加速可觀測性結果

AWS 方案指引

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

AWS 方案指引: 加速可觀測性結果

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon 的商標和商業外觀不得用於任何非 Amazon 的產品或服務，也不能以任何可能造成客戶混淆、任何貶低或使 Amazon 名譽受損的方式使用 Amazon 的商標和商業外觀。所有其他非 Amazon 擁有的商標均為其各自擁有者的財產，這些擁有者可能附屬於 Amazon，或與 Amazon 有合作關係，亦或受到 Amazon 贊助。

Table of Contents

簡介 1
目標對象 1
目標 1
概觀 2
為什麼您需要重新思考可觀測性策略 2
可觀測性工具和架構 2
階段 1：定義您的北極星 4
在開發生命週期的早期整合可觀測性 （向左轉移方法） 4
設定有效的組織和團隊結構 5
追蹤成本分配 6
定義標準 7
建立呈報程序 7
透過訓練提升技能 7
階段 2：實作可觀測性 8
選擇您的可觀測性平台 8
功能集 8
授權和部署模型 9
定價維度 9
團隊技能 7
操作和維護 9
檢測您的應用程式 9
收集遙測 10
實作可觀測性元件 10
驗證可觀測性系統 10
階段 3：檢查、調整和重複 11
實作定期檢閱 11
慶祝勝利 12
從事件中學習 12
後續步驟和資源 13
資源 13
文件歷史紀錄 14
..... xv

加速可觀測性結果

Jyothi Madanlal、Abhaya Chauhan 和 Jose Augusto Ferronato，Amazon Web Services

2025 年 7 月 (文件歷史記錄)

Amazon 副總裁兼技術長 Werner Vogels 博士曾說過：「一切都失敗，一直...所以規劃失敗，什麼都不會失敗」(AWS re：Invent 2024 年主題演講)。採用這種思維是有效且強大的可觀測性實作的核心。

此策略文件會探索如何提升可觀測性狀態，以建立更具彈性且更有效率的系統。它還討論如何定義錨點目標 (北極星)，以在整個組織中提供有效的一致性，從而提高客戶信任、品牌評價和團隊士氣。文件重點介紹要考慮的關鍵事項，以及加速可觀測性結果的建議方法。此方法包括釋放新的商機、提供更好的使用者體驗、最佳化效能和成本，以及改善整個組織的營運效率。無論您可觀測性實作的成熟度為何，本文件都會在追求可觀測性目標時，提供增量且可行的洞見。

目標對象

此資訊適用於下列對象：

- 負責定義可觀測性策略的技術領導者和架構師
- 負責監督卓越營運工程經理和 DevOps 領導者
- 負責建立可觀測性基礎的平台工程團隊
- 負責實作可觀測性實務的站點可靠性工程師 (SREs)

目標

本指南中的資訊可協助您提升可觀測性實務，以建立在整個組織中一致的更具彈性且更有效率的系統。它為您提供了一個架構，用於跨三個支柱採用良好的可觀測性實務：人員、程序和技術。無論您選擇使用何種工具，這些資訊都廣泛適用。如果您對支援可觀測性 AWS 的服務感興趣，請參閱 AWS 網站上的監控和可觀測性。

概觀

為什麼您需要重新思考可觀測性策略

可觀測性發展自監控，專注於收集遙測訊號，例如日誌、指標和追蹤，以協助您偵錯應用程式。由於這種關聯性，可觀測性通常是事後考量，並導致檢測過多或過少、無法關聯訊號、中斷連線可見性，以及多個工具，這些工具通常無法進行整合。這些導致感知到缺乏價值和成本，似乎超過可觀測性的優勢。從業務角度來看，這些問題意味著識別時間較長 (MTTI)、復原時間較長 (MTTR)，以及使用者體驗、信任、品牌評價和收入降低。目前可觀測性不僅與偵錯和診斷應用程式的能力有關，也與驗證應用程式是否完全如預期般運作的能力有關。

