



開發人員指南

# Amazon Simple Workflow Service



API 版本 2012-01-25

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

# Amazon Simple Workflow Service: 開發人員指南

Copyright © 2024 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，這些擁有者可能隸屬於 Amazon，或與 Amazon 有合作關係，或由 Amazon 贊助。

# Table of Contents

什麼是 Amazon SWF ? .....	1
使用 AWS SDK 進行開發 .....	1
考慮 AWS Flow Framework .....	1
使用 HTTP 服務 API 進行建置 .....	2
Amazon SWF 簡介 .....	3
概念 .....	3
工作流程執行 .....	4
設定 .....	5
開始 .....	6
關於工作流程 .....	7
必要條件 .....	7
教學步驟 .....	7
第 1 部分：使用適用 SDK for Ruby 使用 Amazon SWF .....	8
包括 AWS SDK for Ruby .....	8
設定AWS工作階段 .....	8
註冊 Amazon SWF 域 .....	10
後續步驟 .....	11
第 2 部分：實作工作流程 .....	11
設計工作流程 .....	11
設定工作流程程式碼 .....	12
註冊工作流程 .....	13
輪詢決策 .....	15
啟動工作流程執行 .....	17
後續步驟 .....	19
第 3 部分：實作活動 .....	19
定義基本活動類型 .....	20
定義 GetContactActivity .....	21
定義 SubscribeTopicActivity .....	23
定義 WaitForConfirmationActivity .....	26
定義 SendResultActivity .....	29
後續步驟 .....	30
第 4 部分：實作活動任務輪詢器 .....	30
執行工作流程 .....	33
接下來做些什麼？ .....	37

教學課程 .....	38
註冊網域 .....	38
註冊工作流程類型 .....	41
註冊活動類型 .....	46
啟動工作流程執行 .....	51
使用主控台啟動工作流程執行 .....	51
檢視擱置中任務 .....	56
管理工作流程執行 .....	57
管理 Amazon SWF 網域 .....	65
Amazon SWF 運作方式 .....	66
工作流程 .....	66
什麼是工作流程？ .....	66
簡單工作流程範例：電子商務應用程式 .....	67
工作流程註冊和執行 .....	68
另請參閱 .....	68
工作流程歷史記錄 .....	68
動作者 .....	73
什麼是 Amazon SWF 中的動作者？ .....	73
工作流程啟動者 .....	74
決策者 .....	74
活動工作者 .....	75
動作者之間的資料交換 .....	76
任務 .....	76
網域 .....	77
物件識別符 .....	77
任務清單 .....	78
決策任務清單 .....	78
活動任務清單 .....	79
任務路由 .....	79
關閉工作流程執行 .....	79
Amazon SWF 工作流程執行生命週期 .....	80
工作流程執行生命週期 .....	80
輪詢任務 .....	85
進階概念 .....	85
版本控制 .....	86
訊號 .....	86

子工作流程 .....	88
標記 .....	89
標籤 .....	90
實作互斥選項 .....	93
計時器 .....	96
活動任務取消 .....	97
安全性 .....	100
資料保護 .....	100
加密 .....	101
身分和存取權管理 .....	101
物件 .....	102
使用身分驗證 .....	103
使用政策管理存取權 .....	105
存取控制 .....	107
政策動作 .....	107
政策資源 .....	108
政策條件索引鍵 .....	109
ACL .....	109
ABAC .....	110
臨時憑證 .....	110
主體許可 .....	111
服務角色 .....	111
服務連結角色 .....	111
身分型政策 .....	112
資源型政策 .....	112
Amazon 簡單工作流程服務如何與 IAM 配合 .....	112
身分型政策範例 .....	113
基本原則 .....	116
Amazon SWF IAM 政策 .....	117
API 摘要 .....	124
標籤類型政策 .....	132
Amazon VPC 端點 VPC 端點 .....	133
故障診斷 .....	135
記錄和監控 .....	136
Amazon SWF 指標 CloudWatch .....	137
檢視 Amazon SWF 指標 .....	146

錄製到 CloudTrail .....	149
Amazon SWF EventBridge .....	156
AWS 使用者通知與 Amazon SWF 一起使用 .....	164
合規驗證 .....	165
恢復能力 .....	165
基礎設施安全 .....	165
組態與漏洞分析 .....	166
使用 AWS CLI .....	167
使用 API .....	169
提出 HTTP 請求 .....	169
HTTP 標頭內容 .....	169
HTTP 內文內容 .....	171
範例 JSON 請求和回應 .....	172
計算 HMAC-SHA 簽章 .....	173
Amazon SWF 動作列表 .....	175
與活動相關的動作 .....	175
與決策者相關的動作 .....	176
與工作流程執行相關的動作 .....	176
與管理相關的動作 .....	176
可見度動作 .....	177
建立基本工作流程 .....	178
為工作流程和其活動建模 .....	179
註冊網域 .....	179
另請參閱 .....	180
設定逾時值 .....	180
逾時值限額 .....	180
工作流程執行和決策任務逾時 .....	180
活動任務逾時 .....	181
另請參閱 .....	182
註冊工作流程類型 .....	182
另請參閱 .....	182
註冊活動類型 .....	182
另請參閱 .....	183
Lambda 任務 .....	183
關於 AWS Lambda .....	183
使用 Lambda 任務的優點與限制 .....	184

在工作流程中使用 Lambda 任務 .....	184
開發活動工作者 .....	189
輪詢活動任務 .....	190
執行活動任務 .....	190
報告活動任務活動訊號 .....	190
完成或讓活動任務失敗 .....	191
啟動活動工作者 .....	192
開發決策者 .....	193
定義協調性邏輯 .....	194
輪詢決策任務 .....	194
套用協調性邏輯 .....	196
回應決策 .....	197
關閉工作流程執行 .....	198
啟動決策者 .....	199
啟動工作流程執行 .....	199
設定任務優先順序 .....	200
為工作流程設定任務優先順序 .....	201
為活動設定任務優先順序 .....	203
傳回任務優先順序資訊的動作 .....	204
處理錯誤 .....	204
驗證錯誤 .....	205
制定動作或決策時發生錯誤 .....	205
逾時 .....	205
使用者代碼所引發的錯誤 .....	206
與關閉工作流程執行相關的錯誤 .....	206
配額 .....	207
Amazon SWF 的一般帳戶配額 .....	207
工作流程執行配額 .....	208
工作執行配額 .....	208
Amazon SWF 節流配額 .....	210
所有區域的節流配額 .....	210
所有區域的決策配額 .....	212
工作流程層級配額 .....	213
請求提高配額 .....	213
資源 .....	214
逾時類型 .....	214

工作流程和決策任務的逾時 .....	214
活動任務的逾時 .....	215
端點 .....	217
其他 文件 .....	217
Amazon 簡單工作流服務 API 參考 .....	217
AWS Flow Framework 文件 .....	217
AWS SDK 文件集 .....	218
AWS CLI 文件 .....	219
Web 資源 .....	219
Amazon SWF 論壇 .....	220
Amazon SWF 常見問 .....	220
Amazon SWF 視頻 .....	220
Ruby Flow 選擇 .....	220
繼續使用 Ruby Flow Framework .....	221
遷移至 Java Flow Framework .....	221
遷移至 Step Functions .....	221
直接使用 Amazon SWF API .....	222
文件歷史記錄 .....	223
.....	CCXXVII

# 什麼是 Amazon 簡單 workflow 服務？

Amazon 簡單 workflow 服務 (Amazon SWF) 提供建置、執行和擴展具有 parallel 或連續步驟的背景任務的方法。使用 Amazon SWF，您可以協調分散式元件間的工作，追蹤任務狀態。

在 Amazon SWF 中，任務代表由應用程式元件執行的邏輯工作單元。跨應用程式協調工作涉及管理應用程式邏輯流程中的任務間相依性、排程和並行。Amazon SWF 可讓您控制實作任務和協調任務，而不必擔心基本複雜性，例如追蹤任務進度和維護狀態。

使用 Amazon SWF 時，您可以實作工作者來執行任務。工作者可以在雲端基礎設施上執行，例如 Amazon Elastic Compute Cloud (Amazon EC2)，也可以在您自己的內部部署執行。您可以建立長時間執行、可能會失敗、逾時或需要重新啟動的工作，或者建立可能以不同的輸送量和延遲完成的工作。Amazon SWF 會存放任務，並在工作者準備就緒時將任務指派給工作者、追蹤其進度並維持其狀態，包括完成的詳細資料。

為了協調任務，您需要編寫一個程式，該程式會從 Amazon SWF 取得最新任務狀態，並使用該狀態來啟動後續任務。Amazon SWF 可持續維護應用程式的執行狀態，因此您的應用程式能夠抵禦個別元件故障。使用 Amazon SWF，您可以獨立建置、部署、擴展和修改應用程式元件。

Amazon SWF 適合需要協調任務的使用案例，包括媒體處理、Web 應用程式後端、業務程序工作流程和分析管道。

## 使用 AWS SDK 進行開發

Amazon SWF 受到 Java、.NET、Node.js、PHP、Python 和紅寶石的 AWS 開發套件支援，提供了一種以您選擇的程式設計語言使用 Amazon SWF HTTP API 的便捷方式。

您可以使用這些程式庫公開的 API 來開發決策程式、活動工作者或 workflow 起始程式。此外，您可以透過這些程式庫使用可見性操作，以便開發自己的 Amazon SWF 監控和報告工具。

若要下載開發和管理應用程式的 AWS 工具 (包括 SDK)，請前往開發 [人員中心](#)。

如需每個開發套件中 Amazon SWF 方法的詳細資訊，請參閱開發套件的特定語言參考文件。

## 考慮 AWS Flow Framework

這 AWS Flow Framework 是一個增強型開發套件，用於撰寫在 Amazon SWF 上以 workflow 形式執行的分散式非同步程式。該框架可用於 Java 編程語言，並提供用於編寫複雜分佈式程序的類。

使用 AWS Flow Framework，您可以使用預先設定的類型，將工作流程的定義直接對應至程式中的方法。AWS Flow Framework 支援標準物件導向概念，例如以例外狀況為基礎的錯誤處理，這可讓您更輕鬆地實作複雜的工作流程。使用編寫的程式 AWS Flow Framework 可以完全在偏好的編輯器或 IDE 中建立、執行和偵錯。如需詳細資訊，請參閱 [AWS Flow Framework](#) 網站，並參閱 [AWS Flow Framework Java 開發人員指南](#)。

## 使用 HTTP 服務 API 進行建置

Amazon SWF 提供可透過 HTTP 請求存取的服務操作。您可以使用這些操作直接與 Amazon SWF 進行通訊，也可以使用這些操作，以任何可透過 HTTP 與 Amazon SWF 通訊的語言開發自己的程式庫。

您可以使用服務 API 開發決策程式、活動工作者或工作流程啟動器。您還可以通過 API 使用可見性操作來開發自己的監視和報告工具。

如需如何使用 API 的資訊，請參閱「[向 Amazon SWF 提出 HTTP 請求](#)」。

# Amazon SWF 簡介

有越來越多的應用程式依賴非同步和分散式處理。這類應用程式的擴展性是使用此方法的主要動機。藉由設計獨立自主的分散式元件，開發人員可以彈性地在應用程式負載增加時，獨立部署及擴展應用程式的各部分。另一個動機是雲端服務的可用性。當應用程式開發人員開始利用雲端運算時，必須結合其現有的現場部署資產與其他雲端式資產。使用非同步和分散式方法另有一個動機，即是應用程式要建立模型之程序的固有分散性質，如訂單履行商務程序的自動化可能跨越數個系統和人力任務。

開發這類應用程式可能很複雜。您必須協調多個分散式元件的執行，並處理遠端通訊中增加的延遲和不可靠的固有。為了達成這個目標，您通常需要撰寫涉及訊息佇列和資料庫的複雜基礎架構，以及用於同步的複雜邏輯。

Amazon Simple Workflow Service (Amazon SWF) 提供程式設計模型和基礎設施來協調分散式元件並以可靠的方式維持其執行狀態，讓開發非同步和分散式應用程式變得更加輕鬆。透過仰賴 Amazon SWF，您就可以專注於建置與眾不同的應用程式各個層面。

## Simple Workflow 概念

下面介紹了解 Amazon SWF 工作流程所需的基本概念，並在本指南的後續章節中進一步說明。下列討論是工作流程結構和元件的整體概觀。

Amazon SWF 中的基本概念是工作流程。工作流程是一組實現一些目標的「活動」，加上協調這些活動的邏輯。例如，一個工作流程可能會接收客戶的訂單，並採取履行訂單的任何必要動作。每個工作流程都會在稱為網域的 AWS 資源中執行，該資源可控制工作流程的範圍。一個 AWS 帳戶可以有許多網域，每個網域都可以包含多個工作流程，但不同網域中的工作流程無法互動。

設計 Amazon SWF 工作流程時，您可以精確定義每個必要的活動。然後，您可以在 Amazon SWF 中將每個活動註冊為活動類型。當您註冊活動時，您會提供名稱和版本等資訊，並根據活動預期的時間長度來提供一些逾時值。例如，客戶可能預期訂單會在 24 小時內出貨。這類預期會影響您在註冊活動時所指定的逾時值。

在執行工作流程的過程中，有些活動可能需要執行多次，並可能有不同的輸入。例如，在客戶訂單工作流程中，您可能會有處理購買項目的活動。如果客戶購買多個料號，則此作業必須執行多次。Amazon SWF 具有一個活動任務的概念，該任務代表一次活動的調用。在我們的範例中，每個項目的處理會以單一活動任務表示。

「活動工作者」是接收並執行活動任務，然後提供結果的程式。請注意，任務本身可能實際上是由某人執行，在此情況下，這個人會使用活動工作者軟體來接收及處置任務。以統計分析師為例，他們會接收一組資料、加以分析，然後送回分析。

活動任務 — 以及執行它們的活動工作者 — 可以同步或非同步執行。它們可以分散到多部電腦 (可能位於不同的地理區域)，也可以全部在同一部電腦上執行。不同的活動工作者可以使用不同的程式設計語言撰寫，並在不同的作業系統上執行。例如，一個活動工作者可能在亞洲的桌上型電腦上執行，而另一個活動工作者可能在北美洲的掌上型電腦裝置上執行。

工作流程的協調邏輯包含在稱為「決策者」的軟體程式中。決策者會排程活動任務、提供輸入資料給活動工作者、處理在工作流程進行時抵達的事件，最後在完成目標時結束 (或關閉) 工作流程。

Amazon SWF 服務的作用是做為可靠的中央中樞，透過這個中樞可在決策程式、活動工作程式和其他相關實體 (例如管理工作流程的人員) 之間交換資料。Amazon SWF 也會維護每個工作流程執行的狀態，讓您的應用程式不必以持久的方式儲存狀態。

決策程式會接收來自 Amazon SWF 的決策任務，並透過決策回應給 Amazon SWF，藉此引導工作流程。決策代表工作流程後續步驟的一個動作或一組動作。一個典型的決策是排程活動任務。決策也可用來設定計時器，以延遲執行活動任務、請求取消已在進行的活動任務，以及完成或關閉工作流程。

活動工作者和決策程式都接收其任務 (分別為活動任務和決策任務) 的機制是輪詢 Amazon SWF 服務。

Amazon SWF 會在每個決策任務中包含目前工作流程執行歷程記錄的副本，藉此通知決策程式工作流程的狀態。工作流程執行歷史記錄是由事件組成，每個事件代表工作流程執行狀態中的一項重大變更。事件範例包括任務完成、任務逾時通知，或工作流程執行先前設定的計時器過期。歷史記錄是工作流程進度之完整一致的權威記錄。

Amazon SWF 存取控制使用 AWS Identity and Access Management (IAM)，可讓您以受控且有限的方式提供對 AWS 資源的存取。例如，您可以允許使用者存取您的帳戶，但只能執行特定網域中的某些工作流程。

## 工作流程執行

以下概述了在 Amazon SWF 中開發和執行工作流程的步驟，整合上述各節中討論的想法：

1. 撰寫活動工作者，以實作您工作流程中的處理步驟。
2. 撰寫決策者，以實作您工作流程的協調邏輯。
3. 使用 Amazon SWF 註冊您的活動和工作流程。

您可以透過程式設計方式執行此步驟，或使用 AWS Management Console。

4. 啟動您的活動工作者和決策者。

這些演員可以在任何可存取 Amazon SWF 端點的運算裝置上執行。例如，您可以使用雲端中的運算執行個體，例如 Amazon Elastic Compute Cloud (Amazon EC2)、資料中心中的伺服器，甚至是

行動裝置來託管決策程式或活動工作者。啟動之後，決策程式和活動工作者應該開始輪詢 Amazon SWF 是否有任務。

#### 5. 啟動您工作流程的一或多個執行。

這些執行可以透過程式設計的方式或透過 AWS Management Console 啟動。

每個執行都獨立執行，而且您可以為每個執行提供其各自的輸入資料集。啟動執行時，Amazon SWF 會排程初始決策任務。做為回應，您的決策者會開始產生啟動活動任務的決策。執行會繼續進行，直到您的決策者決定關閉執行為止。

#### 6. 使用檢視工作流程執行。 AWS Management Console

您可以篩選及檢視執行中及已完成執行的完整詳細資訊。例如，您可以選取已開啟的執行，了解哪些任務已完成及其結果。

## 設定

您可以選擇使用支援的任何程式設計語言為 Amazon SWF 進行開發 AWS。對於 Java 開發人員，AWS Flow Framework 也可以使用。如需詳細資訊，請參閱[AWS Flow Framework](#)網站，並參閱[AWS Flow Framework Java 開發人員指南](#)。

為了減少延遲並將資料存放在符合您需求的位置，Amazon SWF 在不同區域提供端點。

Amazon SWF 中的每個端點都是完全獨立的。您在某個區域中註冊的任何網域、工作流程和活動都不會與另一個區域中的網域或屬性共用資料或屬性。

當您註冊 Amazon SWF 網域、工作流程或活動時，該網域僅存在於您註冊該網域的區域內。例如，您可以 SWF-Flows-1 在兩個不同的區域中註冊一個名為的網域，但這些網域不會彼此共用資料或屬性，每個網域都是完全獨立的網域。

如需 Amazon SWF 端點的清單，請參閱[區域和端點](#)。

# Amazon SWF 和 Amazon SNS 的訂閱工作流程教程

本節提供一個教學課程，說明如何建立 Amazon SWF 工作流程應用程式，該應用程式由一組四個按順序運作的活動組成。它也涵蓋：

- 設定 default 和 execution-time 工作流程與活動選項。
- 輪詢 Amazon SWF 以取得決策和活動任務。
- 使用 Amazon SWF 在活動和工作流程之間傳遞資料。
- 等待人工任務，並從活動任務向 Amazon SWF 報告活動訊號。
- 使用 Amazon SNS 建立主題、向使用者訂閱主題，然後將訊息發佈到已訂閱的端點。

您可以同時使用 [Amazon 簡易工作流程服務 \(Amazon SWF\)](#) 和 [Amazon Simple Notification Service \(Amazon SNS\)](#) 來模擬「人工任務」工作流程，這個工作流程需要人工執行某些動作，然後與 Amazon SWF 通訊以啟動工作流程中的下一個活動。

由於 Amazon SWF 是雲端式網頁服務，因此與 Amazon SWF 的通訊可以從任何可連線到網際網路的地方產生。在這種情況下，我們將使用 Amazon SNS 透過電子郵件、簡訊或兩者與使用者通訊。

本教學課程使用 [AWS SDK for Ruby](#) 來存取 Amazon SWF 和 Amazon SNS，但有許多可用的開發選項，包括 Ruby，可讓您更輕鬆地與 Amazon SWF 進行協調和通訊。AWS Flow Framework

## Note

本教學課程使用 AWS SDK for Ruby，但我們建議您使 [AWS Flow Framework 用 Java](#)。

## 主題

- [關於工作流程](#)
- [必要條件](#)
- [教學步驟](#)
- [訂閱工作流程教學第 1 部分：將 Amazon SWF 與 AWS SDK for Ruby](#)
- [訂購工作流程教學第 2 部分：實作工作流程](#)
- [訂工作流程教學第 3 部分：實作活動](#)
- [訂工作流程教學第 4 部分：實作活動任務輪詢器](#)
- [訂工作流程教學：執行工作流程](#)

## 關於工作流程

我們將開發的工作流程包含四個主要步驟：

1. 從使用者取得訂閱地址 (電子郵件或簡訊)。
2. 建立 SNS 主題並將提供的端點訂閱到主題。
3. 等待使用者確認訂閱。
4. 如果使用者確認，將發佈賀辭到主題。

這些步驟包含完全自動化的活動 (步驟 2 和 4)，以及需要工作流程等待人員先將某項資料提供給活動才能進行工作流程的活動 (步驟 1 和 3)。

每個步驟都依賴前一個步驟所產生的資料 (您必須要先有端點，才能訂閱到主題，而且您必須要先有主題訂閱，才能等待確認，以此類推)。本教學課程也會說明如何在完成時提供活動結果，以及如何將輸入傳遞至正在排程的工作。Amazon SWF 會處理活動和工作流程之間資訊的協調和交付，反之亦然。

我們也同時使用鍵盤輸入和 Amazon SNS 來處理 Amazon SWF 與提供資料給工作流程的人員之間的通訊。實際上，您可以使用許多不同的技術與人類使用者通訊，但 Amazon SNS 提供了一種非常簡單的方法，可以使用電子郵件或文字訊息通知使用者工作流程中的事件。

## 必要條件

若要遵循本教學，您將需要下列項目：

- [Amazon Web Services 帳戶](#)
- [Ruby 解譯器](#)
- [AWS SDK for Ruby](#)

如果您已經設定好這些項目，代表您已準備就緒。如果您不想執行範例，您仍然可以依照教學課程進行操作 — 無論您選擇的開發選項為何，本教學中的大部分內容都適用於使用 Amazon SWF 和 Amazon SNS。

## 教學步驟

本教學分為下列步驟：

1. [訂閱工作流程教學第 1 部分：將 Amazon SWF 與AWS SDK for Ruby](#)
2. [訂購工作流程教學第 2 部分：實作工作流程](#)
3. [訂工作流程教學第 3 部分：實作活動](#)
4. [訂工作流程教學第 4 部分：實作活動任務輪詢器](#)
5. [訂工作流程教學：執行工作流程](#)

## 訂閱工作流程教學第 1 部分：將 Amazon SWF 與AWS SDK for Ruby

### 主題

- [包括 AWS SDK for Ruby](#)
- [設定AWS工作階段](#)
- [註冊 Amazon SWF 域](#)
- [後續步驟](#)

## 包括 AWS SDK for Ruby

從建立名為 `utils.rb` 的檔案開始。此檔案中的程式碼將取得或建立（如必要）工作流程和活動程式碼使用的 Amazon SWF 網域，並提供位置來放置所有類別都通用的程式碼。

首先，我們需要包含 `aws-sdk-v1` 庫，這樣我們就可以使用適用 SDK for Ruby 提供的功能。

```
require 'aws-sdk-v1'
```

這可讓我們存取AWS命名空間，它提供了設定與全局會話相關的值的功能，例如AWS憑據和區域，還提供了對AWS服務 API。

## 設定AWS工作階段

我們將配置AWS會話，通過設置我們的AWS憑據（用於訪問AWS服務）和AWS要使用的區域。

您可透過多種方式使用[組AWS](#)中的[登入資料AWS適用於 Ruby 的 開發套件](#)：在環境變數（`AWS_ACCESS_KEY_ID` 和 `AWS_SECRET_ACCESS_KEY`）中加以設定，或使用[AWS.config](#)。我們會使用第二種方法，從名為 `aws-config.txt` 的 YAML 組態檔案將其載入，如下所示。

```
---
:access_key_id: REPLACE_WITH_ACCESS_KEY_ID
:secret_access_key: REPLACE_WITH_SECRET_ACCESS_KEY
```

立即創建此文件，替換以用 `_` 替換和您的AWS存取金鑰 ID 和私密存取金鑰。如需您的AWS訪問密鑰，請參閱[如何取得安全登入資料？](#)中的Amazon Web Services 一般參考。

我們也需要設定AWS要使用的區域。因為我們將使用[短訊服務](#)若要使用 Amazon SNS 向用戶的手機發送短信，我們需要確保我們使用的是 Amazon SNS 支持的區域。請參閱[支援的區域和國家](#) (位於 Amazon Simple Notification Service 開發人員指南)。

### Note

如果您無法存取 `us-east-1`，或不在意能否執行啟用簡訊的示範，請隨意使用任一區域。您可從範例移除簡訊功能，並使用電子郵件作為訂 Amazon SNS 主題的唯一端點。

如需關於傳送 SMS 消息的詳細資訊，請參閱[使用 Amazon SNS 傳送和接收 SMS 通知](#)中的 Amazon Simple Notification Service 開發人員指南。

我們現在將一些代碼添加到 `utils.rb` 加載配置文件，取得用戶的登入資料，然後將登入資料和區域提供給[AWS.config](#)。

```
require 'yaml'

# Load the user's credentials from a file, if it exists.
begin
  config_file = File.open('aws-config.txt') { |f| f.read }
rescue
  puts "No config file! Hope you set your AWS credentials in the environment..."
end

if config_file.nil?
  options = { }
else
  options = YAML.load(config_file)
end

# SMS Messaging (which can be used by Amazon SNS) is available only in the
# `us-east-1` region.
$SMS_REGION = 'us-east-1'
options[:region] = $SMS_REGION
```

```
# Now, set the options
AWS.config = options
```

## 註冊 Amazon SWF 域

若要使用 Amazon SWF，您需要設定域：這是容納您工作流程和活動的具名實體。您可以註冊許多 Amazon SWF 域，但它們必須在您的 AWS 帳戶，並且 workflow 無法跨域進行交互：您應用程式的所有 workflow 和活動都必須位於同一個網域，才能彼此互動。

因為我們將在整個應用程序中使用相同的域，所以我們將在 `utils.rb` 呼叫 `init_domain`，它將檢索名為 SWF 樣本域。

當您一註冊網域後，就可以重複使用它處理許多 workflow 執行。但是，「嘗試註冊已存在的網域會發生錯誤」，所以我們的程式碼會先檢查網域是否存在，若可找到現有網域，就會予以使用。如果找不到網域，就會予以建立。

要在適用於 Ruby 的軟件開發工具包中使用 Amazon SWF 域，請使用 [AWS::SimpleWorkflow.域](#)，它返回一個 [DomainCollection](#)，可用於枚舉和註冊域：

- 若要檢查是否已註冊網域，您可查看 [AWS::SimpleWorkflow.domains.registered](#) 提供的清單。
- 若要註冊新的網域，請使用 [AWS::SimpleWorkflow.domains.register](#)。

以下為 `utils.rb` 中的 `init_domain` 程式碼。

```
# Registers the domain that the workflow will run in.
def init_domain
  domain_name = 'SWFSampleDomain'
  domain = nil
  swf = AWS::SimpleWorkflow.new

  # First, check to see if the domain already exists and is registered.
  swf.domains.registered.each do | d |
    if(d.name == domain_name)
      domain = d
      break
    end
  end

  if domain.nil?
    # Register the domain for one day.
```

```
domain = swf.domains.create(  
  domain_name, 1, { :description => "#{domain_name} domain" })  
end  
  
return domain  
end
```

## 後續步驟

接下來，您要在「[訂購工作流程教學第 2 部分：實作工作流程](#)」中建立工作流程和啟動者程式碼。

## 訂購工作流程教學第 2 部分：實作工作流程

到目前為止，我們的程式碼顯得較通用。因此，我們需要開始實際定義工作流程的運作方式，以及實作工作流程所需的活動。

### 主題

- [設計工作流程](#)
- [設定工作流程程式碼](#)
- [註冊工作流程](#)
- [輪詢決策](#)
- [啟動工作流程執行](#)
- [後續步驟](#)

## 設計工作流程

回想一下，此工作流程最初的構想包含了下列步驟：

1. 從使用者取得訂閱地址 (電子郵件或簡訊)。
2. 建立 SNS 主題並將提供的端點訂閱到主題。
3. 等待使用者確認訂閱。
4. 如果使用者確認，將發佈賀辭到主題。

我們可以將工作流程中的每個步驟視為必須執行的「活動」。「工作流程」負責排程每個活動以於適當的時間執行，以及協調活動之間的資料傳輸。

針對此工作流程，我們將為所有這些步驟建立不同的活動，並以描述性的名稱將活動命名為：

1. `get_contact_activity`
2. `subscribe_topic_activity`
3. `wait_for_confirmation_activity`
4. `send_result_activity`

這些活動將會依序執行，且每個步驟中的資料將用於後續步驟。

我們可以設計應用程式，讓所有程式碼都存在於一個原始檔案，但這樣會與 Amazon SWF 的設計用意相反。後者是針對可跨整個網際網路規模的工作流程所設計，因此讓我們將應用程式至少分為兩個不同的執行檔：

- `swf_sns_workflow.rb` - 包含工作流程和工作流程啟動者。
- `swf_sns_activities.rb` - 包含活動和活動啟動者。

工作流程和活動實作可以在不同的視窗、不同的電腦，甚至在世界上不同的區域中執行。因為 Amazon SWF 會記錄您工作流程和活動的詳細資訊，所以您的工作流程可以協調您活動的排程和資料傳輸，而不論其運行位置為何。

## 設定工作流程程式碼

我們將從建立稱為 `swf_sns_workflow.rb` 的檔案開始。在此檔案中，宣告稱為 `SampleWorkflow` 的類別。以下是類別宣告和其建構函數：`initialize` 方法。

```
require_relative 'utils.rb'

# SampleWorkflow - the main workflow for the SWF/SNS Sample
#
# See the file called `README.md` for a description of what this file does.
class SampleWorkflow

  attr_accessor :name

  def initialize(workflowId)

    # the domain to look for decision tasks in.
    @domain = init_domain
```

```
# the task list is used to poll for decision tasks.
@workflowId = workflowId

# The list of activities to run, in order. These name/version hashes can be
# passed directly to AWS::SimpleWorkflow::DecisionTask#schedule_activity_task.
@activity_list = [
  { :name => 'get_contact_activity', :version => 'v1' },
  { :name => 'subscribe_topic_activity', :version => 'v1' },
  { :name => 'wait_for_confirmation_activity', :version => 'v1' },
  { :name => 'send_result_activity', :version => 'v1' },
].reverse! # reverse the order... we're treating this like a stack.

register_workflow
end
```

如您所見，我們會保留下列類別執行個體資料：

- `domain - utils.rb` 中擷取自 `init_domain` 的網域名稱。
- `workflowId` - 傳入 `initialize` 的任務清單。
- `activity_list` - 活動清單，其具有我們將執行之活動的名稱和版本。

Amazon SWF 可以使用域名稱、活動名稱和活動版本，讓 Amazon SWF 積極識別活動類型，因此這些資料就是我們需要保留的所有資料，讓 Amazon SWF 排程活動。

工作流程「決策者」程式碼將會使用任務清單來輪詢決策任務及排程活動。

在此函數結束時，我們呼叫尚未定義的方法：`register_workflow`。我們接下來將定義此方法。

## 註冊工作流程

若要使用工作流程類型，我們必須先予以註冊。如同活動類型，工作流程類型是以其網域、名稱和版本進行識別。另外，如同網域和活動類型，您無法重新註冊現有的工作流程類型。如果您需要變更有關工作流程類型的任何資訊，則必須提供新版本，基本上即會建立新的類型。

