AWS 雲端採用架構：平台觀點

AWS 方案指引

AWS 方案指引: AWS 雲端採用架構：平台觀點

Table of Contents

AWS 雲端採用架構：平台觀點

Amazon Web Services（貢獻者）

2023 年 10 月 (文件歷史記錄)

數位轉型是高階主管改善客戶體驗、創新和彈性的單一最大推動因素。它使用機器學習 (ML)、人工智慧 (AI)、大數據和雲端的速度和規模，以滿足不斷變化的商業條件和不斷變化的客戶需求。

Amazon Web Services (AWS) 是全球最全面且廣泛採用的雲端平台。它可協助您將組織轉型，同時降低業務風險、改善環境、社會和治理 (ESG) 效能、增加收入並改善營運效率。

AWS 雲端採用架構 (AWS CAF) 使用 AWS 最佳實務來協助您加速業務成果。使用 AWS CAF 來識別轉型機會並排定優先順序、評估和改善雲端準備度，以及反覆發展轉型藍圖。

AWS CAF 將其指引分為六個觀點：商業、人員、控管、平台、安全與營運。每個觀點都包含在單獨的指南中。本指南涵蓋平台的觀點，著重於使用企業級、可擴展的混合雲端環境加速交付您的雲端工作負載。

簡介

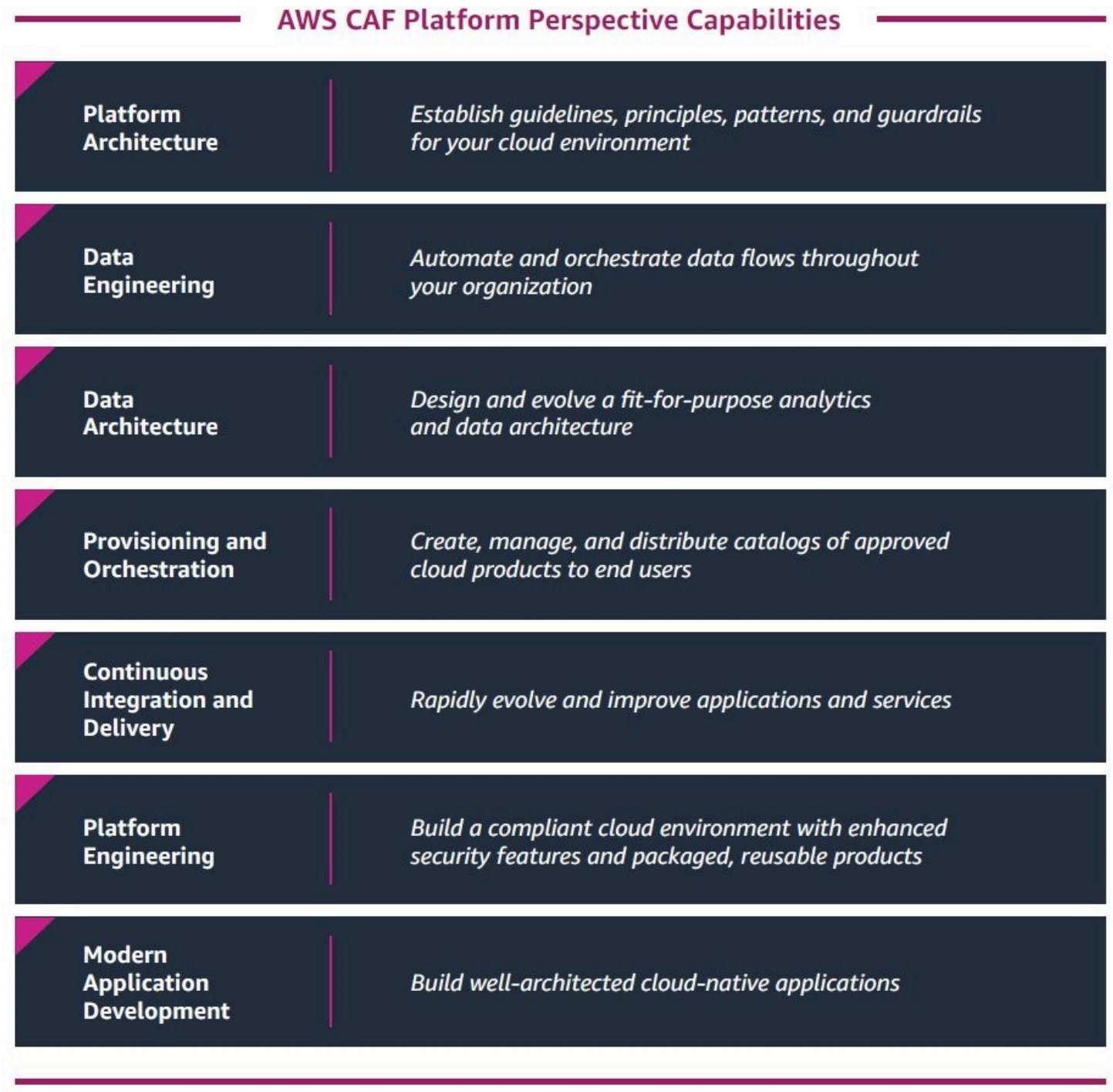
數百萬客戶使用 ，包括成長最快的新創公司、最大的企業和領先的政府組織 AWS。（請參閱 AWS 網站上的客戶成功案例。）他們可以遷移和現代化舊式工作負載、變得更受資料驅動、自動化和最佳化業務流程，以及重塑操作模型。他們能夠透過降低業務風險、改善環境、社會和治理 (ESG) 效能、增加收入並改善營運效率，來改善業務成果。

組織有效使用雲端進行數位轉換的能力（組織雲端整備度）由一組基礎功能提供支援。功能是組織使用程序來部署資源（人員、技術和任何其他有形或無形資產）以實現特定成果的能力。 AWS CAF 會識別這些功能，並提供全球數千個組織已成功用於改善雲端準備程度和加速雲端轉型之旅的規範性指引。

AWS CAF 將其功能分組為六個觀點：

- 商業
- 人物
- 控管
- 平台
- 安全性
- 操作

平台的觀點著重於使用企業級、可擴展的混合雲端環境加速交付您的雲端工作負載。此環境包含下圖所示的七種功能。這些功能由在其雲端轉型旅程中運作相關的利益相關者管理。典型的利益相關者包括技術長 (CTO)、技術領導者、架構師和工程師。



本指南的下列各節會詳細討論這些功能。每個區段提供如何啟動、提升和最終在特定功能中表現出色的指導方針。

- 平台架構
- 平台工程
- 資料架構

- 資料工程
- 佈建和協同運作
- 現代應用程式開發
- 持續整合和持續交付 (CI/CD)

平台視角是 AWS CAF 的關鍵部分。它是所有其他觀點做出的決策匯聚起來以提供業務敏捷性和價值的關聯。此處所做的決策有助於或阻礙您的業務目標達到基礎。 AWS CAF 平台的觀點有助於建立企業級、可擴展的雲端環境，以支持組織的轉型。透過此觀點， AWS CAF 會引導您建立強大的平台，讓您的雲端之旅得以進行，最終帶來重大的業務轉型和成長。

當您完成平台的觀點時，請考慮與需要開發的業務領導者的跨職能聯繫，以及他們為您的團隊和組織帶來的價值。將更多重點放在營運模型變更和團隊拓撲上，以確保符合需求。此外，請尋求開發團隊所需的技能，以建置平台並跨應用程式團隊啟用其使用。當您做出這些決策時，請記住組織的人員、業務、控管、安全和營運目標，這些對於確保平台的採用和工作的成功至關重要。

AWS 和 AWS 合作夥伴網路提供工具和服務，例如研討會和訓練，可協助您在此旅程中實作和改善您的安全狀態。AWS Professional Services 是一個全球專家團隊，可協助您透過一系列符合 AWS CAF 的方案，實現與雲端轉型相關的特定成果。

平台架構

為您的雲端環境建立和維護指導方針、原則、模式和護欄。

架構良好的雲端環境可協助您加速實作、降低風險並推動雲端採用。平台架構功能會在您的組織中建立企業標準的共識，以推動雲端採用。您可以定義最佳實務藍圖和護欄，以促進身分驗證、安全性、聯網以及記錄和監控。此外，您考慮並規劃由於延遲、資料處理或資料駐留需求而需要保留在現場部署的工作負載，並評估混合雲端使用案例，例如雲端爆增、備份和災難復原到雲端、分散式資料處理和邊緣運算。

Start

定義多帳戶策略

良好的多帳戶策略會考慮擴展和營運效率問題。這表示將您的工作負載隔離為最符合您營運需求的邏輯模式。我們建議您從一組基礎帳戶開始，以適應企業中的集中式和分散式服務。您可以集中安全、財務和營運功能，以有效地管理和管理您的分散式和自主團隊和帳戶。您會希望在整個組織中保持一致，以了解平台和工作負載將如何分割和管理。了解此結構可協助您確保安全原則適用於身分驗證和授權，同時符合平台不斷發展的可接受使用政策。