想要為使用者提供最佳體驗以及可觀測性工具和功能的演變的企業之間的溝通需要重新考慮可觀測性並重新排定優先順序。

可觀測性工具和架構

在 OpenTelemetry 在 2019 年推出之前，為應用程式效能監控 (APM) 和數位體驗監控 (DEM) 提供可觀測性解決方案的專用工具，使得遙測訊號之間的中斷連線更加可見，並突顯不良的使用者體驗。

- APM 會即時追蹤和分析軟體應用程式行為。它會測量關鍵指標，例如回應時間、錯誤率和資源使用量，同時監控應用程式元件之間的使用者交易。APM 工具可協助團隊在這些問題影響使用者之前，快速識別效能問題、瓶頸和錯誤。其主要目標是維持最佳應用程式效能和使用者體驗，同時減少解決問題所需的時間。
- DEM 會測量和分析使用者與數位服務互動的品質。它結合了真實使用者監控 (RUM)、合成監控和端點監控，以提供使用者體驗的完整檢視。DEM 會追蹤指標，例如不同裝置、瀏覽器和位置的頁面載入時間、應用程式回應能力和使用者旅程完成度。這有助於組織了解使用者體驗數位服務的方式、識別影響使用者滿意度的效能問題，以及最佳化數位接觸點。洞見可讓企業做出資料驅動型決策，以改善客戶體驗並維持競爭優勢。

2019 年推出的 OpenTelemetry 為產生、收集、管理和匯出遙測資料 提供了開放原始碼、統一的標準。此架構著重於透過新增內容、提供更好的訊號相互關聯，以及提供更好的衍生值，來橋接遙測訊號之間的差距。例如，使用具有新增內容的結構化日誌可協助您從擷取的日誌衍生指標，並以不同的方式分析資訊，以更快地找到根本原因。在 OpenTelemetry 之前，訊號是單獨檢視的。若要新增功能，您必須修改程式碼，將新維度新增至現有指標或建立新的指標、等待程式碼經過開發生命週期，然後等待在適當的環境中觀察指標，才能進行扣除。此程序會延遲可見性，並在必要時影響您將資料與日誌或追蹤建立關聯的能力。

對 OpenTelemetry 的支援，以及來自此支援的工具改進，可協助您從可觀測性平台獲得更好的價值、增強使用者體驗，並改善營運效率和團隊士氣。

如果您想要改善和增強可觀測性狀態，實際開始的位置和方式為何？ 我們建議採用由三個步驟組成的方法，本指南會詳細說明這些步驟：

- 階段 1：定義您的北極星
- 階段 2：實作可觀測性
- 階段 3：檢查、調整和重複

階段 1：定義您的北極星

成功實作可觀測性不僅與操作和工具有關，還與培養擁有權、持續改善和主動解決問題的文化有關。如同任何成功的策略，您的可觀測性策略需要全面考慮三個支柱：人員、程序和技術。

當您想要建立或改善可觀測性狀態時，建議您先定義重要事項、從業務成果中恢復工作，並在業務、團隊和產品演進時持續檢閱、調整和重新調整策略。

在此第一階段中，您會定義並建立您的 North Star，這是對組織好用的一致且充分了解的定義。我們建議您在業務發展、啟動新產品、應用程式或服務，或設計重大架構變更時，重新檢視此階段中的部分或全部活動，以重新評估您的可觀測性平台和組織需求。

在開發生命週期的早期整合可觀測性（向左轉移方法）

讓可觀測性成為工程、營運和產品團隊中每個成員的責任，並將其視為主要功能需求，類似於您處理單元測試或安全的方式。這不會將責任從營運團隊轉移到開發團隊，而是強調多個團隊所需的協作。這有助於團隊在開發生命週期的早期協同合作中執行下列活動。您可能想要根據每個票證、每個功能或每個產品來執行這些操作。