下列 `register_workflow` 程式碼用來擷取我們在先前執行上註冊的現有工作流程類型，或者註冊尚未註冊的工作流程。

```
# Registers the workflow
def register_workflow
  workflow_name = 'swf-sns-workflow'
  @workflow_type = nil
```

```
# a default value...
workflow_version = '1'

# Check to see if this workflow type already exists. If so, use it.
@domain.workflow_types.each do | a |
  if (a.name == workflow_name) && (a.version == workflow_version)
    @workflow_type = a
  end
end

if @workflow_type.nil?
  options = {
    :default_child_policy => :terminate,
    :default_task_start_to_close_timeout => 3600,
    :default_execution_start_to_close_timeout => 24 * 3600 }

  puts "registering workflow: #{workflow_name}, #{workflow_version},
#{options.inspect}"
  @workflow_type = @domain.workflow_types.register(workflow_name, workflow_version,
options)
end

puts "*** registered workflow: #{workflow_name}"
end
```

首先要藉由逐一查看網域的 [workflow\\_types](#) 集合，確認是否已註冊工作流程名稱和版本。如果我們找到相符項目，就會使用已註冊的工作流程類型。

如果我們找不到匹配項，則註冊一個新的 workflow 類型 ( 通過調用 [註冊](#) 在相同的 workflow\_types 集合，我們正在搜索的工作流 ) 名稱為 'swf-sns-Workflow'，版本 '1'，以及以下選項。

```
options = {
  :default_child_policy => :terminate,
  :default_task_start_to_close_timeout => 3600,
  :default_execution_start_to_close_timeout => 24 * 3600 }
```

在註冊期間傳遞的選項會用來設定 workflow 類型的「預設行為」，因此不需要在每次開始新的 workflow 執行時設定這些值。

我們在這裡只會設定一些逾時值：從任務開始到結束所需的最長時間 (一小時)，以及 workflow 執行完成所需的最長時間 (24 小時)。如果超過這兩個時間中的其中一個，則任務或 workflow 將會逾時。

如需逾時值的詳細資訊，請參閱「[Amazon SWF 超時類型](#)」。

## 輪詢決策

每個工作流程執行的核心即為「決策者」。決策者的責任在於管理工作流程本身的執行。決策者會收到並回應「決策任務」，方法是排程新活動、取消並重新啟動活動，或是將工作流程執行的狀態設定為完成、已取消或失敗。

決策者會使用工作流程執行的「任務清單」名稱來接收要回應的決策任務。若要輪詢決策任務，請呼叫 [民意調查](#) 在域的 [決策任務](#) 集合來循環可用的決策任務。您接著可以逐一查看 [new\\_events](#) 集合，來檢查決策任務中的新事件。

傳回的事件為 [AWS::SimpleWorkflow::HistoryEvent](#) 物件，而使用所傳回事件的 [event\\_type](#) 成員即可取得事件類型。如需歷史記錄事件類型的清單和描述，請參 [HistoryEvent](#) 佈中的 Amazon Simple Workflow Service API 參考。

以下是決策任務輪詢器邏輯的開始。工作流程類別中的新方法稱為 `poll_for_decisions`。

```
def poll_for_decisions
  # first, poll for decision tasks...
  @domain.decision_tasks.poll(@workflowId) do | task |
    task.new_events.each do | event |
      case event.event_type
```

我們現在將根據收到的 `event_type` 來分支處理決策者的執行。我們可能會收到的第一個類型為 `WorkflowExecutionStarted`。收到此事件時，表示 Amazon SWF 會對您的決策者發出訊號，通知決策者應開始工作流程執行。首先，對輪詢時收到的任務呼叫 [schedule\\_activity\\_task](#) 以排程第一個活動。

我們會將活動清單中宣告的第一個活動傳遞給它，但因為我們會反轉清單如堆疊一般使用，所以第一個活動會佔用清單上的 `last` 位置。我們定義的「活動」就只是包含名稱和版本編號的對應，但這些足以讓 Amazon SWF 識別排程活動，並假設活動已註冊。

```
when 'WorkflowExecutionStarted'
  # schedule the last activity on the (reversed, remember?) list to
  # begin the workflow.
  puts "*** scheduling activity task: #{@activity_list.last[:name]}"

  task.schedule_activity_task( @activity_list.last,
    { :workflowId => "#{@workflowId}-activities" } )
```

當我們安排活動時，Amazon SWF 會發送活動任務傳入的活動任務清單，並通知排程任務時傳入的動作任務清單，並通知任務開始。我們將在「[訂工作流程教學第 3 部分：實作活動](#)」中處理活動任務，但需注意的是我們並不會在此執行任務。我們只會告知 Amazon SWF 應計劃。

我們需要解決的下一個活動是 ActivityTaskCompleted 事件，而這會在 Amazon SWF 收到來自活動任務的活動完成回應時會發生。

```
when 'ActivityTaskCompleted'
  # we are running the activities in strict sequential order, and
  # using the results of the previous activity as input for the next
  # activity.
  last_activity = @activity_list.pop

  if(@activity_list.empty?)
    puts "!! All activities complete! Sending complete_workflow_execution..."
    task.complete_workflow_execution
    return true;
  else
    # schedule the next activity, passing any results from the
    # previous activity. Results will be received in the activity
    # task.
    puts "*** scheduling activity task: #{@activity_list.last[:name]}"
    if event.attributes.has_key?('result')
      task.schedule_activity_task(
        @activity_list.last,
        { :input => event.attributes[:result],
          :workflowId => "#{@workflowId}-activities" } )
    else
      task.schedule_activity_task(
        @activity_list.last, { :workflowId => "#{@workflowId}-activities" } )
    end
  end
end
```

由於我們會以線性形式執行任務，且一次僅執行一個活動，因此我們可藉機從 `activity_list` 堆疊中跳出已完成的任務。如果此結果為空白清單，則表示我們的工作流程已完成。在這種情況下，我們向 Amazon SWF 通過調用 [完成工作流程執行](#) 在任務上。

如果清單上仍有項目，我們會排程清單上的下一個活動（一樣會在最後一個位置）。不過，這次我們會透過選用的索引鍵，確認前一個活動在完成後是否有任何結果資料傳回，而這會以事件屬性提供給工作流程。result 金鑰。如果活動有結果產生，我們會將之做為 input 選項連同活動任務清單一起傳遞給下一個排程的活動。

透過擷取已完成活動的 `result` 值，及設定已排程活動的 `input` 值，我們可以根據活動的結果，將資料從某個活動傳遞給下一個活動，或使用某個活動的資料來變更決策者的行為。

基於本教學的用途，這兩個事件類型在定義工作流程的行為時最為重要。不過，活動可能產生 `ActivityTaskCompleted` 以外的事件。我們將通過提供演示處理程序代碼來結束我們的決策者代碼 `ActivityTaskTimedOut` 和活動任務失敗事件，以及工作流程執行已完成事件，該事件將在 Amazon SWF 處理 `complete_workflow_execution` 當我們運行的活動用完時，我們進行的調用。

```
when 'ActivityTaskTimedOut'
  puts "!! Failing workflow execution! (timed out activity)"
  task.fail_workflow_execution
  return false

when 'ActivityTaskFailed'
  puts "!! Failing workflow execution! (failed activity)"
  task.fail_workflow_execution
  return false

when 'WorkflowExecutionCompleted'
  puts "## Yesss, workflow execution completed!"
  task.workflow_execution.terminate
  return false
end
end
end
end
```

## 啟動工作流程執行

在產生工作流程要輪詢的任何決策任務以前，我們得先啟動工作流程執行。

要開始工作流程執行，請調用 [啟動執行](#) 註冊的工作流程類型 ( [AWS:: SimpleWorkflow:: Workflow Simple Workflow](#) )。我們將對此定義小型包裝函式，以利用我們在類別建構函數中擷取的 `workflow_type` 執行個體成員。

```
def start_execution
  workflow_execution = @workflow_type.start_execution( {
    :workflowId => @workflowId } )
  poll_for_decisions
end
end
```

工作流程執行之時，決策事件會開始出現在工作流程的任務清單上，而任務清單在 [start\\_execution](#) 中會以工作流程執行選項傳遞。

與註冊工作流程類型時提供的選項不同，傳遞給 `start_execution` 的選項不會視為工作流程類型的一部分。您可以自由地針對個別的工作流程執行變更這些項目，而不需要變更工作流程版本。

因為我們想要在執行檔案時開始執行工作流程，所以請新增可實例化類別的程式碼，然後呼叫剛才定義的 `start_execution` 方法。

```
if __FILE__ == $0
  require 'securerandom'

  # Use a different task list name every time we start a new workflow execution.
  #
  # This avoids issues if our pollers re-start before SWF considers them closed,
  # causing the pollers to get events from previously-run executions.
  workflowId = SecureRandom.uuid

  # Let the user start the activity worker first...

  puts ""
  puts "Amazon SWF Example"
  puts "-----"
  puts ""
  puts "Start the activity worker, preferably in a separate command-line window, with"
  puts "the following command:"
  puts ""
  puts "> ruby swf_sns_activities.rb #{workflowId}-activities"
  puts ""
  puts "You can copy & paste it if you like, just don't copy the '>' character."
  puts ""
  puts "Press return when you're ready..."

  i = gets

  # Now, start the workflow.

  puts "Starting workflow execution."
  sample_workflow = SampleWorkflow.new(workflowId)
  sample_workflow.start_execution
end
```

為了避免任何任務清單命名衝突，我們將使用 `SecureRandom.uuid` 產生可用做任務清單名稱的隨機 UUID，藉以保證每個工作流程執行用的是不同的任務清單名稱。

#### Note

任務清單會用來記錄工作流程執行的事件，因此如果您在相同工作流程類型的多次執行當中使用相同的任務清單，則可能會取得上一次執行所產生的事件，尤其會發生在近乎連續執行工作流程的時候，大多在試用新程式碼或執行測試的情況下。

為了避免必須處理先前執行之成品的問題，我們可以針對每次執行都使用新的任務清單，在我們開始工作流程執行時予以指定。

這裡也有一些程式碼可提供說明給執行它的人員 (可能是您)，以及提供任務清單的「活動」版本。決策者使用此任務清單名稱來排程工作流程的活動，而活動實作將會接聽此任務清單名稱的活動事件，知道何時開始排程的活動，以及提供活動執行的更新。

程式碼也會在啟動工作流程執行「之前」，等待使用者開始執行活動啟動者，因此活動任務開始出現於提供的任務清單時，活動啟動者就已準備好回應。

## 後續步驟

您已實作工作流程。接下來，您會在「[訂工作流程教學第 3 部分：實作活動](#)」中定義活動和啟動者程式碼。

## 訂工作流程教學第 3 部分：實作活動

現在我們將從一些提供活動程式碼之常用功能的基底類別開始，以實作工作流程中的每個活動。

### 主題

- [定義基本活動類型](#)
- [定義 `GetContactActivity`](#)
- [定義 `SubscribeTopicActivity`](#)
- [定義 `WaitForConfirmationActivity`](#)
- [定義 `SendResultActivity`](#)
- [後續步驟](#)

## 定義基本活動類型

設計工作流程時，我們已識別下列活動：

- `get_contact_activity`
- `subscribe_topic_activity`
- `wait_for_confirmation_activity`
- `send_result_activity`

我們將會立即實作所有這些活動。由於我們的活動會共享一些功能，因此我們需要打好基礎，並建立一些活動能共享的常用程式碼。我們稱之為 `BasicActivity`，並於稱為 `basic_activity.rb` 的新檔案中加以定義。

與其他原始檔案相同，我們將包含 `utils.rb` 以存取 `init_domain` 函數，藉此設定範例網域。

```
require_relative 'utils.rb'
```

接下來，我們將宣告基本活動類別，及對每個活動感興趣的一些常用資料。我們將在類別屬性中儲存活動的 [AWS::SimpleWorkflow::ActivityType](#) 執行個體、「名稱」和「結果」。

```
class BasicActivity

  attr_accessor :activity_type
  attr_accessor :name
  attr_accessor :results

end
```

這些屬性訪問類中定義的實例數據 `initialize` 方法，它接受一個活動名稱，以及一個可選版和對應選項，以便在 Amazon SWF 註冊活動時使用。

```
def initialize(name, version = 'v1', options = nil)

  @activity_type = nil
  @name = name
  @results = nil

  # get the domain to use for activity tasks.
  @domain = init_domain

end
```

```
# Check to see if this activity type already exists.
@domain.activity_types.each do | a |
  if (a.name == @name) && (a.version == version)
    @activity_type = a
  end
end

if @activity_type.nil?
  # If no options were specified, use some reasonable defaults.
  if options.nil?
    options = {
      # All timeouts are in seconds.
      :default_task_heartbeat_timeout => 900,
      :default_task_schedule_to_start_timeout => 120,
      :default_task_schedule_to_close_timeout => 3800,
      :default_task_start_to_close_timeout => 3600 }
  end
  @activity_type = @domain.activity_types.register(@name, version, options)
end
end
```

與工作流程類型註冊相同，如果某活動類型已註冊，即可透過查看網域的 [activity\\_types](#) 集合加以擷取。如果找不到活動，則會註冊活動。

另外，與工作流程類型相同，您可以在註冊活動類型時設定與其一起存放的「預設選項」。

基本活動的最後一項要點為一致的執行方式。我們將定義可採用活動任務的 `do_activity` 方法。如下所示，我們可以使用傳入的活動任務，透過其 `input` 執行個體屬性來接收資料。

```
def do_activity(task)
  @results = task.input # may be nil
  return true
end
end
```

這會包裝 `BasicActivity` 類別。現在，我們將用以定義簡單且一致的活動。

## 定義 GetContactActivity

在工作流程執行期間運行的第一個活動是 `get_contact_activity`，以檢索使用者的 Amazon SNS 主題訂資訊。

建立名為 `get_contact_activity.rb`，並要求 `yaml`，我們將使用它來準備傳遞給 Amazon SWF 的字符串，以及 `basic_activity.rb`，我們將使用它作為這個 `GetContact` 活動類別。

```
require 'yaml'
require_relative 'basic_activity.rb'

# **GetContactActivity** provides a prompt for the user to enter contact
# information. When the user successfully enters contact information, the
# activity is complete.
class GetContactActivity < BasicActivity
```

由於我們將活動註冊代碼放在基本活動，`initialize`方法 `GetContact` 活動很簡單。我們只需要使用活動名稱 `get_contact_activity` 來呼叫基底類別建構函數。這就是註冊活動所需的作業。

```
# initialize the activity
def initialize
  super('get_contact_activity')
end
```

我們現在將定義 `do_activity` 方法，提示輸入使用者的電子郵件和 (或) 電話號碼。

```
def do_activity(task)
  puts ""
  puts "Please enter either an email address or SMS message (mobile phone) number
to"
  puts "receive SNS notifications. You can also enter both to use both address
types."
  puts ""
  puts "If you enter a phone number, it must be able to receive SMS messages, and
must"
  puts "be 11 digits (such as 12065550101 to represent the number
1-206-555-0101)."
```

```
  input_confirmed = false
  while !input_confirmed
    puts ""
    print "Email: "
    email = $stdin.gets.strip

    print "Phone: "
    phone = $stdin.gets.strip
```

```
puts ""
if (email == '') && (phone == '')
  print "You provided no subscription information. Quit? (y/n)"
  confirmation = $stdin.gets.strip.downcase
  if confirmation == 'y'
    return false
  end
else
  puts "You entered:"
  puts " email: #{email}"
  puts " phone: #{phone}"
  print "\nIs this correct? (y/n): "
  confirmation = $stdin.gets.strip.downcase
  if confirmation == 'y'
    input_confirmed = true
  end
end
end

# make sure that @results is a single string. YAML makes this easy.
@results = { :email => email, :sms => phone }.to_yaml
return true
end
end
```

`do_activity` 結束時，我們會採用自使用者擷取的電子郵件和電話號碼、放入對應中，然後使用 `to_yaml` 將整個對應轉換為 YAML 字串。這麼做的重要原因在於：您在完成活動時傳遞給 Amazon SWF 的任何結果都必須是僅限字串資料。Ruby 可輕鬆地將物件轉換為 YAML 字串後再重新轉換回物件，此功能最適合此用途。

這是 `get_contact_activity` 實作的尾聲。此資料接下來將用於 `subscribe_topic_activity` 實作。

## 定義 SubscribeTopicActivity

我們現在將探索 Amazon SNS，並建立活動以使用 `get_contact_activity` 讓使用者訂 Amazon SNS 主題。

建立稱為 `subscribe_topic_activity.rb` 的新檔案、新增用於 `get_contact_activity` 的相同需求、宣告類別，並提供其 `initialize` 方法。

```
require 'yaml'
```

```
require_relative 'basic_activity.rb'

# **SubscribeTopicActivity** sends an SMS / email message to the user, asking for
# confirmation. When this action has been taken, the activity is complete.
class SubscribeTopicActivity < BasicActivity

  def initialize
    super('subscribe_topic_activity')
  end
end
```

既然我們已經建立了用於設定和註冊活動的代碼，我們將新增一些代碼以建立 Amazon SNS 主題。為此，我們將使用 [AWS::SNS::Client](#) 物件的 [create\\_topic](#) 方法。

新增 `create_topic` 方法傳入至類別，以採用傳入的 Amazon SNS 客戶端對象。

```
def create_topic(sns_client)
  topic_arn = sns_client.create_topic(:name => 'SWF_Sample_Topic')[[:topic_arn]]

  if topic_arn != nil
    # For an SMS notification, setting `DisplayName` is *required*. Note that
    # only the *first 10 characters* of the DisplayName will be shown on the
    # SMS message sent to the user, so choose your DisplayName wisely!
    sns_client.set_topic_attributes( {
      :topic_arn => topic_arn,
      :attribute_name => 'DisplayName',
      :attribute_value => 'SWFSample' } )
  else
    @results = {
      :reason => "Couldn't create SNS topic", :detail => "" }.to_yaml
    return nil
  end

  return topic_arn
end
```

有了主題的 Amazon Resource Name (ARN) 之後，就能將其搭配 Amazon SNS 客戶端的 [設置主題屬性](#) 方法來設置主題的 `DisplayName`，此為使用 Amazon SNS 傳送簡訊的必要項目。

最後，我們將定義 `do_activity` 方法。我們將從收集在排程活動時透過 `input` 選項所傳遞的任何資料開始。如前所述，這必須以使用 `to_yaml` 所建立的字串傳遞。進行擷取時，我們會使用 `YAML.load` 將資料轉換為 Ruby 物件。

以下是 `do_activity` 的開始，內容為擷取輸入資料。

```
def do_activity(task)
  activity_data = {
    :topic_arn => nil,
    :email => { :endpoint => nil, :subscription_arn => nil },
    :sms => { :endpoint => nil, :subscription_arn => nil },
  }

  if task.input != nil
    input = YAML.load(task.input)
    activity_data[:email][:endpoint] = input[:email]
    activity_data[:sms][:endpoint] = input[:sms]
  else
    @results = { :reason => "Didn't receive any input!", :detail => "" }.to_yaml
    puts(" #{@results.inspect}")
    return false
  end

  # Create an SNS client. This is used to interact with the service. Set the
  # region to $SMS_REGION, which is a region that supports SMS notifications
  # (defined in the file `utils.rb`).
  sns_client = AWS::SNS::Client.new(
    :config => AWS.config.with(:region => $SMS_REGION))
end
```

如果我們未收到任何輸入，即無作業可進行，因此會直接讓活動失敗。

假設沒有任何問題，我們會繼續填入do\_activity方法，Amazon SNS 取具有AWS SDK for Ruby，並將其傳遞給我們的create\_topic方法來建立 Amazon SNS 主題。

```
# Create the topic and get the ARN
activity_data[:topic_arn] = create_topic(sns_client)

if activity_data[:topic_arn].nil?
  return false
end
```

以下是一些值得注意的事項：

- 我們使用[AWS.config.with](#)設定 Amazon SNS 用戶端的區域。我們想要傳送簡訊，因此會使用 `utils.rb` 中所宣告的啟用簡訊功能的區域。
- 我們將主題的 ARN 儲存至 `activity_data` 對應。這是會傳遞給工作流程中「下一個」活動的部分資料。

最後，此活動會使用傳入的端點（電子郵件和簡訊）讓使用者訂 Amazon SNS 主題。我們不需要使用者進入「兩個」端點，但至少需要進入其中一個。

```
# Subscribe the user to the topic, using either or both endpoints.
[:email, :sms].each do | x |
  ep = activity_data[x][:endpoint]
  # don't try to subscribe an empty endpoint
  if (ep != nil && ep != "")
    response = sns_client.subscribe( {
      :topic_arn => activity_data[:topic_arn],
      :protocol => x.to_s, :endpoint => ep } )
    activity_data[x][:subscription_arn] = response[:subscription_arn]
  end
end
```

[AWS::SNS::Client.subscribe](#) 採用主題 ARN 和「協定」（其將巧妙地喬裝為對應端點的 `activity_data` 對應索引鍵）。

最後，我們會以 YAML 格式重新封裝用於下一個活動的資訊，以便我們將之送回 Amazon SWF。

```
# if at least one subscription arn is set, consider this a success.
if (activity_data[:email][:subscription_arn] != nil) or (activity_data[:sms]
[:subscription_arn] != nil)
  @results = activity_data.to_yaml
else
  @results = { :reason => "Couldn't subscribe to SNS topic", :detail =>
"" }.to_yaml
  puts(" #{@results.inspect}")
  return false
end
return true
end
```

至此完成了 `subscribe_topic_activity` 的實作。接下來，我們將定義 `wait_for_confirmation_activity`。

## 定義 WaitForConfirmationActivity

使用者在訂 Amazon SNS 主題之後，仍需要確認訂請求。在此情況下，我們將等待使用者透過電子郵件或簡訊確認。

等待使用者確認訂閱的活動稱為 `wait_for_confirmation_activity`，我們將在此予以定義。若要開始，請建立稱為 `wait_for_confirmation_activity.rb` 的新檔案，並如同設定先前活動的方式加以設定。

```
require 'yaml'
require_relative 'basic_activity.rb'

# **WaitForConfirmationActivity** waits for the user to confirm the SNS
# subscription. When this action has been taken, the activity is complete. It
# might also time out...
class WaitForConfirmationActivity < BasicActivity

  # Initialize the class
  def initialize
    super('wait_for_confirmation_activity')
  end
end
```

接下來，我們將開始定義 `do_activity` 方法，並將任何輸入資料擷取到稱為 `subscription_data` 的本機變數。

```
def do_activity(task)
  if task.input.nil?
    @results = { :reason => "Didn't receive any input!", :detail => "" }.to_yaml
    return false
  end

  subscription_data = YAML.load(task.input)
```

既然已經有了主題 ARN，即可建立新的 [AWS::SNS::Topic](#) 執行個體並對其傳遞 ARN 以擷取主題。

```
topic = AWS::SNS::Topic.new(subscription_data[:topic_arn])

if topic.nil?
  @results = {
    :reason => "Couldn't get SWF topic ARN",
    :detail => "Topic ARN: #{topic.arn}" }.to_yaml
  return false
end
```

現在，我們將檢查主題，確認使用者是否已使用其中一個端點確認訂閱。我們只需要確認該端點將活動視為成功即可。

Amazon SNS 主題會維護訂閱，我們可以通過檢查訂的 ARN 是否設定為以外的任何項目，就能檢查使用者是否已確認特定訂 PendingConfirmation。

```
# loop until we get some indication that a subscription was confirmed.
subscription_confirmed = false
while(!subscription_confirmed)
  topic.subscriptions.each do | sub |
    if subscription_data[sub.protocol.to_sym][:endpoint] == sub.endpoint
      # this is one of the endpoints we're interested in. Is it subscribed?
      if sub.arn != 'PendingConfirmation'
        subscription_data[sub.protocol.to_sym][:subscription_arn] = sub.arn
        puts "Topic subscription confirmed for (#{sub.protocol}:
#{sub.endpoint})"
        @results = subscription_data.to_yaml
        return true
      else
        puts "Topic subscription still pending for (#{sub.protocol}:
#{sub.endpoint})"
      end
    end
  end
end
```

如果我們取得訂閱的 ARN，則會將之儲存至活動結果資料、轉換為 YAML，然後從 do\_activity 傳回 true，指出活動已順利完成。

因為等待確認訂閱可能需要一些時間，所以我們偶爾會對活動任務呼叫 record\_heartbeat。此為 Amazon SWF 的訊號，即活動仍在處理，也可以用來提供活動進度的更新 (如果您執行的是類似處理檔案等可報告進度的作業)。

```
task.record_heartbeat!(
  { :details => "#{topic.num_subscriptions_confirmed} confirmed,
#{topic.num_subscriptions_pending} pending" })
# sleep a bit.
sleep(4.0)
end
```

這會結束 while 迴圈。如果我們以某種方式離開 while 迴圈而未成功，則會報告失敗，並完成 do\_activity 方法。

```
if (subscription_confirmed == false)
  @results = {
    :reason => "No subscriptions could be confirmed",
```

```

      :detail => "#{topic.num_subscriptions_confirmed} confirmed,
#{topic.num_subscriptions_pending} pending" }.to_yaml
    return false
  end
end
end
end

```

至此完成了 `wait_for_confirmation_activity` 的實作。我們只需要再定義一個活動：`send_result_activity`。

## 定義 `SendResultActivity`

如果工作流程進展至此，則我們已成功讓使用者訂 Amazon SNS 主題，且使用者已確認訂。

最後一個活動 `send_result_activity` 會使用使用者所訂閱的主題及使用者用來確認訂閱的端點，將成功訂閱主題的確認傳送給使用者。

請建立稱為 `send_result_activity.rb` 的新檔案，並如同設定先前所有活動的方式加以設定。

```

require 'yaml'
require_relative 'basic_activity.rb'

# **SendResultActivity** sends the result of the activity to the screen, and, if
# the user successfully registered using SNS, to the user using the SNS contact
# information collected.
class SendResultActivity < BasicActivity

  def initialize
    super('send_result_activity')
  end
end

```

`do_activity` 方法一開始也同樣會從工作流程取得輸入資料，並從 YAML 轉換之，然後使用主題 ARN 建立 [AWS::SNS::Topic](#) 執行個體。

```

def do_activity(task)
  if task.input.nil?
    @results = { :reason => "Didn't receive any input!", :detail => "" }
    return false
  end

  input = YAML.load(task.input)

```

```
# get the topic, so we publish a message to it.
topic = AWS::SNS::Topic.new(input[:topic_arn])

if topic.nil?
  @results = {
    :reason => "Couldn't get SWF topic",
    :detail => "Topic ARN: #{topic.arn}" }
  return false
end
```

有了主題之後，就要將訊息[發佈](#)到該主題 (也會反應至螢幕)。

```
@results = "Thanks, you've successfully confirmed registration, and your
workflow is complete!"

# send the message via SNS, and also print it on the screen.
topic.publish(@results)
puts(@results)

return true
end
end
```

發佈至 Amazon SNS 主題時，會將您提供的訊息傳送給所有已訂和已確認的主題的端點。因此，如果使用者透過電子郵件和簡訊號碼「兩者」進行確認，則會收到兩則確認訊息，即一個端點一則。

## 後續步驟

至此完成了 `send_result_activity` 的實作。在接下來的「[訂工作流程教學第 4 部分：實作活動任務輪詢器](#)」中，您可以在處理活動任務的活動應用程式中將所有這些活動都繫結在一起，並且能啟動活動予以回應。

## 訂工作流程教學第 4 部分：實作活動任務輪詢器

在 Amazon SWF 中，運作中之工作流程執行的活動任務會出現在活動任務列表，而此功能是在您排定工作流程中的活動時所提供。

我們將實作基本活動輪詢器來處理工作流程的這些任務，並在 Amazon SWF 在活動任務清單上放置任務來啟動活動時，用以啟動活動。

若要開始，請建立稱為 `swf_sns_activities.rb` 的新檔案。我們將會用來：

- 實例化我們已建立的活動類別。
- 向註冊 Amazon SWF 的每個活動。
- 會輪詢活動，並在每個活動的名稱出現在活動任務清單時呼叫每個活動的 `do_activity`。

在 `swf_sns_activities.rb` 中，新增下列陳述式以要求我們定義的每個活動類別。

```
require_relative 'get_contact_activity.rb'  
require_relative 'subscribe_topic_activity.rb'  
require_relative 'wait_for_confirmation_activity.rb'  
require_relative 'send_result_activity.rb'
```

現在，我們將建立類別，並提供某個初始化程式碼。

```
class ActivitiesPoller  
  
  def initialize(domain, workflowId)  
    @domain = domain  
    @workflowId = workflowId  
    @activities = {}  
  
    # These are the activities we'll run  
    activity_list = [  
      GetContactActivity,  
      SubscribeTopicActivity,  
      WaitForConfirmationActivity,  
      SendResultActivity ]  
  
    activity_list.each do | activity_class |  
      activity_obj = activity_class.new  
      puts "*** initialized and registered activity: #{activity_obj.name}"  
      # add it to the hash  
      @activities[activity_obj.name.to_sym] = activity_obj  
    end  
  end  
end
```

除了儲存傳入的「網域」和「任務清單」之外，此程式碼還會實例化我們建立的每個活動類別。因為每個類都註冊其關聯的活動（請參閱 `basic_activity.rb`），則建議您讓 Amazon SWF 瞭解我們將執行的所有活動。

對於每個實例化的活動，我們使用活動名稱（例如 `get_contact_activity`）做為索引鍵，將之存放於地圖上，因此可以使用接下來定義的活動輪詢器程式碼輕鬆地查詢這些活動。

建立名為的新方法 `poll_for_activities` 並呼叫 [民意調查](#) 在 [活動任務](#) 以獲取活動任務。

```
def poll_for_activities
  @domain.activity_tasks.poll(@workflowId) do | task |
    activity_name = task.activity_type.name
```

從任務的 [activity\\_type](#) 成員可以取得活動名稱。接下來，我們將使用與此任務相關聯的活動名稱來查詢要執行 `do_activity` 的類別，並將任務傳遞給它 (包含應該傳送給活動的任何輸入資料)。

```
# find the task on the activities list, and run it.
if @activities.key?(activity_name.to_sym)
  activity = @activities[activity_name.to_sym]
  puts "*** Starting activity task: #{activity_name}"
  if activity.do_activity(task)
    puts "++ Activity task completed: #{activity_name}"
    task.complete!({ :result => activity.results })
    # if this is the final activity, stop polling.
    if activity_name == 'send_result_activity'
      return true
    end
  else
    puts "-- Activity task failed: #{activity_name}"
    task.fail!(
      { :reason => activity.results[:reason],
        :details => activity.results[:detail] } )
  end
else
  puts "couldn't find key in @activities list: #{activity_name}"
  puts "contents: #{@activities.keys}"
end
end
end
end
```

程式碼只需等待 `do_activity` 完成，然後根據傳回碼，對任務呼叫 [complete!](#) 或 [fail!](#)。

#### Note

此程式碼會在啟動最後一個活動之後結束輪詢器，因為它已完成其任務，並已啟動所有活動。在您的 Amazon SWF 程式碼中，如果您的活動可能會重新執行，則建議您讓活動輪詢器無限期執行。