定義預防性控制

使用一組內嵌的預設控制項 (護欄) 規劃安全的多帳戶環境。開始了解並使用服務控制政策 (SCPs) 等機制來管理整個組織的服務使用，包括 AWS 區域 可在雲端平台內使用的 。政策提供集中式機制，用於控制所有帳戶的可用許可上限，並確保他們遵守組織的存取控制準則。

定義組織單位結構

組織單位 (OUs) 是根據法規要求和軟體開發生命週期 (SDLC) 環境管理和分類帳戶的實用方式。透過使用 OUs，組織可簡化在其雲端基礎設施中申請適當政策和許可的程序。工作負載 OUs專為支援應用程式基礎設施資源的帳戶而設計，並確保強制執行正確的政策。使用 OUs和 SCPs 有助於增強組織的雲端基礎設施的安全性和合規性，同時確保應用程式和服務的運作順暢。這最終會導致更有效率且強大的雲端採用程序。

定義網路連線

網路連線是任何雲端基礎設施的重要層面，可支援建立安全、可擴展且高可用性的網路，以支援應用程式和工作負載。設計良好的網路可提供一致的高效能，並確保跨不同環境的無縫操作。

當您設計網路架構時，請考慮您是否因為延遲、資料處理或資料駐留需求而想要在內部部署中保留工作負載。透過評估混合雲端使用案例，例如雲端暴增、雲端備份和災難復原、分散式資料處理和邊緣運算，您可以識別下列方面的關鍵需求：

- 往返網際網路的連線能力。這方面涉及在您的應用程式或工作負載與網際網路之間提供安全可靠的連線。這種連線對於促進對 Web 資源的存取、啟用使用者和應用程式之間的通訊，並確保您的服務在需要時可供大眾存取至關重要。
- 跨雲端環境的連線能力。此區域著重於在您的雲端基礎設施內的各種元件和服務之間建立強大的連線。它可確保資料和資源可在不同的雲端服務之間輕鬆共用和存取，進而促進高效率的協同合作和更順暢的操作。此處的關鍵考量是您使用虛擬私有雲端 (VPCs)。為了保持簡單，請考慮建立如何建立和追蹤 VPCs 的標準。請考慮以程式設計方式建立這些標準，並計劃使用 IP 地址管理 (IPAM) 解決方案。配置足夠的 IP 空間以允許增長，並設計子網路結構，以便在使用多個可用區域時輕鬆進行故障診斷。當您設計和實作網路連線時，請務必遵循 VPCs 的安全最佳實務。
- 內部部署網路與雲端環境之間的連線。此方面處理現場部署基礎設施與雲端環境的整合。透過在兩者之間建立安全可靠的連線，組織可以從混合架構的優勢中受益。例如，您可以同時使用內部部署資源和雲端服務，以提高效能、可擴展性和成本最佳化。

透過解決網路連線的這三個關鍵領域，您可以建置強大的雲端基礎設施，以有效支援您的應用程式和工作負載，讓您可以利用雲端採用的優勢。請注意聯網需求，並建立簡單的設計，讓您能夠根據多帳戶策略進行擴展。

定義 DNS 策略

妥善規劃的 DNS 策略可協助您避免雲端環境成長時的複雜性。如果您維護內部部署 DNS 功能，建議您針對任何雲端型 DNS 需求，設計使用內部部署 DNS 基礎設施和雲端 DNS 的混合 DNS 架構。使用解析程式端點和轉送規則，將 DNS 解析與內部部署 DNS 環境整合。使用私有託管區域來保留有關您希望雲端 DNS 如何回應一個或多個網路中網域及其子網域的查詢的資訊。

定義標記標準

標記資源是有效管理成本並識別資源擁有權的重要實務。請考慮您的組織如何進一步允許雲端使用，包括使用平台內的特定服務。定義標記策略，追蹤哪些團隊正在部署哪些資源。從 AWS CAF Operations 的觀點取得輸入，並使用標籤來自動化已部署基礎設施的任務。

此外，透過標記具有相關中繼資料的資源，您可以根據 AWS CAF 控管角度中雲端財務管理 (CFM) 功能中指定的組織需求來分組和追蹤您的花費。識別支援會計和財務實務的報告機制，包括違反財務政策時要採取的動作。

定義可觀測性策略

建立可觀測性策略是最佳化和保護雲端架構的關鍵步驟。此策略涉及將雲端服務產生的指標和日誌轉換為可行的洞察，以進行策略決策。排定監控關鍵績效指標的優先順序，並設定提醒來預先解決潛在問題。為了防止工具擴散、最佳化成本，並專注於對組織最重要的事項，請在您的平台和應用程式中整合此可觀測性策略。如需進一步指引，請參閱我們有關制定可觀測性策略的簡報 (AWS re：Invent 2022)。

進階

定義主動和偵測性控制

若要提升，您的組織必須識別環境中主動和偵測性控制 (護欄） 的需求。建立政策來定義角色和使用者在組織單位 (OU) 內帳戶中所擁有的護欄或限制。檢閱平台的任何預設偵測護欄，然後選擇要套用的護欄。視需要建立額外的預防性和偵測性控制，並依 OUs 分組，使其與您的多帳戶策略保持一致。考慮您需要哪些組織工具和機制來檢查偵測控制項識別的不合規資源。

定義服務加入的標準

建立可接受使用平台的標準，以及與服務使用量相關的模式，以及如何控管。考慮允許使用哪些初始服務。建立概述這些標準的文件，並將其發佈至平台的使用者和運算子。確保這些標準隨著時間調整，以滿足組織不斷變化的目標和雲端運算不斷發展的功能。

定義模式和原則

考慮使用應用程式擁有者的輸入，在您的組織內允許哪些架構模式，並開始定義標準化的藍圖。當您在雲端中擴展時，標準化可實現更大的控管和更低的管理負擔。定義將使用基礎設施做為程式碼 (IaC) 的模式，並使用整合到您的變更控制程序和 IT 服務管理 (ITSM) 系統的服務目錄來規劃簡化的部署模型。定義如何使用這些藍圖，以及允許例外狀況的情況。規劃這些例外狀況及其控管，並考量身分驗證、安全監控和防護機制。

Excel

定義修復模式

考慮如何註釋偵測性護欄調查結果並排定優先順序，以便根據您的安全與合規架構進行修復。 計劃使用自動化來偵測out-of-policy佈建，包括違反預算和標記政策的項目。識別在更新 Runbook 和程序手冊

時設定和測量服務層級目標所需的功能。設定這些實務和意見回饋機制的定期審查，以擷取與平台演變相關的資料。定義相應地建立和更新 Runbook 和程序手冊的機制。

溝通和精簡政策

為所有文件建立集中式內容管理系統，並將其分發給平台的使用者和運算子。建立機制來擷取意見回饋，以供未來考慮政策的變更。

了解財務管理功能

當組織保持對預算的透明和全面了解時，他們就會茁壯成長。這讓他們能夠做出明智的決策、有效率地配置資源，以及實現其策略目標。透過促進明智的決策、有效的資源分配、成本控制、效能衡量以及責任和合規維護，清晰的預算檢視有助於組織實現卓越。這最終會產生更有效率、財務穩定且繁榮的組織。當您有成功的標記策略時，您可以在 中使用成本篩選條件AWS Budgets，根據資源標籤篩選費用。這可協助您建立針對特定專案、部門、環境或其他條件量身打造的預算，進一步增強財務管理功能。您可以將成本分配標籤和AWS Cost Categories與標籤建立關聯，以便在報告成本時推動財務洞察和透明度。

平台工程

使用封裝、可重複使用的雲端產品，建置安全且合規的多帳戶雲端環境。

為了讓開發團隊能夠支援創新，平台需要快速調整以因應業務需求。（請參閱 AWS CAF Business 的觀點。）它必須這樣做， 同時具有足夠彈性以適應產品管理需求、足夠嚴格以遵守安全限制，以及足夠快速以啟用營運需求。此程序需要建置具有增強安全性功能的 合規多帳戶雲端環境，以及封裝、可重複使用的雲端產品。

有效的雲端環境可讓您的團隊輕鬆佈建新帳戶，同時確保這些帳戶符合組織政策。一組精選的雲端產品可讓您編寫最佳實務、協助您管理，並協助提高雲端部署的速度和一致性。部署您的最佳實務藍圖，以及偵測和預防性護欄。整合您的雲端環境與現有的環境，以啟用所需的混合雲端使用案例。

自動化帳戶佈建工作流程，並使用多個帳戶來支援您的安全和控管目標。設定內部部署和雲端環境之間以及不同雲端帳戶之間的連線。在現有身分提供者 (IdP) 和雲端環境之間實作聯合，讓使用者可以使用其現有的登入憑證進行身分驗證。集中記錄、建立跨帳戶安全稽核、建立傳入和傳出 DNS 解析程式，以及取得您帳戶和護欄的儀表板可見性。