- 識別利益相關者。如果此功能或產品無法如預期般運作，誰是利益相關者，以及對他們而言什麼是重要的？ 當您識別利益相關者時，請考慮功能、可用性、安全性、成本、銷售和產品使用量等層面。利益相關者可以包括您的團隊、產品的客戶、內部業務利益相關者、平台營運團隊成員和應用程式開發人員。根據情況，您的安全和財務團隊也可以是利益相關者。
- 識別關鍵結果。確定關鍵成果及其對業務和每個利益相關者的影響。識別每個結果和利益相關者的成功和失敗。結果通常定義為服務層級目標 (SLOs)，且必須可量化。SLO 是每個結果的指標。良好的 SLO 具有應作為目標努力或維護的目標值。SLO 可以是使用者滿意度的指標。服務層級指標 (SLI) 是用來判斷您是否符合 SLO 的實際測量或指標：它是您針對目標追蹤的可量化資料點。範例包括將 MTTR 降低 60%、將應用程式可用性維持在 99.99%，或將開發人員生產力提高 30%。

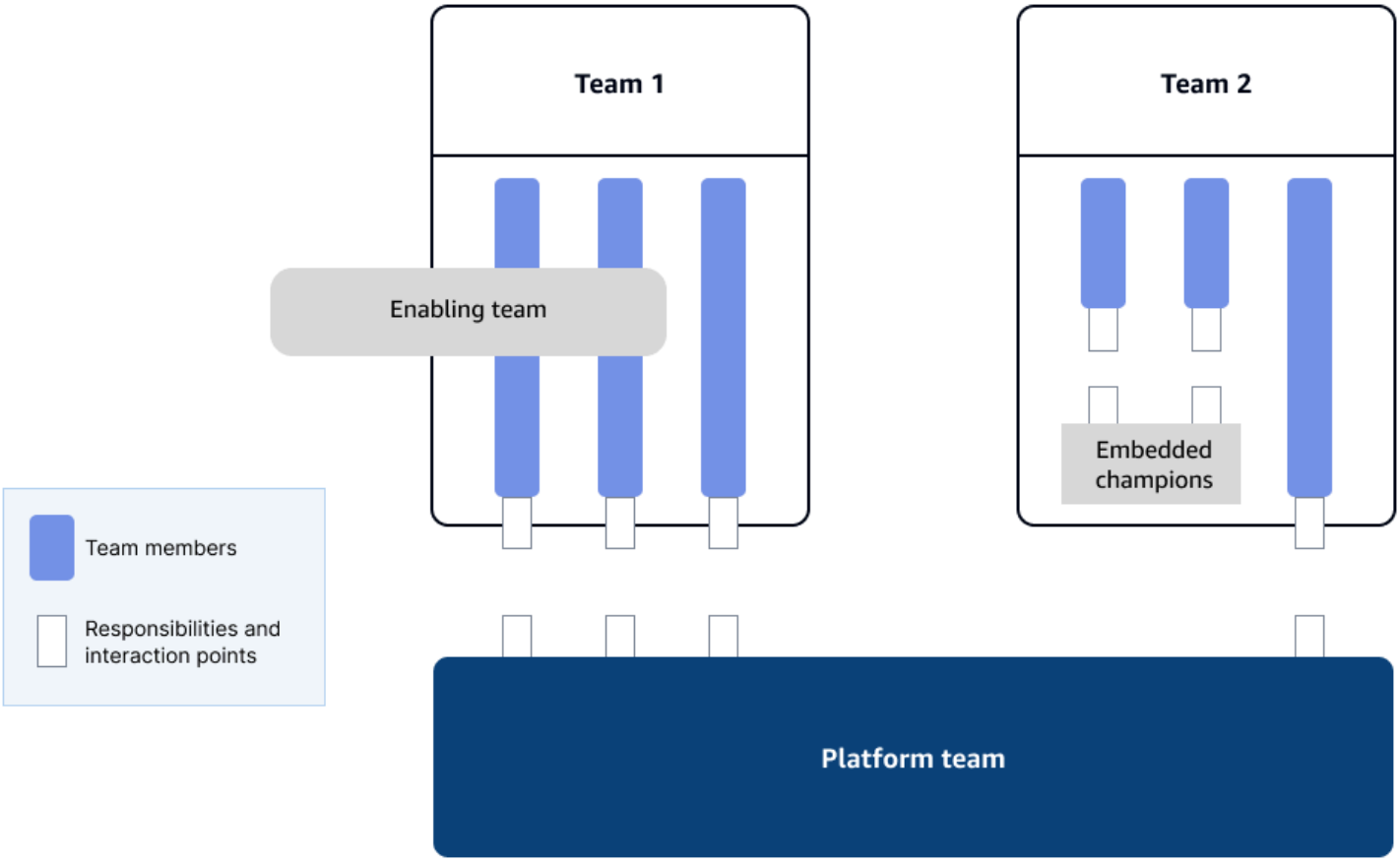
  讓我們舉出將應用程式可用性維持在 99.99% 的範例，並定義衡量和驗證成功所需的 SLO、SLI 和指標。在此範例中，讓我們考慮 RESTful 應用程式，並將應用程式可用性定義為成功完成所有傳入的請求。這需要測量對應用程式的請求總數，以及每個請求的完成狀態。當您將這些轉換為 SLO 和 SLI 時，您需要一個擷取傳入請求的指標，另一個擷取請求狀態的指標。如果所有請求都成功完成，則應用程式會被視為可用。如果一或多個請求導致錯誤，應用程式會被視為無法使用。因此，SLI 會是錯誤請求完成的總和，除以 5 分鐘間隔內傳入請求的總和，實際上是錯誤率。您可以將目標新增至此 SLI，以將其轉換為 SLO；例如：連續 3 個 5 分鐘間隔的錯誤率達到低於 0.1%。