這是 `ActivitiesPoller` 類別的程式碼結尾，但我們將在檔案結尾再新增一些程式碼，以允許使用者從命令列執行它。

```
if __FILE__ == $0
  if ARGV.count < 1
    puts "You must supply a task-list name to use!"
    exit
  end
  poller = ActivitiesPoller.new(init_domain, ARGV[0])
  poller.poll_for_activities
  puts "All done!"
end
```

如果使用者從命令列執行檔案 (將活動任務清單做為第一個引數傳遞給它)，則此程式碼將會實例化輪詢器類別，並開始輪詢活動。輪詢器完成之後 (啟動最後一個活動之後)，就會列印訊息並結束。

活動輪詢器的運作就是如此。您只需要執行程式碼，並在「[訂工作流程教學：執行工作流程](#)」中查看其運作方式。

## 訂工作流程教學：執行工作流程

現在您已完成實作工作流程、活動，以及工作流程與活動輪詢器，所以您可隨時執行工作流程。

如果您尚未提供，請先執行這項動作，請先執行這項動作。AWS訪問密鑰`aws-config.txt`文件，如[設定AWS工作階段](#)本教學第 1 部分中提供。

現在請前往您的命令列，變更到教學來源檔案所在的目錄。您應有下列檔案：

```
.
|-- aws-config.txt
|-- basic_activity.rb
|-- get_contact_activity.rb
|-- send_result_activity.rb
|-- subscribe_topic_activity.rb
|-- swf_sns_activities.rb
|-- swf_sns_workflow.rb
|-- utils.rb
`-- wait_for_confirmation_activity.rb
```

現在，請使用以下命令啟動工作流程。

```
ruby swf_sns_workflow.rb
```

這會開始工作流程，並應該會使用您可複製貼入不同命令列視窗的程式碼來列印訊息 (或甚至是其他電腦，如果您已將教學來源檔案複製到該電腦)。

```
Amazon SWF Example
```

```
-----
```

```
Start the activity worker, preferably in a separate command-line window, with the following command:
```

```
> ruby swf_sns_activities.rb 87097e76-7c0c-41c7-817b-92527bb0ea85-activities
```

```
You can copy & paste it if you like, just don't copy the '>' character.
```

```
Press return when you're ready...
```

工作流程程式碼會耐心等待您在不同的視窗中啟動活動輪詢器。

開啟新的命令列視窗，再次變更到來源檔案所在的目錄，然後使用 `swf_sns_workflow.rb` 檔案提供的命令來啟動活動輪詢器。例如，如果您收到前述輸出，您可以輸入 (或貼上) 以下命令。

```
ruby swf_sns_activities.rb 87097e76-7c0c-41c7-817b-92527bb0ea85-activities
```

一旦您開始執行活動輪詢器，它就會開始輸出活動註冊的相關資訊。

```
** initialized and registered activity: get_contact_activity
** initialized and registered activity: subscribe_topic_activity
** initialized and registered activity: wait_for_confirmation_activity
** initialized and registered activity: send_result_activity
```

您現在可以返回原來的命令列視窗，按 `return` (傳回) 開始您的工作流程執行。它會註冊工作流程及排程第一項活動。

```
Starting workflow execution.
** registered workflow: swf-sns-workflow
** scheduling activity task: get_contact_activity
```

回到您的活動輪詢器執行所在的另一個視窗。第一個執行的活動結果隨即顯示，提示您輸入您的電子郵件或簡訊電話號碼。輸入這兩段資料的任一段或全部輸入，然後確認您輸入的文字。

```
activity task received: <AWS::SimpleWorkflow::ActivityTask>
```

```
** Starting activity task: get_contact_activity
```

```
Please enter either an email address or SMS message (mobile phone) number to receive Amazon SNS notifications. You can also enter both to use both address types.
```

```
If you enter a phone number, it must be able to receive SMS messages, and must be 11 digits (such as 12065550101 to represent the number 1-206-555-0101).
```

```
Email: me@example.com
```

```
Phone: 12065550101
```

```
You entered:
```

```
  email: me@example.com
```

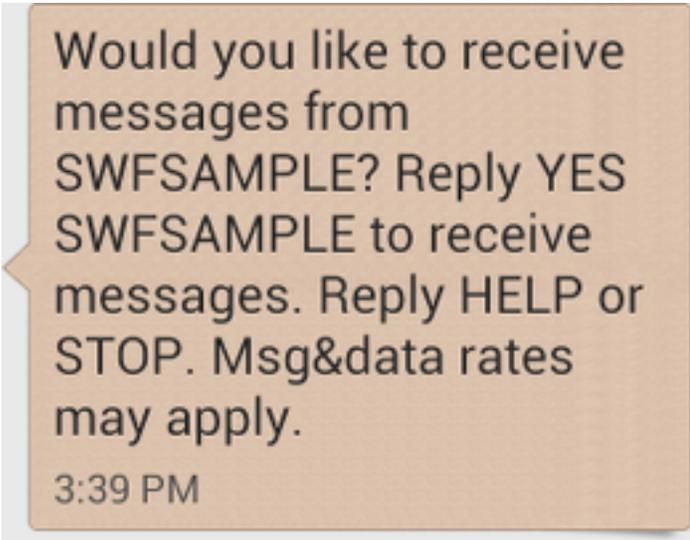
```
  phone: 12065550101
```

```
Is this correct? (y/n): y
```

#### Note

提供的電話號碼為虛構，僅供示範之用。請在這裡使用您自己的電話號碼和電子郵件地址。

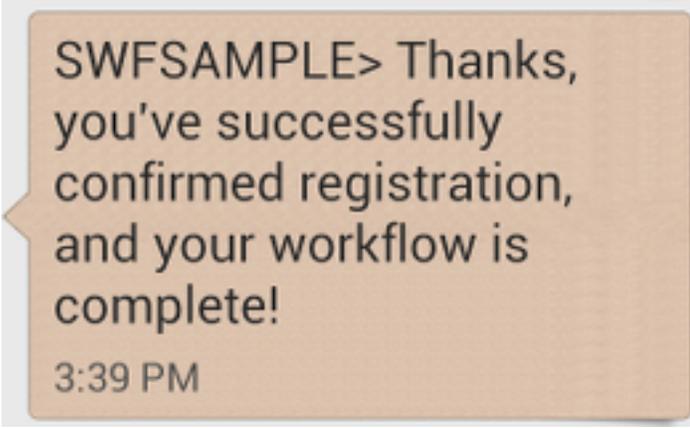
輸入此資訊之後，您應該會立即收到的電子郵件或簡 Amazon SNS，要求您確認您的主題訂。如曾輸入簡訊號碼，您會在電話中看到類似以下的內容。



Would you like to receive messages from SWFSAMPLE? Reply YES SWFSAMPLE to receive messages. Reply HELP or STOP. Msg&data rates may apply.

3:39 PM

如果您使用 YES 回覆此訊息，您會收到我們在 `send_result_activity` 中提供的回應。



```
SWFSAMPLE> Thanks,  
you've successfully  
confirmed registration,  
and your workflow is  
complete!
```

```
3:39 PM
```

當所有一切發生時，您是否看到您的命令列視窗中發生了什麼？工作流程和活動輪詢器都在努力工作。

以下為工作流程輪詢器的輸出。

```
** scheduling activity task: subscribe_topic_activity  
** scheduling activity task: wait_for_confirmation_activity  
** scheduling activity task: send_result_activity  
!! All activities complete! Sending complete_workflow_execution...
```

以下是活動輪詢器的輸出，它會同時發生在另一個命令列視窗中。

```
++ Activity task completed: get_contact_activity  
** Starting activity task: subscribe_topic_activity  
++ Activity task completed: subscribe_topic_activity  
** Starting activity task: wait_for_confirmation_activity  
Topic subscription still pending for (email: me@example.com)  
Topic subscription confirmed for (sms: 12065550101)  
++ Activity task completed: wait_for_confirmation_activity  
** Starting activity task: send_result_activity  
Thanks, you've successfully confirmed registration, and your workflow is complete!  
++ Activity task completed: send_result_activity  
All done!
```

恭喜您完成您的工作流程和此教學！

建議您再次重新執行工作流程，查看逾時如何運作，或輸入不同的資料。只要記住，一旦您訂閱主題，「若未取消訂閱就會一直是訂閱狀態」。先重新執行工作流程再取消訂閱主題可能會造成自動成功，因為 `wait_for_confirmation_activity` 會看到您已確認訂閱。

## 取消訂 Amazon SNS 主題

- 以負值 (傳送 STOP) 回應簡訊。
- 選擇您電子郵件中收到的取消訂閱連結。

您現已就緒可再次重新訂閱主題。

## 接下來做些什麼？

本教學包含許多基礎，但有關AWS SDK for Ruby、Amazon SWF 或 Amazon SNS。如需詳細資訊及更多範例，請參閱各官方文件：

- [AWS SDK for Ruby 文件](#)
- [Amazon Simple Notification Service 文檔](#)
- [Amazon Simple Workflow Service 文檔](#)



## New console

使用主控台註冊 Amazon SWF 網域

1. 如果尚未註冊網域，請在主窗格的中央選擇 [註冊網域]。
2. 在 [註冊網域] 頁面上，輸入網域的 [名稱]、[保留期間]，以及選擇性地輸入 [說明]。

這些值會對應到 Register Domain 動作中名稱相似的參數。

### Domain details

**Name**

Up to 256 characters. Must not contain a : (colon), / (slash), | (vertical bar), or any control characters.

**Retention period**

Duration for which the record of workflow executions should be kept. Executions will not be available after the specified retention period elapses.

 days  
Must be 0-90 days.

**Description - optional**

Up to 1024 characters.

► **Tags - optional** [Learn more](#) 

**Cancel** **Register domain**

3. 選擇註冊網域。
4. 註冊網域後，主控台會顯示一個標題為您的網域名稱的頁面。您也可以在此頁面上執行下列動作：
  - [在您的網域中註冊工作流程](#)
  - [在您的網域中註冊活動](#)
  - [開始執行工作流程](#)

- [管理您的工作流程執行](#)
- [棄用或重新評估網域](#)

## Old console

使用主控台註冊 Amazon SWF 網域

1. 如果尚未註冊網域，請在主窗格的中央選擇 [註冊網域]。

如已註冊至少一個網域，請在儀表板檢視選擇 Manage Domains (管理網域) 按鈕，然後在 Manage Domains (管理網域) 對話方塊中，選擇 Register New (註冊新網域)。

2. 在 Register Domain (註冊網域) 對話方塊中，為 Name (名稱)、Retention Period (保留期) 和 Description (描述) 輸入內容。這些值會對應到 RegisterDomain 動作中名稱相似的參數。

**Register Domain** Cancel X

Provide the details of your new Domain below, then click Register

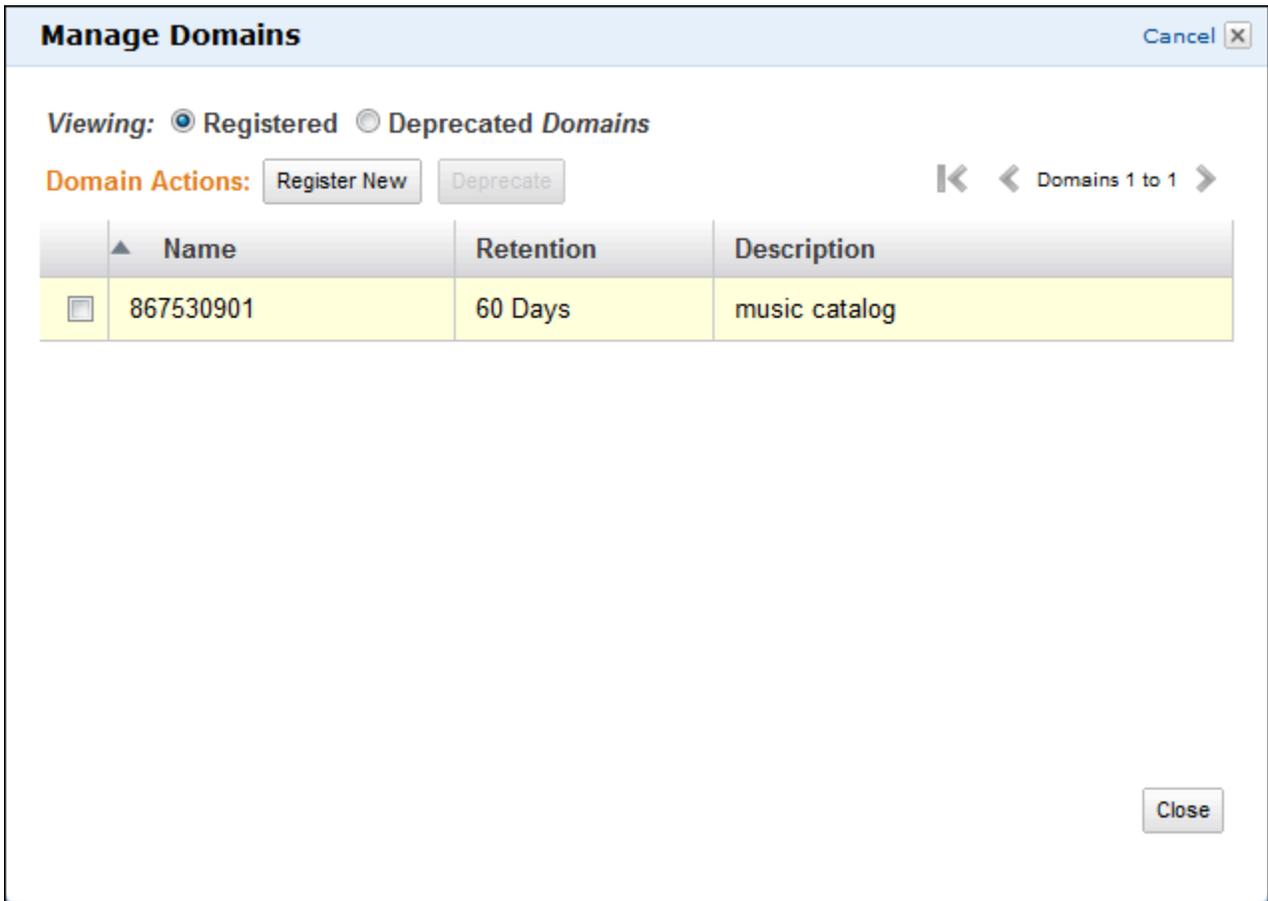
**Name:\***

**Workflow Execution Retention Period:\***  Days

**Description:**

Register

3. 選擇 Register (註冊)。
4. 註冊網域後，主控台會顯示 Manage Domains (管理網域) 對話方塊。



## 註冊工作流程類型

您可以使用 Amazon SWF 主控台註冊工作流程類型。您必須至少註冊一個網域，才能註冊工作流程。

### New console

使用主控台註冊 Amazon SWF 工作流程

1. 開啟您要在其中註冊工作流程的網域。
2. 選擇登錄，接著選擇註冊工作流程。
3. 在「」註冊工作流程頁面中，輸入工作流程名稱和工作流程版本。或者，您也可以選擇指定[預設工作清單](#)將用於排程決策任務，以執行此工作流程。

## Register workflow [Learn more](#)

A workflow is a set of activities that carry out an objective and contains the logic that coordinates the activities. Workflows coordinate and manage the execution of activities that can be run asynchronously across multiple computing devices and feature both sequential and parallel processing methods.

### Workflow details

#### Domain

#### Workflow name

Up to 256 characters. Must not contain a : (colon), / (slash), | (vertical bar), or any control characters.

#### Workflow version

Up to 64 characters. Must not contain a : (colon), / (slash), | (vertical bar), or any control characters.

#### Description - *optional*

Up to 1024 characters.

#### Default task list - *optional* [Learn more](#)

Used for decision tasks generated for executions of this workflow.

Up to 256 characters. Must not contain a : (colon), / (slash), | (vertical bar), or any control characters.

#### 4. (選用) 選擇進階選項為工作流程指定下列詳細資訊：

- [預設工作優先順序](#)— 要指派給工作流程的預設任務優先順序。
- [預設工作優先順序](#)— 要指派給工作流程的預設任務優先順序。
- [預設執行開始關閉逾時](#)— 此工作流程執行的預設最長持續時間。
- [默認任務開始關閉超時](#)— 該工作流程的決策任務的默認最長持續時間。
- [預設子政策](#)— 用於子工作流程執行的預設原則。
- [預設 Lambda 角色](#)— 附加到此工作流程的預設 IAM 角色。

▼ **Advanced options**

**Default Task priority** [Learn more](#) 

Specify the priority to schedule decision tasks generated for this execution. A higher number indicates higher priority. Default value is 0.

  
Must be from -2147483648 to 2147483647

**Default Execution start to close timeout** [Learn more](#) 

Executions of this workflow will be timed out if not completed within this duration. Default value is 3600 seconds

  ▼  
Up to 8 characters. Use NONE for unlimited duration.

**Default Task start to close timeout** [Learn more](#) 

Decision task will be timed out if not completed within this duration. Default value is 600 seconds

  ▼  
Up to 8 characters. Use NONE for unlimited duration.

**Default Child policy** [Learn more](#) 

Determines a policy for child workflow executions if this execution is terminated.

Terminate  
 Request cancel  
 Abandon

**Default Lambda role - optional**

It gives Amazon SWF the ability to invoke any Lambda function associated with your account in place of, or alongside traditional activities.

 ▼

[Cancel](#) [Register workflow](#)

## 5. 選擇註冊工作流程。

### Old console

#### 使用主控台註冊 Amazon SWF 工作流程

1. 在 Amazon Simple Workflow Service 儀表板，下快速連結，選擇註冊新的工作流程型態。

在 Workflow Details (工作流程詳細資訊) 對話方塊中，輸入以下資訊。

- 網域
- Workflow Name (工作流程名稱)

- Workflow Version (工作流程版本)
- 預設任務清單
- Default Execution Run Time (預設執行的執行時間)
- Default Task Run Time (預設任務執行時間)

有星號 ((\*)) 標記的欄位為必要項。

**Register New Workflow** Cancel X

WORKFLOW DETAILS    ADDITIONAL OPTIONS    REVIEW

Provide the details of your new Workflow below:

**Domain\*:** 867530901

**Workflow Name\*:** customerOrderWorkflow

**Workflow Version\*:** 1.0

**Default Task List\*:** mainTaskList

**Default Execution Run Time\*:** 3600 seconds ▼

**Default Task Run Time\*:** 600 seconds ▼

Continue

選擇 Continue (繼續)。

2. 在 Additional Options (其他選項) 對話方塊中，輸入 Description (描述)，然後指定 Default Child Policy (預設子政策)。選擇 Review (檢閱)。

The screenshot shows a dialog box titled "Register New Workflow" with a "Cancel" button in the top right corner. A progress bar at the top indicates three steps: "WORKFLOW DETAILS", "ADDITIONAL OPTIONS" (which is currently selected), and "REVIEW". Below the progress bar, the text reads "Provide additional options for your new Workflow below:". There is a text input field labeled "Description:" containing the text "Handle customer orders". Below this is a dropdown menu labeled "Default Child Policy:" with "Terminate" selected. At the bottom left is a "Back" button with a left-pointing arrow, and at the bottom right is a "Review" button.

3. 在 Review (檢閱) 對話方塊中，檢閱您在上一個對話方塊中輸入的資訊。若資訊無誤，請選擇 Register Workflow (註冊工作流程)。否則，請選擇 Back (上一步) 變更資訊。

**Register New Workflow** Cancel X

WORKFLOW DETAILS    ADDITIONAL OPTIONS    **REVIEW**

Please review the information below, then click **Register Workflow**

**Domain:** 867530901

**Workflow Name:** customerOrderWorkflow

**Workflow Version:** 1.0

**Default Task List:** mainTaskList

**Default Child Policy:** TERMINATE

**Default Execution Run Time:** 1 hour

**Default Task Run Time:** 10 minutes

**Description:** Handle customer orders

[Back](#) Register Workflow

4. 選擇註冊工作流程。

## 註冊活動類型

您可以使用 Amazon SWF 主控台註冊活動類型。您必須至少註冊一個網域，才能註冊活動。

### New console

1. 開啟您要在其中註冊活動的網域。
2. 選擇登錄，接著選擇登錄活動。
3. 在「」登錄活動頁面中，輸入[活動名稱](#)和[活動版本](#)。或者，您也可以選擇指定[預設工作清單](#)將用於安排此活動的任務。

## Register activity [Learn more](#)

Activities are tasks which you want your workflow to coordinate and execute (for example: verify customer's order, charge credit card etc.). The order in which activities are performed is determined by the workflow's coordination logic.

### Activity details

#### Domain

#### Activity name

Up to 256 characters. Must not contain a : (colon), / (slash), | (vertical bar), or any control characters.

#### Activity version

Up to 64 characters. Must not contain a : (colon), / (slash), | (vertical bar), or any control characters.

#### Description - *optional*

*Enter description for this activity*

Up to 1024 characters.

#### Default task list - *optional* [Learn more](#)

Used for tasks of this activity

Up to 256 characters. Must not contain a : (colon), / (slash), | (vertical bar), or any control characters.

#### 4. (選用) 選擇進階選項為您的活動指定下列詳細資訊：

- [預設工作優先順序](#)— 要指派給活動的預設任務優先順序。
- [啟動逾時的預設作業排程](#)— 此活動的任務在指派給 Worker 之前可以等待的預設最長持續時間。
- [預設工作開始關閉逾時](#)— Worker 處理此活動任務所能採取的預設最長持續時間。
- [關閉逾時的預設作業排程](#)— 此活動任務的預設最長持續時間。
- [預設工作活動訊號逾時](#)— 處理此類型任務的 Worker 必須在此時間前透過呼叫來報告進度的預設最長時間 [RecordActivityTaskHeartbeat](#)。

**▼ Advanced options****Default Task priority** [Learn more](#)

Specify the priority to schedule activity tasks generated for this execution. A higher number indicates higher priority. Default value is 0.

Must be from -2147483648 to 2147483647

**Default task schedule to start timeout** [Learn more](#)

Tasks of this activity will be timed out if not started within this duration. Default value is 1800 seconds.

Up to 8 characters. Use NONE for unlimited duration.

**Default Task start to close timeout** [Learn more](#)

Tasks of this activity will be timed out if not completed within this duration. Default value is 600 seconds.

Up to 8 characters. Use NONE for unlimited duration.

**Default task schedule to close timeout** [Learn more](#)

Tasks of this activity will be timed out if not started and completed within this duration. Default value is 5400 seconds.

Up to 8 characters. Use NONE for unlimited duration.

**Default task heartbeat timeout** [Learn more](#)

Maximum time before which a worker processing a task must report progress. Default value is 1800 seconds.

Up to 8 characters. Use NONE for unlimited duration.

[Cancel](#)[Register activity](#)

## 5. 選擇登錄活動。

### Old console

若要使用主控台註冊 Amazon SWF 活動類型

1. 在 Amazon Simple Workflow Service，下快速連結，選擇註冊新活動類型。

在 Activity Details (活動詳細資訊) 對話方塊中輸入以下資訊。

- 網域
- 活動名稱

- 活動版本
- 預設任務清單
- 啟動逾時的任務排程
- 結束逾時的任務排程

有星號 ((\*)) 標記的欄位為必要項。

**Register New Activity** Cancel X

ACTIVITY DETAILS    ADDITIONAL OPTIONS    REVIEW

Provide the details of your new Activity below:

**Domain\***: 867530901

**Activity Name\***: activityVerify

**Activity Version\***: 1.0

**Task List**: mainTaskList

**Task Schedule to Start Timeout**: 5 minutes

**Task Start to Close Timeout**: 15 minutes

Continue

選擇 Continue (繼續)。

2. 在 Additional Options (其他選項) 對話方塊中輸入 Description (描述)，並指定 Heartbeat Timeout (活動訊號逾時) 和 Task Schedule to Close Timeout (結束逾時的任務排程)。選擇 Review (檢閱)。

The screenshot shows the 'Register New Activity' dialog box with a progress bar at the top. The progress bar has three steps: 'ACTIVITY DETAILS', 'ADDITIONAL OPTIONS' (which is currently selected and highlighted with an orange circle), and 'REVIEW'. Below the progress bar, the text reads 'Provide additional options for your new Activity below:'. There is a large text area labeled 'Description:' containing the text 'Verify the customer credit'. Below this are two input fields: 'Heartbeat Timeout:' with a value of '2' and a dropdown menu set to 'minutes', and 'Task Schedule to Close Timeout:' with a value of '15' and a dropdown menu set to 'minutes'. At the bottom left is a blue '< Back' link, and at the bottom right is a grey 'Review' button. A 'Cancel X' button is located in the top right corner of the dialog box.

3. 在 Review (檢閱) 對話方塊中，檢閱您在上一個對話方塊中輸入的資訊。若資訊無誤，請選擇 Register Activity (註冊活動)。否則，請選擇 Back (上一步) 變更資訊。

## Register New Activity Cancel X

ACTIVITY DETAILS    ADDITIONAL OPTIONS    REVIEW

Please review the information below, and then click **Register**

**Domain:** 867530901  
**Activity Name:** activityVerify  
**Activity Version:** 1.0  
**Task List:** mainTaskList  
**Task Schedule to Close Timeout:** 15 minutes  
**Task Schedule to Start Timeout:** 5 minutes  
**Task Start to Close Timeout:** 10 minutes  
**Task Heartbeat Timeout:** 2 minutes

**Description:**

[← Back](#) Register Activity

4. 選擇登錄活動。

## 啟動工作流程執行

您可以從 Amazon SWF 主控台開始執行工作流程。您必須至少註冊一個工作流程，才能啟動工作流程執行。

### 使用主控台啟動工作流程執行

#### New console

1. 開啟 Amazon SWF 主控台，然後在左側導覽窗格選擇網域。
2. 在網域名稱下方，選擇工作流程。

3. 在「 工作流程」頁面，選擇您要執行的工作流程。
4. 選擇 Start execution (開始執行)。
5. 在「」開始執行頁面中，輸入[工作流程名稱](#)和執行識別碼以名稱識別執行身份。或者，您也可以選擇指定[工作清單](#)將用於為此工作流程執行所產生的決策任務。

## Start execution [Learn more](#)

### Execution details

**Domain**

**Workflow**

**Execution ID**  
This will be used to associate a custom identifier with the workflow execution.

Up to 256 characters. Must not contain a : (colon), / (slash), | (vertical bar), or any control characters.

**Task list** [Learn more](#)  
Used for decision tasks generated for this execution.

Up to 256 characters. Must not contain a : (colon), / (slash), | (vertical bar), or any control characters.

**Input - optional**

*Enter input for execution*

---

Up to 32768 characters.

► **Advanced options**

6. (選用) 選擇進階選項，以指定工作流程執行的下列詳細資訊：
  - [任務優先順序](#)— 用於此工作流程執行的作業優先順序。
  - [執行開始關閉逾時](#)— 此工作流程執行的總持續時間。
  - [任務開始關閉超時](#)— 此工作流程執行的決策任務的最長持續時間。

- [兒童政策](#)— 用於此工作流程執行的子工作流程執行 (如果終止) 的原則，方法是呼叫 [TerminateWorkflowExecution](#) 明確行動或由於逾時過期而導致的動作。
- [Lambda 角色](#)— 要附加至此工作流程執行的 IAM 角色。

▼ **Advanced options**

**Task priority** [Learn more](#)   
Specify the priority to schedule decision tasks generated for this execution. A higher number indicates higher priority.

  
Must be from -2147483648 to 2147483647

**Execution start to close timeout** [Learn more](#)   
Executions of this workflow will be timed out if not completed within this duration.

  ▼  
Up to 8 characters. Use NONE for unlimited duration.

**Task start to close timeout** [Learn more](#)   
Decision task will be timed out if not completed within this duration.

  ▼  
Up to 8 characters. Use NONE for unlimited duration.

**Child policy** [Learn more](#)   
Determines a policy for child workflow executions if this execution is terminated.

Terminate  
 Request cancel  
 Abandon

**Lambda role - optional**  
It gives Amazon SWF the ability to invoke any Lambda function associated with your account in place of, or alongside traditional activities.

 ▼

## 7. 選擇 Start execution (開始執行)。

### Old console

1. 在 Amazon Simple Workflow Service 儀表板，下快速連結，選擇啟動新的工作流程執行。

在 Execution Details (執行詳細資訊) 對話方塊中輸入以下資訊。

- 網域

- Workflow Name (工作流程名稱)
- Workflow Version (工作流程版本)
- 工作流程 ID
- 任務清單
- 執行的執行時間上限
- 結束逾時的任務排程

有星號 ((\*)) 標記的欄位為必要項。

**Start New Execution** Cancel X

EXECUTION DETAILS    ADDITIONAL OPTIONS    REVIEW

Provide the details of your Execution below:

**Domain\***: 867530901

**Workflow Name\***: customerOrderWorkflow

**Workflow Version\***: 1.0

**Workflow ID\***: 20110927-T-1

**Task List**: specialTaskList

**Max. Execution Run Time**: 1800 seconds

**Task Start to Close Timeout**: 600 seconds

Continue

選擇 Continue (繼續)。

2. 在 Additional Options (其他選項) 對話方塊中指定：
  - 一組 Tags (標籤)，建立與工作流程執行的關聯。您可以使用這些標籤來查詢您的工作流程執行資訊。
  - 對執行有意義的 Input (輸入) 字串。Amazon SWF 不會解譯此字串。
  - 子政策。

## Start New Execution Cancel X

EXECUTION DETAILS **ADDITIONAL OPTIONS** REVIEW

Provide additional options for your Execution below:

**Tags:** music purchase, digital, ricoh-the-dog

**Input:** arbitrary-string-that-is-meaningful-to-the-workflow

**Child Policy:** Terminate

[Back](#) Review

3. 在 Review (檢閱) 對話方塊中，檢閱您在上一個對話方塊中輸入的資訊。若資訊無誤，請選擇 Start Execution (開始執行)。否則，請選擇 Back (上一步) 變更資訊。

### Start New Execution Cancel

EXECUTION DETAILS    ADDITIONAL OPTIONS    REVIEW

Please review the information below, then click **Start**

**Domain:** 867530901

**Workflow Name:** customerOrderWorkflow

**Workflow Version:** 1.0

**Workflow ID:** 20110927-T-1

**Max. Execution Run Time:** 30 minutes

**Task Start to Close Timeout:** 10 minutes

**Task List:** specialTaskList

**Child Policy:** TERMINATE

**Tags:**

**Input:**

[Back](#)

## 檢視擱置中任務

### Note

只有舊版 Amazon SWF 主控台中才能使用這項工作。

從 Amazon 簡單工作流程服務儀表板，您可以檢視與特定任務清單相關聯的擱置任務數目。

1. 選取此任務清單為 Decider Task List (決策者任務清單) 或 Activity Task List (活動任務清單)。

2. 在文字方塊中輸入任務清單名稱。
3. 選擇 View Backlog (檢視後端記錄)。

Enter the Task List Name:

Decider Task List  Activity Task List

*Task List "specialTaskList" has a backlog of 3 tasks.*

## 管理工作流程執行

在 Amazon SWF 主控台上，您可以管理目前正在執行和/或關閉的工作流程執行。

### New console

若要管理您的工作流程執行

1. 開啟網域以管理其工作流程執行。
2. 選擇尋找執行。
3. 在「」工作流程執行頁面上，選擇按屬性過濾執行，然後在屬性選擇下列其中一個篩選條件：

選擇	若要套用此篩選
工作流程	<p>選擇此篩選器可列出特定工作流程的執行項目。例如，若要檢視fiction-books-order-workflow，執行下列動作：</p> <ol style="list-style-type: none"> <li>1. 選擇工作流程。</li> <li>2. 下運算子，選擇等於。</li> <li>3. 下工作流程，選擇fiction-books-order-workflow。</li> <li>4. (選用) 選擇清除篩選刪除過濾器並開始新的執行搜索。</li> </ol>
狀態	<p>選擇此篩選器以列出具有特定狀態的執行項目。例如，若要檢視具有狀態的執行項目失敗，執行下列動作：</p> <ol style="list-style-type: none"> <li>1. 選擇狀態。</li> <li>2. 下運算子，選擇等於。</li> </ol>

選擇	若要套用此篩選
	<ol style="list-style-type: none"> <li>3. 下狀態，選擇失敗。</li> <li>4. (選用) 選擇清除篩選刪除過濾器並開始新的執行搜索。</li> </ol>
執行代碼	<p>選擇此篩選器可根據工作流程 ID 檢視工作流程執行。例如，要使用 ID 查看執行fiction-books-order-category1，執行下列動作：</p> <ol style="list-style-type: none"> <li>1. 選擇執行代碼。</li> <li>2. 下運算子，選擇等於。</li> <li>3. 下執行 ID，選擇fiction-books-order-category1。</li> <li>4. (選用) 選擇清除篩選刪除過濾器並開始新的執行搜索。</li> </ol>
標籤	<p>選擇此篩選器以列出具有特定標記的執行項目。例如，若要檢視具有狀態的執行項目purchaseOrder，執行下列動作：</p> <ol style="list-style-type: none"> <li>1. 選擇標籤。</li> <li>2. 下運算子，選擇等於。</li> <li>3. 下標籤，選擇採購訂單。</li> <li>4. (選用) 選擇清除篩選刪除過濾器並開始新的執行搜索。</li> </ol>

4. (選擇性) 套用必要的篩選器以列出工作流程執行之後，您可以對作用中執行：
  - 信號— 使用此選項可傳送正在執行的工作流程執行其他資料。若要執行此作業：
    1. 選擇您要將其他資料傳送至哪一個執行。
    2. 選擇信號，然後在信號執行對話方塊。
    3. 選擇信號。

## Signal execution [Learn more](#) ✕

**Name**

Up to 256 characters

**Input - *optional***

Value to attach to the WorkflowExecutionSignaled event in the target execution's history.