評估和認證雲端服務的耗用量，以符合公司標準和組態管理。將企業標準封裝為自助式可部署產品和消耗品服務，並持續改善。利用基礎設施即程式碼 (IaC) 以宣告方式定義組態。建立支援團隊，將平台傳播給開發人員和商業使用者，並讓他們能夠建置整合，以加速整個組織的採用。

完成以下章節中討論的任務需要您建置功能和團隊，讓您的組織發展為現代平台工程。如需技術詳細資訊，請參閱在白皮書上建立您的雲端基礎 AWS。

Start

建置登陸區域並部署護欄

當您開始成熟平台工程的旅程時，您必須先使用平台架構功能中定義的偵測性和預防性護欄來部署登陸區域。護欄可確保不會因為應用程式擁有者使用雲端資源而違反組織標準。透過此機制，您可以自動化帳戶佈建工作流程，以使用多個支援安全和控管目標的帳戶。 https://docs.aws.amazon.com/whitepapers/latest/overview-aws-cloud-adoption-framework/security-perspective.html https://docs.aws.amazon.com/whitepapers/latest/overview-aws-cloud-adoption-framework/governance-perspective.html

建立身分驗證

根據 AWS CAF 安全角度中規定的標準，在所有環境、系統、工作負載和程序中實作身分管理和存取控制。對於人力身分，請限制 AWS Identity and Access Management (IAM) 使用者的使用，並改用可讓您在集中位置管理身分的身分提供者。這可讓您更輕鬆地管理多個應用程式與服務的存取權，因為您是從單一位置建立、管理和撤銷存取權。使用現有程序來管理建立、更新和移除存取權，以包含您的 AWS 環境。

部署您的網路

根據您的平台架構設計，建立集中式網路帳戶，以控制進出您環境的傳入和傳出流量。我們建議您設計網路，以便在內部部署網路和 AWS 環境之間、網際網路之間以及整個 AWS 環境中快速佈建連線。集中網路管理可讓您部署網路控制，透過使用預防性和被動控制來隔離整個環境的網路和連線。

收集、彙總和保護事件和日誌資料

使用 Amazon CloudWatch 跨帳戶可觀測性。它提供統一的界面來搜尋、視覺化和分析連結帳戶的指標、日誌和追蹤，並消除帳戶界限。

如果您的組織有集中式日誌控制和安全性的特定合規要求，請考慮設定專用日誌封存帳戶。這提供專門用於日誌資料的集中式加密儲存庫。透過定期輪換加密金鑰來增強此封存的安全性。

實作保護敏感日誌資料的強大政策，並視需要使用遮罩技術。將日誌彙總用於合規、安全性和稽核日誌，並確保使用嚴格的護欄和身分建構，以防止對日誌組態進行未經授權的變更。

建立控制項

根據 AWS CAF 安全角度的定義，部署符合您業務需求的基礎安全功能。部署額外的預防性和偵測性控制，並視需要以程式設計方式和一致地在您的所有帳戶中佈建這些控制。將偵測性控制整合到平台架構功能定義的操作工具中，以便操作機制可以檢閱不合規的資源。

實作雲端財務管理

根據 AWS CAF 治理的觀點，實作成本分配標籤和 AWS Cost Categories，使組織的標記策略符合雲端消費的財務責任。 AWS Cost Categories可讓您使用 中發佈的 AWS Cost Explorer和 帳單資料等工具，向內部成本中心收費或顯示雲端費用AWS Cost and Usage Report。

進階

建置基礎設施自動化

在繼續之前，請評估和認證雲端服務，以符合您的平台架構。然後，將企業標準封裝並持續改善為可部署產品和消耗性服務，並使用基礎設施做為程式碼 (IaC)，以宣告的方式定義組態。基礎設施自動化透過角色型存取控制 (RBAC) 或屬性型存取控制 (ABAC) 允許存取每個帳戶中的特定服務，來模擬軟體開發週期。部署方法以快速佈建新帳戶，並使用 APIs 或開發自助式功能，使其與您的服務和事件管理功能保持一致。在建立帳戶時自動化網路整合和 IP 配置，以確保合規性和網路安全。使用設定為使用 的原生連接器，將新帳戶與您的 IT 服務管理 (ITSM) 解決方案整合 AWS。視需要更新您的手冊和執行手冊。

提供集中式可觀測性服務

為了實現有效的雲端可觀測性，您的平台應支援本機和集中式日誌資料的即時搜尋和分析。隨著您的操作擴展，平台索引、視覺化和解譯日誌、指標和追蹤的能力，是將原始資料轉換為可行洞見的關鍵。

透過關聯日誌、指標和追蹤，您可以擷取可行的結論，並開發有針對性的明智回應。建立規則，允許主動回應日誌、指標或追蹤中識別的安全事件或模式。當您 AWS 的解決方案擴展時，請確定您的監控策略會串聯擴展，以維護和增強您的可觀測性功能。

實作系統管理和 AMI 控管

使用 Amazon Elastic Compute Cloud (Amazon EC2) 執行個體的組織廣泛需要操作工具來大規模管理執行個體。軟體資產管理、端點偵測和回應、庫存管理、漏洞管理和存取管理是許多組織的基礎功能。 這些功能通常透過安裝在執行個體上的軟體代理程式提供。開發將代理程式和其他自訂組態封裝到 Amazon Machine Image (AMIs) 的功能，並將這些 AMIs 提供給雲端平台的消費者。使用可控管這些 AMIs 使用的預防性和偵測性控制。AMIs 應包含可大規模管理長時間執行之 EC2 執行個體的工具，特別是不定期使用新 AMIs 的可變 Amazon EC2 工作負載。您可以使用 AWS Systems Manager 大規模自動化代理程式升級、收集系統庫存、遠端存取 EC2 執行個體，以及修補作業系統漏洞。

管理登入資料使用

根據 AWS CAF 安全的觀點，實作角色和臨時登入資料。使用工具透過使用預先安裝的代理程式來管理執行個體或內部部署系統的遠端存取，而不儲存秘密。減少對長期憑證的依賴，並掃描 IaC 範本中的硬式編碼憑證。如果您無法使用臨時登入資料，請使用應用程式字符和資料庫密碼等程式設計工具來自動化登入資料輪換和管理。使用 IaC 的最低權限原則來編碼使用者、群組和角色，並使用護欄防止手動建立身分帳戶。

建立安全工具

安全監控工具應支援跨基礎設施、應用程式和工作負載的精細安全監控，並提供彙總檢視以進行模式分析。與所有其他安全管理工具一樣，您應該擴展延伸偵測和回應 (XDR) 工具，以提供功能， AWS 以根據 AWS CAF 安全觀點中定義的要求來評估、偵測、回應和修復 上的應用程式、資源和環境的安全性。

Excel

使用自動化來來源和分發身分建構

Codify 和版本身分建構，例如使用 IaC 工具的角色、政策和範本。使用政策 驗證工具來檢查安全警告、錯誤、一般警告、IAM 政策的建議變更，以及其他問題清單。在適當的情況下，部署和移除以自動化方式提供環境臨時存取權的身分建構，並禁止使用主控台的個人進行部署。

針對跨環境的異常模式新增偵測和提醒

主動評估環境是否有已知漏洞，並新增異常事件和活動模式的偵測。檢閱問題清單並向平台架構團隊提出建議，以取得可進一步提高效率和創新的變更。

分析威脅並建立模型

根據 AWS CAF Security 觀點的要求，針對產業和安全性基準實作持續監控和衡量。當您實作檢測方法時，請判斷哪些類型的事件資料和資訊最適合通知您的安全管理函數。此監控包含數個攻擊向量，包括服務使用量。您的安全基礎應包含跨多帳戶環境進行安全記錄和分析的全方位功能，其中包含關聯來自多個來源事件的能力。使用特定控制項和護欄防止變更此組態。

持續收集、檢閱和精簡許可

記錄身分角色和許可的變更，並在偵測護欄偵測到與您預期組態狀態的偏差時實作警示。使用彙總和模式識別工具來檢閱您的集中式事件集合，並視需要精簡許可。

選取、測量並持續改善您的平台指標

若要啟用成功的平台操作，請建立並定期檢閱完整的指標。確保它們符合組織目標和利益相關者需求。追蹤平台效能和改善指標，並使用團隊啟用和工具採用指標來結合修補程式、備份和合規等操作參數。

使用 CloudWatch 跨帳戶可觀測性進行有效率的指標管理。此服務可簡化資料彙總和視覺化 ，以啟用明智的決策和目標增強功能。使用這些指標作為成功指標和變革驅動因素，以培養持續改進的環境。

資料架構

設計並發展fit-for-purpose的資料和分析架構。

精心設計的資料和分析架構對於獲得可行的洞察至關重要。透過設計和發展fit-for-purpose的資料和分析架構，組織可以降低複雜性、成本和技術負債，同時從不斷增長的資料量中釋放寶貴的洞見。透過符合 AWS CAF 原則，企業可以建立與現有平台無縫整合的資料架構。這種一致性可讓組織利用現代資料處理和分析技術提供的優勢。