- 排定關鍵結果的優先順序。 根據您為每個結果設定的優先順序，您可以選擇先專注於影響最大的結果，而不是同時執行所有操作。以小幅度開始小型、反覆並改善可觀測性狀態。可觀測性是一種程序，需要持續審查、稽核、增強和改進，以提高成熟度和優勢。優先順序也可以讓您有機會定義已識別結果的增量里程碑。
- 識別所需的檢測。如先前步驟所述，架構或實作的哪些元件和相關功能會影響重要的結果？ 例如，當您在 Amazon Elastic Compute Cloud (Amazon EC2) 執行個體上執行應用程式時，核心和可用 RAM 的數量可能會影響應用程式的回應能力和輸送量。在此階段，判斷您使用的工具或程式庫是否已經提供一些此檢測也很有幫助。執行一系列初步檢閱或將下列問題新增至票證的就緒 (DoR) 定義，可讓此活動成為標準程序的一部分。
  - 如果此操作失敗，您需要知道什麼才能解決失敗？ 典型或有問題的操作如何影響涉及的元件？ 此操作應傳送哪種訊號：日誌、指標或追蹤？ 相較於其值，此檢測的成本是多少？ 在不違反 SLOs 的情況下，可以接受哪種彙總？
  - 可能導致此操作失敗的元件和相依性有哪些？ 如何識別導致故障的元件或相依性？ 這些元件和相依性有哪些不同的組態控制桿，以及每個設定控制桿如何影響操作？
  - 確保可準確測量 SLI 和 SLO 所需的指標精細程度和取樣率是多少？
- 定義成功條件。對於每個排定優先順序的結果，定義與達到或未達到目標的影響相符的閾值。當團隊回應提醒時，成功條件會提供其他內容給團隊。它們還可讓您預測和權衡檢測的成本，以獲得所需的可見性。

設定有效的組織和團隊結構

根據您業務的架構複雜性和大小，您可能需要設定專注於可觀測性的專用團隊。此團隊將負責設定可觀測性工具，並為其他團隊設定可觀測性平台。如果您選擇標準 OpenTelemetry 實作，我們也建議您設定專用團隊。在較小的組織中，您可以將可觀測性指派為每位團隊成員的額外責任，也可以指派可觀測性擁護者，以在整個團隊中宣傳和強制執行最佳實務。這些擁護者會在一天中擔任志願者，為組織定義程序和設定標準。它們可以作為自我協調團隊運作，也可以由專門的可觀測性專家領導。下圖顯示您的投資如何決定您的組織方法。

擁護者可以完全嵌入團隊（如下圖所示的團隊 2)，也可以成為讓團隊輪換以建立和提升最佳實務的團隊（下圖中的團隊 1) 的一部分。



追蹤成本分配

組織應在指標、日誌和追蹤之間實作全面的成本追蹤和可見性，同時建立資源用量和成本的團隊特定責任。財務操作 (FinOps) 實務的成功整合需要具有預算提醒的自動化監控系統，搭配系統性資料保留和