Up to 32768 characters

**Execution ID (1)**

**Cancel** **Send signal**

- 嘗試取消— 使用此選項可嘗試取消工作流程執行。與其終止工作流程執行，最好的做法是將之取消。取消讓工作流程執行得以執行任何清除任務，然後妥當地予以關閉。
  1. 選擇您要取消的執行。
  2. 選擇嘗試取消。
- 終止— 使用此選項可終止工作流程執行。請注意，最好的做法為取消工作流程執行，而不是終止它。
  1. 選擇您要終止的執行。
  2. 對於兒童政策，請確定終止已選取。
  3. (選用) 指定原因和詳細資訊用於終止執行。
  4. 選擇 Terminate (終止)。

## Terminate execution [Learn more](#) ✕

**Child policy**  
Determines a policy for child workflow executions if this execution is terminated.

Terminate  
 Request cancel  
 Abandon

**Reason - optional**

Cancellation unsuccessful

Up to 256 characters

**Details - optional**

Attempted cancellation of the workflow execution was unsuccessful.

Up to 32768 characters

**Execution ID (1)**

• trial-fiction-1

**Cancel** **Terminate**

5. (選用)重新執行— 使用此選項可重新執行已關閉的工作流程執行。
  1. 在工作流程執行清單中，選取要重新執行的已關閉執行。當您選取已關閉的執行時，重新執行按鈕變為啟用狀態。選擇重新執行。
  2. 在「重新執行」頁面上，指定工作流程執行的詳細資訊，如中所述[啟動工作流程執行](#)。

### Old console

若要存取我的工作流程執行檢視中，選擇尋找執行 (秒)Amazon SWF 儀表板中的按鈕。

Enter the Workflow ID:

20110927-T-1 **Find Execution(s)** ▶

如果您先輸入工作流程 ID，則主控台將會顯示具有該工作流程 ID 的執行。否則，如果您選擇 Find Execution(s) (尋找執行)，則 My Workflow Executions (我的工作流程執行) 檢視將可讓您根據工作流程執行的啟動時間、是否仍在執行和其相關聯的中繼資料來查詢工作流程執行。對於指定的查詢，您可以選取下列任一類型的中繼資料：

- 工作流程 ID
- 工作流程類型
- Tags (標籤)
- 關閉狀態

如果關閉工作流程執行，則關閉狀態是下列其中一個值，指出關閉工作流程執行的情況：

- 已完成
- 失敗
- 已取消
- 逾時
- 以新項目繼續執行

 Note

您必須先從 Domain (網域) 下拉式清單中選取網域，才能列舉工作流程執行。

**My Workflow Executions**

Domain: 867530901 Manage Domains Refresh Help

**Workflow Execution List Parameters**

Filter by: Workflow Id

Workflow ID\*: Workflow Id  
Workflow Type  
Close Status  
Tag  
No Filter

Execution Status: *List Active and Closed*

Started between 2011 Jan 21 21:50:49 and 2012 Jan 21 23:59:59

List Executions

**Execution Actions:** Signal Try-Cancel Terminate Re-Run 1 to 3 of 3 items

<input type="checkbox"/>	Workflow ID	Run ID	Name (Version)	Tags	Execution Status	Start Time	Close Time
<input checked="" type="checkbox"/>	20110927-T-1	817e8eb0-353b-47de-90b0-5f6754bf278a	customerOrderWorkflow (1.0)	ricoh-the-dog	Active	Sat Jan 21 21:25:20 GMT-800 2012	
<input type="checkbox"/>	20110927-T-1	c4f8b600-f3a7-4c6a-be2b-440f92b7afe0	customerOrderWorkflow (1.0)	music purchase,digital,ricoh-the-dog	Timed Out	Fri Dec 23 14:25:14 GMT-800 2011	Fri Dec 23 14:55:14 GMT-800 2011
<input type="checkbox"/>	20110927-T-1	6c585d49-82ca-4b3e-adcb-852768dabfcd	customerOrderWorkflow (1.0)	music purchase,digital,ricoh-the-dog	Timed Out	Tue Dec 20 22:13:21 GMT-800 2011	Tue Dec 20 22:43:21 GMT-800 2011

列舉工作流程執行清單之後，即可執行下列操作。

- 指示工作流程執行 — 也就是傳送執行中的工作流程執行其他資料。

**Signal Execution** Cancel X

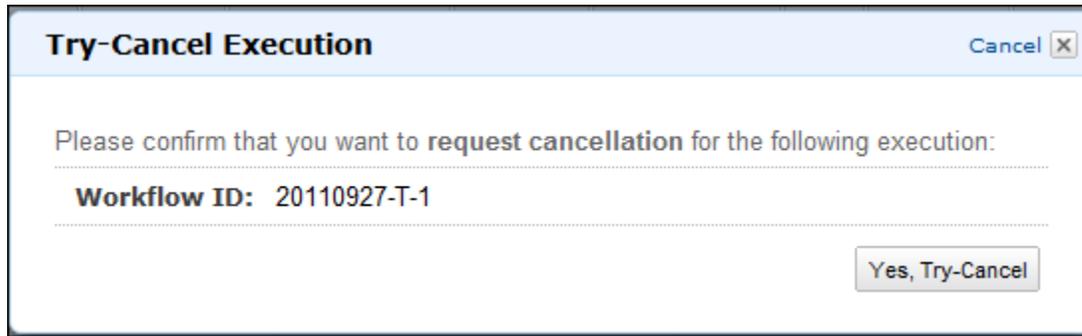
Provide the details of Signal action:

**Name\*:**

**Input:**

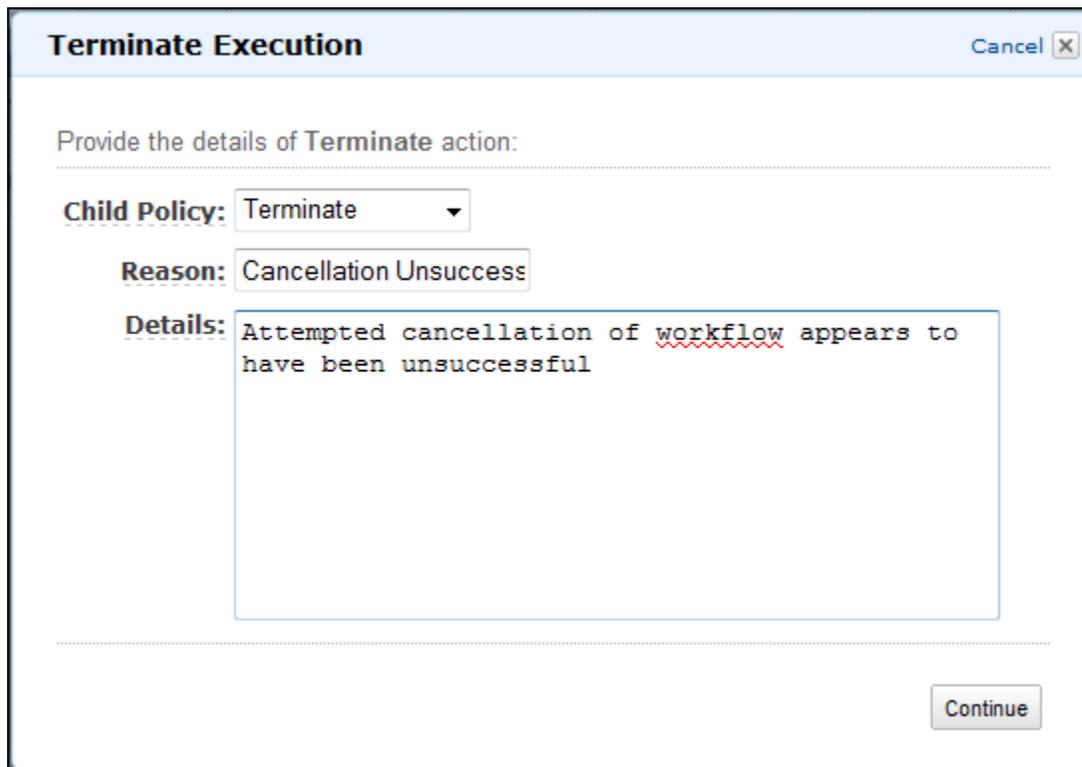
Continue

- 嘗試取消工作流程執行。與其終止工作流程執行，最好的做法是將之取消。取消讓工作流程執行得以執行任何清除任務，然後妥當地予以關閉。



The screenshot shows a dialog box titled "Try-Cancel Execution" with a "Cancel" button in the top right corner. The main text asks the user to confirm they want to request cancellation for a specific execution. Below this, the "Workflow ID: 20110927-T-1" is displayed. At the bottom right, there is a button labeled "Yes, Try-Cancel".

- 終止工作流程執行。請注意，最好的做法為取消工作流程執行，而不是終止它。



The screenshot shows a dialog box titled "Terminate Execution" with a "Cancel" button in the top right corner. The main text asks the user to provide details of the terminate action. Below this, there are three fields: "Child Policy" set to "Terminate", "Reason" set to "Cancellation Unsuccess", and "Details" containing the text "Attempted cancellation of workflow appears to have been unsuccessful". At the bottom right, there is a button labeled "Continue".

- 重新執行已關閉的工作流程執行。

The screenshot shows the 'My Workflow Executions' page in the AWS console. The domain is set to '867530901'. The 'Workflow Execution List Parameters' section includes a 'Filter by' dropdown set to 'Tag', a 'Tag' input field with 'ricoh-the-dog', and 'Execution Status' radio buttons for 'Active' and 'Closed' (selected). The 'Started between' range is from '2011 Dec 15 12:00:00' to '2011 Dec 23 23:59:59'. A 'List Executions' button is present. Below this, the 'Execution Actions' section shows buttons for 'Signal', 'Try-Cancel', 'Terminate', and 'Re-Run', with a red arrow pointing to 'Re-Run'. A table below lists one execution:

<input checked="" type="checkbox"/>	Workflow ID	Run ID	Name (Version)	Tags	Execution Sta	Start Time	Close Time
<input checked="" type="checkbox"/>	20110927-T-1	6c585d49-82ca-4b3e-adcb-852768dabfcd	customerOrderWorkflow (1.0)	music purchase,digital,ricoh-the-dog	Timed Out	Tue Dec 20 22:13:21 GMT-800 2011	Tue Dec 20 22:43:21 GMT-800 2011

## 重新執行已關閉的工作流程執行

1. 在工作流程執行清單中，選取要重新執行的已關閉執行。當您選取已關閉的執行時，Re-Run (重新執行) 按鈕會變成可使用的狀態。選擇 Re-Run (重新執行)。

即會出現 Re-Run Execution (重新執行執行) 序列的對話方塊。

2. 在 Execution Details (執行詳細資訊) 對話方塊中，指定下列資訊。此對話方塊已填入原始執行中的資訊。

- 網域
- Workflow Name (工作流程名稱)
- Workflow Version (工作流程版本)
- 工作流程 ID

選擇 Advanced Options (進階選項) 連結，即可指定下列其他選項。

- 任務清單
- 執行的執行時間上限
- 結束逾時的任務排程

選擇 Continue (繼續)

3. 在 Additional Options (其他選項) 對話方塊中，指定執行的輸入字串。選擇 Advanced Options (進階選項) 連結，即可指定 Tags (標籤) 建立與此執行或工作流程執行的關聯，以及變更執行 Child Policy (子政策)。如同前一個對話方塊，已填入原始執行中的資訊。

選擇 Review (檢閱)。

4. 在 Review (檢閱) 對話方塊中，確認所有資訊皆正確。如果資訊正確，請選擇 Re-Run Execution (重新執行執行)。否則，請選擇 Back (上一步) 變更資訊。

## 管理 Amazon SWF 網域

您可以使用 Amazon SWF 主控台執行網域管理動作，例如淘汰和未分配網域。在您取代網域之後，您便無法用它來建立新的工作流程執行，或註冊新的工作流程。取代網域也會棄用在網域中註冊的所有活動和工作流程。在取代網域之前啟動的執行會繼續執行。

您也可以取消先前已取代的網域。在未分析網域之後，您可以使用它來[在您的網域中註冊新的工作流程](#)或者[啟動新的工作流程執行](#)。

如需這些網域管理動作的詳細資訊，請參閱「[DeprecateDomain](#)和[UndeprecateDomain](#)」。

### Note

下列 Amazon SWF 網域管理程序適用於新的主控台體驗。

### 使用主控台取代網域

1. 開啟您要取代的 Amazon SWF 網域。
2. 選擇管理網域，然後選擇棄用。
3. 選擇棄用在取消網域用於確認動作的對話方塊。

### 使用主控台取消網域

1. 開啟您要取代的 Amazon SWF 網域。
2. 選擇管理網域，然後選擇十一酸。

或者，您也可以選擇十一酸網域在頁面顯示的訊息中。

3. 選擇十一酸在十一酸網域用於確認動作的對話方塊。

# Amazon SWF 運作方式

本章概念提供 Amazon Simple Workflow Service 服務及其主要功能。雖然本章的一些主題中含有 Amazon SWF 使用的範例，請參閱[使用 Amazon SWF API](#)，瞭解實現此處描述的功能的更多具體示例。

## 主題

- [Amazon SWF 工作流程](#)
- [Amazon SWF 工作流程歷史記錄](#)
- [Amazon SWF 動作者](#)
- [Amazon SWF 任務](#)
- [Amazon SWF 網域](#)
- [Amazon SWF 物件識別符](#)
- [Amazon SWF 任務清單](#)
- [關閉 Amazon SWF 工作流程執行](#)
- [Amazon SWF 工作流程執行生命週期](#)
- [在 Amazon SWF 中輪詢任務](#)
- [Amazon SWF 中的高級概念](#)

# Amazon SWF 工作流程

## 主題

- [什麼是工作流程？](#)
- [簡單工作流程範例：電子商務應用程式](#)
- [工作流程註冊和執行](#)
- [另請參閱](#)

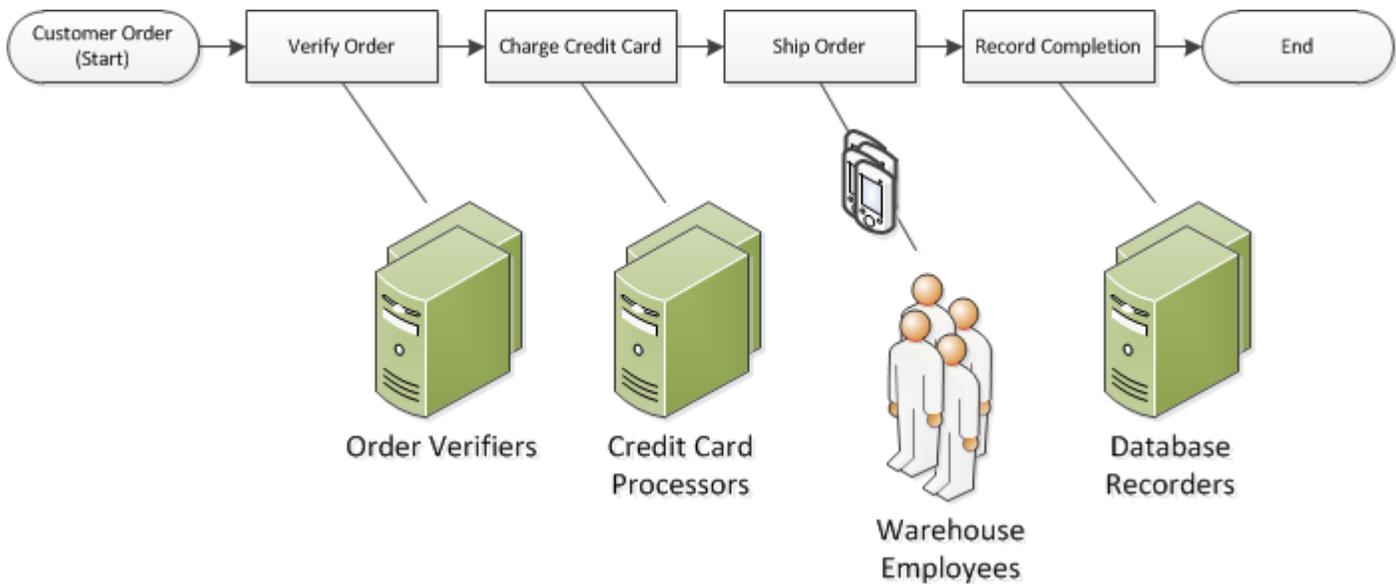
## 什麼是工作流程？

使用 Amazon Simple Workflow Service (Amazon SWF)，您可以將分散式非同步應用程式實作為工作流程。工作流程會協調和管理活動的執行，且活動可以跨多個運算裝置非同步執行，也可以具備循序和平行處理。

設計工作流程時，您可以分析應用程式來識別其元件「任務」。在 Amazon SWF 中，這些任務是以活動。活動的執行順序是根據工作流程的協調性邏輯所決定。

## 簡單工作流程範例：電子商務應用程式

例如，下圖顯示包含人員和自動化程序的簡單電子商務訂單處理工作流程。



客戶下單時，會啟動此工作流程。其中包含四個「任務」。

1. 確認訂單。
2. 如果訂單有效，將會向客戶收取費用。
3. 如果付款已完成，將送出訂單。
4. 如果送出訂單，將儲存訂單詳細資訊。

此工作流程中的任務是「循序的」：必須先確認訂單，才能收取信用卡費用；必須先順利收取信用卡費用，才能送出訂單，而且必須先送出訂單，才能加以記錄。即使如此，Amazon SWF 支援分散式程序，所以可以在不同的位置執行這些任務。如果任務具有程式設計的性質，也可使用不同的程式設計語言或工具加以撰寫。

Amazon SWF 還支援平行處理任務的工作流程。平行任務會同時執行，而且可以由不同的應用程式或人類工作者獨立地執行。您的工作流程會在完成一或多個平行任務之後，做出應如何繼續的決策。

## 工作流程註冊和執行

設計協調性邏輯和活動之後，請向 Amazon SWF 將這些組件註冊為工作流程和活動類型。註冊時，您會為每種類型指定名稱、版本以及一些預設組態值。

Amazon SWF 只有已註冊的工作流程和活動類型才能用於。在電子商務範例中，您將註冊 CustomerOrder 工作流程類型，以及 VerifyOrder、ChargeCreditCard、ShipOrder 和 RecordCompletion 活動類型。

工作流程類型一經註冊後，即可供您不限次數地執行。「工作流程執行」為運作中的工作流程執行個體。在電子商務範例中，每個客戶訂單會啟動新的工作流程執行。

任何程序或應用程式，甚至是另一個工作流程執行，都可以啟動工作流程執行。在電子商務範例中，由何種類型的應用程式啟動工作流程取決於客戶如何下單。工作流程可能由網站或行動應用程式，或者客戶服務代表使用公司的內部應用程式來啟動。

使用 Amazon SWF，您可以關聯一個標識符（稱為 workflowId，以便您可以將現有業務標識符集成到工作流程中。在電子商務範例中，可能會使用客戶發票號碼來識別每個工作流程執行。

除了您提供的標識符外，Amazon SWF 還會關聯一個唯一的系統生成的標識符 —runId— 用於每個工作流程執行。Amazon SWF 允許在任何給定時間運行具有此標識符的工作流程執行；儘管您可以執行相同工作流程類型的多個工作流程執行，但每個工作流程執行都有一個不同的 runId。

### 另請參閱

[Amazon SWF 工作流程歷史記錄](#)

## Amazon SWF 工作流程歷史記錄

Amazon SWF 將記錄每個工作流程執行的進度記錄在工作流程歷史記錄中。工作流程歷史記錄是每個事件自工作流程執行自開始以來之詳細、完整和一致的記錄。事件代表您工作流程執行狀態的不連續變更，例如正在排程的新活動或即將完成的執行中活動。工作流程歷史記錄包含的每一個事件，都會造成工作流程執行的執行狀態變更，例如排程和完成的活動、任務逾時和訊號。

不變更工作流程執行狀態的操作，一般不會顯示在工作流程歷史記錄中。例如，工作流程歷史記錄不顯示輪詢嘗試或可見度操作的使用。

工作流程歷史記錄有數個重要優勢：

- 讓應用程式成為無狀態，因為工作流程執行的所有資訊都存放在其工作流程歷史記錄中。

- 歷史記錄為每個工作流程執行提供一筆記錄，包括已排程的活動、它們目前的狀態及其結果。工作流程執行使用此資訊決定後續步驟。
- 歷史記錄提供詳細的稽核線索，您可用來監控執行中的工作流程執行，以及驗證完成的工作流程執行。

以下是電子商務工作流程歷史記錄的概念檢視。

```
Invoice0001

Start Workflow Execution

Schedule Verify Order
Start Verify Order Activity
Complete Verify Order Activity

Schedule Charge Credit Card
Start Charge Credit Card Activity
Complete Charge Credit Card Activity

Schedule Ship Order
Start Ship Order Activity
```

在前面的範例中，訂單等待出貨。在下列範例中，訂單已完成。因為工作流程歷史記錄是累積的，所以會附加較新的事件：

```
Invoice0001

Start Workflow Execution

Schedule Verify Order
Start Verify Order Activity
Complete Verify Order Activity

Schedule Charge Credit Card
Start Charge Credit Card Activity
Complete Charge Credit Card Activity

Schedule Ship Order
Start Ship Order Activity

Complete Ship Order Activity
```

*Schedule Record Order Completion*  
*Start Record Order Completion Activity*  
*Complete Record Order Completion Activity*

*Close Workflow*

透過編寫程式的方式，工作流程執行歷史記錄中的事件是表示為 JavaScript 物件標記法 (JSON) 物件。歷史記錄本身是這些物件的 JSON 陣列。每個事件都有下列項目：

- 類型，例如 [WorkflowExecutionStarted](#) 或 [ActivityTaskCompleted](#)
- Unix 時間格式的時間戳記
- 唯一識別事件的 ID

此外，每個事件類型都有適合該類型的不同描述屬性集。例如，`ActivityTaskCompleted` 事件有包含事件 ID 的屬性，對應活動任務排程的時間和開始的時間，以及保留結果資料的屬性。

您可以使用 [GetWorkflowExecutionHistory](#) 動作，取得工作流程執行歷史記錄目前狀態的複本。此外，Amazon SWF 和您工作流程決策者之間互動的一部分，決策者會定期收到歷史記錄的複本。

以下是 JSON 格式範例工作流程執行歷史記錄的一部分。

```
[ {
  "eventId": 11,
  "eventTimestamp": 1326671603.102,
  "eventType": "WorkflowExecutionTimedOut",
  "workflowExecutionTimedOutEventAttributes": {
    "childPolicy": "TERMINATE",
    "timeoutType": "START_TO_CLOSE"
  }
}, {
  "decisionTaskScheduledEventAttributes": {
    "startToCloseTimeout": "600",
    "taskList": {
      "name": "specialTaskList"
    }
  },
  "eventId": 10,
  "eventTimestamp": 1326670566.124,
  "eventType": "DecisionTaskScheduled"
}, {
```

```

    "activityTaskTimedOutEventAttributes": {
      "details": "Waiting for confirmation",
      "scheduledEventId": 8,
      "startedEventId": 0,
      "timeoutType": "SCHEDULE_TO_START"
    },
    "eventId": 9,
    "eventTimestamp": 1326670566.124,
    "eventType": "ActivityTaskTimedOut"
  }, {
    "activityTaskScheduledEventAttributes": {
      "activityId": "verification-27",
      "activityType": {
        "name": "activityVerify",
        "version": "1.0"
      },
      "control": "digital music",
      "decisionTaskCompletedEventId": 7,
      "heartbeatTimeout": "120",
      "input": "5634-0056-4367-0923,12/12,437",
      "scheduleToCloseTimeout": "900",
      "scheduleToStartTimeout": "300",
      "startToCloseTimeout": "600",
      "taskList": {
        "name": "specialTaskList"
      }
    },
    "eventId": 8,
    "eventTimestamp": 1326670266.115,
    "eventType": "ActivityTaskScheduled"
  }, {
    "decisionTaskCompletedEventAttributes": {
      "executionContext": "Black Friday",
      "scheduledEventId": 5,
      "startedEventId": 6
    },
    "eventId": 7,
    "eventTimestamp": 1326670266.103,
    "eventType": "DecisionTaskCompleted"
  }, {
    "decisionTaskStartedEventAttributes": {
      "identity": "Decider01",
      "scheduledEventId": 5
    },
  },

```

```
    "eventId": 6,
    "eventTimestamp": 1326670161.497,
    "eventType": "DecisionTaskStarted"
  }, {
    "decisionTaskScheduledEventAttributes": {
      "startToCloseTimeout": "600",
      "taskList": {
        "name": "specialTaskList"
      }
    },
    "eventId": 5,
    "eventTimestamp": 1326668752.66,
    "eventType": "DecisionTaskScheduled"
  }, {
    "decisionTaskTimedOutEventAttributes": {
      "scheduledEventId": 2,
      "startedEventId": 3,
      "timeoutType": "START_TO_CLOSE"
    },
    "eventId": 4,
    "eventTimestamp": 1326668752.66,
    "eventType": "DecisionTaskTimedOut"
  }, {
    "decisionTaskStartedEventAttributes": {
      "identity": "Decider01",
      "scheduledEventId": 2
    },
    "eventId": 3,
    "eventTimestamp": 1326668152.648,
    "eventType": "DecisionTaskStarted"
  }, {
    "decisionTaskScheduledEventAttributes": {
      "startToCloseTimeout": "600",
      "taskList": {
        "name": "specialTaskList"
      }
    },
    "eventId": 2,
    "eventTimestamp": 1326668003.094,
    "eventType": "DecisionTaskScheduled"
  }
]
```

有關工作流程執行歷史記錄中可顯示的不同類型事件的詳細列表，請參閱[HistoryEvent](#) 佈資料類型 Amazon Simple Workflow Service API 參考。

Amazon SWF 會在執行結束後會存放所有工作流程執行的完整歷史記錄一段時間，天數可設定。這段期間稱為工作流程歷史記錄保留期，當您為工作流程註冊「網域」時會指定。本節後文會詳細討論網域。

## Amazon SWF 動作者

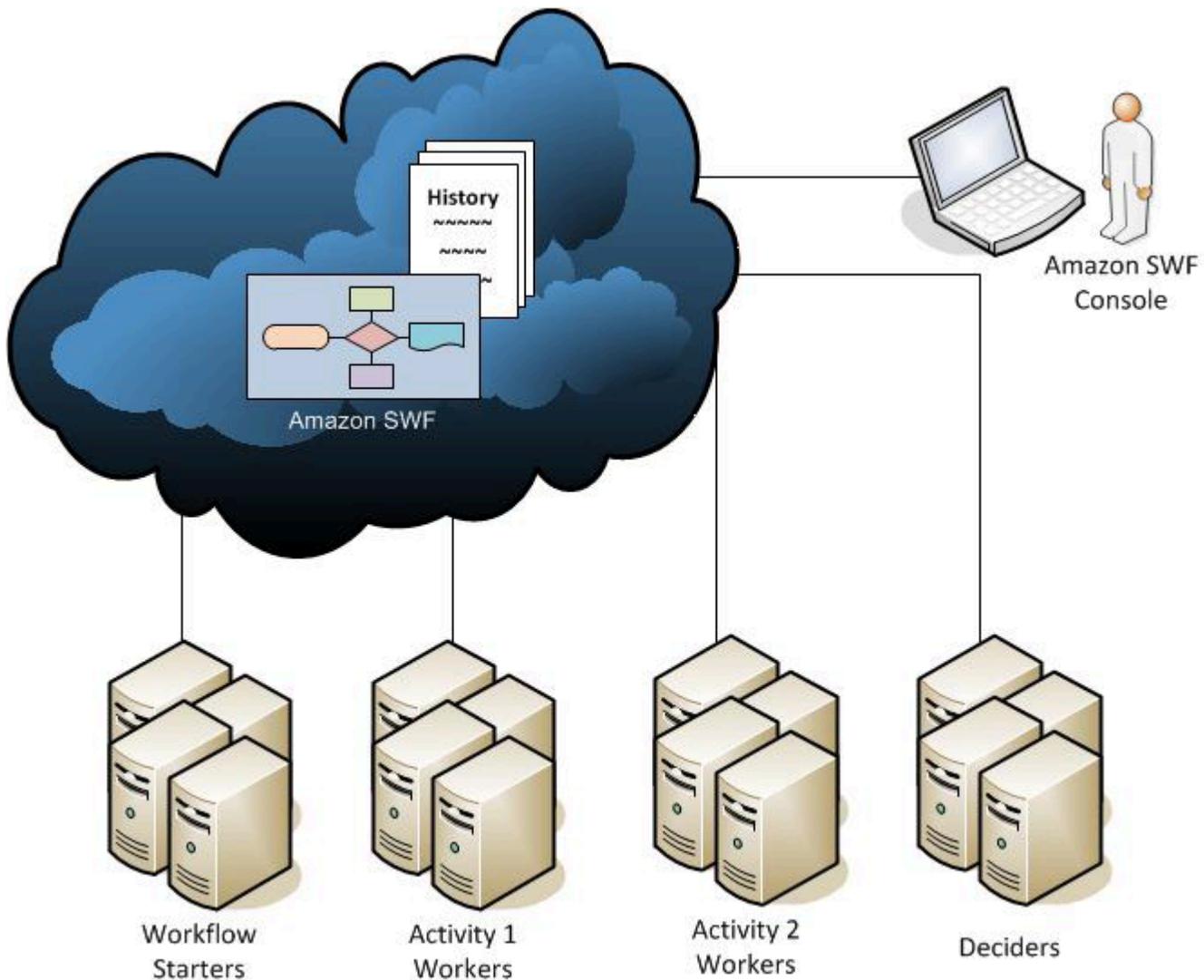
### 主題

- [什麼是 Amazon SWF 中的動作者？](#)
- [工作流程啟動者](#)
- [決策者](#)
- [活動工作者](#)
- [動作者之間的資料交換](#)

