 Amazon API Gateway 逐步現代化舊版 Microsoft ASP.NET (ASMX) Web 服務。

子網

您 VPC 中的 IP 地址範圍。子網必須位於單一可用區域。

監控控制和資料擷取 (SCADA)

在製造中，使用硬體和軟體來監控實體資產和生產操作的系統。

對稱加密

使用相同金鑰來加密及解密資料的加密演算法。

合成測試

以模擬使用者互動的方式測試系統，以偵測潛在問題或監控效能。您可以使用 Amazon CloudWatch Synthetics 來建立這些測試。

系統提示

一種向 LLM 提供內容、指示或指導方針以指示其行為的技術。系統提示有助於設定內容，並建立與使用者互動的規則。

T

標籤

做為中繼資料以組織 AWS 資源的鍵值對。標籤可協助您管理、識別、組織、搜尋及篩選資源。如需詳細資訊，請參閱標記您的 AWS 資源。

目標變數

您嘗試在受監督的 ML 中預測的值。這也被稱為結果變數。例如，在製造設定中，目標變數可能是產品瑕疵。

任務清單

用於透過執行手冊追蹤進度的工具。任務清單包含執行手冊的概觀以及要完成的一般任務清單。對於每個一般任務，它包括所需的預估時間量、擁有者和進度。

測試環境

請參閱 環境。

訓練

為 ML 模型提供資料以供學習。訓練資料必須包含正確答案。學習演算法會在訓練資料中尋找將輸入資料屬性映射至目標的模式 (您想要預測的答案)。它會輸出擷取這些模式的 ML 模型。可以使用 ML 模型，來預測您不知道的目標新資料。

tool

代理程式可以叫用以在外部系統中執行操作的函數或 API。

傳輸閘道

可以用於互連 VPC 和內部部署網路的網路傳輸中樞。如需詳細資訊，請參閱 AWS Transit Gateway 文件中的什麼是傳輸閘道。

主幹型工作流程

這是一種方法，開發人員可在功能分支中本地建置和測試功能，然後將這些變更合併到主要分支中。然後，主要分支會依序建置到開發環境、生產前環境和生產環境中。

受信任的存取權

將許可授予您指定的服務，以代表您在組織中 AWS Organizations 及其帳戶中執行任務。受信任的服務會在需要該角色時，在每個帳戶中建立服務連結角色，以便為您執行管理工作。如需詳細資訊，請參閱 文件中的 AWS Organizations 搭配使用 AWS Organizations 與其他 AWS 服務。

調校

變更訓練程序的各個層面，以提高 ML 模型的準確性。例如，可以透過產生標籤集、新增標籤、然後在不同的設定下多次重複這些步驟來訓練 ML 模型，以優化模型。

雙比薩團隊

兩個比薩就能吃飽的小型 DevOps 團隊。雙披薩團隊規模可確保軟體開發中的最佳協作。

U

不確定性

這是一個概念，指的是不精確、不完整或未知的資訊，其可能會破壞預測性 ML 模型的可靠性。有兩種類型的不確定性：認知不確定性是由有限的、不完整的資料引起的，而隨機不確定性是由資料中固有的噪聲和隨機性引起的。

未區分的任務

也稱為繁重工作，這是建立和操作應用程式的必要工作，但不為最終使用者提供直接價值或提供競爭優勢。未區分任務的範例包括採購、維護和容量規劃。

較高的環境

請參閱 環境。

V

清空

一種資料庫維護操作，涉及增量更新後的清理工作，以回收儲存並提升效能。

版本控制

追蹤變更的程序和工具，例如儲存庫中原始程式碼的變更。

VPC 對等互連

兩個 VPC 之間的連線，可讓您使用私有 IP 地址路由流量。如需詳細資訊，請參閱 Amazon VPC 文件中的什麼是 VPC 對等互連。

漏洞

危害系統安全性的軟體或硬體瑕疵。

W

暖快取

包含經常存取的目前相關資料的緩衝快取。資料庫執行個體可以從緩衝快取讀取，這比從主記憶體或磁碟讀取更快。

暖資料

不常存取的資料。查詢這類資料時，通常可接受中等緩慢的查詢。

視窗函數

SQL 函數，對與目前記錄在某種程度上相關的資料列群組執行計算。視窗函數適用於處理任務，例如根據目前資料列的相對位置計算移動平均值或存取資料列的值。

工作負載

提供商業價值的資源和程式碼集合，例如面向客戶的應用程式或後端流程。

工作串流

遷移專案中負責一組特定任務的功能群組。每個工作串流都是獨立的，但支援專案中的其他工作串流。例如，組合工作串流負責排定應用程式、波次規劃和收集遷移中繼資料的優先順序。組合工作串流將這些資產交付至遷移工作串流，然後再遷移伺服器和應用程式。

WORM

請參閱寫入一次，讀取許多。

WQF

請參閱AWS 工作負載資格架構。

寫入一次，讀取許多 (WORM)

儲存模型，可一次性寫入資料，並防止刪除或修改資料。授權使用者可以視需要多次讀取資料，但無法變更資料。此資料儲存基礎設施被視為不可變。

Z

零時差入侵

利用零時差漏洞的攻擊，通常是惡意軟體。

零時差漏洞

生產系統中未緩解的缺陷或漏洞。威脅行為者可以使用這種類型的漏洞來攻擊系統。開發人員經常因為攻擊而意識到漏洞。

零鏡頭提示

提供 LLM 執行任務的指示，但沒有可協助引導任務的範例 (快照)。LLM 必須使用其預先訓練的知識來處理任務。零鏡頭提示的有效性取決於任務的複雜性和提示的品質。另請參閱少量擷取提示。

殭屍應用程式

CPU 和記憶體平均使用率低於 5% 的應用程式。在遷移專案中，通常會淘汰這些應用程式。

本文為英文版的機器翻譯版本，如內容有任何歧義或不一致之處，概以英文版為準。