收集最佳化。工程和財務團隊應該透過共用儀表板和定期審查來調整其目標。組織受益於實作明確的退款模型和成本分配策略，以推動擁有權和責任。

定義標準

識別和定義應用程式所需的基本訊號和遙測，包括提醒和儀表板策略。為每個應用程式建立檢查清單或正式檢閱程序。AWS 可觀測性最佳實務網站提供警示和儀表板建立的指導方針，例如設定適當的警示閾值、將警示疲勞降至最低、建立具有足夠每個角色內容的儀表板等。如需連線和策劃的可觀測性體驗，請參閱 Amazon CloudWatch 文件中的應用程式訊號。

建立呈報程序

請務必建立並強制執行呈報機制、警示擁有權和回應程序。我們建議您提升不爭用呈報的文化。

透過訓練提升技能

找出提升現有和新團隊成員技能的最佳方式，強調可觀測性的重要性，並培養持續改進的文化。根據您的組織需求，您可以選擇由可觀測性擁護者或專家提供的預先錄製、隨需訓練或課堂訓練。 AWS 帳戶您的團隊可以提供深入的實作培訓課程，例如 One Observability Workshop 或 GameDays，以指導和改善可觀測性技能和最佳實務。此外， 納入機制以強化最佳實務並提升組織定義的標準。

階段 2：實作可觀測性

在此階段，您會開始讓團隊逐步前往北極星的程序。

選擇您的可觀測性平台

第一步是識別正確的工具來擷取、視覺化和分析訊號，以及傳送提醒。選取工具時，請考慮其功能集、授權模型、價格、技能需求和維護。

功能集

以下是一些要考慮的問題：

- 可設定性和自訂。該工具提供哪些功能來簡化調查體驗，並協助減少 MTTR？ 此工具是否提供警示關聯性、指標數學、處理遺失遙測的彈性，或異常偵測？
- 精細程度。支援的遙測訊號擷取和視覺化精細程度為何？
- 角色。此工具是否支援您想要提供給開發人員、平台工程師和其他角色的體驗？ 它是否適用於技術和業務角色？
- 小工具。儀表板支援哪些類型的小工具？ 工具是否允許建立自訂小工具？
- 預先建置的解決方案。該工具提供哪些預先建置的可觀測性解決方案，以縮短實現價值的時間？
- 自動化和生成式 AI。該工具提供哪些功能，可協助您和您的團隊自動化或減少混沌？ 例如，自動異常偵測、預測分析和其他生成式 AI 功能有助於減輕假設和未知因素的壓力 （也就是您不知道或完全了解的事物）。此工具是否支援使用生成式 AI/ML 模型來增強大規模資料分析？ 它是否為您提供自動化和實作 AIOps 的選項？
- 安全性。 工具支援哪些類型的身分驗證和授權整合？ 使用者和登入體驗是否符合組織的需求？
- OpenTelemetry 支援。工具和代理程式是否支援 OpenTelemetry？ 大多數可觀測性平台支援擷取 OpenTelemetry 相容訊號，但並非所有代理器都提供組態選項，將這些訊號轉送至可觀測性平台。
- 整合。工具提供哪些整合？ 考慮是否需要傳送提醒給 Slack、頁面團隊成員或自動化解決方案。
- 延展性。 工具的可擴展性和效能如何？ 可觀測性解決方案必須隨著您的需求和用量增加而擴展，因此即使您的應用程式無法使用，它也可以提供診斷。
- 支援。提供哪些支援模型？ 即使應用程式失敗，您的可觀測性工具也必須可用，如此您才能符合 MTTR 和應用程式可用性目標或服務層級協議 (SLAs)。開放原始碼解決方案可能提供有限的正式支援。

授權和部署模型

考慮解決方案的授權模式（開放原始碼或商業）和部署模式（自我託管或雲端型）。開放原始碼選項通常具有較低的預付成本，但在提供價值之前，可能需要更多時間進行部署、設定和組態、維護和團隊升級。如果您正在考慮開放原始碼選項，您可能需要一個專門的可觀測性專家團隊。商業軟體通常會以更高的前期成本提供更快的價值時間，而且專用可觀測性團隊的需求會隨著採用率、複雜性和成熟度的增加而隨時間演進。與雲端型解決方案相比，自我託管解決方案需要更多時間進行部署、設定和組態、維護和操作開銷。