## 什麼是 Amazon SWF 中的動作者？

在操作期間，Amazon SWF 會與許多不同類型程式設計的演員。動作者可以是[工作流程啟動者](#)、[決策者](#)或[活動工作者](#)。這些動作者會透過的 API 與 Amazon SWF 進行通訊。您可以使用任何程式設計語言來開發這些動作者。

下圖顯示 Amazon SWF 架構，包含 Amazon SWF 及其動作者。



## 工作流程啟動者

工作流程啟動者是任何可起始工作流程執行的應用程式。以電子商務為例，一個工作流程啟動者可以是客戶下單的網站。另一個工作流程啟動者可以是客戶服務代表用來代替客戶下單的行動應用程式或系統。

## 決策者

決策者是工作流程協調性邏輯的實作。決策者負責控制工作流程執行內之活動任務的流程。只要在工作流程執行期間有所更動 (例如完成工作)，就會將決策任務 (包含整個工作流程歷史記錄) 傳遞給決策者。決策者收到來自 Amazon SWF 的決策任務時，會分析工作流程執行歷史記錄，以判斷工作流程執行中的下一個適當步驟。決策者將這些步驟傳遞回 Amazon SWF，使用決定。決策是 Amazon SWF 資料類型，可代表各種後續動作。如需可能決策的清單，請前往[決策](#)，位於 Amazon Simple Workflow Service API 參考中。

以下範例是以 JSON 格式呈現的決策，而此格式的決策可傳輸至 Amazon SWF。此決策會新的活動任務進行排程。

```
{
  "decisionType" : "ScheduleActivityTask",
  "scheduleActivityTaskDecisionAttributes" : {
    "activityType" : {
      "name" : "activityVerify",
      "version" : "1.0"
    },
    "activityId" : "verification-27",
    "control" : "digital music",
    "input" : "5634-0056-4367-0923,12/12,437",
    "scheduleToCloseTimeout" : "900",
    "taskList" : {
      "name": "specialTaskList"
    },
    "scheduleToStartTimeout" : "300",
    "startToCloseTimeout" : "600",
    "heartbeatTimeout" : "120"
  }
}
```

在 workflow 執行啟動時，及每次 workflow 執行內的狀態變更時，決策者會接收到決策任務。決策者會繼續接收決策任務並回應 Amazon SWF，直至決策者判斷 workflow 執行完成，使 workflow 執行繼續進展。直到最後，決策者會回應關閉 workflow 執行的決策。workflow 執行關閉之後，Amazon SWF 就不會為該執行排程其他任務。

在電子商務範例中，決策者會判斷是否正確地執行每個步驟，然後排程下一個步驟或管理任何錯誤情況。

決策者代表單一電腦程序或執行緒。多個決策者可以處理相同 workflow 類型的任務。

## 活動工作者

活動工作者是一種程序或執行緒，可執行屬於 workflow 一部分的「活動任務」。活動任務代表您在應用程式中識別的其中一個任務。

若要在 workflow 中使用活動任務，您必須使用 Amazon SWF 主控台或 [RegisterActivityType](#) 動作。

每個活動工作者都會輪詢 Amazon SWF，以獲取適合該活動工作者執行的新任務；某些任務只能由特定活動工作者來執行。收到任務之後，活動工作者會處理任務，完成後向 Amazon SWF 報告任務已完

成並提供結果。然後活動工作者會輪詢新任務。與工作流程執行相關聯的活動工作者會以此方式繼續處理任務，直到工作流程執行本身完成為止。在電子商務範例中，活動工作者是信用卡處理者和倉儲員工這類人員所使用的獨立程序和應用程式，可在程序中執行個別步驟。

活動工作者代表單一電腦程序 (或執行緒)。多個活動工作者可以處理相同活動類型的任務。

## 動作者之間的資料交換

啟動工作流程執行時，即可將輸入資料提供給工作流程執行。同樣地，活動工作者為活動任務進行排程時，即可將輸入資料提供給活動工作者。活動任務完成時，活動工作者可以將結果傳回給 Amazon SWF。同樣地，決策者可以在工作流程執行完成時報告其結果。每個動作者都皆可透過字串 (使用者定義的形式) 在 Amazon SWF 中傳送和接收資料。根據資料的大小和敏感度，您可以直接傳遞資料，或傳遞指標至存放在另一個系統或服務 (例如 Amazon S3 或 DynamoDB) 上的資料。直接傳遞的資料以及其他資料存放區的指標會記錄在工作流程執行歷史記錄中；不過，Amazon SWF 不會複製或快取外部存放區中的任何資料來做為歷史記錄。

由於 Amazon SWF 會維護每個工作流程執行的完成執行狀態 (包含輸入和任務結果)，因此所有動作者都可以是無狀態。工作流程處理因而極具擴展性。隨著系統負載的成長，您只要新增其他動作者即可增加容量。

## Amazon SWF 任務

Amazon SWF 將稱為任務的工作指派提供給活動工作者和決策者，以與其互動。Amazon SWF 中有三種類型的任務：

- 個活動任務— 一個活動任務都會告知活動工作者執行其功能，例如檢查庫存或使用信用卡付費。活動任務包含活動工作者執行其功能所需的所有資訊。
- Lambda 任務— 一個 Lambda 任務與「活動」任務類似，但執行 Lambda 函數，而非傳統 Amazon SWF 活動。如需如何定義 Lambda 任務的詳細資訊，請參閱「[AWS Lambda 任務](#)」。
- 決策任務— 一個決策任務會告知決策者工作流程執行狀態已變更，因此決策者可以判斷下一個需要執行的活動。決策任務包含目前工作流程歷史記錄。

工作流程啟動時，以及工作流程的狀態變更時 (例如活動任務完成後)，Amazon SWF 會排程決策任務。每個決策任務皆包含以分頁檢視的整個工作流程執行歷史記錄。決策者會分析工作流程執行歷史記錄，並將一組決策回報 Amazon SWF，而這組決策指定應該在工作流程執行中發生的下一個任務。基本上，每個決策任務都會讓決策者有機會評估工作流程，並將指示提供回 Amazon SWF。

為確定不會處理到衝突決策，Amazon SWF 只會將每個決策任務指派給一個決策者，且於工作流程執行中一次只允許一個作用中的決策任務。

下表示範與工作流程和決策者相關之不同建構間的關係。

邏輯設計	註冊為	執行者	收到並執行	產生
工作流程	工作流程類型	決策者	決策任務	決策

活動工作者完成活動任務時，會向 Amazon SWF 報告任務已完成，並包含任何產生的相關結果。Amazon SWF 會使用指出任務完成的事件來更新工作流程執行歷史記錄，然後排程決策任務，將已更新的歷史記錄傳輸給決策者。

Amazon SWF 只會將每個活動任務指派給一個活動工作者。任務指派之後，就沒有其他活動工作者可以宣告或執行該任務。

下表示範與活動相關之不同建構間的關係。

邏輯設計	註冊為	執行者	收到並執行	產生
活動	活動類型	活動工作者	個活動任務	結果資料

## Amazon SWF 網域

網域提供方法，將 Amazon SWF 資源限定為您 AWS 帳戶。工作流程的所有元件，例如工作流程類型和活動類型，必須指定在一個網域內。一個網域中可能有多個工作流程，但不同網域中的工作流程彼此無法互動。

設定新的工作流程時，在您設定任何其他工作流程元件之前，如果您尚未註冊網域，則需要註冊網域。

當您註冊網域時，您要指定「工作流程歷史記錄保留期」。這個期間是 Amazon SWF 在工作流程執行完成後繼續保留工作流程執行相關資訊的時間。

## Amazon SWF 物件識別符

下列清單描述 Amazon SWF 物件（例如工作流程執行）唯一識別。

- 工作流程類型— 已註冊的工作流程類型依其網域、名稱和版本來識別。工作流程類型指定於對 `RegisterWorkflowType` 的呼叫中。
- 活動類型— 已註冊的活動類型依其網域、名稱和版本來識別。活動類型指定於對 `RegisterActivityType` 的呼叫中。
- 決策任務和活動任務— 每一項決策任務和活動任務皆依其唯一任務字符來識別。Amazon SWF 生成任務字符，並隨其他任務信息一起在 `PollForDecisionTask` 或者 `PollForActivityTask`。雖然接到任務的程序最常使用字符，但該程序可以將字符傳遞給其他程序，然後該其他程序可回報該任務完成或失敗。
- 工作流程執行— 單一的工作流程執行依網域、工作流程 ID 和執行 ID 來識別。前兩項是傳遞至 [StartWorkflowExecution](#) 的參數。執行 ID 則由 `StartWorkflowExecution` 傳回。

## Amazon SWF 任務清單

任務清單可讓您組織與工作流程相關聯的各種任務。您可以將任務清單想成像動態佇列。當您向 Amazon SWF 中排程任務時，您可以指定要放入任務的隊列 (任務清單)。同樣地，當您向輪詢 Amazon SWF 時，您會指定要從中取得任務的隊列 (任務清單)。

任務清單提供彈性的機制，可視您使用案例的需求將任務路由給工作者。任務清單是動態的，因此您不需要註冊任務清單或透過動作明確加以建立：只要排程任務，即可建立任務清單 (如果尚未存在)。

「活動」任務和「決策」任務有不同的清單。一個任務一律只能在一個任務清單中排程，無法跨清單共享任務。此外，如同活動和工作流程，任務清單的範圍為特定 AWS 地區和 Amazon SWF 域。

### 主題

- [決策任務清單](#)
- [活動任務清單](#)
- [任務路由](#)

## 決策任務清單

每個工作流程執行都與特定決策任務清單相關聯。註冊工作流程類型之後 ([RegisterWorkflowType](#) 動作)，您可為該工作流程類型的執行指定預設任務清單。當工作流程啟動者啟動工作流程執行時 (`StartWorkflowExecution` 動作)，可選擇為該工作流程執行指定不同的任務清單。

當決策者輪詢新的決策任務時 (`PollForDecisionTask` 動作)，決策者會指定要從中擷取的決策任務清單。單一決策者可以透過多次呼叫 `PollForDecisionTask`，每次呼叫都使用不同的任務清單，且

每個任務清單都專屬於特定工作流程執行，來提供多個工作流程執行。或者，決策者可以輪詢為多個工作流程執行提供決策任務的單一決策任務清單。您也可以透過讓全部決策者向任務清單輪詢工作流程執行，來提供單一工作流程執行。

## 活動任務清單

單一活動任務清單可包含不同活動類型的任務。任務在任務列表上按順序計劃。Amazon SWF 會盡最大努力依序傳回清單中的任務。在某些情況下，任務可能不會依序離開清單。

註冊活動類型之後 ([RegisterActivityType](#) 動作)，您可為該活動類型指定預設任務清單。根據預設，此類型的活動任務會在指定的任務清單中排程，不過，當決策者排程活動任務時 ([ScheduleActivityTask](#) 決策)，決策者可選擇性指定不同的任務清單來排程任務。如果決策者未指定任務清單，則會使用預設任務清單。因此，您可以根據任務的屬性，將活動任務放在特定任務清單中。例如，您可以將指定信用卡類型之活動任務的所有執行個體放在特定任務清單中。

## 任務路由

當活動工作者輪詢新的任務時 ([PollForActivityTask](#) 動作)，可指定要從中擷取的活動任務清單。如果這樣做，活動工作者就只會接受來自該清單的任務。透過此方法，您可以確保特定任務只會指派給特定活動工作者。例如，您可以建立任務清單，其中包含需要使用高效能電腦的任務。只有在適當硬體上執行的活動工作者才會輪詢該任務清單。另一個範例是建立特定地理區域的任務清單。您可以接著確保只有部署在該區域中的工作者才會領取這些任務。或者，您可以建立高優先順序的任務清單，並一律先檢查該清單。

將特定任務指派給特定活動工作者的這種方式稱為「任務路由」。任務路由是選擇性的，如果您未在排程活動任務時指定任務清單，則會自動將任務放在預設任務清單中。

## 關閉 Amazon SWF 工作流程執行

一旦啟動工作流程執行，它就會開啟。開啟的工作流程執行可以完成、取消、失敗或逾時等名目關閉。也可以做為新的執行繼續，也可以終止。工作流程執行可由決策者、管理工作流程的人員或由 Amazon SWF 關閉。

如果決策者決定工作流程活動已完成，它應該會使用 [RespondDecisionTaskCompleted](#) 動作並傳遞 [CompleteWorkflowExecution](#) 決策，以完成的名目關閉工作流程執行。

或者，決策者可能以取消或失敗的名目關閉工作流程執行。為取消執行，決策者應該使用 [RespondDecisionTaskCompleted](#) 動作並傳遞 [CancelWorkflowExecution](#) 決策。

如果它進入一般完成範圍外的狀態，決策者應該無法繼續工作流程執行。為使執行成為失敗狀態，決策者應該使用 `RespondDecisionTaskCompleted` 動作並傳遞 [FailWorkflowExecution](#) 決策。

Amazon SWF 監控工作流程執行以確保它們不超出任何使用者指定的逾時設定。如果工作流程執行逾時，Amazon SWF 會自動關閉它。如需逾時值的詳細資訊，請參閱「[Amazon SWF 超時類型](#)」一節。

決策者也可能關閉執行，使用 `RespondDecisionTaskCompleted` 動作並傳遞 [ContinueAsNewWorkflowExecution](#) 決策，以邏輯方式將它當做新的執行繼續。這對歷史記錄會隨著時間成長變大之長期執行的工作流程執行，是很有用的策略。

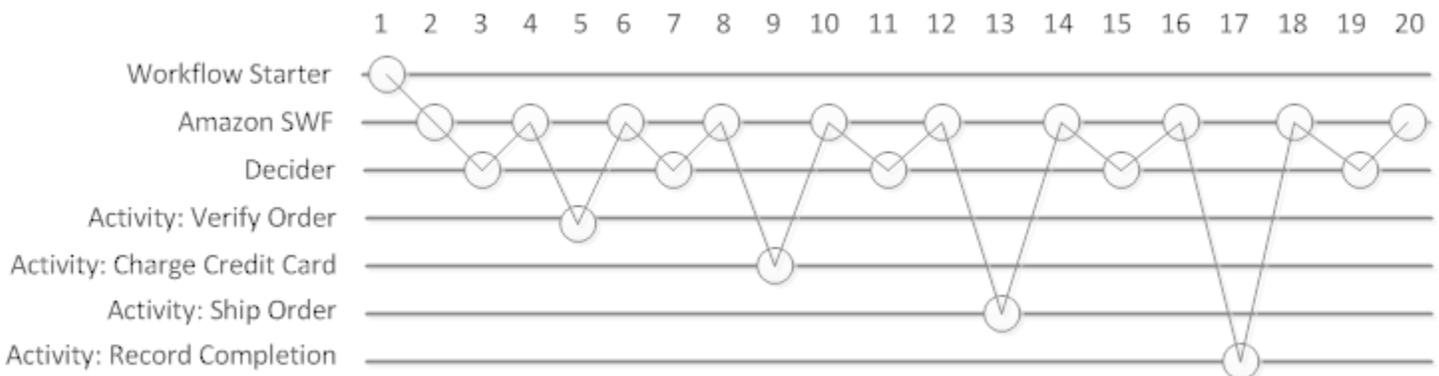
最後，您可直接從 Amazon SWF 主控台終止工作流程執行，或使用 [TerminateWorkflowExecution](#) API。終止會強制關閉工作流程執行。取消優於終止，因為您的決策者可管理工作流程執行的關閉。

如果執行超過某些服務定義的限制，Amazon SWF 將終止工作流程執行。如果父工作流程已終止，且適用的子政策指出子工作流程也應終止，則會終止子工作流程。

## Amazon SWF 工作流程執行生命週期

從工作流程執行的開始到完成，Amazon SWF 會與動作者互動，方法是將適當的任務 (活動任務或決策任務) 指派給它們。

下圖是從處理訂單處理工作流程執行的元件觀點，顯示該執行的生命週期。



### 工作流程執行生命週期

下表說明上圖中的每個任務。

描述	動作、決策或事件
1. 工作流程啟動者會呼叫適當的 Amazon SWF 操作以啟動訂單的工作流程執行，並提供訂單資訊。	<a href="#">StartWorkflowExecution</a> 動作。
2. Amazon SWF 收到啟動工作流程執行請求，然後排定第一個決策任務。	<a href="#">WorkflowExecutionStarted</a> 事件和 <a href="#">DecisionTaskScheduled</a> 事件。
3. 決策者收到來自 Amazon SWF 的任務、檢歷史記錄、套用協調性邏輯以確定未發生先前的活動、套用協調性邏輯以確認訂單活動工作者處理任務所需的資訊來排定「確認訂單」活動，以及將決策傳回給 Amazon SWF。	<a href="#">PollForDecisionTask</a> 動作。 <a href="#">RespondDecisionTaskCompleted</a> 動作和 <a href="#">ScheduleActivityTask</a> 決策。
4. Amazon SWF 收到決策、排定「確認訂單」活動任務，以及等待活動任務完成或超時。	<a href="#">ActivityTaskScheduled</a> 事件
5. 可執行「確認訂單」活動的活動工作者收到任務、執行任務，並將結果傳回給 Amazon SWF。	<a href="#">PollForActivityTask</a> 動作和 <a href="#">RespondActivityTaskCompleted</a> 動作。

描述	動作、決策或事件
6. Amazon SWF 收到「確認訂單」活動的結果、新增到工作流程歷史記錄，以及排定決策任務。	<a href="#">ActivityTaskCompleted</a> 事件和 <a href="#">DecisionTaskScheduled</a> 事件。
7. 決策者收到來自 Amazon SWF 的任務、檢歷史記錄、套用協調性邏輯、進行決策以使用活動工作者處理任務所需的資訊來排定 ChareCreditCard 活動任務，以及將決策傳回給 Amazon SWF。	<a href="#">PollForDecisionTask</a> 動作。 <a href="#">RespondDecisionTaskCompleted</a> 動作和 <a href="#">ScheduleActivityTask</a> 決策。
8. Amazon SWF 收到決策、排定 ChareCreditCard 活動任務，以及等待其完成或超時。	<a href="#">DecisionTaskCompleted</a> 事件和 <a href="#">ActivityTaskScheduled</a> 事件。
9. 可執行 ChareCreditCard 活動的活動工作者收到任務、執行任務，並將結果傳回給 Amazon SWF。	<a href="#">PollForActivityTask</a> 和 <a href="#">RespondActivityTaskCompleted</a> 動作。
10. Amazon SWF 收到 ChareCreditCard 活動任務的結果、新增到工作流程歷史記錄，以及排定決策任務。	<a href="#">ActivityTaskCompleted</a> 事件和 <a href="#">DecisionTaskScheduled</a> 事件。

描述	動作、決策或事件
<p>11. 決策者收到來自 Amazon SWF 的任務、檢歷史記錄、套用協調性邏輯、進行決策以使用活動工作者執行任務所需的資訊來排定 ShiOrder 活動任務，以及將決策傳回給 Amazon SWF。</p>	<p><a href="#">PollForDecisionTask</a> 動作。<a href="#">RespondDecisionTaskCompleted</a> 與 <a href="#">ScheduleActivityTask</a> 決策。</p>
<p>12. Amazon SWF 收到決策、排定 ShiOrder 活動任務，以及等待其完成或超時。</p>	<p><a href="#">DecisionTaskCompleted</a> 事件和 <a href="#">ActivityTaskScheduled</a> 事件。</p>
<p>13. 可執行 ShiOrder 活動的活動工作者收到任務、執行任務，並將結果傳回給 Amazon SWF。</p>	<p><a href="#">PollForActivityTask</a> 動作和 <a href="#">RespondActivityTaskCompleted</a> 動作。</p>
<p>14. Amazon SWF 收到 ShiOrder 活動任務的結果、新增到工作流程歷史記錄，以及排定決策任務。</p>	<p><a href="#">ActivityTaskCompleted</a> 事件和 <a href="#">DecisionTaskScheduled</a> 事件。</p>

描述	動作、決策或事件
<p>15. 決策者收到來自 Amazon SWF 的任務、檢歷史記錄、套用協調性邏輯、進行決策以使用活動工作者執行任務所需的資訊來排定 RecordCompletion 活動任務，以及將決策傳回給 Amazon SWF。</p>	<p><a href="#">PollForDecisionTask</a> 動作。<a href="#">RespondDecisionTaskCompleted</a> 動作和 <a href="#">ScheduleActivityTask</a> 決策。</p>
<p>16. Amazon SWF 收到決策、排定 RecordCompletion 活動任務，以及等待其完成或超時。</p>	<p><a href="#">DecisionTaskCompleted</a> 事件和 <a href="#">ActivityTaskScheduled</a> 事件。</p>
<p>17. 可執行 RecordCompletion 活動的活動工作者收到任務、執行任務，並將結果傳回給 Amazon SWF。</p>	<p><a href="#">PollForActivityTask</a> 動作和 <a href="#">RespondActivityTaskCompleted</a> 動作。</p>
<p>18. Amazon SWF 收到 RecordCompletion 活動任務的結果、新增到工作流程歷史記錄，以及排定決策任務。</p>	<p><a href="#">ActivityTaskCompleted</a> 事件和 <a href="#">DecisionTaskScheduled</a> 事件。</p>

描述	動作、決策或事件
19. 決策者收到來自 Amazon SWF 的任務、檢歷史記錄、套用協調性邏輯、進行決策以關閉工作流程執行，以及將決策與任何結果傳回給 Amazon SWF。	<a href="#">PollForDecisionTask</a> 動作。 <a href="#">RespondDecisionTaskCompleted</a> 動作和 <a href="#">CompleteWorkflowExecution</a> 決策。
20. Amazon SWF 關閉工作流程執行，並將歷史記錄傳回給未來參考。	<a href="#">WorkflowExecutionCompleted</a> 事件。

## 在 Amazon SWF 中輪詢任務

決策者和活動工作 Amazon SWF 員使用長輪詢。決策者或活動工作者會定期啟動與 Amazon SWF 的通訊，並通知 Amazon SWF 是否能接受任務，然後指定要從中取得任務的任務清單。

如果指定的任務清單上有任務，Amazon SWF 會在回應中立即傳回任務。如果沒有任務，則 Amazon SWF 會持續開啟 TCP 連接最多 60 秒，以便如果在該期間有任務，即可在相同的連接中傳回。如果在 60 秒內仍沒有任務，則會傳回空回應，並關閉連接。(空回應是 taskToken 值為空白字串的 Task 結構)。如果發生這種狀況，決策者或活動工作者應該會重新輪詢。

長時間輪詢十分適合大量任務處理。決策者和活動工作者可以管理它們自己的容量，而且在決策者和活動工作者受到防火牆保護時很容易使用。

如需詳細資訊，請參閱 [輪詢決策任務](#) 和 [輪詢活動任務](#)。

## Amazon SWF 中的高級概念

「[Amazon SWF 運作方式](#)」小節中的電子商務範例呈現了簡化的工作流程藍本。實際上，您可能想要工作流程執行並行任務 (在授權信用卡時傳送訂單確認電子郵件)、記錄主要事件 (包裝所有項目)、更動訂單 (新增或移除項目)，以及在工作流程執行期間做出更多進階決策。本節所說明的進階工作流程功能可用來建構健全和複雜的工作流程。

## 主題

- [版本控制](#)
- [訊號](#)
- [子工作流程](#)
- [標記](#)
- [標籤](#)
- [使用 Amazon Simple Workflow Service 實作互斥選項](#)
- [Amazon Simple Workflow Service Pimple Work](#)
- [Amazon Simple Workflow Service 活動任務取消](#)

## 版本控制

商業需求通常需要您在同時執行的相同工作流程或活動中，具備不同的實作或變化。例如，您可能想要在另一個工作流程進入生產階段的同時，測試新的工作流程實作。或許您也想要執行兩個不同功能集 (例如基本和高級實作) 的兩個不同實作。版本控制可讓您並行執行多個工作流程與活動的實作，滿足任何您所需的用途。

工作流程和活動類型會有在註冊時指定的相關聯版本。版本是任意形式的字串，而且您可以選擇自己的版本控制方案。若要建立已註冊類型的新版本，您應該以相同的名稱來註冊不同的版本。「[Amazon SWF 任務清單](#)」(如前所述) 可以進一步協助您實作版本控制。試想您已有指定類型之長時間運作的工作流程執行正在進行中的狀況，以及需要您修改工作流程的情況 (例如新增功能)。您可以建立新的活動類型與工作者版本及新的決策者，來實作新的功能。您接著可以使用另一組的任務清單，啟動新工作流程版本的執行。如此一來，您便能同時執行不同版本的工作流程執行，而彼此不互相影響。

## 訊號

訊號可讓您將資訊注入運作中的工作流程執行。在某些情況下，您可能想要將資訊新增到運作中的工作流程執行，予以告知某個項目已變更或在發生外部事件時予以通知。任何程序都可以傳送訊號給已開啟的工作流程執行。例如，某個工作流程執行可能會對另一個工作流程執行發出訊號。

### Note

嘗試傳送訊號給未開啟的工作流程執行會導致 `SignalWorkflowExecution` 失敗並出現 `UnknownResourceFault`。

若要使用信號，請定義要傳遞至信號的訊號名稱和資料 (如果有的話)。然後，對決策程序進行編程以識別歷史中的信號事件 ( [WorkflowExecutionSignaled](#) ) 並對其進行適當處理。當程序想要向工作流程執行發出指示時，會呼叫 Amazon SWF (使用 [SignalWorkflowExecution](#) 動作，或者在決策程式的情況下，使用 [SignalExternalWorkflowExecution](#) 決策程式) 來指定目標工作流程執行的識別碼、訊號名稱和訊號資料。然後，Amazon SWF 會接收訊號，將其記錄在目標工作流程執行的歷史記錄中，並為其排程決策任務。決策者收到決策任務時，也會在工作流程執行歷史記錄內收到訊號。決策者接著可以根據訊號和其資料來採取適當的動作。

有時候，您可能想要等候訊號。例如，用戶可以通過發送信號來取消訂單，但只能在下訂單後一小時內取消訂單。Amazon SWF 沒有使決策程序能夠等待來自服務的信號的原始文件。決策者本身必須實作暫停功能。為了暫停，決策者應該使用 `StartTimer` 決策啟動計時器，這會指定決策者將等候訊號，同時繼續輪詢決策任務的時間。決策者收到決策任務時，應該檢查歷史記錄，了解是否已收到訊號或計時器已觸發。如果已收到訊號，則決策者應該取消計時器。不過反之，如果計時器已觸發，則表示訊號未在指定時間內抵達。總結來說，為了等候特定訊號，請執行下列動作。

1. 針對決策者應該等候的時間長度，建立計時器。
2. 收到決策任務後，檢查歷史記錄，了解訊號是否已抵達或計時器是否已觸發。
3. 如果訊號已抵達，請使用 `CancelTimer` 決策取消計時器並處理訊號。根據計時，歷史記錄可包含 `TimerFired` 和 `WorkflowExecutionSignaled` 事件。在此情況下，您可以依據事件在歷史記錄中的相對順序，來決定哪個事件先出現。
4. 如果計時器在收到訊號之前就已觸發，則決策者等候訊號已逾時。您可以停止執行，或執行任何適合您使用案例的其他邏輯。

對於應取消工作流程的情況 (例如，訂單本身已被客戶取消)，應該使用 `RequestCancelWorkflowExecution` 動作，而不是傳送訊號至工作流程。

訊號的一些應用如下：

- 在收到訊號之前，暫停進行工作流程執行 (例如，等待存貨運送)。
- 提供資訊給工作流程執行，從而可能會影響決策者如何做出決策的邏輯。這適合用在受外部事件影響的工作流程 (例如，在市場關閉之後嘗試完成股票銷售)。
- 在您預期可能發生變更時，更新工作流程執行 (例如，在下單之後和運送之前變更訂單數量)。

在下列範例中，工作流程執行會傳送取消訂單的訊號。

```
https://swf.us-east-1.amazonaws.com  
SignalWorkflowExecution
```

```
{"domain": "867530901",  
  "workflowId": "20110927-T-1",  
  "runId": "f5ebbac6-941c-4342-ad69-dfd2f8be6689",  
  "signalName": "CancelOrder",  
  "input": "order 3553"}
```

如果工作流程執行收到訊號，Amazon SWF 會傳回與下列類似的成功 HTTP 回應。Amazon SWF 將產生決策任務，以通知決策程式處理訊號。

```
HTTP/1.1 200 OK  
Content-Length: 0  
Content-Type: application/json  
x-amzn-RequestId: bf78ae15-3f0c-11e1-9914-a356b6ea8bdf
```

## 子工作流程

使用子工作流程，能夠將複雜工作流程細分為較小、更好管理且能重複使用的元件。子工作流程是另一個 (父) 工作流程執行所啟動的工作流程執行。若要啟動子工作流程，父工作流程的決策者會使用 `StartChildWorkflowExecution` 決策。透過歷史記錄，子工作流程可以使用利用此決策所指定的輸入資料。

`StartChildWorkflowExecution` 決定的屬性也指定了子政策，也就是說，Amazon SWF 應如何處理父工作流程執行在子工作流程執行之前終止的情況。有三個可能值：