資料和分析架構是組織從資料中衍生價值之功能的藍圖。它有助於組織獲得新的業務洞察，並且是業務成長的促進因素。為了支援業務需求，現代資料架構應符合短期和長期業務目標，並符合組織的文化和內容需求。在現今的世界中，資料和分析架構的成功實作和採用，是以在正確的時間為正確的消費者啟用正確資料的原則為基礎。

這是透過規劃和組織組織資料資產在實體上或邏輯上如何建模、如何保護資料，以及這些資料模型如何彼此互動，以解決業務問題並衍生未知模式並產生洞見來實現的。

Start

定義總體功能

在目前的商業環境中，現代資料分析平台必須從資料衍生價值，以支援組織中的各種網域。現代資料架構應包含工具集和模式，這些工具集專為特定使用案例而建置和最佳化，而不是採用單一資料架構方法。架構應該能夠發展並包含基本的建置區塊，例如可擴展的資料湖、專用分析服務、統一的資料存取和統一控管。

組織資料區域

如何組織和存放資料以快速且輕鬆地存取，是資料架構的關鍵層面。這可以透過在資料湖中設定自訂資料區域來實現。資料區域分類如下：

- 從異質來源收集的原始資料
- 整理和轉換資料以支援每個網域的分析需求
- 報告需求的使用案例或產品型資料規格
- 具有安全與合規控制的外部公開資料

規劃資料的敏捷性和民主化

分析平台的有效性取決於佈建資料的速度，以及將佈建的資料普及化以供取用。資料佈建敏捷性是透過資料架構以各種方式取得和處理資料的能力，例如根據使用案例的即時、近乎即時、批次、微型批次或混合。資料民主化是透過定義由資料管理員監控的資料共用和存取控制工作流程來實現的。實作資料市集是讓資料普及的其中一個推動因素。

定義安全資料交付

現代資料架構是外在安全方面的堡壘，但允許員工或資料使用者輕鬆存取其工作職能所定義的 ，並遵守健康保險流通與責任法案 (HIPAA)、個人身分識別資訊 (PII)、一般資料保護法規 (GDPR) 等合規限制。這是透過角色型存取控制 (RBAC) 和標籤型存取控制 (TBAC) 方法來實現的。在 上 AWS，標籤用於控制對資料的存取，以簡化存取控制管理。這樣做符合 AWS CAF 安全觀點中概述的原則。

規劃成本效益

傳統資料倉儲提供緊密耦合的運算和儲存，具有高成本的資源使用率。現代架構會分離運算和儲存體，並根據資料生命週期實作分層儲存體。例如， AWS您可以使用 Amazon Simple Storage Service (Amazon S3) 控制成本，並將資料儲存與運算分離。Amazon S3 儲存體方案專為為不同存取模式提供最低成本的儲存體而打造。此外， AWS 運算工具 （ 例如 Amazon Athena、AWS Glue、Amazon Redshift 和 Amazon SageMaker 執行期) 是無伺服器，因此您不必管理基礎設施，只需為使用量付費。

進階

現代資料架構可以進一步增強，以增加資料用量的廣度，包括支援業務和營運功能的標準分析，以及支援預測和洞見的更複雜功能，並有助於支援更快的決策。為了達成此目的， 架構支援下列各節所述的功能。

了解功能工程

特徵工程使用機器學習，並涉及設定特徵存放區或特徵規格。資料科學團隊會為監督式和非監督式學習模型建立新的功能 （ 衍生屬性 ），並將其存放在功能組合中，以簡化轉換並增強資料準確性。企業可以在多個分析模型中重複使用這些功能，進而加快上市速度。

計劃取消標準化資料集

建構非標準化資料集或資料封送可以大幅簡化商業使用者的資料集，方法是在單一位置隨時提供所需的資料，並提高分析速度。如果精心設計，則一筆記錄可以支援多個用量模型，並減少整體開發生命

週期。非標準化資料集的有效控管也很重要，原因有兩個。實作非標準化資料可能會建立大量備援資料集，這可能會成為大規模管理的挑戰。此外，如果資料集未正確建模，則可能越來越難以重新利用這些資料集。

設計可攜性和可擴展性

大型組織很少在單一資料平台上擁有其所有應用程式和使用者。其應用程式和資料存放區通常會分散在舊式內部部署和雲端平台，讓分析團隊難以混合和合併資料。我們建議您根據網域、地理位置、商業使用案例等特性容器化資料。此容器化可提高各種平台和應用程式之間的可攜性，並支援更有效的取用。將資料分割成容器並透過 APIs公開，可協助您更輕鬆地擴展資料架構。它可啟用混合end-to-end的資料流程，並協助內部部署和雲端型應用程式順暢運作。

Excel

隨著現代分析架構在組織內不斷發展，透過引入可重複使用的概念來管理該變更非常重要。這些概念可提高耐用性和採用率，同時控制成本。以下各節將討論一些要考慮的概念。

設計可設定的架構

組織通常會建立多個複雜的模型，以滿足其獨特的業務需求。這些模型需要建立多個資料管道和工程設計功能。隨著時間的推移，這會產生大量的備援並提高營運成本。建立包含一組參數驅動、可設定的基礎模型的架構，可降低開發時間和營運成本。分析引擎可以實作這些可設定的模型，以提供所需的輸出。

計劃建置統一的分析引擎

業務問題是唯一的，通常需要自訂技術來解決需求，從而在組織中產生多個分析引擎。設計和開發可支援多個程式設計範例的統一 AI 型分析引擎界面，可簡化用量並降低成本。

定義 DataOps

大多數資料專業人員會花費大量時間執行資料操作，例如尋找正確的資料、轉換、建模等。擁有敏捷的資料操作 (DataOps) 可以透過打破資料工程師、資料科學家、資料擁有者和分析師的孤島，大幅增強資料架構。DataOps 可讓團隊之間進行更好的通訊、縮短週期時間，並確保高資料品質。由於業務需求和技術發展不斷變化，資料和分析架構隨著時間經歷了許多轉型。組織必須努力開發、實作和維護資料和分析架構，該架構會隨著時間演進並支援其業務。

資料工程

自動化和協調整個組織的資料流程。

使用中繼資料來自動化處理原始資料的管道，並產生最佳化輸出。利用 AWS CAF 平台架構和平台工程功能以及營運觀點中定義的現有架構防護機制和安全控制。與平台工程支援團隊合作，為簡化管道部署的常見模式開發可重複使用的藍圖。

Start

部署資料湖

為結構化和非結構化資料使用適當的儲存解決方案，以建立基礎資料儲存功能。這可讓您從各種來源收集和存放資料，並讓資料可供進一步處理和分析。資料儲存是資料工程策略的關鍵元件。精心設計的資料儲存架構可讓組織以有效且符合成本效益的方式存放、管理和存取其資料。 AWS 提供各種資料儲存服務，以滿足特定業務需求。

例如，您可以將 Amazon Simple Storage Service (Amazon S3) 用於物件儲存、將 Amazon Relational Database Service (Amazon RDS) 用於關聯式資料庫，以及將 Amazon Redshift 用於資料倉儲，以建立基礎資料儲存功能。這些服務可協助您以安全且符合成本效益的方式存放資料，並可輕鬆存取資料以進行進一步處理和分析。我們建議您也實作資料儲存最佳實務，例如資料分割和壓縮，以改善效能並降低成本。

開發資料擷取模式

若要自動化和協調資料流程，請建立資料擷取程序，以從各種來源收集資料，包括資料庫、檔案和 APIs。您的資料擷取程序應支援業務敏捷性，並考量控管控制。

協調器應該能夠執行以雲端為基礎的服務，並提供自動化排程機制。它應該提供任務之間的條件式連結和相依性選項，以及輪詢和錯誤處理功能。此外，它應該與提醒和監控系統無縫整合，以確保管道順利執行。

一些熱門的協同運作機制包括：

- 時間型協同運作會以遞迴間隔和定義的頻率啟動工作流程。
- 事件型協同運作會根據事件的發生啟動工作流程，例如建立檔案或 API 請求。
- 輪詢會實作一種機制，其中任務或工作流程會呼叫服務 （ 例如，透過 API)，並等待定義的回應，然後再繼續進行下一個步驟。

現代架構設計強調利用可簡化雲端基礎設施管理的受管服務，並減少開發人員和基礎設施團隊的負擔。此方法也適用於資料工程。我們建議您在適用的情況下使用 受管服務來建置資料擷取管道，以加速資料工程程序。這些服務類型的兩個範例是 Amazon Managed Workflows for Apache Airflow (Amazon MWAA) 和 AWS Step Functions：