定價維度

隨著您的應用程式獲得新的使用者、更大的架構足跡或新的功能和應用程式，工具的定價模型將如何影響您的總體擁有成本 (TCO)？ 例如，某些典型的授權模式是永久的或基於訂閱、具名使用者數量、耗用量或磁碟區。考慮您的應用程式和可觀測性工具如何擴展用量，以及授權模型如何影響工具的成本。

團隊技能

根據團隊目前的技能集和成熟度，您需要判斷需要多少提升技能。考慮廠商提供哪些支援來訓練您的團隊。同時考慮您的組織結構是否支援您選擇的工具的組態和管理。例如，如果您選擇 OpenTelemetry，您應該考慮設定專門提供可觀測性的單獨團隊。

操作和維護

評估下列問題：

- 可觀測性代理程式或收集器提供哪些部署選項？ 這些選項是否符合架構的要求？ 例如，如果您使用可觀測性工具的容器化部署，它是否支援協助程式集或附屬項目？ 營運團隊需要採取或使用哪些額外的步驟或工具，以確保符合安全性和所有其他程序？
- 維護解決方案所需的工作量為何？ 更新代理程式或收集器的程序有多簡單或自動化？ 與自我部署和託管應用程式相比，完全受管和雲端型可觀測性界面通常具有較低的操作開銷，但代理程式或收集器的管理保持不變。將您的團隊結構納入考量，並考量維護您選擇的選項的人力成本。

檢測您的應用程式

上一節問題的答案為您提供檢測應用程式所需的資訊，也就是新增程式碼以擷取應用程式的遙測訊號，以及測量、觀察和驗證行為。您可以使用SDKs應用程式語言的 OpenTelemetry SDK 等 SDK 來自動檢測應用程式。您可能仍然需要新增手動檢測程式碼，以涵蓋任何差距，並確保end-to-end可見性。請特別注意您新增的遙測，並確保您可以將它連接到您在上一個階段建立的一或多個 SLIs 和 SLOs。

收集遙測

設定遙測收集器或代理程式來擷取相關的遙測訊號，以符合您在階段 1 中排定優先順序的結果。

實作可觀測性元件

當遙測正在流動並擷取到可觀測性平台時，請建立儀表板、警示、手冊和 Runbook。

- 儀表板：建立包含相關資訊的儀表板，包括與優先結果相關聯的目前和歷史趨勢視覺化呈現。將這些儀表板提供給您在階段 1 中定義的利益相關者。如需詳細資訊，請參閱 Amazon Builders' Library網站上的建置儀表板以取得操作可見性。
- 提醒：定義提醒，在結果處於風險或違規時通知您的團隊。請考慮新增安全性和效能問題的提醒。採用下列方式來最佳化警示，以減少警示疲勞和成本：
  - 使用異常偵測來避免設定硬閾值，這需要頻繁調整，並減少未知的發生。
  - 使用智慧提醒組合，一起查看多個相關指標，而不是為每個指標設定個別提醒。例如，不針對 CPU、記憶體和回應時間設定個別提醒，而是建立一個有意義的提醒，只在這些指標共同指出實際問題時觸發。這種方法可大幅減少警示雜訊，並協助您的團隊專注於實際的服務影響問題，而不必回應隔離的指標峰值。
  - 只有在使用者體驗或結果處於風險時，才會產生提醒。例如，當您的應用程式沒有作用中的使用者時，請避免收到由自動升級引起的 CPU 峰值提醒。
- 手冊：手冊為回應事件或警示的人員提供心理模型和內容，並協助他們更快識別根本原因。考慮為高度耦合、複雜的應用程式和缺少檢測但直接影響您在階段 1 中識別和優先結果的應用程式建立手冊。
- Runbook：使用 Runbook 定義解決事件或警示所需的步驟。