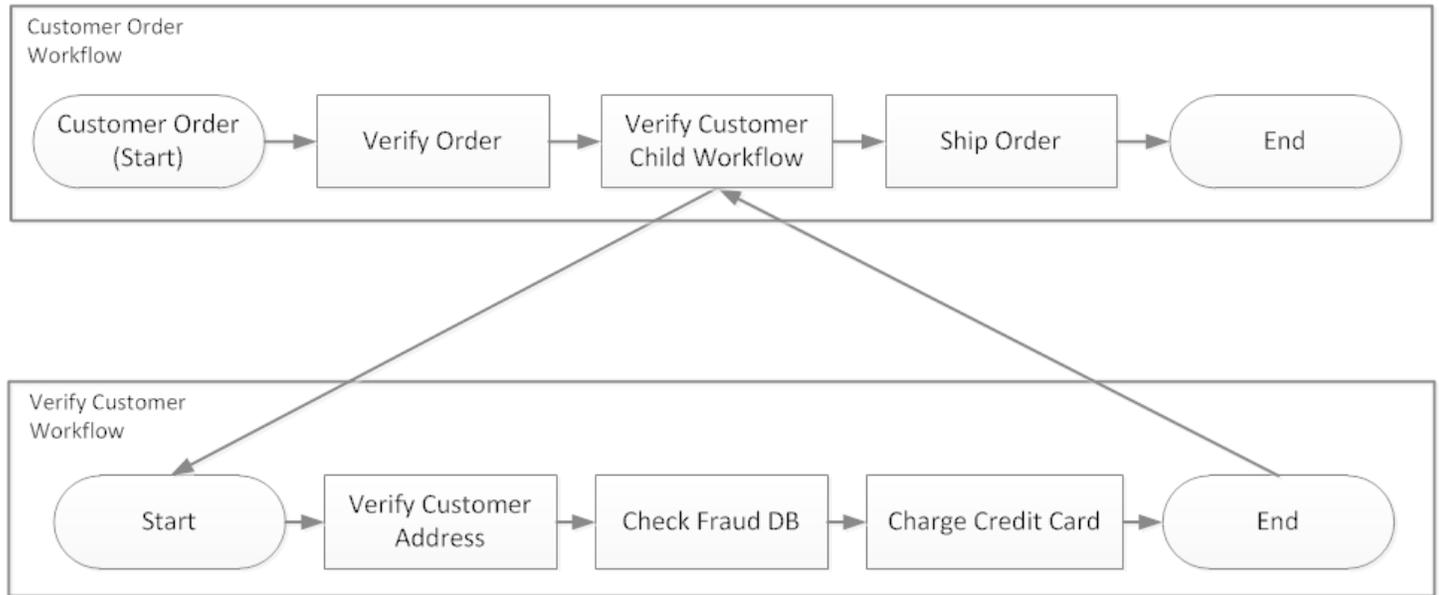
- 終止：Amazon SWF 將終止子項執行。
- 請求 \_ 取消：Amazon SWF 會嘗試透過在子系的工作流程執行歷程記錄中放置 `WorkflowExecutionCancelRequested` 事件來取消子項執行。
- 放棄：Amazon SWF 不會採取任何動作；子項執行將繼續執行。

子工作流程執行在啟動之後，會像一般執行一樣運作。完成時，Amazon SWF 會在父工作流程執行的工作流程歷程記錄中記錄完成及其結果。子工作流程範例如下：

- 不同網站中工作流程所使用的信用卡處理子工作流程
- 驗證客戶電子郵件地址、檢查選擇退出清單、傳送電子郵件以及驗證其未遭退信或失敗的電子郵件子工作流程。
- 可結合連線、設定、交易和驗證的資料庫儲存與擷取子工作流程。
- 可結合建置、封裝和驗證的來源碼編譯子工作流程。

在電子商務範例中，您可能想要將「使用信用卡付費」活動設為子工作流程。若要執行此作業，您可以註冊新的「驗證客戶」工作流程、註冊「驗證客戶地址」和「檢查詐欺資料庫」活動，以及定義任務的協調性邏輯。然後，「客戶訂單」工作流程中的決策者排定可指定此工作流程類型的 `StartChildWorkflowExecution` 決策，藉以啟動「驗證客戶」子工作流程。

下圖顯示包含新「驗證客戶」子工作流程的客戶訂單工作流程，包含檢查客戶地址、檢查詐欺資料庫，以及使用信用卡付費。



多個工作流程可以使用相同的工作流程類型來建立子工作流程執行。例如，「驗證客戶」子工作流程也可以用於組織的其他部分。子工作流程的事件是包含在自己的工作流程歷史記錄中，而不是包含在父工作流程歷史記錄中。

因為子工作流程只是決策者啟動的工作流程執行，所以啟動方式也如同一般獨立工作流程執行。

## 標記

有時候，您會想要將您使用案例中所使用的特定工作流程之資訊，記錄到工作流程歷史記錄中。標記能夠讓您將可供任何自訂或藍本專屬用途的資訊，記錄到工作流程執行歷史記錄中。

若要使用標記，決策程式會使用 `RecordMarker` 決策、命名標記、將所需資料附加至決策，以及使用 `RespondDecisionTaskCompleted` 動作通知 Amazon SWF。Amazon SWF 會接收請求、在工作流程歷程記錄中記錄標記，並在請求中制定任何其他決策。自此以後，決策者可以在工作流程歷史記錄中看到標記，並以您所設計的方式使用之。

記錄標記本身並不會啟動決策任務。為了防止工作流程執行停頓，必須發生某個事項，以持續執行工作流程。例如，這可能包含決策者排程另一個活動任務、工作流程執行接收訊號，或完成先前已排程的活動任務。

標記範例如下：

- 計算遞迴工作流程中迴圈數目的計數器。
- 根據活動結果的工作流程執行進度。
- 從先前工作流程歷史記錄事件摘要說明的資訊。

在電子商務範例中，您可以新增活動，每天檢查庫存量，並每次增加標記中的計數。接著，您可以新增決策邏輯，將電子郵件傳送給客戶，或在計數超過五時通知經理，而不需要檢閱整個歷史記錄。

在下列範例中，決策者完成決策任務，並回應包含 RecordMarker 決策的 RespondDecisionTaskCompleted 動作。

```
https://swf.us-east-1.amazonaws.com
RespondDecisionTaskCompleted
{
  "taskToken":"12342e17-80f6-FAKE-TASK-TOKEN32f0223",
  "decisions":[{"
    "decisionType":"RecordMarker",
    "recordMarkerDecisionAttributes":{"
      "markerName":"customer elected special shipping offer"
    }
  }],
}
```

如果 Amazon SWF 成功記錄標記，它會傳回與下列類似的成功 HTTP 回應。

```
HTTP/1.1 200 OK
Content-Length: 0
Content-Type: application/json
x-amzn-RequestId: 6c0373ce-074c-11e1-9083-8318c48dee96
```

## 標籤

Amazon SWF 支援標記工作流程執行。這一點在擁有許多資源時特別實用。

Amazon SWF 支援使用最多五個標籤來標記工作流程執行。每個標籤都是任意形式的字串，而且最長可能為 256 個字元。如果您想要使用標籤，則必須在啟動工作流程執行時指派它們。您無法在啟動工作流程執行之後將標籤新增至工作流程執行，也無法編輯或移除已指派給工作流程執行的標籤。

IAM 支援根據標籤來控制對 Amazon SWF 網域的存取。若要根據標籤控制存取，請在 IAM 政策的條件 (condition) 元素中，提供您標籤的資訊。

## 管理標籤

使用 AWS 開發套件或直接與 Amazon SWF API 互動來管理 Amazon 簡單工作流程服務標籤。使用 API，您可以在註冊網域時新增標籤、列出現有網域的標籤，以及新增或刪除現有網域的標籤。

### Note

每個資源限制為 50 個標籤。請參閱 [Amazon SWF 的一般帳戶配額](#)

- [RegisterDomain](#)
- [ListTagsForResource](#)
- [TagResource](#)
- [UntagResource](#)

如需詳細資訊，[使用 Amazon SWF API](#)，請參閱和 [Amazon 簡單工作流程服務 API 參考](#)。

## 標記工作流程執行

使用 Amazon SWF，您可以將標籤與工作流程執行產生關聯，然後根據這些標籤查詢工作流程執行。您可以在使用可見性作業時篩選 list。通過仔細選擇分配給執行項目的標籤，您可以使用它們來提供有意義的列表。

例如，假設您執行數個履行中心。使用標籤，您可以列出特定出貨中心中發生的流程。或者，如果客戶要轉換不同類型的媒體文件，則在轉換視頻，音頻和圖像文件時，標籤可能表示不同的過程。

在您使用 StartWorkflowExecution 動作、StartChildWorkflowExecution 決策或 ContinueAsNewWorkflowExecution 決策來啟動工作流程執行時，最多可以將五個標籤關聯至這些工作流程執行。當您使用可見度動作來列出或計算工作流程執行項目時，您可以根據標籤篩選結果。

## 使用標記

1. 設計標記策略。請考量您的商業需求，並建立對您有意義的標籤清單。判斷哪些執行會取得哪些標籤。即使最多可以將五個標籤指派給執行，您的標籤程式庫還是可以有任意數目的標籤。因為每個標籤可以是任意字串值，最長可達 256 個字元，所以標籤幾乎可以描述任何商業概念。
2. 在您建立執行時，標記執行的上限為最多五個標籤。
3. 使用 `ListOpenWorkflowExecutions`、`ListClosedWorkflowExecutions`、`CountClosedWorkflowExecutions` 動作指定 `tagFilter` 參數，以列出或計算使用特定標籤標記的執行項目。`CountOpenWorkflowExecutions` 動作將根據指定的標籤來篩選執行。

當您建立標籤與工作流程執行的關聯時，標籤便與該執行建立永久的關聯，無法予以移除。

您只能在具有 `ListWorkflowExecutions` 的 `tagFilter` 參數中指定一個標籤。另外，標籤比對區分大小寫，因此只會傳回完全相符的項目結果。

假設您已設定如下標記的兩個執行。

執行名稱	指派的標籤
Execution-One	Consumer, 2011-February
Execution-Two	Wholesale, 2011-March

您可以篩選 `ListOpenWorkflowExecutions` 對於 `Consumer` (消費者) 標籤所傳回的執行清單。`oldestDate` 和 `latestDate` 值指定為 [Unix 時間值](#)。

```
https://swf.us-east-1.amazonaws.com
RespondDecisionTaskCompleted
{
  "domain": "867530901",
  "startTimeFilter": {
    "oldestDate": 1262332800,
    "latestDate": 1325348400
  },
  "tagFilter": {
    "tag": "Consumer"
  }
}
```

## 使用標籤控制網域的存取

您可以在 IAM 中參考與 Amazon SWF 網域關聯的標籤，以控制對 Amazon 簡單工作流程服務網域的存取。例如，您可以限制包含索引鍵所在標籤 environment 且值為的網域 production：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": "swf:*",
      "Resource": "arn:aws:swf:*:123456789012:/domain/*",
      "Condition": {
        "StringEquals": {"aws:ResourceTag/environment": "production"}
      }
    }
  ]
}
```

此政策將 Deny (拒絕) 對已標記為 environment/production 之任何網域的存取權。

如需詳細資訊，請參閱：

- [使用 IAM 標籤控制存取](#)
- [標籤類型政策](#)

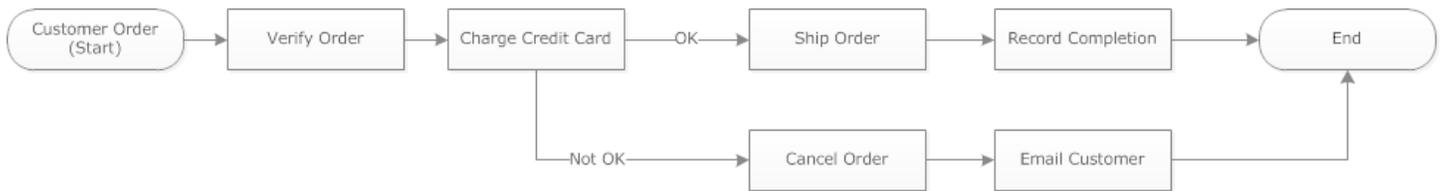
## 使用 Amazon Simple Workflow Service 實作互斥選項

在某些情況下，您可能想要根據先前活動的結果來排定一組不同的活動。互斥選項模式可讓您為複雜的應用程式要求建立相符的彈性工作流程。

Amazon Simple Workflow Service (Amazon SWF) 沒有特定的互斥選項操作。若要使用互斥選項，您只需要撰寫決策者邏輯，從而根據先前活動的結果做出不同的決策。互斥選項的一些應用如下：

- 如果先前活動的結果未成功，則執行清理活動
- 根據客戶購買基本還是進階方案來排定不同的活動
- 根據客戶的訂購歷史記錄來執行不同的客戶身份驗證活動

在電子商務範例中，您可以根據收取信用卡費用的結果，使用互斥選項來送出或取消訂單。在下圖中，如果成功收取信用卡費用，則決策者將排定「送出訂單」和「記錄完成」活動任務。否則，決策者將排定「取消訂單」和「發送電子郵件給客戶」活動任務。



如果成功收取信用卡費用，則決策者會排定 ShipOrder 活動。否則，決策者會排定 CancelOrder 活動。

在此情況下，以程式設計決策者解譯歷史記錄，並判斷是否成功收取信用卡費用。若要這麼做，您的邏輯可能與下面類似：

```

IF lastEvent = "WorkflowExecutionStarted"
  addToDecisions ScheduleActivityTask(ActivityType = "VerifyOrderActivity")

ELSIF lastEvent = "ActivityTaskCompleted"
  AND ActivityType = "VerifyOrderActivity"
  addToDecisions ScheduleActivityTask(ActivityType = "ChargeCreditCardActivity")

#Successful Credit Card Charge Activities
ELSIF lastEvent = "ActivityTaskCompleted"
  AND ActivityType = "ChargeCreditCardActivity"
  addToDecisions ScheduleActivityTask(ActivityType = "ShipOrderActivity")

ELSIF lastEvent = "ActivityTaskCompleted"
  AND ActivityType = "ShipOrderActivity"
  addToDecisions ScheduleActivityTask(ActivityType = "RecordOrderCompletionActivity")

ELSIF lastEvent = "ActivityTaskCompleted"
  AND ActivityType = "RecordOrderCompletionActivity"
  addToDecisions CompleteWorkflowExecution

#Unsuccessful Credit Card Charge Activities
ELSIF lastEvent = "ActivityTaskFailed"
  AND ActivityType = "ChargeCreditCardActivity"
  addToDecisions ScheduleActivityTask(ActivityType = "CancelOrderActivity")

ELSIF lastEvent = "ActivityTaskCompleted"
  AND ActivityType = "CancelOrderActivity"
  addToDecisions ScheduleActivityTask(ActivityType = "EmailCustomerActivity")
  
```

```
ELSIF lastEvent = "ActivityTaskCompleted"
    AND ActivityType = "EmailCustomerActivity"
    addToDecisions CompleteWorkflowExecution

ENDIF
```

如果成功收取信用卡費用，則決策者應該回應 RespondDecisionTaskCompleted 來排定 ShipOrder 活動。

```
https://swf.us-east-1.amazonaws.com
RespondDecisionTaskCompleted
{
  "taskToken": "12342e17-80f6-FAKE-TASK-TOKEN32f0223",
  "decisions":[
    {
      "decisionType":"ScheduleActivityTask",
      "scheduleActivityTaskDecisionAttributes":{
        "control":"OPTIONAL_DATA_FOR_DECIDER",
        "activityType":{
          "name":"ShipOrder",
          "version":"2.4"
        },
        "activityId":"3e2e6e55-e7c4-fee-deed-aa815722b7be",
        "scheduleToCloseTimeout":"3600",
        "taskList":{
          "name":"SHIPPING"
        },
        "scheduleToStartTimeout":"600",
        "startToCloseTimeout":"3600",
        "heartbeatTimeout":"300",
        "input": "123 Main Street, Anytown, United States"
      }
    }
  ]
}
```

如果未成功收取信用卡費用，則決策者應該回應 RespondDecisionTaskCompleted 來排定 CancelOrder 活動。

```
https://swf.us-east-1.amazonaws.com
RespondDecisionTaskCompleted
```

```
{
  "taskToken": "12342e17-80f6-FAKE-TASK-TOKEN32f0223",
  "decisions": [
    {
      "decisionType": "ScheduleActivityTask",
      "scheduleActivityTaskDecisionAttributes": {
        "control": "OPTIONAL_DATA_FOR_DECIDER",
        "activityType": {
          "name": "CancelOrder",
          "version": "2.4"
        },
        "activityId": "3e2e6e55-e7c4-fee-deed-aa815722b7be",
        "scheduleToCloseTimeout": "3600",
        "taskList": {
          "name": "CANCELLATIONS"
        },
        "scheduleToStartTimeout": "600",
        "startToCloseTimeout": "3600",
        "heartbeatTimeout": "300",
        "input": "Out of Stock"
      }
    }
  ]
}
```

如果 Amazon SWF 能驗證 RespondDecisionTaskCompleted 操作，Amazon SWF 會傳回與下列類似的成功 HTTP 響應。

```
HTTP/1.1 200 OK
Content-Length: 11
Content-Type: application/json
x-amzn-RequestId: 93cec6f7-0747-11e1-b533-79b402604df1
```

## Amazon Simple Workflow Service Pimple Work

計時器能讓您通知決策者已經過的時間長短。回應決策任務時，決策者可選擇使用 StartTimer 決策回應。此決策指定計時器在多長時間後應該觸發。在指定時間過去後，Amazon SWF 將添加 TimerFired 事件設置為工作流程執行歷史記錄，並排程決策任務。然後決策者可以使用此資訊通知進一步的決策。計時器的一個常見應用程式是延遲活動任務的執行。例如，客戶可能想要延遲交付某個項目。

## Amazon Simple Workflow Service 活動任務取消

活動任務取消可讓決策者結束不再需要執行的活動。Amazon SWF 使用合作取消機制，不會強制中斷正在執行的活動任務。您必須以程式設計活動工作者，處理取消請求。

決策者可以在活動任務處理決策任務之時，決定取消活動任務。若要取消活動任務，決策者會使用 RespondDecisionTaskCompleted 動作與 RequestCancelActivityTask 決策。

如果活動任務未有活動工作者取走，則服務會取消任務。請注意，可能的競爭條件是活動工作者隨時都可以取得任務。如果任務已指派給活動工作者，則會請求活動工作者取消任務。

在此範例中，會將取消訂單的訊號傳送給工作流程執行。

```
https://swf.us-east-1.amazonaws.com
SignalWorkflowExecution
{"domain": "867530901",
 "workflowId": "20110927-T-1",
 "runId": "9ba33198-4b18-4792-9c15-7181fb3a8852",
 "signalName": "CancelOrder",
 "input": "order 3553"}
```

如果工作流程執行收到訊號，Amazon SWF 會傳回與下列類似的成功 HTTP 回應。Amazon SWF 將會產生決策任務，以通知決策者處理訊號。

```
HTTP/1.1 200 OK
Content-Length: 0
Content-Type: application/json
x-amzn-RequestId: 6c0373ce-074c-11e1-9083-8318c48dee96
```

決策者處理決策任務並在歷史記錄中看到訊號時，決策者會嘗試取消具有 ShipOrderActivity0001 活動 ID 的未決活動。在排程活動任務事件的工作流程歷史記錄中，會提供活動 ID。

```
https://swf.us-east-1.amazonaws.com
RespondDecisionTaskCompleted
{
  "taskToken": "12342e17-80f6-FAKE-TASK-TOKEN32f0223",
  "decisions": [{
    "decisionType": "RequestCancelActivityTask",
    "RequestCancelActivityTaskDecisionAttributes": {
      "ActivityID": "ShipOrderActivity0001"
    }
  ]
}
```

```
    }  
  }  
]  
}
```

如果 Amazon SWF 成功收到取消請求，則會傳回與下列類似的成功 HTTP 回應：

```
HTTP/1.1 200 OK  
Content-Length: 0  
Content-Type: application/json  
x-amzn-RequestId: 6c0373ce-074c-11e1-9083-8318c48dee96
```

取消嘗試會以 `ActivityTaskCancelRequested` 事件形式載於歷史記錄中。

如果任務已成功取消，如 `ActivityTaskCanceled` 事件-對決策者進行編程，以便在取消任務之後採取適當的步驟，例如關閉工作流程執行。

如果無法取消活動任務（例如，任務完成、失敗或超時而不是取消），則決策者應接受活動的結果，或執行使用案例所需的任何清理或緩解。

如果活動任務已由活動工作者取得，則會透過任務活動訊號機制傳輸取消請求。活動工作人員可以定期使用 `RecordActivityTaskHeartbeat` 向 Amazon SWF 報告任務仍在進行中。

請注意，活動工作者不需要發出活動訊號，但建議針對長時間執行的任務發出活動訊號。任務取消需要記錄定期活動訊號；如果工作者未發出活動訊號，則無法取消任務。

如果決策者請求取消任務，則 Amazon SWF 會為 `cancelRequest` 對象設定為 `true`。`cancelRequest` 物件是 `ActivityTaskStatus` 物件的一部分，而服務會傳回後者以回應 `RecordActivityTaskHeartbeat`。

Amazon SWF 不會阻止其順利完成已請求取消的活動任務；活動必須自行判斷如何處理取消請求。依據您的需求，程式設計活動工作者，以取消活動任務或忽略取消請求。

如果您想要活動工作者指出已取消活動任務的工作，請程式設計該活動工作者以用 `RespondActivityTaskCanceled` 回應。如果您想要活動工作者完成任務，請程式設計該活動工作者以用標準 `RespondActivityTaskCompleted` 回應。

當 Amazon SWF 收到 `RespondActivityTaskCompleted` 或 `RespondActivityTaskCanceled` 請求時，會更新工作流程執行歷史記錄，並為決策任務排程以通知決策者。

程式設計決策者以處理決策任務，並傳回任何額外決策。如果成功取消活動任務，請程式設計決策者，以執行繼續或關閉工作流程執行所需的任務。如果未成功取消活動任務，請程式設計決策者，以接受結果、忽略結果，或為任何必要的清理排程。

# Amazon Simple Workflow Service

本節提供 Amazon Simple Workflow Service 的相關資訊。

## 主題

- [Amazon 簡易工作流程服務中的資料保護](#)
- [Amazon 簡單 workflow 服務中的 Identity and Access Management](#)
- [記錄和監控](#)
- [Amazon 簡易工作流程服務的合規驗證](#)
- [Amazon Simple Workflow Service 中的彈性](#)
- [Amazon Simple Workflow Service 的基礎設施安全](#)
- [Amazon Simple Workflow Service 中的組態與漏洞分析](#)

Amazon SWF 使用 IAM 控制對其他 AWS 服務和資源的存取。如需 IAM 運作方式的概觀，請參閱 [IAM 使用者指南中的存取管理概觀](#)。如需安全登入資料的概觀，請參閱 [AWS Amazon Web Services 一般參考](#)。

## Amazon 簡易工作流程服務中的資料保護

AWS [共同責任模型](#) 適用於 Amazon 簡單 workflow 服務中的資料保護。如此模型所述，AWS 負責保護執行所有 AWS 雲端的全球基礎設施。您負責維護在此基礎設施上託管內容的控制權。您也必須負責您所使用 AWS 服務的安全組態和管理任務。如需有關資料隱私權的更多相關資訊，請參閱 [資料隱私權常見問答集](#)。如需有關歐洲資料保護的相關資訊，請參閱 AWS 安全性部落格上的 [AWS 共同的責任模型](#) 和 [GDPR](#) 部落格文章。

基於資料保護目的，建議您使用 AWS IAM Identity Center 或 AWS Identity and Access Management (IAM) 保護 AWS 帳戶憑證，並設定個人使用者。如此一來，每個使用者都只會獲得授與完成其任務所必須的許可。我們也建議您採用下列方式保護資料：

- 每個帳戶均要使用多重要素驗證 (MFA)。
- 使用 SSL/TLS 與 AWS 資源通訊。我們需要 TLS 1.2 並建議使用 TLS 1.3。
- 使用 AWS CloudTrail 設定 API 和使用者活動日誌記錄。
- 使用 AWS 加密解決方案，以及 AWS 服務內的所有預設安全控制項。
- 使用進階的受管安全服務（例如 Amazon Macie），協助探索和保護儲存在 Amazon S3 的敏感資料。

- 如果您在透過命令列介面或 API 存取 AWS 時，需要 FIPS 140-2 驗證的加密模組，請使用 FIPS 端點。如需有關 FIPS 和 FIPS 端點的更多相關資訊，請參閱[聯邦資訊處理標準 \(FIPS\) 140-2 概觀](#)。

我們強烈建議您絕對不要將客戶的電子郵件地址等機密或敏感資訊，放在標籤或自由格式的文字欄位中，例如 Name (名稱) 欄位。這包括當您使用主控台、API 或 AWS 開發套件 AWS 服務使用 Amazon SWF 或其他軟體時。AWS CLI 您在標籤或自由格式文字欄位中輸入的任何資料都可能用於計費或診斷日誌。如果您提供外部伺服器的 URL，我們強烈建議請勿在驗證您對該伺服器請求的 URL 中包含憑證資訊。

## Amazon 簡單工作流程服務中的加密

### 靜態加密

Amazon SWF 一律會加密您的靜態資料。Amazon 簡單工作流程服務中的資料會使用透明伺服器端加密進行靜態加密。這可協助降低保護敏感資料所涉及的操作負擔和複雜性。您可以透過靜態加密，建立符合加密合規和法規要求，而且對安全性要求甚高的應用程式。

### 傳輸中加密

在 Amazon SWF 和其他服務之間傳遞的所有資料都會使用傳輸層安全性 (TLS) 加密。

## Amazon 簡單工作流服務中的 Identity and Access Management

存取 Amazon SWF 需要 AWS 可用來驗證您的請求的登入資料。這些認證必須具有存取 AWS 資源的權限，例如從其他 AWS 資源擷取事件資料。以下各節提供如何使用 [AWS Identity and Access Management \(IAM\)](#) 和 Amazon SWF 透過控制資源的存取來協助保護資源的詳細資訊。

AWS Identity and Access Management (IAM) 可協助系統管理員安全地控制 AWS 資源存取權。AWS 服務 IAM 管理員控制哪些人可以通過身份驗證 (登入) 和授權 (具有權限) 來使用 Amazon SWF 資源。IAM 是您可以使用的 AWS 服務，無需額外付費。

### 主題

- [物件](#)
- [使用身分驗證](#)
- [使用政策管理存取權](#)
- [存取控制](#)
- [Amazon SWF 的政策動作](#)

- [Amazon SWF 的政策資源](#)
- [Amazon SWF 的政策條件金鑰](#)
- [Amazon SWF 中的 ACL](#)
- [Amazon SWF 的 ABAC](#)
- [搭配 Amazon SWF 使用臨時登入資料](#)
- [Amazon SWF 的跨服務主體許可](#)
- [Amazon SWF 的服務角色](#)
- [Amazon SWF 的服務連結角色](#)
- [Amazon SWF 的基於身份的政策](#)
- [Amazon SWF 中基於資源的政策](#)
- [Amazon 簡單工作流程服務如何與 IAM 配合](#)
- [Amazon 簡易工作流程服務的身分型政策範例](#)
- [基本原則](#)
- [Amazon SWF IAM 政策](#)
- [API 摘要](#)
- [標籤類型政策](#)
- [Amazon SWF 的 VPC 端點 VPC 端點 VPC 端點](#)
- [疑難排解 Amazon 簡單工作流程服務身分識別](#)

## 物件

您使用 AWS Identity and Access Management (IAM) 的方式會因您在 Amazon SWF 中執行的工作而有所不同。

**服務使用者** — 如果您使用 Amazon SWF 服務執行工作，則管理員會為您提供所需的登入資料和許可。當您使用更多 Amazon SWF 功能完成工作時，可能需要額外的許可。了解存取許可的管理方式可協助您向管理員請求正確的許可。如果您無法存取 Amazon SWF 中的某個功能，請參閱[疑難排解 Amazon 簡單工作流程服務身分識別](#)。

**服務管理員** — 如果您負責公司的 Amazon SWF 資源，您可能可以完整存取 Amazon SWF。您的任務是決定服務使用者應存取哪些 Amazon SWF 功能和資源。接著，您必須將請求提交給您的 IAM 管理員，來變更您服務使用者的許可。檢閱此頁面上的資訊，了解 IAM 的基本概念。若要進一步了解貴公司如何將 IAM 與 Amazon SWF 搭配使用，請參閱[Amazon 簡單工作流程服務如何與 IAM 配合](#)。

IAM 管理員 — 如果您是 IAM 管理員，您可能想要了解如何撰寫政策以管理 Amazon SWF 存取權的詳細資訊。若要檢視可在 IAM 中使用的 Amazon SWF 以身分識別為基礎的政策範例，請參閱 [Amazon 簡易工作流程服務的身分型政策範例](#)

## 使用身分驗證

驗證是您 AWS 使用身分認證登入的方式。您必須以 IAM 使用者身分或假設 IAM 角色進行驗證 (登入 AWS)。AWS 帳戶根使用者

您可以使用透過 AWS 身分識別來源提供的認證，以聯合身分識別身分登入。AWS IAM Identity Center (IAM 身分中心) 使用者、貴公司的單一登入身分驗證，以及您的 Google 或 Facebook 登入資料都是聯合身分識別的範例。您以聯合身分登入時，您的管理員先前已設定使用 IAM 角色的聯合身分。當您使 AWS 用同盟存取時，您會間接擔任角色。

根據您的使用者類型，您可以登入 AWS Management Console 或 AWS 存取入口網站。如需有關登入的詳細資訊 AWS，請參閱《AWS 登入 使用指南》AWS 帳戶中 [的如何登入](#) 您的。

如果您 AWS 以程式設計方式存取，請 AWS 提供軟體開發套件 (SDK) 和命令列介面 (CLI)，以使用您的認證以加密方式簽署要求。如果您不使用 AWS 工具，則必須自行簽署要求。如需使用建議的方法自行簽署請求的詳細資訊，請參閱 IAM 使用者指南中的 [簽署 AWS API 請求](#)。

無論您使用何種身分驗證方法，您可能都需要提供額外的安全性資訊。例如，AWS 建議您使用多重要素驗證 (MFA) 來增加帳戶的安全性。如需更多資訊，請參閱 AWS IAM Identity Center 使用者指南中的 [多重要素驗證](#) 和 IAM 使用者指南中的 [在 AWS 中使用多重要素驗證 \(MFA\)](#)。

## AWS 帳戶 根使用者

當您建立時 AWS 帳戶，您會從一個登入身分開始，該身分可完整存取該帳戶中的所有資源 AWS 服務和資源。此身分稱為 AWS 帳戶 root 使用者，可透過使用您用來建立帳戶的電子郵件地址和密碼登入來存取。強烈建議您不要以根使用者處理日常任務。保護您的根使用者憑證，並將其用來執行只能由根使用者執行的任務。如需這些任務的完整清單，了解需以根使用者登入的任務，請參閱 IAM 使用者指南中的 [需要根使用者憑證的任務](#)。

## 聯合身分

最佳作法是要求人類使用者 (包括需要系統管理員存取權的使用者) 使用與身分識別提供者的同盟，才能使用臨時認證 AWS 服務 來存取。

聯合身分識別是來自企業使用者目錄的使用者、Web 身分識別提供者、Identity Center 目錄，或使用透過身分識別來源提供的認證進行存取 AWS 服務 的任何使用者。AWS Directory Service 同盟身分存取時 AWS 帳戶，他們會假設角色，而角色則提供臨時認證。

對於集中式存取權管理，我們建議您使用 AWS IAM Identity Center。您可以在 IAM Identity Center 中建立使用者和群組，也可以連線並同步到自己身分識別來源中的一組使用者和群組，以便在所有應用程式 AWS 帳戶 和應用程式中使用。如需 IAM Identity Center 的詳細資訊，請參閱 AWS IAM Identity Center 使用者指南中的 [什麼是 IAM Identity Center？](#)。

## IAM 使用者和群組

[IAM 使用者](#)是您內部的身分，具 AWS 帳戶 有單一人員或應用程式的特定許可。建議您盡可能依賴暫時憑證，而不是擁有建立長期憑證 (例如密碼和存取金鑰) 的 IAM 使用者。但是如果特定使用案例需要擁有長期憑證的 IAM 使用者，建議您輪換存取金鑰。如需更多資訊，請參閱 [IAM 使用者指南](#)中的為需要長期憑證的使用案例定期輪換存取金鑰。

[IAM 群組](#)是一種指定 IAM 使用者集合的身分。您無法以群組身分簽署。您可以使用群組來一次為多名使用者指定許可。群組可讓管理大量使用者許可的程序變得更為容易。例如，您可以擁有一個名為 IAMAdmins 的群組，並給予該群組管理 IAM 資源的許可。

使用者與角色不同。使用者只會與單一人員或應用程式建立關聯，但角色的目的是在由任何需要它的人員取得。使用者擁有永久的長期憑證，但角色僅提供暫時憑證。如需進一步了解，請參閱IAM 使用者指南中的[建立 IAM 使用者 \(而非角色\) 的時機](#)。

## IAM 角色

[IAM 角色](#)是您 AWS 帳戶 內部具有特定許可的身分。它類似 IAM 使用者，但不與特定的人員相關聯。您可以[切換角色，在中暫時擔任 IAM 角色](#)。AWS Management Console 您可以透過呼叫 AWS CLI 或 AWS API 作業或使用自訂 URL 來擔任角色。如需使用角色的方法詳細資訊，請參閱 IAM 使用者指南中的[使用 IAM 角色](#)。

使用暫時憑證的 IAM 角色在下列情況中非常有用：

- 聯合身分使用者存取 — 如需向聯合身分指派許可，請建立角色，並為角色定義許可。當聯合身分進行身分驗證時，該身分會與角色建立關聯，並獲授予由角色定義的許可。如需有關聯合角色的相關資訊，請參閱 [IAM 使用者指南](#)中的為第三方身分提供者建立角色。如果您使用 IAM Identity Center，則需要設定許可集。為控制身分驗證後可以存取的內容，IAM Identity Center 將許可集與 IAM 中的角色相關聯。如需有關許可集的資訊，請參閱 AWS IAM Identity Center 使用者指南中的[許可集](#)。
- 暫時 IAM 使用者許可 – IAM 使用者或角色可以擔任 IAM 角色來暫時針對特定任務採用不同的許可。
- 跨帳戶存取權：您可以使用 IAM 角色，允許不同帳戶中的某人 (信任的主體) 存取您帳戶的資源。角色是授予跨帳戶存取權的主要方式。但是，對於某些策略 AWS 服務，您可以將策略直接附加到資源 (而不是使用角色作為代理)。若要了解跨帳戶存取角色和以資源為基礎的政策之間的差異，請參閱 IAM 使用者指南中的 [IAM 中的跨帳戶資源存取](#)。

- 跨服務訪問 — 有些 AWS 服務 使用其他 AWS 服務功能。例如，當您在服務中進行呼叫時，該服務通常會在 Amazon EC2 中執行應用程式或將物件儲存在 Amazon Simple Storage Service (Amazon S3) 中。服務可能會使用呼叫主體的許可、使用服務角色或使用服務連結角色來執行此作業。
- 轉寄存取工作階段 (FAS) — 當您使用 IAM 使用者或角色在中執行動作時 AWS，您會被視為主體。使用某些服務時，您可能會執行某個動作，進而在不同服務中啟動另一個動作。FAS 會使用主體呼叫的權限 AWS 服務，並結合要求 AWS 服務 向下游服務發出要求。只有當服務收到需要與其 AWS 服務 他資源互動才能完成的請求時，才會發出 FAS 請求。在此情況下，您必須具有執行這兩個動作的許可。如需提出 FAS 請求時的政策詳細資訊，請參閱 [《轉發存取工作階段》](#)。
- 服務角色 – 服務角色是服務擔任的 [IAM 角色](#)，可代表您執行動作。IAM 管理員可以從 IAM 內建立、修改和刪除服務角色。如需詳細資訊，請參閱 IAM 使用者指南中的 [建立角色以委派許可給 AWS 服務服務](#)。
- 服務連結角色 — 服務連結角色是連結至 AWS 服務服務可以擔任代表您執行動作的角色。服務連結角色會顯示在您的中，AWS 帳戶 且屬於服務所有。IAM 管理員可以檢視，但不能編輯服務連結角色的許可。
- 在 Amazon EC2 上執行的應用程式 — 您可以使用 IAM 角色來管理在 EC2 執行個體上執行的應用程式以及發出 AWS CLI 或 AWS API 請求的臨時登入資料。這是在 EC2 執行個體內儲存存取金鑰的較好方式。若要將 AWS 角色指派給 EC2 執行個體並提供給其所有應用程式，請建立連接至執行個體的執行個體設定檔。執行個體設定檔包含該角色，並且可讓 EC2 執行個體上執行的程式取得暫時憑證。如需詳細資訊，請參閱 IAM 使用者指南中的 [利用 IAM 角色來授予許可給 Amazon EC2 執行個體上執行的應用程式](#)。

如需了解是否要使用 IAM 角色或 IAM 使用者，請參閱 IAM 使用者指南中的 [建立 IAM 角色 \(而非使用者\) 的時機](#)。

## 使用政策管理存取權

您可以透 AWS 過建立原則並將其附加至 AWS 身分識別或資源來控制中的存取。原則是一個物件 AWS，當與身分識別或資源相關聯時，會定義其權限。AWS 當主參與者 (使用者、root 使用者或角色工作階段) 提出要求時，評估這些原則。政策中的許可決定是否允許或拒絕請求。大多數原則會 AWS 以 JSON 文件的形式儲存在中。如需 JSON 政策文件結構和內容的詳細資訊，請參閱 IAM 使用者指南中的 [JSON 政策概觀](#)。

管理員可以使用 AWS JSON 政策來指定誰可以存取哪些內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

預設情況下，使用者和角色沒有許可。若要授予使用者對其所需資源執行動作的許可，IAM 管理員可以建立 IAM 政策。然後，管理員可以將 IAM 政策新增至角色，使用者便能擔任這些角色。

IAM 政策定義該動作的許可，無論您使用何種方法來執行操作。例如，假設您有一個允許 `iam:GetRole` 動作的政策。具有該原則的使用者可以從 AWS Management Console AWS CLI、或 AWS API 取得角色資訊。

## 身分型政策

身分型政策是可以附加到身分 (例如 IAM 使用者、使用者群組或角色) 的 JSON 許可政策文件。這些政策可控制身分在何種條件下能對哪些資源執行哪些動作。若要了解如何建立身分類型政策，請參閱 IAM 使用者指南中的 [建立 IAM 政策](#)。

身分型政策可進一步分類成內嵌政策或受管政策。內嵌政策會直接內嵌到單一使用者、群組或角色。受管理的策略是獨立策略，您可以將其附加到您的 AWS 帳戶。受管政策包括 AWS 受管政策和客戶管理的策略。如需了解如何在受管政策及內嵌政策間選擇，請參閱 IAM 使用者指南中的 [在受管政策和內嵌政策間選擇](#)。

## 資源型政策

資源型政策是連接到資源的 JSON 政策文件。資源型政策的最常見範例是 IAM 角色信任政策和 Amazon S3 儲存貯體政策。在支援資源型政策的服務中，服務管理員可以使用它們來控制對特定資源的存取權限。對於附加政策的資源，政策會定義指定的主體可以對該資源執行的動作以及在何種條件下執行的動作。您必須在資源型政策中 [指定主體](#)。主參與者可以包括帳戶、使用者、角色、同盟使用者或。AWS 服務

資源型政策是位於該服務中的內嵌政策。您無法在以資源為基礎的政策中使用 IAM 的 AWS 受管政策。

## 存取控制清單 (ACL)

存取控制清單 (ACL) 可控制哪些主體 (帳戶成員、使用者或角色) 擁有存取某資源的許可。ACL 類似於資源型政策，但它們不使用 JSON 政策文件格式。

Amazon S3 和 Amazon VPC 是支援 ACL 的服務範例。AWS WAF 如需進一步了解 ACL，請參閱 Amazon Simple Storage Service 開發人員指南中的 [存取控制清單 \(ACL\) 概觀](#)。

## 其他政策類型

AWS 支援其他較不常見的原則類型。這些政策類型可設定較常見政策類型授予您的最大許可。

- 許可界限 – 許可範圍是一種進階功能，可供您設定身分型政策能授予 IAM 實體 (IAM 使用者或角色) 的最大許可。您可以為實體設定許可界限。所產生的許可會是實體的身分型政策和其許可界限的交集。會在 Principal 欄位中指定使用者或角色的資源型政策則不會受到許可界限限制。所有這類政

策中的明確拒絕都會覆寫該允許。如需許可界限的詳細資訊，請參閱 IAM 使用者指南中的 [IAM 實體許可界限](#)。

- 服務控制策略 (SCP) — SCP 是 JSON 策略，用於指定中組織或組織單位 (OU) 的最大權限。AWS Organizations 是一種用於分組和集中管理您企業擁有的多個 AWS 帳戶的服務。若您啟用組織中的所有功能，您可以將服務控制政策 (SCP) 套用到任何或所有帳戶。SCP 限制成員帳戶中實體的權限，包括每個 AWS 帳戶根使用者帳戶。如需 Organizations 和 SCP 的詳細資訊，請參閱 AWS Organizations 使用者指南中的 [SCP 運作方式](#)。
- 工作階段政策 – 工作階段政策是一種進階政策，您可以在透過編寫程式的方式建立角色或聯合使用者的暫時工作階段時，作為參數傳遞。所產生工作階段的許可會是使用者或角色的身分型政策和工作階段政策的交集。許可也可以來自資源型政策。所有這類政策中的明確拒絕都會覆寫該允許。如需詳細資訊，請參閱 IAM 使用者指南中的 [工作階段政策](#)。

## 多種政策類型

將多種政策類型套用到請求時，其結果形成的許可會更為複雜、更加難以理解。要了解如何在涉及多個政策類型時 AWS 確定是否允許請求，請參閱《IAM 使用者指南》中的 [政策評估邏輯](#)。

## 存取控制

您可以擁有有效的登入資料來驗證請求，但除非您擁有許可，否則無法建立或存取 Amazon SWF 資源。例如，您必須擁有與 Amazon SWF 規則相關聯的叫用 AWS Lambda、亞馬遜簡單通知服務 (Amazon SNS) 和亞馬遜簡單佇列服務 (Amazon SQS) 目標的許可。

以下各節說明如何管理 Amazon SWF 的許可。我們建議您先閱讀概觀。

- [基本原則](#)
- [Amazon SWF IAM 政策](#)
- [Amazon SWF 政策示例](#)

## Amazon SWF 的政策動作

支援政策動作 是

管理員可以使用 AWS JSON 政策來指定誰可以存取哪些內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

JSON 政策的 Action 元素描述您可以用來允許或拒絕政策中存取的動作。原則動作通常與關聯的 AWS API 作業具有相同的名稱。有一些例外狀況，例如沒有相符的 API 操作的僅限許可動作。也有一些作業需要政策中的多個動作。這些額外的動作稱為相依動作。

政策會使用動作來授予執行相關聯動作的許可。

若要查看 Amazon SWF 動作清單，請參閱服務授權參考資料中的 Amazon 簡單工作流程服務 [定義](#) 的資源。

Amazon SWF 中的政策動作會在動作之前使用下列前置詞：