- Apache Airflow 是編寫、排程和監控工作流程的熱門協同運作工具。 AWS 提供 Amazon Managed Workflows for Apache Airflow (Amazon MWAA) 作為受管服務，可讓開發人員專注於建置，而不是管理協同運作工具的基礎設施。Amazon MWAA 可讓您使用 Python 指令碼輕鬆撰寫工作流程。有向無環圖 (DAG) 代表工作流程做為任務的集合，其方式會顯示每個任務的關係和相依性。您可以擁有任意數量DAGs，Apache Airflow 會根據每個任務的關係和相依性來執行這些 DAG。
- AWS Step Functions 協助開發人員建置低程式碼視覺化工作流程，以自動化 IT 和業務流程。您使用 Step Functions 建置的工作流程稱為狀態機器，而工作流程的每個步驟都稱為狀態。您可以使用 Step Functions 建立工作流程，以進行內建錯誤處理、參數傳遞、建議的安全性設定和狀態管理。這些可減少您必須撰寫和維護的程式碼數量。任務會透過與另一個 AWS 服務或您在內部部署或雲端環境中託管的應用程式進行協調來執行工作。

加速資料處理

資料處理是了解現代組織所收集的大量資料的重要步驟。若要開始使用資料處理， AWS 提供 等受管服務AWS Glue，可提供強大的擷取、轉換和載入 (ETL) 功能。組織可以使用這些服務開始處理和轉換原始資料，包括清理、標準化和彙總資料，以準備進行分析。

資料處理從簡單的技術開始，例如彙總和篩選，以執行初始資料轉換。隨著資料處理需求的演進，您可以實作更進階的 ETL 程序，讓您從各種來源擷取資料、轉換資料以符合您的特定需求，並將其載入集中式資料倉儲或資料庫以進行統一分析。此方法可確保資料準確、完整，並可及時進行分析。

透過使用 AWS 受管服務進行資料處理，組織可以從更高層級的自動化、可擴展性和成本效益中受益。這些服務可自動化許多例行資料處理任務，例如結構描述探索、資料分析和資料轉換，並為更具策略性的活動釋放寶貴的資源。此外，這些服務會自動擴展，以支援不斷增長的資料磁碟區。

提供資料視覺化服務

尋找方法，讓使用資料視覺化以有意義的方式快速解譯資料的決策者可以使用資料。透過視覺化，您可以解譯模式並提高各種利益相關者的參與度，無論他們的技術技能如何。良好的平台可讓資料工程團隊佈建資源，以快速且低成本的方式提供資料視覺化。您也可以使用工具輕鬆查詢資料存放區，而不需要工程專業知識，以提供自助服務功能。請考慮使用內建工具，透過資料視覺效果和互動式儀表板提供無伺服器商業智慧，並使用自然語言查詢後端資料。

進階

實作近乎即時的資料處理

資料處理是任何資料工程管道的重要元件，可讓組織將原始資料轉換為有意義的洞見。除了傳統的批次處理之外，即時資料處理在現今快節奏的商業環境中變得越來越重要。即時資料處理可讓組織在事件發生時回應事件，並改善決策和營運效率。

驗證資料品質

資料品質會直接影響衍生自資料的洞見和決策的準確性和可靠性。實作資料驗證和清理程序對於確保您使用高品質且值得信賴的資料進行分析至關重要。

資料驗證涉及透過比對預先定義的規則和條件來驗證資料的準確性、完整性和一致性。這有助於識別資料中的任何差異或錯誤，並確保其符合用途。資料清理涉及識別和更正資料中的任何不準確、不一致或重複。

透過實作資料品質程序和工具，組織可以改善從資料衍生之洞見的準確性和可靠性，進而獲得更好的決策和營運效率。這不僅可以增強組織的效能，還可以提高利益相關者對產生的資料和分析的信心和信任。

證明資料轉換服務

資料轉換會準備進階分析和機器學習模型的資料。它涉及使用資料正規化、擴充和重複資料刪除等技術，以確保資料乾淨、一致且已準備好進行分析。

- 資料標準化涉及將資料組織成標準格式、消除冗餘，以及確保資料在不同來源之間保持一致。這可讓您更輕鬆地分析和比較來自多個來源的資料，並讓組織更全面地了解其操作。
- 資料擴充涉及使用來自外部來源的其他資訊增強現有資料，例如人口統計資料或市場趨勢。這可提供有關客戶行為或產業趨勢的寶貴洞見，這些趨勢可能不單從內部資料來源看出。
- 重複資料刪除涉及識別和移除重複的資料項目，並確保資料準確且無錯誤。這在處理大型資料集時特別重要，其中即使只有一小部分的重複項目可能會扭曲分析結果。

透過使用進階資料轉換技術，組織可以確保其資料具有高品質、準確且已準備好進行更複雜的分析。這有助於做出更好的決策、提高營運效率，並在市場中獲得競爭優勢。

啟用資料民主化

透過讓資料可供所有員工存取、理解和使用，促進資料普及文化。資料普及化可協助員工做出資料驅動型決策，並有助於組織的資料驅動型文化。這表示打破孤島並建立一種文化，讓所有員工共用和使用資料來推動決策。

整體而言，資料民主化是關於建立一種文化，讓組織中的每個人都能重視、存取和理解資料。透過啟用資料民主化，組織可培養資料驅動型文化，以推動創新、改善決策，並最終帶來業務成功。

Excel

提供以 UI 為基礎的協同運作

若要建置敏捷且使用有效方法的組織，請務必規劃現代協同運作平台，供跨業務單位的開發和營運資源使用。目標是開發、部署和共用資料管道和工作流程，而不依賴單一團隊、技術或支援模型。這可透過 UI 型協同運作等功能實現。drag-and-drop互動等功能，可讓技術知識不多的使用者建構 DAGs 和狀態機器資料流程。然後，這些元件可以產生可協調資料管道的可執行程式碼。

DataOps 有助於克服資料管理的複雜性，並確保跨組織的無縫資料流程。中繼資料驅動型方法可確保資料品質和合規符合組織的要求。投資微服務、容器化和無伺服器函數等工具集可改善可擴展性和敏捷性。

仰賴資料工程團隊從資料中產生價值，並將day-to-day基礎設施任務留給自動化，讓組織在自動化和協同運作方面取得卓越成果。資料流程管理任務近乎即時的監控和記錄支援立即的修補動作，並改善資料流程管道的效能和安全性。這些原則有助於實現可擴展性和效能，同時確保安全的資料共用模型，並使組織在未來取得成功。

整合 DataOps

DataOps 是一種現代化的資料工程方法，強調開發和操作程序的整合，以簡化資料管道的建立、測試和部署。為了實作 DataOps 最佳實務，組織會使用基礎設施做為程式碼 (IaC) 和持續整合和持續交付 (CI/CD) 工具。這些工具支援自動化管道建立、測試和部署，可大幅提升效率並減少錯誤。DataOps 團隊會與平台工程支援團隊合作來建置這些自動化，因此每個團隊都可以專注於他們最擅長的工作。

實作 DataOps 方法有助於促進資料工程師、資料科學家和商業使用者的協作環境，並快速開發、部署和監控資料管道和分析解決方案。這種方法提供跨團隊更順暢的溝通和協作，進而加快創新速度並取得更好的成果。

若要充分利用 DataOps 的優勢，請務必簡化資料工程程序。這是透過使用平台工程團隊的最佳實務來實現的，包括程式碼檢閱、持續整合和自動化測試。透過實作這些實務，組織可以確保資料管道可靠、可擴展且安全，並且同時符合業務和技術利益相關者的需求。

佈建和協同運作

建立、管理和分發核准的雲端產品目錄給使用者。

隨著組織的成長，以一致、可擴展且可重複的方式佈建基礎設施變得更具挑戰性。簡化佈建和協同運作可協助您實現一致的控管，並滿足您的合規要求，同時允許使用者僅部署核准的雲端產品。

在組織中重複使用預先核准的產品，可讓開發人員更快速且更一致地建置應用程式，同時滿足組織的安全和管理需求。

Start

部署hub-and-spoke目錄模型

在服務目錄中管理為產品組合的軟體資產，會以hub-and-spoke模式與一或多個帳戶中的使用者共用。您可以使用私有市集和私有優惠來策劃各種第三方解決方案，並將其與您的基礎設施一起分發為程式碼 (IaC) 範本。

若要讓您的建置器能夠使用預先核准的產品，請定義程序來檢閱、核准這些產品並將其發佈給您的使用者。首先，設計和實作包含這些預先核准產品的集中受管儲存庫。當您組織中的使用者需要取用每個產品時，設計一個系統來授予此儲存庫中授權和產品的存取權。

允許組織中的建置者提交產品以核准發佈機制，因此這些產品在核准後可供組織中的所有使用者使用。

整理範本以重複使用

當您為您的解決方案編纂 IaC 範本並定義hub-and-spoke模型時，您應該為每個輻條帳戶定義兩類範本：佈建/強制執行且可供取用。佈建/強制執行的範本會直接從管理帳戶佈建到每個成員帳戶，做為基礎功能。可供建置者以自助方式瀏覽和佈建的 範本。