驗證可觀測性系統

在整個軟體開發生命週期 (SDLC) 中，驗證儀表板是否在系統測試期間提供預期的行為和更新。實作混沌工程並驗證手冊和 Runbook 中記錄的步驟，以確保它們準確並滿足其目的。您也應該驗證警示擁有權和呈報路徑。

階段 3：檢查、調整和重複

實作可觀測性系統之後，建議您持續檢閱、評估、學習、調整和改善實作。您可以使用AWS 可觀測性成熟度模型作為工具，以評估實作的成熟度，並識別需要改進的領域並排定其優先順序。

實作定期檢閱

可觀測性是一種反覆程序。它需要定期稽核和評估現有元件，以及變更和增強功能，以推動持續改進。我們建議您執行定期審查，以重新評估 SLOs、警示閾值、儀表板、指標精細程度、保留政策、抽樣策略等，以確保這些都為您的團隊和業務帶來價值。透過將可觀測性成本連接到特定團隊和服務，您可以啟用有關涵蓋範圍和資源配置的資料驅動型決策。

在 Amazon，我們會每週執行營運準備度審查 (ORRs)，根據最佳實務稽核團隊的程序和可觀測性狀態。這是非封鎖練習，符合 Amazon 的服務數量和發行頻率。

根據您的組織大小，您也可以擁有照常營運 (BAU) 名單，其中每個團隊中的一位成員負責報告異常和趨勢、發現未知情況、移除不必要的檢測和提醒、改善儀表板，並確保可觀測性解決方案繼續為團隊工作，並與團隊的目標和成功指標保持一致。這也有機會重新評估警示策略，以更迅速、主動且更接近使用者。這些檢閱的目標是建立良性循環，如下圖所示，並改善可觀測性姿勢成熟度的成熟度，如AWS 可觀測性成熟度模型中所述。

識別最常存取的手冊，並考慮改善您的應用程式或新增更多檢測。識別最常執行的 Runbook，並考慮自動化這些 Runbook。

這些檢閱的學習也會與可觀測性小組和專家分享，以強調集中計劃和可觀測性平台的改善。例如，根據部署觸發事件的頻率，您可以決定優先改善部署管道，而不是其他元件。如果 MTTR 因為監控差距而較高，您可以優先改善可觀測性平台及其組態。

慶祝勝利

分享來自使用可觀測性工具之團隊的成功案例。例如，強調使用可觀測性指標來實作替代解決方案的團隊成功，該解決方案更有效率，並降低延遲或成本。傳達此成功強調可觀測性的重要性，並鼓勵其他團隊改善可觀測性狀態，並努力取得類似的成功。

從事件中學習

進行與 Amazon 錯誤校正 (COE) 程序類似的無責事後練習，以識別需要改進的領域並防止未來發生問題。與獲勝一樣，本練習的學習可以與其他團隊廣泛分享，以強化可觀測性和最佳實務的價值。

後續步驟和資源

透過從使用者和業務成果開始，並專注於影響，您可以從可觀測性解決方案中獲得更好的價值，同時控制成本並避免工具擴散。請記住，可觀測性是一種反覆程序，而不是目的地。設定您的團隊和程序，以持續迭代和改善您的可觀測性狀態、納入最佳實務，並在架構、業務需求和團隊發展時縮小差距，並以明確定義的目標做為指南。透過規劃失敗、整合持續改進的文化，並採用本指南中討論的最佳實務，您應該能夠將 Werner Vogels 醫生的指導付諸實踐：「一切都會一直失敗...所以規劃失敗，什麼都不會失敗。」

如需協助您定義可觀測性策略的實作研討會，請聯絡您的 AWS 代表，以提供可觀測性策略研討會。

資源

如需在 實作有效可觀測性策略的詳細資訊和指引 AWS，請參閱下列資源：

- 監控和可觀測性 （ 可AWS 觀測性服務 ）
- AWS 可觀測性最佳實務
- AWS Distro for OpenTelemetry
- 卓越營運支柱 (AWS Well-Architected Framework)
- 建置儀表板以實現營運可見性 (Amazon Builders' Library)
- 可觀測性 （ 在 DevOps 指引中， AWS Well-Architected Framework)

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	—	2025 年 7 月 16 日

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。