```
swf
```

若要在單一陳述式中指定多個動作，請用逗號分隔。

```
"Action": [  
  "swf:action1",  
  "swf:action2"  
]
```

若要檢視 Amazon SWF 以身分識別為基礎的政策範例，請參閱 [Amazon 簡易工作流程服務的身分型政策範例](#)

## Amazon SWF 的政策資源

支援政策資源 是

管理員可以使用 AWS JSON 政策來指定誰可以存取哪些內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

Resource JSON 政策元素可指定要套用動作的物件。陳述式必須包含 Resource 或 NotResource 元素。最佳實務是使用其 [Amazon Resource Name \(ARN\)](#) 來指定資源。您可以針對支援特定資源類型的動作 (稱為資源層級許可) 來這麼做。

對於不支援資源層級許可的動作 (例如列出操作)，請使用萬用字元 (\*) 來表示陳述式適用於所有資源。

```
"Resource": "*"
```

若要查看 Amazon SWF 資源類型及其 ARN 的清單，請參閱服務授權參考資料中的 [Amazon 簡單工作流程服務定義的動作](#)。若要了解可以使用哪些動作指定每個資源的 ARN，請參閱 [Amazon 簡單工作流程服務定義的資源](#)。

若要檢視 Amazon SWF 以身分識別為基礎的政策範例，請參閱 [Amazon 簡易工作流程服務的身分型政策範例](#)

## Amazon SWF 的政策條件金鑰

支援服務特定政策條件金鑰	是
--------------	---

管理員可以使用 AWS JSON 政策來指定誰可以存取哪些內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

Condition 元素 (或 Condition 區塊) 可讓您指定使陳述式生效的條件。Condition 元素是選用項目。您可以建立使用 [條件運算子](#) 的條件運算式 (例如等於或小於)，來比對政策中的條件和請求中的值。

若您在陳述式中指定多個 Condition 元素，或是在單一 Condition 元素中指定多個索引鍵，AWS 會使用邏輯 AND 操作評估他們。如果您為單一條件索引鍵指定多個值，請使用邏輯 OR 運算來 AWS 評估條件。必須符合所有條件，才會授與陳述式的許可。

您也可以在指定條件時使用預留位置變數。例如，您可以只在使用者使用其 IAM 使用者名稱標記時，將存取資源的許可授予該 IAM 使用者。如需更多資訊，請參閱 IAM 使用者指南中的 [IAM 政策元素：變數和標籤](#)。

AWS 支援全域條件金鑰和服務特定條件金鑰。若要查看所有 AWS 全域條件金鑰，請參閱《IAM 使用者指南》中的 [AWS 全域條件內容金鑰](#)。

若要查看 Amazon SWF 條件金鑰清單，請參閱服務授權參考中的 Amazon 簡單工作流程服務的 [條件金鑰](#)。若要了解可以使用條件金鑰的動作和資源，請參閱 [Amazon 簡單工作流程服務定義的資源](#)。

若要檢視 Amazon SWF 以身分識別為基礎的政策範例，請參閱 [Amazon 簡易工作流程服務的身分型政策範例](#)

## Amazon SWF 中的 ACL

支援 ACL	否
--------	---

存取控制清單 (ACL) 可控制哪些主體 (帳戶成員、使用者或角色) 擁有存取某資源的許可。ACL 類似於資源型政策，但它們不使用 JSON 政策文件格式。

## Amazon SWF 的 ABAC

支援 ABAC (政策中的標籤)

部分

屬性型存取控制 (ABAC) 是一種授權策略，可根據屬性來定義許可。在中 AWS，這些屬性稱為標籤。您可以將標籤附加到 IAM 實體 (使用者或角色) 和許多 AWS 資源。為實體和資源加上標籤是 ABAC 的第一步。您接著要設計 ABAC 政策，允許在主體的標籤與其嘗試存取的資源標籤相符時操作。

ABAC 在成長快速的環境中相當有幫助，並能在政策管理變得繁瑣時提供協助。

如需根據標籤控制存取，請使用 `aws:ResourceTag/key-name`、`aws:RequestTag/key-name` 或 `aws:TagKeys` 條件索引鍵，在政策的 [條件元素](#) 中，提供標籤資訊。

如果服務支援每個資源類型的全部三個條件金鑰，則對該服務而言，值為 Yes。如果服務僅支援某些資源類型的全部三個條件金鑰，則值為 Partial。

如需 ABAC 的詳細資訊，請參閱 IAM 使用者指南中的 [什麼是 ABAC?](#)。如要查看含有設定 ABAC 步驟的教學課程，請參閱 IAM 使用者指南中的 [使用屬性型存取控制 \(ABAC\)](#)。

## 搭配 Amazon SWF 使用臨時登入資料

支援臨時憑證

是

當您使用臨時憑據登錄時，某些 AWS 服務不起作用。如需其他資訊，包括哪些 AWS 服務與臨時登入資料 [搭配 AWS 服務使用](#)，請參閱 IAM 使用者指南中的 IAM。

如果您使用除了使用者名稱和密碼以外的任何方法登入，則您正在 AWS Management Console 使用臨時認證。例如，當您 AWS 使用公司的單一登入 (SSO) 連結存取時，該程序會自動建立暫時認證。當您以使用者身分登入主控台，然後切換角色時，也會自動建立臨時憑證。如需切換角色的詳細資訊，請參閱 IAM 使用者指南中的 [切換至角色 \(主控台\)](#)。

您可以使用 AWS CLI 或 AWS API 手動建立臨時登入資料。然後，您可以使用這些臨時登入資料來存取 AWS。AWS 建議您動態產生臨時登入資料，而不是使用長期存取金鑰。如需詳細資訊，請參閱 [IAM 中的暫時性安全憑證](#)。

## Amazon SWF 的跨服務主體許可

支援轉寄存取工作階段 (FAS) 是

當您使用 IAM 使用者或角色在中執行動作時 AWS，您會被視為主體。使用某些服務時，您可能會執行某個動作，進而在不同服務中啟動另一個動作。FAS 會使用主體呼叫的權限 AWS 服務，並結合要求 AWS 服務 向下游服務發出要求。只有當服務收到需要與其 AWS 服務 他資源互動才能完成的請求時，才會發出 FAS 請求。在此情況下，您必須具有執行這兩個動作的許可。如需提出 FAS 請求時的政策詳細資訊，請參閱 [《轉發存取工作階段》](#)。

## Amazon SWF 的服務角色

支援服務角色 是

服務角色是服務擔任的 [IAM 角色](#)，可代您執行動作。IAM 管理員可以從 IAM 內建立、修改和刪除服務角色。如需詳細資訊，請參閱 IAM 使用者指南中的 [建立角色以委派許可給 AWS 服務服務](#)。

### Warning

變更服務角色的權限可能會中斷 Amazon SWF 的功能。只有在 Amazon SWF 提供指導時，才能編輯服務角色。

## Amazon SWF 的服務連結角色

支援服務連結角色。 否

服務連結角色是一種連結至 AWS 服務服務可以擔任代表您執行動作的角色。服務連結角色會顯示在您的中，AWS 帳戶 且屬於服務所有。IAM 管理員可以檢視，但不能編輯服務連結角色的許可。

如需建立或管理服務連結角色的詳細資訊，請參閱 [可搭配 IAM 運作的 AWS 服務](#)。在表格中尋找服務，其中包含服務連結角色欄中的 Yes。選擇是連結，以檢視該服務的服務連結角色文件。

## Amazon SWF 的基於身份的政策

支援身分型政策

是

身分型政策是可以連接到身分 (例如 IAM 使用者、使用者群組或角色) 的 JSON 許可政策文件。這些政策可控制身分在何種條件下能對哪些資源執行哪些動作。若要了解如何建立身分類型政策，請參閱《IAM 使用者指南》中的[建立 IAM 政策](#)。

使用 IAM 身分型政策，您可以指定允許或拒絕的動作和資源，以及在何種條件下允許或拒絕動作。您無法在身分型政策中指定主體，因為這會套用至連接的使用者或角色。如要了解您在 JSON 政策中使用的所有元素，請參閱《IAM 使用者指南》中的[IAM JSON 政策元素參考](#)。

### Amazon SWF 的基於身份的政策範例

若要檢視 Amazon SWF 以身分識別為基礎的政策範例，請參閱。[Amazon 簡易工作流程服務的身分型政策範例](#)

## Amazon SWF 中基於資源的政策

支援以資源基礎的政策

否

資源型政策是附加到資源的 JSON 政策文件。資源型政策的最常見範例是 IAM 角色信任政策和 Amazon S3 儲存貯體政策。在支援資源型政策的服務中，服務管理員可以使用它們來控制對特定資源的存取權限。對於附加政策的資源，政策會定義指定的主體可以對該資源執行的動作以及在何種條件下執行的動作。您必須在資源型政策中[指定主體](#)。主參與者可以包括帳戶、使用者、角色、同盟使用者或。AWS 服務

如需啟用跨帳戶存取權，您可以指定在其他帳戶內的所有帳戶或 IAM 實體，作為資源型政策的主體。新增跨帳戶主體至資源型政策，只是建立信任關係的一半。當主體和資源位於不同時 AWS 帳戶，受信任帳戶中的 IAM 管理員也必須授與主體實體 (使用者或角色) 權限，才能存取資源。其透過將身分型政策連接到實體來授與許可。不過，如果資源型政策會為相同帳戶中的主體授予存取，這時就不需要額外的身分型政策。如需詳細資訊，請參閱 IAM 使用者指南中的[IAM 中的跨帳戶資源存取](#)。

### Amazon 簡單工作流程服務如何與 IAM 配合

在您使用 IAM 管理 Amazon SWF 的存取權限之前，請先了解哪些 IAM 功能可以搭配 Amazon SWF 使用。

您可以搭配 Amazon 簡易工作流程服務使用的 IAM 功能

IAM 功能	Amazon SWF 支持
<a href="#">身分型政策</a>	是
<a href="#">資源型政策</a>	否
<a href="#">政策動作</a>	是
<a href="#">政策資源</a>	是
<a href="#">政策條件索引鍵 (服務特定)</a>	是
<a href="#">ACL</a>	否
<a href="#">ABAC(政策中的標籤)</a>	部分
<a href="#">臨時憑證</a>	是
<a href="#">主體許可</a>	是
<a href="#">服務角色</a>	是
<a href="#">服務連結角色</a>	否

若要深入瞭解 Amazon SWF 和其他 AWS 服務如何搭配大多數 IAM 功能使用，請參閱 IAM 使用者指南中的搭配 IAM 使用的[AWS 服務](#)。

## Amazon 簡易工作流程服務的身分型政策範例

依預設，使用者和角色沒有建立或修改 Amazon SWF 資源的權限。他們也無法使用 AWS Management Console、AWS Command Line Interface (AWS CLI) 或 AWS API 來執行工作。若要授予使用者對其所需資源執行動作的許可，IAM 管理員可以建立 IAM 政策。然後，管理員可以將 IAM 政策新增至角色，使用者便能擔任這些角色。

若要了解如何使用這些範例 JSON 政策文件建立 IAM 身分型政策，請參閱《IAM 使用者指南》中的[建立 IAM 政策](#)。

如需 Amazon SWF 定義的動作和資源類型的詳細資訊，包括每種資源類型的 ARN 格式，請參閱服務授權參考中的[Amazon 簡單工作流程服務的動作、資源和條件金鑰](#)。

## 主題

- [政策最佳實務](#)
- [使用 Amazon SWF 控制台](#)
- [允許使用者檢視他們自己的許可](#)

## 政策最佳實務

以身分識別為基礎的政策可決定使用者是否可以在您的帳戶中建立、存取或刪除 Amazon SWF 資源。這些動作可能會讓您的 AWS 帳戶產生費用。當您建立或編輯身分型政策時，請遵循下列準則及建議事項：

- 開始使用 AWS 受管原則並邁向最低權限權限 — 若要開始將權限授與使用者和工作負載，請使用可授與許多常見使用案例權限的 AWS 受管理原則。它們可用在您的 AWS 帳戶。建議您透過定義特定於您使用案例的 AWS 客戶管理政策，進一步降低使用權限。如需更多資訊，請參閱 IAM 使用者指南中的 [AWS 受管政策](#) 或 [任務職能的 AWS 受管政策](#)。
- 套用最低權限許可 – 設定 IAM 政策的許可時，請僅授予執行任務所需的許可。為實現此目的，您可以定義在特定條件下可以對特定資源採取的動作，這也稱為最低權限許可。如需使用 IAM 套用許可的更多相關資訊，請參閱 IAM 使用者指南中的 [IAM 中的政策和許可](#)。
- 使用 IAM 政策中的條件進一步限制存取權 – 您可以將條件新增至政策，以限制動作和資源的存取。例如，您可以撰寫政策條件，指定必須使用 SSL 傳送所有請求。您也可以使用條件來授與對服務動作的存取權 (如透過特定 AWS 服務) 使用 AWS CloudFormation。如需詳細資訊，請參閱 IAM 使用者指南中的 [IAM JSON 政策元素：條件](#)。
- 使用 IAM Access Analyzer 驗證 IAM 政策，確保許可安全且可正常運作 – IAM Access Analyzer 驗證新政策和現有政策，確保這些政策遵從 IAM 政策語言 (JSON) 和 IAM 最佳實務。IAM Access Analyzer 提供 100 多項政策檢查及切實可行的建議，可協助您編寫安全且實用的政策。如需更多資訊，請參閱 IAM 使用者指南中的 [IAM Access Analyzer 政策驗證](#)。
- 需要多因素身份驗證 (MFA) — 如果您的案例需要 IAM 使用者或根使用者 AWS 帳戶，請開啟 MFA 以獲得額外的安全性。如需在呼叫 API 操作時請求 MFA，請將 MFA 條件新增至您的政策。如需更多資訊，請參閱 [IAM 使用者指南](#) 中的設定 MFA 保護的 API 存取。

如需 IAM 中最佳實務的相關資訊，請參閱 IAM 使用者指南中的 [IAM 安全最佳實務](#)。

## 使用 Amazon SWF 控制台

若要存取 Amazon 簡單工作流程服務主控台，您必須擁有最少一組許可。這些權限必須允許您列出和檢視有關 AWS 帳戶。如果您建立比最基本必要許可更嚴格的身分型政策，則對於具有該政策的實體（使用者或角色）而言，主控台就無法如預期運作。

您不需要為僅對 AWS CLI 或 AWS API 進行呼叫的使用者允許最低主控台權限。反之，只需允許存取符合他們嘗試執行之 API 操作的動作就可以了。

為確保使用者和角色仍然可以使用 Amazon SWF 主控台，請同時將 Amazon SWF *ConsoleAccess* 或 *ReadOnly* AWS 受管政策附加到實體。如需詳細資訊，請參閱《IAM 使用者指南》中的[新增許可到使用者](#)。

### 允許使用者檢視他們自己的許可

此範例會示範如何建立政策，允許 IAM 使用者檢視附加到他們使用者身分的內嵌及受管政策。此原則包含在主控台上或以程式設計方式使用 AWS CLI 或 AWS API 完成此動作的權限。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupsWithUser",
        "iam:ListAttachedUserPolicies",
        "iam:ListUserPolicies",
        "iam:GetUser"
      ],
      "Resource": ["arn:aws:iam::*:user/${aws:username}"]
    },
    {
      "Sid": "NavigateInConsole",
      "Effect": "Allow",
      "Action": [
        "iam:GetGroupPolicy",
        "iam:GetPolicyVersion",
        "iam:GetPolicy",
        "iam:ListAttachedGroupPolicies",
        "iam:ListGroupPolicies",

```

```
        "iam:ListPolicyVersions",
        "iam:ListPolicies",
        "iam:ListUsers"
    ],
    "Resource": "*"
}
]
```

## 基本原則

Amazon SWF 存取控制主要以兩種類型的權限為基礎：

- 資源許可：使用者可以存取哪些 Amazon SWF 資源。  
您只能表達網域的資源許可。
- API 權限：使用者可以呼叫哪些 Amazon SWF 動作。

最簡單的方法是授予完整帳戶存取權限 (呼叫任何網域中的任何 Amazon SWF 動作)，或是完全拒絕存取。不過，IAM 支援更精細的存取控制方法，通常更有用。例如，您可以：

- 允許使用者不受限制地呼叫任何 Amazon SWF 動作，但只能在指定的網域中呼叫。您可以使用這類政策允許正在開發的工作流程應用程式使用任何動作，但僅限「沙盒」網域。
- 允許使用者存取任何網域，但限制使用者使用 API 的方式。您可以使用這類政策允許「稽核員」應用程式呼叫任何網域中的 API，但僅允許讀取存取。
- 允許使用者在特定網域中只呼叫一組有限的動作。您可以使用這類政策允許工作流程啟動者在指定的網域中僅呼叫 `StartWorkflowExecution` 動作。

Amazon SWF 存取控制是以下列原則為基礎：

- 存取控制決策僅根據 IAM 政策；所有政策稽核和操作都是透過 IAM 完成。
- 存取控制模型使用 deny-by-default 原則；任何未明確允許的存取都會遭到拒絕。
- 您可以透過將適當的 IAM 政策附加到工作流程的參與者，以控制對 Amazon SWF 資源的存取。
- 僅能表達網域的資源許可。
- 您可以將條件套用至一或多個參數，從而進一步限制部分動作的使用。
- 如果您授與使用權限 [RespondDecisionTaskCompleted](#)，則可以對該動作中包含的決策清單表示權限。

每個決策都會有一或多個參數，有如一般的 API 呼叫。若要允許政策更易讀取，您可以表達決策的許可，有如實際 API 呼叫一樣，包含將條件套用至一些參數。這些類型的許可稱為「虛擬 API」許可。

如需可使用條件限制之一般和虛擬 API 參數的摘要，請參閱「[API 摘要](#)」。

## Amazon SWF IAM 政策

IAM 政策包含一或多個Statement元素，每個元素都包含一組定義政策的元素。如需完整的元素清單以及如何建構原則的一般討論，請參閱[存取原則語言](#)。Amazon SWF 存取控制是以下列元素為基礎：

### Effect

(必要) deny 或 allow 陳述式的影響。

#### Note

您必須明確允許存取；IAM 預設會拒絕存取。

### 資源

(必要) 該陳述式適用的資源 — 使用者可以與之互動的 AWS 服務中的實體。

您只能表達網域的資源許可。例如，政策可以僅允許存取您帳戶中的特定網域。`#####  
###Resource### Amazon ### (ARN)#####arn: aw: swf: ##:AccountID: /#  
#/##DomainName#`域是 AWS 區域，`Account` ID 是沒有破折號的帳戶 ID，`DomainName`是網域名稱。

### 動作

(必要) 套用陳述式的動作，您可以使用下列格式參照：`serviceId:action`。如果swf是 Amazon SWF，請將## ID 設定為。例如，swf:StartWorkflowExecution參照「[StartWorkflow執行](#)」動作，可用來控制允許哪些使用者啟動工作流程。

如果您授與使用權限 [RespondDecisionTaskCompleted](#)，您也可以使用來表示虛擬 API 的權限Action來控制對內含決策清單的存取權。由於 IAM 預設會拒絕存取，因此必須明確允許決策程式的決定，否則將不被接受。您可以使用 \* 值來允許所有決策。

## 條件

(選用) 表達一或多個動作參數的限制條件，這樣會限制允許的值。

Amazon SWF 動作的範圍通常很廣，您可以使用 IAM 條件來減少這些動作。例如，若要限制允許存取 [PollForActivityTask](#) 動作的工作清單，您可以包含 a Condition 並使用 `swf:taskList.name` 索引鍵來指定允許的清單。

您可以表達下列實體的限制條件。

- 工作流程類型。名稱和版本有不同的索引鍵。
- 活動類型。名稱和版本有不同的索引鍵。
- 任務清單。
- 標籤。您可以為一些動作指定多個標籤。在該情況下，每個標籤都會有不同的索引鍵。