套用預設參數以重複使用

實作 IaC 範本，其中包含建置器可以預先選取的預設參數。這可讓建置器符合控管，而無需評估每個參數的詳細資訊，並防止他們做出不正確的選擇。此方法只會公開設定所需的內容。例如， AWS Service Catalog 實作此方法時具有限制功能，可控制套用至特定產品組合中產品的規則。當建置器團隊使用 範本的自助式佈建時，會預先設定此自訂。

建立核准程序

如果使用者有使用該產品的業務理由，他們應該能夠提交請求來存取他們未核准的產品。建置通知系統，在使用者使用的產品有更新可用時通知他們，讓他們可以遵守最新的安全性更新。

為建置器建立工作流程，透過自助式入口網站提交新產品以供檢閱。建置器可以使用 入口網站來定義產品的受眾，並識別應可存取產品的使用者群組。對於每個提交，請使用您定義的程序來檢閱、核准產品，並將產品發佈到自助式入口網站。

進階

建立自助式入口網站

建立自助式入口網站以分發、瀏覽和使用核准的雲端產品。組織中的使用者可以使用此入口網站來搜尋他們建置基礎設施所需的產品，並將應用程式部署至其環境。為有權存取入口網站中產品的使用者建立許可界限，並設定使用者可使用授權產品的次數限制。定義一組基本資源，這些資源可以直接佈建或做為每個口語帳戶中的自助式模型提供，因為帳戶是透過使用自訂等解決方案建立的 AWS Control Tower。

啟用私有市集

私有市集提供購買產品（軟體、資料和專業服務）的精選目錄，並以hub-and-spoke模式（具有一個管理帳戶和多個成員帳戶）實作，以便輻條帳戶只能訂閱核准的軟體。此產品控管有助於控制軟體成本，並簡化法律和合約審查。在管理帳戶層級建立私有市集，以做為主要中樞。

管理權利

啟用控制項，僅允許授權的使用者和工作負載在廠商定義的限制內使用授權。這有助於降低昂貴稽核和意外授權調校的風險。

Excel

與採購系統整合

透過將現有採購程序整合到 來補充這些程序AWS Marketplace。這可透過將您的採購系統 (Coupa 或 SAPriba) 擴展到私有市場來完成，以便您的使用者可以遵循現有的採購和核准程序來取得軟體。建立適當的 IAM 受管許可、使用 AWS Marketplace 產生必要的資訊來設定您的採購解決方案，以及設定您

的採購解決方案以完成整合。例如，您可以設定打孔、將採購訂單連接至 AWS 發票，然後調整您的採購程序以使用標準佈建解決方案。

讓您的建置器能夠透過內部 API 存取預先核准的產品，讓使用者可以將產品納入其應用程式，或建置自己的個人化入口網站，讓團隊使用產品。整合用於建立新產品的提交和發佈程序，並允許使用者透過APIs請求新的授權和產品存取權。

與您的 ITSM 工具整合

如果適用，請使用 IT 服務管理 (ITSM) 工具連線，並自動化對組態管理資料庫 (CMDB) 的任何更新。建立程序和機制來評估組織使用的產品。建立機制，通知使用者需要更新以取得合規的預先核准產品。使用您的 ITSM 工具來分析您的環境，並在需要重大更新時，將安全性和合規更新推送至整個組織的產品。

實作生命週期管理和版本分佈系統

在整個開發生命週期中維護 IaC 範本的版本，以及從範本佈建的服務版本。您可以使用針對目錄實作的hub-and-spoke模型來定義是否需要在輻條層級強制更新（例如，如果同時版本可用於自助式佈建），以及哪些版本需要標記為過時。使用hub-and-spoke目錄也有助於根據需要管理新版本的稽核和分發。

現代應用程式開發

建置架構良好的雲端原生應用程式。

現代應用程式開發實務對於組織建置架構良好的雲端原生應用程式並保持競爭至關重要。企業可以使用容器和無伺服器運算等雲端原生技術來建立可擴展且靈活的應用程式，以適應不斷變化的市場需求。這些技術可讓組織最佳化資源使用率、降低成本，並改善其應用程式的效能。

當您設計現代應用程式時，請為操作和開發開發開發靈活的解決方案。現代應用程式會自動回應客戶需求的變化，並對故障具有彈性。工程師可以快速開發和部署變更，並監控應用程式效能。現代應用程式旨在自我修復，並且能夠視需要擴展至大型或小型流量，包括無成本的流量。

建置架構良好的雲端原生應用程式需要深入了解基礎技術及其最佳實務。組織應採用微服務架構，並將其應用程式設計為模組化且鬆散耦合，允許獨立部署和可擴展性。這種方法可讓組織將其應用程式分解為更小型、更易於管理的元件，這些元件可快速獨立開發、測試和部署。

Start

探索現代方法

首先，請調查容器、無伺服器技術，以及其他可開發微服務的方法，以增強資源效率、協助改善安全性，並將基礎設施費用降至最低。選擇修改現有的差異化和企業應用程式，以提高效率並最大化現有投資的價值。考慮轉換（將您的自我管理容器、資料庫或訊息代理程式轉換為受管雲端服務），並根據價值導向的決策重構（重新開發您的應用程式以採用雲端原生架構）。

當您更新現有的雲端型應用程式時，成功的方法涉及使用 strangler fig 模式，逐步將您的架構分解為微服務。此程序有助於採用現代應用程式方法，因此您可以實現固有的優勢，並向大型組織展示其價值。考慮將您的應用程式建構為不同的微服務，在適用的情況下利用事件驅動的架構。確保您的架構將不可變更的服務配額和實體資源納入考量，以避免影響工作負載效能或可靠性。

採用雲端原生運算功能

雲端原生運算功能是現代應用程式開發的關鍵。這種方法需要組織考慮如何託管其運算單位，並識別每個使用案例或服務的最佳選項。例如， AWS Lambda提供無伺服器機制來執行您的應用程式程式碼，並在事件驅動的架構中扮演關鍵角色。Lambda 函數會隨需啟動，並平行執行至定義的並行上限，因此可以擴展以執行各種任務。

使用容器化

在現代軟體開發中，管理應用程式及其相依性已成為越來越複雜的任務，尤其是當您認為需要維持各種環境的一致性時。為了解決這些挑戰，Docker 等容器化技術已成為封裝應用程式及其相依性的有效解決方案。無論您應用程式的執行時間環境為何，容器都能確保一致且可重現的部署，因此在本機環境中的開發與雲端環境中的生產開發運作方式相同。此方法可減少因環境或其組態不相符而造成的錯誤。

使用現代資料庫

當您使用現代資料庫時，應用程式內的每個微服務都可以使用符合其需求的正確專用資料庫，這可提高敏捷性和效能，同時降低成本。例如，一個微服務可能會在儲存工作階段資料時使用 NoSQL 資料庫來達到高輸送量，另一個微服務可能會使用關聯式資料庫來執行複雜的資料表聯結，而另一個微服務可能會使用量子分類帳資料庫來追蹤區塊鏈的變更。

現代資料庫提供可擴展性和彈性。它們也有助於提供比傳統資料庫更好的安全性、合規性和可靠性。它們可讓組織更有效率地存放和管理資料，並確保應用程式可以在正確的時間存取正確的資料，進而獲得更好的效能和使用者體驗。

遷移至現代資料庫是現代應用程式開發的關鍵元件。透過使用正確的資料儲存解決方案，組織可以最佳化其資料管理功能，並提供更有效率且可靠的應用程式。透過讓每個微服務獨立，並為每個微服務選擇正確的技術，組織可以進一步最佳化其資料功能，以實現最高效率和可擴展性，同時最大限度地降低成本。

進階

最佳化您的現代架構

若要進一步最佳化，請精簡無伺服器技術的實作，並開發可以使用 Amazon API Gateway 和 等 AWS 服務獨立擴展和部署的架構AWS Lambda。使用 Amazon Route 53 和 實作服務探索AWS Cloud Map，以確保元件之間的無縫通訊。

採用 API 版本控制、快取和速率限制，以維持不同應用程式版本的相容性和效能。使用 AWS Identity and Access Management (IAM) 和資源政策增強安全性。這些有助於確保您的基礎設施受到保護，並且只會將存取權授予授權的實體。

如果可能，請使用無伺服器服務來執行容器，而不必管理基礎基礎設施。這可讓您專注於開發核心應用程式，並實現更好的資源管理和效能。它還可協助您充分利用可擴展性、靈活性和成本效益的優勢。

透過深入了解無伺服器架構的複雜性並整合這些進階實務，組織可以發現改進和微調的機會，最終最大限度地發揮雲端原生應用程式的潛力。此追求有助於採用更複雜的應用程式模式，進一步提升整體使用者體驗。它還讓組織在其軟體開發過程中變得更加靈活和有效率。

使用服務網格技術

隨著組織越來越採用微服務架構來建置和部署應用程式，管理這些服務之間的複雜性、安全性和通訊變得至關重要。Istio、Linkerd 或 Consul 等服務網格技術在協助增強微服務的安全性、可觀測性和可靠性方面扮演關鍵角色。

確保可見性和可追蹤性

現代實務在開發過程中提供更高的可見性和可追蹤性，並可讓您更輕鬆地遵守產業標準和最佳實務。可見性和監控對於現代應用程式開發至關重要。實作監控和記錄解決方案，以提供對應用程式效能的寶貴洞見，可讓組織識別需要改進的領域並最佳化其應用程式。我們建議您與平台工程團隊合作，確保工具可用於提供end-to-end可見性和監控，以便您可以快速偵測、診斷和解決問題。

Excel

接受微服務

對於許多組織而言，現代應用程式開發是業務成功的同義詞。微服務是此轉型的核心，組織可以從採用這些強大的架構模式中受益。

微服務提供高度可擴展性、彈性和敏捷的應用程式架構。透過將應用程式分成小型、可獨立部署的服務，組織可以選擇在特定元件上快速迭代，而不會影響應用程式的其他部分。進階彈性模式，例如斷路器和隔板，在確保這些應用程式的高可用性方面扮演重要角色。

斷路器可做為安全機制，透過暫時停止或從運作狀態不佳的服務轉移通訊來防止串聯失敗，以便復原。大量隔離資源並限制潛在故障的影響範圍。這些模式共同建立強大的架構，可承受不可預見的中斷並維持最佳效能。

實作微服務的另一個關鍵層面是採用網域驅動型設計 (DDD) 原則。DDD 專注於建立對商業網域的共同理解，並將其轉換為結構良好的軟體設計。此方法可帶來更具凝聚力且可維護的微服務，並確保應用程式隨著組織需求逐步演進。

在微服務型應用程式中，最佳化服務間通訊也很重要。透過實作 gRPC 或 GraphQL 等進階通訊協定，組織可以大幅提升服務之間的通訊效率。這些通訊協定提供類型安全、低延遲和彈性等功能，有助於改善應用程式的整體效能和可維護性。

採用微服務的組織提供一個促進創新、敏捷性和協作的環境。開發團隊通常會根據業務能力進行組織，並高度專注於持續整合和持續交付 (CI/CD) 實務。他們有權快速做出決策、實驗和反覆運算，並且接受共同責任和責任的文化。

持續整合和持續交付

比使用傳統軟體開發和基礎設施管理程序的組織更快地發展和改善應用程式和服務。

採用具有持續整合和持續交付 (CI/CD) 的 DevOps 實務，可提升建置、測試和部署應用程式的簡化、自動化和高效率程序。CI/CD 可讓您快速交付軟體、降低部署錯誤的風險，並確保應用程式隨時掌握最新功能和錯誤修正。主要目標是透過使用傳統軟體開發和基礎設施管理程序，以更快的速度發展和改善應用程式和服務。

Start

採用軟體元件管理

軟體元件管理是管理用於建置軟體的所有個別元件的實務，包括程式庫、架構、原始程式碼儲存庫、模組、成品和第三方相依性。我們建議您使用 Git 或 Apache Subversion 等版本控制系統來管理原始碼、啟用協同合作，以及維護程式碼變更的歷史記錄。您可以監控儲存庫中的變更和事件，以自動化程序、建立管道、管理您的程式碼，並視需要將工作流程與其他 服務整合。

建立 CI/CD 管道

CI/CD 管道是由遞交至版本控制系統的變更所啟動的一組自動化指示。它們通常包含建置應用程式、執行自動化測試，以及將程式碼部署到特定環境的指示。您可以使用 AWS CodePipeline、Jenkins、GitLab 或 CircleCI 等工具來設定自動化 CI/CD 管道。您也可以直接在支援管道產生版本的控制系統中設定它們。

從用於持續整合的最低可行管道開始，然後轉換到包含更多動作和階段的持續交付管道。將您的持續交付組態視為程式碼。您可以為每個分支和團隊使用多個不同的管道，因此請考慮您需要設定哪些組態變數，以及如何最好地支援將使用管道的團隊。

考慮部署時段 – 您要部署程式碼的日期和時間。請考慮您系統的低需求時數，因此如果您必須轉返，它將對您的客戶影響最小。其他最佳實務包括避免在星期五進行部署，以及在高尖峰日期或假日之前實作程式碼凍結。當遞交的作者無法使用時（例如假期），請考慮定義有關部署程式碼的規則。請記住，部署失敗，您可能需要依賴外部協助。評估不同的部署方法，例如就地部署、滾動部署、不可變部署和藍/綠部署。請考慮將全受管服務用於持續交付工作流程，以提高可用性和安全性，同時將複雜性和管理降至最低。

部署自動化測試

現代實務建議向左轉移（將測試移至更靠近開發人員和 IDE，以及更早的生命週期），以在問題遞交至儲存庫並啟動管道之前偵測和修復問題。此實務涉及開發人員的快速意見回饋迴圈，因為在開發人員進行編碼時偵測到錯誤。向左轉移與降低成本相關聯，因為測試不需要執行管道，這可能會導致非同步意見回饋和更高的營運費用。

自動化測試會在開發程序初期發現錯誤，並包含單元測試、整合測試和功能測試。建議您鼓勵開發人員儘早使用工具來建立單元測試，並在將程式碼推送至中央儲存庫之前執行它們。此外，請確定您的自動化程序包含靜態程式碼分析、效能基準測試和安全應用程式測試。

建立文件

除了實作 CI/CD 管道以簡化開發工作流程之外，您應該維護清晰且全面的文件，以確保管道的持續有效性、可維護性和可擴展性。文件是 CI/CD 管道的重要層面，因為它可讓開發團隊清楚了解管道的設計、元件和程序。當您建立文件時，請從管道概觀開始、解釋架構和設計權衡、描述正在使用的工具和技術、指定初始組態和設定、概述安全措施和存取控制，以及包含疑難排解和維護資訊。

使用基礎設施做為程式碼

使用 Terraform、Ansible 或 等工具AWS CloudFormation來管理基礎設施，並確保一致且可重現的環境。將您的基礎設施視為程式碼，確保您追蹤基礎設施的變更，並避免直接在主控台中進行變更。定義所有基礎設施，包括資料庫佈建，即程式碼，並使用管道部署這些變更。考慮在具有一小部分已淨化生產資料的管道中，以程式碼形式執行資料庫整合。如果可能，請進行變更，並在程式碼中追蹤這些變更。

如同軟體程式碼，請遵循基礎設施程式碼的這些最佳實務：

- 使用版本控制。
- 利用錯誤追蹤和票證系統。
- 讓對等檢閱變更後再套用。
- 建立基礎設施程式碼模式和設計。
- 測試基礎設施變更。

保留和追蹤標準指標

為了維持高水準的效能，請針對關鍵指標進行開發和追蹤，以了解管道的運作狀態和業務影響，包括：

- 組建頻率。建置數量可讓您深入了解團隊的生產力和變更的複雜性。
- 部署頻率。定期部署表示運作狀態良好、敏捷的開發程序。
- 變更的前置時間。測量變更達到生產的平均時間，可協助您識別部署過程中的瓶頸。
- 管道的平均時間。從初始管道階段到每個後續階段的平均時間，有助於最佳化您的工作流程。
- 生產變更量。追蹤到達生產環境的變更數量，可讓您深入了解生產環境的穩定性。
- 建置時間。平均建置時間可能表示程式碼庫或基礎設施中的潛在問題。

進階

使用組態管理

組態管理工具在自動化部署、組態和管理軟體和基礎設施方面扮演重要角色。它們提供一種系統性方法來處理變更，並維護各種環境中基礎設施、軟體和組態的所需狀態。這些工具可讓開發人員使用宣告性或命令性語言來定義系統所需的狀態。然後，組態管理工具會自動將這些組態套用至目標系統的程序，以確保一致性和可重複性。

使用組態管理工具來自動化軟體和基礎設施的部署、組態和管理。AWS Systems Manager State Manager 是一種安全且可擴展的組態管理服務，可將受管節點和其他 AWS 資源維持在您定義的狀態的程序自動化。

整合監控和記錄

將監控和記錄解決方案整合到 CD 管道，可為開發團隊和整體軟體開發程序帶來許多好處。這些解決方案可以提供應用程式效能的即時洞見、更快速地識別和解決問題，並促進持續改進，以協助確保應用程式在整個生命週期中保持可靠性、效能和可擴展性。投資監控和記錄解決方案是維護強大且有效率的 CD 管道的關鍵層面，最終有助於成功交付高品質的軟體。

建立合併的節奏

每天至少對主線（幹線或主）分支遞交或合併程式碼變更一次，或最好在每個任務之後每天多次。此節奏會導致多個每日管道調用。提取式分支工作流程模型符合此方法。使用特徵標記、暗啟動和類似技術來自訂客戶使用的功能。

擷取部署後行為

部署之後，請使用自動化合成測試擷取生產行為，並將結果與持續交付管道同步，以確保立即執行修正動作。開發人員的首要任務應該是盡快修正管道中發現的錯誤、將程式碼變更遞交至原始程式碼儲存庫，以及驗證管道中的錯誤解析。

部署後最佳實務包括觀察最重要的關鍵績效指標 (KPIs)，並驗證生產環境中沒有錯誤。自動化錯誤處理和評估部署後 KPIs，以量化版本的影響。自動產生開發人員可用來進行改善的速度、安全性和穩定性指標。如需詳細資訊，請參閱 上的 DevOps 監控儀表板解決方案 AWS。

Excel

採用尖端實務和技術以獲得最佳效能。持續精簡 CI/CD 程序可協助您改善軟體品質、縮短上市時間，並提高敏捷性。新的技術和工具會持續出現，這使得您的組織必須隨時掌握資訊並適應以維持競爭優勢。

若要保持適應性，請考慮下列事項：

- 將一切定義為程式碼，包括您的應用程式、組態、基礎設施、資料、 AWS 帳戶和組織、部署管道、聯網，以及安全與合規控制。
- 為運算映像、共用服務和應用程式建立對應的部署管道。
- 考慮 GitOps 模型，其中提取型請求會透過比較現有基礎設施狀態與所需狀態來啟動工作流程以部署變更，如程式碼中所述。
- 考慮使用 CD 管道來部署機器學習 (ML)、資料、物聯網 (IoT) 和其他工作負載。
- 以數位方式簽署所有建置成品，並將其存放在安全儲存庫中。
- 透過自動產生軟體物料清單來追蹤軟體來源，該清單會建立部署給客戶的所有版本化和數位簽章成品的記錄。
- 在您消除軟體交付程序中的所有手動活動之後，請移除手動檢閱板。

對於已自動化整個軟體交付程序的應用程式和服務，請考慮持續部署，其中團隊會部署變更，將管道中的所有檢查傳遞給生產環境中的客戶。如需視覺效果，請參閱 網站上的什麼是持續交付？的第一個 AWS 圖表。

整合 AI/ML 技術

將人工智慧 (AI) 和機器學習 (ML) 技術整合至 CI/CD 管道提供數種優點，包括下列項目：

- 自動化測試產生
- 智慧測試優先順序
- 偵測問題的預測分析
- 異常偵測和根本原因分析
- 程式碼檢閱和品質保證

- 部署最佳化

如需詳細資訊，請參閱 AWS 網站上的新增智慧到您的開發人員操作。

採用混沌工程實務

混沌工程涉及刻意將故障注入系統，以測試其承受意外事件並從中復原的能力。透過識別弱點並主動解決它們，組織可以改善其整體系統可靠性，並將潛在問題的影響降至最低。

採用混沌工程實務，使用 Gremlin、Chaos Monkey 或 Litmus 等工具來測試系統的彈性。定期執行受控實驗，以識別漏洞、驗證容錯能力，並確保您的應用程式正常處理非預期的故障。這種主動方法有助於提高系統可靠性，並有助於更強大的 CI/CD 管道。

最佳化效能

使用分析工具、即時監控和回饋迴圈，持續最佳化應用程式的效能。套用下列技術，以確保您的應用程式可以處理增加的流量和需求：

- 程式碼最佳化
- 分析
- 即時監控
- 回饋迴圈
- 快取
- 負載平衡
- 可擴展性和效能測試

實作進階可觀測性

提升雲端基礎設施的可觀測性，不僅止於收集、彙總和分析指標、日誌和追蹤的基本概念。當使用 Amazon CloudWatch 和 等工具增強可觀測性時AWS X-Ray，它會發展為策略實務，以推動持續交付和創新。

在強大的 CI/CD 管道中，進階可觀測性可讓您發現洞見，不只是關於應用程式和基礎設施，也關於整個系統的效能和運作狀態，包括管道本身。這些洞見可協助您：

- 快速識別、了解和解決潛在問題，以改善應用程式穩定性並減少停機時間
- 簡化 CI/CD 程序，以建立更快且更可靠的交付

- 深入了解程式碼變更和部署的影響，以推動明智的決策
- 最佳化資源使用率，以提高營運效率和成本效益

若要提升可觀測性：

- 將可觀測性嵌入應用程式和基礎設施的每一層，以建立系統效能、行為和運作狀態的完整檢視。
- 使用 Amazon CloudWatch 等工具集中資料收集、儲存和分析，以統一您的可觀測性資料，以便於存取和解譯。
- 使用 AWS X-Ray 進行分散式追蹤，以了解應用程式及其基礎服務的效能。
- 建立回饋迴圈以持續改進，並使用可觀測性資料來推動系統的反覆增強。

採用進階可觀測性不只是維護您的系統，而是實現卓越營運並推動組織中持續創新的策略發展。

實作 GitOps 實務

實作 GitOps 實務，透過使用 Git 儲存庫作為單一事實來源來管理基礎設施和應用程式組態。此方法可簡化變更管理、增強可追蹤性，並確保跨環境的一致性。

結論

本指南可做為成功實作和管理雲端採用基礎的手冊。它討論了如何：

- 直接解決平台架構中的技術挑戰和複雜性，為您的雲端環境和其中的資料建立健全的指導方針和原則。
- 使用強大的佈建和協同運作來建置平台工程。
- 啟用合規的多帳戶雲端環境，以可擴展且可重複的方式管理和分發核准的雲端產品給使用者。
- 使用資料工程所需的工具來支援資料架構決策，以推動資料驅動型決策。
- 將這些功能與現代應用程式開發策略和 CI/CD 程序配對，以提高組織中的靈活性、效率和創新。
- 在您自己的決策中建立跨職能關係，並從其他 AWS CAF 觀點取得意見，以確保平台及其背後團隊的成功。

深入閱讀

AWS 雲端採用架構 (AWS CAF) 資源：

- eBook
- 電子書
- 資訊圖表
- AWS CAF 適用於人工智慧、Machine Learning和生成式 AI
- 業務觀點
- 人員觀點
- 控管觀點
- 操作觀點
- 安全性觀點

其他資源：

- AWS 架構中心
- AWS 案例研究
- AWS 一般參考
- AWS 詞彙表
- AWS 知識中心
- AWS 方案指引
- AWS 合作夥伴解決方案 （先前稱為 Quick Starts)
- AWS 安全文件
- AWS 解決方案程式庫
- AWS 訓練和認證
- AWS Well-Architected
- AWS 白皮書和指南
- 入門 AWS
- Amazon Web Services 概觀

貢獻者

本指南的貢獻者包括：

- Tony Santiago，資深合作夥伴解決方案架構師 AWS
- Matias Undurraga，企業技術專家， AWS
- Alex Torres，資深解決方案架構師 AWS
- Michael Rhyndress，資深 DevSecOps 顧問， AWS
- Alex Livingstone，首席解決方案架構師和 CloudOps 專家， AWS
- Bruce Cooper，首席 SDE， AWS
- Ravinder Thota，資深諮詢顧問， AWS
- Sausan Yazji，資深實務經理 AWS
- Paul Duvall，DevSecOps 總監 AWS
- Jeremy Tennant，首席雲端交付經理， AWS
- Sneh Shah，首席基礎設施負責人， AWS
- Sasa Baskarada， AWS 全球雲端採用架構主管， AWS

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
初次出版	—	2023 年 10 月 25 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：將您的現場部署 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

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。

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

經常用於經常性問題的解決方案，其中解決方案具有反生產力、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、低延遲網路連線能力。

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 將指導方針組織到六個重點領域：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。有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) 執行試驗，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。

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

部落格文章中的 Stephen Orban 定義了這些階段： AWS 雲端 企業策略部落格上的邁向雲端優先之旅和採用階段。如需有關它們如何與 AWS 遷移策略相關的詳細資訊，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)

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的 AI 欄位。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變得不合規，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

在整個生命週期中追蹤資料的原始伺服器和歷史記錄的程序，例如資料的產生、傳輸和儲存方式。

資料主體

正在收集和處理其資料的個人。

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

將人類可讀取的純文字資料轉換為加密文字的運算程序。

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 () 文件中的信封加密。 AWS Key Management Service AWS KMS

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邊界會限制故障的影響，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給 LLM。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。對於需要特定格式、推理或網域知識的任務，少量的提示非常有效。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，已在廣義和未標記資料的大量資料集上進行訓練。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。

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。

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)

在遇到挑戰或災難時，工作負載能夠在不介入的情況下持續運作。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，以及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。

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

將基礎設施視為程式碼。

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的部署最佳實務。

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

由 Klaus Schwab 於 2016 年推出的術語，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 （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 個 R。

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。

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

除了屬於組織一部分的管理帳戶 AWS 帳戶 之外的所有 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)

一種 AWS 計畫，提供諮詢支援、訓練和服務，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 中的操作準備度審查 (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，OT 和資訊技術 (IT) 系統的整合是工業 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)

當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 Rs。

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)、已諮詢 (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 Rs。

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 。

依設計的安全性

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

由接收資料的 AWS 服務 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。

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。

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，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

危及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，通常可接受中等速度的查詢。

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瑕疵或漏洞。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。