### Note

對於 Amazon SWF，這些值都是字串，因此您可以使用字串運算子來限制參數，例如 `StringEquals`，這會將參數限制為指定的字串。不過，`StringEquals` 這類一般字串比較運算子需要所有請求都包含該參數。如果您未明確地包含參數，而且在類型註冊期間沒有提供的預設任務清單這類預設值，則會拒絕存取權。

這通常用來將條件設定為選用項目，因此您可以呼叫動作，而不需要包含相關聯的參數。例如，您可能希望允許決策程序指定一組 [RespondDecisionTaskCompleted](#) 決策，但也允許它為任何特定調用僅指定其中之一。在該情況下，您可以使用 `StringEqualsIfExists` 運算子限制適當的參數，而這在參數滿足條件時允許存取權，但在缺少參數時拒絕存取權。

如需可限制之參數和相關聯之索引鍵的完整清單，請參閱「[API 摘要](#)」。

下一節提供如何建構 Amazon SWF 政策的範例。如需詳細資訊，請參閱 [字串條件](#)。

## Amazon SWF 政策示例

工作流程由多個演員組成 — 活動、決策程式等。您可以透過附加適當的 IAM 政策來控制每個執行者的存取權限。本節將提供一些範例。以下顯示最簡單的情況：

```
{
  "Version": "2012-10-17",
```

```
"Statement" : [ {  
  "Effect" : "Allow",  
  "Action" : "swf:*",  
  "Resource" : "arn:aws:swf:*:123456789012:/domain/*"  
} ]  
}
```

如果您將此政策連接至動作者，其會具有所有區域的完整帳戶存取權。您可以使用萬用字元，讓單一值代表多個資源、動作或區域。

- Resource 值中的第一個萬用字元 (\*) 指出資源許可套用至所有區域。若要將權限限制為單一區域，請以適當的區域字串取代萬用字元，例如 us-east-1。
- Resource 值中的第二個萬用字元 (\*) 允許動作者存取指定區域中的任意帳戶網域。
- Action 值中的萬用字元 (\*) 可讓演員呼叫任何 Amazon SWF 動作。

如需如何使用萬用字元的詳細資訊，請參閱[元素描述](#)

下列各節顯示以更精細方式授予許可的政策範例。

#### 網域許可

如果您想要將部門的工作流程限定為特定網域，可以使用下列類似內容：

```
{  
  "Version": "2012-10-17",  
  "Statement": [ {  
    "Effect" : "Allow",  
    "Action" : "swf:*",  
    "Resource" : "arn:aws:swf:*:123456789012:/domain/department1"  
  } ]  
}
```

如果您將此政策連接至動作者，動作者即可呼叫任何動作，但僅限定 department1 網域。

如果您想要讓動作者存取多個網域，您可以分別表達每個網域的許可，如下所示：

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {
```

```
    "Effect" : "Allow",
    "Action" : "swf:*",
    "Resource" : "arn:aws:swf:*:123456789012:/domain/department1"
  }, {
    "Effect" : "Allow",
    "Action" : "swf:*",
    "Resource" : "arn:aws:swf:*:123456789012:/domain/department2"
  }
]
```

如果您將此政策附加到演員，則可以在department1和department2網域中使用任何 Amazon SWF 動作。您有時也可以使用萬用字元來代表多個網域。

### API 許可和限制

您可以控制動作者能與 Action 元素使用的動作。您可以使用 Condition 元素選擇性地限制動作的允許參數值。

如果您想要約束動作者使用特定動作，您可以使用下列類似內容：

```
{
  "Version": "2012-10-17",
  "Statement": [ {
    "Effect" : "Allow",
    "Action" : "swf:StartWorkflowExecution",
    "Resource" : "arn:aws:swf:*:123456789012:/domain/department2"
  } ]
}
```

如果您將此政策連接至動作者，動作者即可呼叫 StartWorkflowExecution 來啟動 department2 網域中的工作流程。動作者無法使用其他動作，或啟動其他網域中的工作流程。

您可以限制一或多個 StartWorkflowExecution 參數值，藉此進一步限制動作者可啟動的工作流程，如下所示：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect" : "Allow",
```

```

    "Action" : "swf:StartWorkflowExecution",
    "Resource" : "arn:aws:swf:*:123456789012:/domain/department1",
    "Condition" : {
      "StringEquals" : {
        "swf:workflowType.name" : "workflow1",
        "swf:workflowType.version" : "version2"
      }
    }
  ]
}

```

本政策會限制 StartWorkflowExecution 動作的 name 和 version 參數。如果您將此政策連接至動作者，動作者將只能執行 department1 網域中 workflow1 的 version2，而且兩個參數都必須包含在請求中。

您可以使用 StringEqualsIfExists 運算子來限制參數，而不需要將參數包含在請求中，如下所示：

```

{
  "Version": "2012-10-17",
  "Statement" : [ {
    "Effect" : "Allow",
    "Action" : "swf:StartWorkflowExecution",
    "Resource" : "arn:aws:swf:*:123456789012:/domain/some_domain",
    "Condition" : {
      "StringEqualsIfExists" : { "swf:taskList.name" : "task_list_name" }
    }
  } ]
}

```

此政策允許動作者在啟動工作流程執行時選擇性地指定任務清單。

您可以為一些動作限制一份標籤清單。在該情況下，每個標籤都會有不同的索引鍵，因此您可以使用 swf:tagList.member.0 限制清單中的第一個標籤、swf:tagList.member.1 限制清單中的第二個標籤，以此類推，最多可以使用 5 個。不過，您必須注意標籤清單的限制方式。例如，以下是「不」建議使用的政策範例：

```

{
  "Version": "2012-10-17",

```

```

"Statement" : [ {
  "Effect" : "Allow",
  "Action" : "swf:StartWorkflowExecution",
  "Resource" : "arn:aws:swf:*:123456789012:/domain/some_domain",
  "Condition" : {
    "StringEqualsIfExists" : {
      "swf:tagList.member.0" : "some_ok_tag", "another_ok_tag"
    }
  }
} ]
}

```

此政策可讓您選擇性指定 `some_ok_tag` 或 `another_ok_tag`。不過，此政策只會限制標籤清單的第一個元素。此清單可能含有因下列原因而允許之任意值的其他元素：此政策未將任何條件套用至 `swf:tagList.member.1`、`swf:tagList.member.2` 等。

處理此問題的其中一種方法為不允許使用標籤清單。下列政策以清單只需要一個元素為由，確保僅允許 `some_ok_tag` 或 `another_ok_tag`。

```

{
  "Version": "2012-10-17",
  "Statement" : [ {
    "Effect" : "Allow",
    "Action" : "swf:StartWorkflowExecution",
    "Resource" : "arn:aws:swf:*:123456789012:/domain/some_domain",
    "Condition" : {
      "StringEqualsIfExists" : {
        "swf:tagList.member.0" : "some_ok_tag", "another_ok_tag"
      },
      "Null" : { "swf:tagList.member.1" : "true" }
    }
  } ]
}

```

## 虛擬 API 許可和限制

如果您想要限制可供 `RespondDecisionTaskCompleted` 使用的決策，則必須先允許動作者呼叫 `RespondDecisionTaskCompleted`。您接著可以使用與一般 API 相同的語法，來表達適當虛擬 API 成員的許可，如下所示：

```

{

```

```
"Version": "2012-10-17",
"Statement" : [
  {
    "Resource" : "arn:aws:swf:*:123456789012:/domain/*",
    "Action" : "swf:RespondDecisionTaskCompleted",
    "Effect" : "Allow"
  }, {
    "Resource" : "*",
    "Action" : "swf:ScheduleActivityTask",
    "Effect" : "Allow",
    "Condition" : {
      "StringEquals" : { "swf:activityType.name" : "SomeActivityType" }
    }
  }
]
}
```

如果您將此政策連接至動作者，則第一個 Statement 元素允許動作者呼叫 RespondDecisionTaskCompleted。第二個元素可讓演員使用 ScheduleActivityTask 決定來引導 Amazon SWF 排程活動任務。要允許所有決定，請將「swf:ScheduleActivityTask」替換為「swf:\*」。

您可以使用 Condition 運算子限制參數，如同一般 API。此 Condition 中的 StringEquals 運算子允許 RespondDecisionTaskCompleted 排程 SomeActivityType 活動的活動任務，且其必須排程該任務。如果您想要允許 RespondDecisionTaskCompleted 使用參數值，但為非必要，則可以改為使用 StringEqualsIfExists 運算子。

## AWS 受管理的策略：SimpleWorkflowFullAccess

您可將 SimpleWorkflowFullAccess 政策連接到 IAM 身分。

此政策提供對 Amazon SWF 組態服務的完整存取權。

許可詳細資訊

此政策包含以下許可。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
```

```
    "Action": [
      "swf:*"
    ],
    "Resource": "*"
  }
]
```

## IAM 政策的服務模式限制

建立 IAM 政策時，您必須考慮服務模型限制。您可以建立代表無效 Amazon SWF 請求的語法有效 IAM 政策；在存取控制方面允許的請求仍可能會失敗，因為它是無效的請求。

例如，針對 [ListOpenWorkflowExecutions](#) 「不」建議使用以下政策：

```
{
  "Version": "2012-10-17",
  "Statement" : [ {
    "Effect" : "Allow",
    "Action" : "swf:ListOpenWorkflowExecutions",
    "Resource" : "arn:aws:swf:*:123456789012:/domain/domain_name",
    "Condition" : {
      "StringEquals" : {
        "swf:typeFilter.name" : "workflow_name",
        "swf:typeFilter.version" : "workflow_version",
        "swf:tagFilter.tag" : "some_tag"
      }
    }
  } ]
}
```

Amazon SWF 服務模型不允許在相同的 `ListOpenWorkflowExecutions` 請求中使用 `typeFilter` 和 `tagFilter` 參數。因此，原則允許服務拒絕的呼叫 (藉由擲回) `ValidationException` 做為無效要求。

## API 摘要

本節簡要說明如何使用 IAM 政策來控制演員如何使用每個 API 和虛擬 API 存取 Amazon SWF 資源。

- 對於 `RegisterDomain` 和 `ListDomains` 以外的所有動作，您可以透過表達網域資源的許可，來允許或拒絕存取帳戶中的任何或所有網域。

- 您可以允許或拒絕任意數目之一般 API 的許可，再者，如果您授予 [RespondDecisionTaskCompleted](#) 的呼叫許可，則為虛擬 API 的任何成員。
- 您可以使用 Condition 限制一些參數的允許值。

下列各節列出可針對一般和虛擬 API 的每個成員所限制的參數、提供相關聯的索引鍵，並記下如何控制網域存取權的任何限制。

## 一般 API

本節列出一般 API 成員，並簡短地說明可限制的參數和相關聯的索引鍵。本節也會記下如何控制網域存取權的任何限制。

### [CountClosedWorkflowExecutions](#)

- `tagFilter.tag`-字符串約束。索引鍵為 `swf:tagFilter.tag`
- `typeFilter.name`-字符串約束。金鑰為 `swf:typeFilter.name`。
- `typeFilter.version`-字符串約束。金鑰為 `swf:typeFilter.version`。

#### Note

`CountClosedWorkflowExecutions` 需要 `typeFilter` 和 `tagFilter` 互斥。

### [CountOpenWorkflowExecutions](#)

- `tagFilter.tag`-字符串約束。索引鍵為 `swf:tagFilter.tag`
- `typeFilter.name`-字符串約束。金鑰為 `swf:typeFilter.name`。
- `typeFilter.version`-字符串約束。金鑰為 `swf:typeFilter.version`。

#### Note

`CountOpenWorkflowExecutions` 需要 `typeFilter` 和 `tagFilter` 互斥。

### [CountPendingActivityTasks](#)

- `taskList.name`-字符串約束。金鑰為 `swf:taskList.name`。

## [CountPendingDecisionTasks](#)

- `taskList.name`-字符串約束。金鑰為 `swf:taskList.name`。

## [DeleteActivityType](#)

- `activityType.name`-字符串約束。金鑰為 `swf:activityType.name`。
- `activityType.version`-字符串約束。金鑰為 `swf:activityType.version`。

## [DeprecateActivityType](#)

- `activityType.name`-字符串約束。金鑰為 `swf:activityType.name`。
- `activityType.version`-字符串約束。金鑰為 `swf:activityType.version`。

## [DeprecateDomain](#)

- 您無法限制此動作的參數。

## [DeleteWorkflowType](#)

- `workflowType.name`-字符串約束。金鑰為 `swf:workflowType.name`。
- `workflowType.version`-字符串約束。金鑰為 `swf:workflowType.version`。

## [DeprecateWorkflowType](#)

- `workflowType.name`-字符串約束。金鑰為 `swf:workflowType.name`。
- `workflowType.version`-字符串約束。金鑰為 `swf:workflowType.version`。

## [DescribeActivityType](#)

- `activityType.name`-字符串約束。金鑰為 `swf:activityType.name`。
- `activityType.version`-字符串約束。金鑰為 `swf:activityType.version`。

## [DescribeDomain](#)

- 您無法限制此動作的參數。

## [DescribeWorkflowExecution](#)

- 您無法限制此動作的參數。

## [DescribeWorkflowType](#)

- workflowType.name-字符串約束。金鑰為 swf:workflowType.name。
- workflowType.version-字符串約束。金鑰為 swf:workflowType.version。

## [GetWorkflowExecutionHistory](#)

- 您無法限制此動作的參數。

## [ListActivityTypes](#)

- 您無法限制此動作的參數。

## [ListClosedWorkflowExecutions](#)

- tagFilter.tag-字符串約束。索引鍵為 swf:tagFilter.tag
- typeFilter.name-字符串約束。金鑰為 swf:typeFilter.name。
- typeFilter.version-字符串約束。金鑰為 swf:typeFilter.version。

### Note

ListClosedWorkflowExecutions 需要 typeFilter 和 tagFilter 互斥。

## [ListDomains](#)

- 您無法限制此動作的參數。

## [ListOpenWorkflowExecutions](#)

- tagFilter.tag-字符串約束。索引鍵為 swf:tagFilter.tag
- typeFilter.name-字符串約束。金鑰為 swf:typeFilter.name。

- `typeFilter.version`-字符串約束。金鑰為 `swf:typeFilter.version`。

 Note

`ListOpenWorkflowExecutions` 需要 `typeFilter` 和 `tagFilter` 互斥。

### [ListWorkflowTypes](#)

- 您無法限制此動作的參數。

### [PollForActivityTask](#)

- `taskList.name`-字符串約束。金鑰為 `swf:taskList.name`。

### [PollForDecisionTask](#)

- `taskList.name`-字符串約束。金鑰為 `swf:taskList.name`。

### [RecordActivityTaskHeartbeat](#)

- 您無法限制此動作的參數。

### [RegisterActivityType](#)

- `defaultTaskList.name`-字符串約束。金鑰為 `swf:defaultTaskList.name`。
- `name`-字符串約束。金鑰為 `swf:name`。
- `version`-字符串約束。金鑰為 `swf:version`。

### [RegisterDomain](#)

- `name`— 正在註冊的域名可作為此操作的資源使用。

### [RegisterWorkflowType](#)

- `defaultTaskList.name`-字符串約束。金鑰為 `swf:defaultTaskList.name`。

- name-字符串約束。金鑰為 swf:name。
- version-字符串約束。金鑰為 swf:version。

### RequestCancelWorkflowExecution

- 您無法限制此動作的參數。

### RespondActivityTaskCanceled

- 您無法限制此動作的參數。

### RespondActivityTaskCompleted

- 您無法限制此動作的參數。

### RespondActivityTaskFailed

- 您無法限制此動作的參數。

### RespondDecisionTaskCompleted

- decisions.member.N— 通過偽 API 權限間接限制。如需詳細資訊，請參閱 [虛擬 API](#)。

### SignalWorkflowExecution

- 您無法限制此動作的參數。

### StartWorkflowExecution

- tagList.member.0-字符串約束。索引鍵為 swf:tagList.member.0
- tagList.member.1-字符串約束。索引鍵為 swf:tagList.member.1
- tagList.member.2-字符串約束。索引鍵為 swf:tagList.member.2
- tagList.member.3-字符串約束。索引鍵為 swf:tagList.member.3
- tagList.member.4-字符串約束。索引鍵為 swf:tagList.member.4
- taskList.name-字符串約束。金鑰為 swf:taskList.name。

- `workflowType.name`-字符串約束。金鑰為 `swf:workflowType.name`。
- `workflowType.version`-字符串約束。金鑰為 `swf:workflowType.version`。

#### Note

您無法限制五個以上的標籤。

### TerminateWorkflowExecution

- 您無法限制此動作的參數。

## 虛擬 API

本節列出虛擬 API 成員，代表 [RespondDecisionTaskCompleted](#) 中所含的決策。如果您已授予 `RespondDecisionTaskCompleted` 的使用許可，您的政策即可表達此 API 之成員的許可，方式如同一般 API。您可以設定一或多個參數的條件，藉此進一步限制一些虛擬 API 的成員。本節列出虛擬 API 成員，並簡短地說明可限制的參數和相關聯的索引鍵。

#### Note

`aws:SourceIP`、`aws:UserAgent` 和 `aws:SecureTransport` 索引鍵不適用於虛擬 API。如果您欲使用的安全政策需要這些索引鍵控制對虛擬 API 的存取權，您可以將它們與 `RespondDecisionTaskCompleted` 動作使用。

### CancelTimer

- 您無法限制此動作的參數。

### CancelWorkflowExecution

- 您無法限制此動作的參數。

### CompleteWorkflowExecution

- 您無法限制此動作的參數。

## ContinueAsNewWorkflowExecution

- `tagList.member.0`-字符串約束。索引鍵為 `swf:tagList.member.0`
- `tagList.member.1`-字符串約束。索引鍵為 `swf:tagList.member.1`
- `tagList.member.2`-字符串約束。索引鍵為 `swf:tagList.member.2`
- `tagList.member.3`-字符串約束。索引鍵為 `swf:tagList.member.3`
- `tagList.member.4`-字符串約束。索引鍵為 `swf:tagList.member.4`
- `taskList.name`-字符串約束。金鑰為 `swf:taskList.name`。
- `workflowTypeVersion`-字符串約束。金鑰為 `swf:workflowTypeVersion`。

### Note

您無法限制五個以上的標籤。

## FailWorkflowExecution

- 您無法限制此動作的參數。

## RecordMarker

- 您無法限制此動作的參數。

## RequestCancelActivityTask

- 您無法限制此動作的參數。

## RequestCancelExternalWorkflowExecution

- 您無法限制此動作的參數。

## ScheduleActivityTask

- `activityType.name`-字符串約束。金鑰為 `swf:activityType.name`。
- `activityType.version`-字符串約束。金鑰為 `swf:activityType.version`。

- `taskList.name`-字符串約束。金鑰為 `swf:taskList.name`。

## SignalExternalWorkflowExecution

- 您無法限制此動作的參數。

## StartChildWorkflowExecution

- `tagList.member.0`-字符串約束。索引鍵為 `swf:tagList.member.0`
- `tagList.member.1`-字符串約束。索引鍵為 `swf:tagList.member.1`
- `tagList.member.2`-字符串約束。索引鍵為 `swf:tagList.member.2`
- `tagList.member.3`-字符串約束。索引鍵為 `swf:tagList.member.3`
- `tagList.member.4`-字符串約束。索引鍵為 `swf:tagList.member.4`
- `taskList.name`-字符串約束。金鑰為 `swf:taskList.name`。
- `workflowType.name`-字符串約束。金鑰為 `swf:workflowType.name`。
- `workflowType.version`-字符串約束。金鑰為 `swf:workflowType.version`。

### Note

您無法限制五個以上的標籤。

## StartTimer

- 您無法限制此動作的參數。

## 標籤類型政策

Amazon SWF 支援以標籤為基礎的政策。例如，您可以限制包含帶有鍵 `environment` 和值的標籤的 Amazon SWF 網域 `production`：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
```

```
    "Action": "swf:*",
    "Resource": "arn:aws:swf:*:123456789012:/domain/*",
    "Condition": {
      "StringEquals": {"aws:ResourceTag/environment": "production"}
    }
  ]
}
```

此政策將 Deny (拒絕) 對已標記為 environment/production 之任何網域的存取權。

如需標記的詳細資訊，請參閱：

- [標籤](#)
- [使用 IAM 標籤控制存取](#)

## Amazon SWF 的 VPC 端點 VPC 端點 VPC 端點

### Note

AWS PrivateLink目前僅在最高機密-東AWS部，AWS秘密地區和中國區域提供支持。

如果您使用 Amazon Virtual Private Cloud (Amazon VPC) 來託管您的AWS資源，可以在您的 Amazon VPC 與 Amazon 簡單工作流程服務之間建立連線。您可以將此連線與 Amazon SWF 工作流程使用此連線，而不需要透過公有網際網路。

Amazon VPC 可讓您在自訂虛擬網路中啟動AWS資源。您可利用 VPC 來控制您的網路設定，例如 IP 地址範圍、子網路、路由表和網路閘道。如需需要有需要 VPC 的詳細資訊，請參閱 [Amazon VPC 使用者指南](#)。

若要將您的 Amazon VPC 連接到 Amazon SWF，您必須先定義一個介面虛擬私人雲端端點，以便將 VPC 連接到其他端點AWS 服務。端點可提供可靠、可擴展的連線能力，且不需要網際網路閘道、網路地址轉譯 (NAT) 執行個體或 VPN 連接。如需詳細資訊，請參閱《Amazon VPC 使用者指南》中的 [界面 VPC 端點 \(AWS PrivateLink\)](#)。

### 建立端點

您可以使用、AWS Command Line Interface (AWS CLI)AWS Management Console、開AWS發套件、Amazon SWF API 或AWS CloudFormation。

如需使用 Amazon VPC 主控台或 AWS CLI，請參閱 Amazon VPC 使用者指南中的[建立界面端點](#)。

#### Note

建立端點時，請將 Amazon SWF 指定為您要 VPC 連接的服務。在 Amazon VPC 主控台中，服務名稱取決於所選 AWS 區域。例如，在最 AWS 高機密-東部區域，Amazon SWF 的服務名稱是 `com.amazonaws.us-iso-east-1.swf`。

如需使用 AWS CloudFormation 建立和設定端點的詳細資訊，請參閱 AWS CloudFormation 使用者指南中的 [AWS::EC2::VPCEndpoint](#) 資源。

## 亞馬遜 VPC 端點政策

若要控制對 Amazon SWF 的連線存取，您可以在建立 Amazon VPC 端點時附加 AWS Identity and Access Management (IAM) 端點政策。您可以透過附加多個端點政策來建立複雜的 IAM 規則。如需詳細資訊，請參閱：

- [Amazon SWF 的 Virtual Private Cloud 端點政策政策政策](#)
- [使用 VPC 端點控制對服務的存取](#)

## Amazon SWF 的 Virtual Private Cloud 端點政策政策政策

您可以為 Amazon VPC 端點建立政策建立政策，在 Amazon SWF 中您可以指定以下內容：

- 可執行動作的主體。
- 可執行的動作。
- 可供執行動作的資源。

下列範例顯示 Amazon VPC 端點政策，該政策允許針對特定 IAM 角色在單一網域上執行所有 Amazon SWF 操作。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "swf:*",
      "Resource": "arn:aws:swf:*:123456789012:/domain/myDomain",
    }
  ]
}
```

```
"Principal": {
  "AWS": "arn:aws:iam::123456789012:role/MyRole"
}
}
```

- 如需有需要建立端點策略的詳細資訊，請參閱[使用 VPC 端點控制服務的存取](#)。
- 如需如需有需要如需如需如需如需如需如需如需如需如需如需 Amazon SWF 如需如需如[Amazon 簡單 workflow 服務中的 Identity and Access Management](#)需如需AWS

## 疑難排解 Amazon 簡單 workflow 服務身分識別

使用下列資訊可協助您診斷和修正使用 Amazon SWF 和 IAM 時可能會遇到的常見問題。

### 主題

- [我沒有授權在 Amazon SWF 中執行操作](#)
- [我沒有授權執行 iam : PassRole](#)
- [我想允許我以外的人訪 AWS 帳戶 問我的 Amazon SWF 資源](#)

### 我沒有授權在 Amazon SWF 中執行操作

如果您收到錯誤，告知您未獲授權執行動作，您的政策必須更新，允許您執行動作。

下列範例錯誤會在 mateojackson 使用者嘗試使用主控台檢視一個虛構 *my-example-widget* 資源的詳細資訊，但卻無虛構 `swf:GetWidget` 許可時發生。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
swf:GetWidget on resource: my-example-widget
```

在此情況下，必須更新 Mateo 政策，允許他使用 `swf:GetWidget` 動作存取 *my-example-widget* 資源。

如果您需要協助，請聯絡您的 AWS 系統管理員。您的管理員提供您的簽署憑證。

### 我沒有授權執行 iam : PassRole

如果您收到未獲授權執行 `iam:PassRole` 動作的錯誤訊息，則必須更新您的政策以允許您將角色傳遞給 Amazon SWF。

有些 AWS 服務 允許您將現有角色傳遞給該服務，而不是建立新的服務角色或服務連結角色。如需執行此作業，您必須擁有將角色傳遞至該服務的許可。

當名為的 IAM 使用者marymajor嘗試使用主控台在 Amazon SWF 中執行動作時，會發生下列範例錯誤。但是，動作請求服務具備服務角色授予的許可。Mary 沒有將角色傳遞至該服務的許可。

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

在這種情況下，Mary 的政策必須更新，允許她執行 iam:PassRole 動作。

如果您需要協助，請聯絡您的 AWS 系統管理員。您的管理員提供您的簽署憑證。

## 我想允許我以外的人訪 AWS 帳戶 問我的 Amazon SWF 資源

您可以建立一個角色，讓其他帳戶中的使用者或您組織外部的人員存取您的資源。您可以指定要允許哪些信任物件取得該角色。針對支援基於資源的政策或存取控制清單 (ACL) 的服務，您可以使用那些政策來授予人員存取您的資源的許可。

如需進一步了解，請參閱以下內容：

- 若要了解 Amazon SWF 是否支援這些功能，請參閱[Amazon 簡單工作流程服務如何與 IAM 配合](#)。
- 若要了解如何提供對您所擁有資源 AWS 帳戶 的存取權，請參閱 [IAM 使用者指南中您擁有的另一 AWS 帳戶 個 IAM 使用者提供存取權限](#)。
- 若要了解如何將資源存取權提供給第三方 AWS 帳戶，請參閱 IAM 使用者指南中的[提供第三方 AWS 帳戶 擁有的存取權](#)。
- 如需了解如何透過聯合身分提供存取權，請參閱 IAM 使用者指南中的[將存取權提供給在外部進行身分驗證的使用者 \(聯合身分\)](#)。
- 若要了解跨帳戶存取使用角色和以資源為基礎的政策之間的差異，請參閱 IAM 使用者指南中的 [IAM 中的跨帳戶資源存取](#)。

## 記錄和監控

本節提供有關記錄和監控 Amazon SWF 的資訊。

### 主題

- [Amazon SWF 指標 CloudWatch](#)
- [使用 CloudWatch 的 Amazon SWF 指標AWS Management Console](#)

- [使用記錄 API 呼叫 AWS CloudTrail](#)
- [Amazon SWF EventBridge 執行狀態更改](#)
- [使AWS 使用者通知用 Amazon 簡單工作流服務](#)

## Amazon SWF 指標 CloudWatch

Amazon SWF 現在可提供指標 CloudWatch，讓您用來追蹤工作流程和活動，並根據您選擇的閾值設定警示。您可以使用檢視量度 AWS Management Console。如需詳細資訊，請參閱 [使用 CloudWatch 的 Amazon SWF 指標AWS Management Console](#)。

### 主題

- [Amazon SWF 指標的報告單元](#)
- [API 和決策事件指標](#)
- [Amazon SWF 指標](#)
- [Amazon SWF 非 ASCII 資源名稱和 CloudWatch維度](#)

## Amazon SWF 指標的報告單元

### 報告時間間隔的指標

某些 Amazon SWF 指標 CloudWatch 是時間間隔，一律以毫秒為單位。該 CloudWatch 單位被報告為Time。這些指標通常對應至您可以設定工作流程和活動逾時的工作流程執行階段，並且具有類似的名稱。

例如，DecisionTaskStartToCloseTime 指標會測量決策任務開始執行後至完成所耗費的時間，這與您可以為 DecisionTaskStartToCloseTimeout 值設定的期間相同。

如需所有這些工作流程執行階段的圖表，及了解每個階段在工作流程和活動生命週期發生的時間，請參閱「[Amazon SWF 超時類型](#)」。

### 報告計數的指標

一些用於 CloudWatch 報告結果的 Amazon SWF 指標計算為計數。例如，WorkflowsCanceled 會將結果記錄為 1 或 0，指出是否取消工作流程。值為 0 並不代表該指標未報告，意思是該指標描述的條件並未發生。

CloudWatch 該報告Count中的某些 Amazon SWF 指標 CloudWatch 是每秒計數。例如ProvisionedRefillRate，報告為 Count in CloudWatch，代表每秒請求Count的速率。

對於計數指標，最小值和最大值將一律為零或一，但平均值會是介於零到一的值。

## API 和決策事件指標

您可以在中監視 API 和決策事件，CloudWatch 以深入了解您的使用情況和容量。[請參閱 Amazon SWF 運作方式本節中的決策程序](#)，以及 [Amazon 簡單工作流程服務 API 參考中的決策主題](#)。

您也可以監控這些限制，以便在接近 Amazon SWF 節流限制時發出警示。如需這些限制和其預設設定的描述，請參閱「[Amazon SWF 節流配額](#)」。這些限制旨在預防不正確的工作流程使用過多的系統資源。若要請求提高限制，請參閱「[???](#)」。

最佳做法是，您應該在 API 或決策事件容量的 60% 左右設定 CloudWatch 警示。這可讓您在啟用 Amazon SWF 節流之前調整工作流程或要求提高服務限制。根據呼叫的[高載性](#)，您可以設定不同的警示，以便於達到服務限制時發出通知：

- 如果您的流量會有遽增情形，請設定在 ProvisionedBucketSize 達到 60% 的限制時發出警示。
- 如果您呼叫的速率相當穩定，則請為相關 API 和決策事件設定在 ProvisionedRefillRate 達到 60% 的限制時發出警示。

## Amazon SWF 指標

以下是可用於 Amazon SWF 的指標：

指標	描述
DecisionTaskScheduleStartTime	<p>決策任務排程後，到由工作者挑選並開始該任務之間的時間間隔 (毫秒)。</p> <p>CloudWatch 單位：Time</p> <p>維度：Domain, WorkflowTypeName, WorkflowTypeVersion</p> <p>有效的統計資訊：Average, Minimum, Maximum</p>
DecisionTaskStartToCloseTime	<p>決策任務從開始到其關閉之間的時間間隔 (毫秒)。</p> <p>CloudWatch 單位：Time</p>

指標	描述
	<p>維度 : Domain, WorkflowTypeName, WorkflowTypeVersion</p> <p>有效的統計資訊 : Average, Minimum, Maximum</p>
DecisionTasksCompleted	<p>已完成的決策作業計數。</p> <p>CloudWatch 單位 : Count</p> <p>維度 : Domain, WorkflowTypeName, WorkflowTypeVersion</p> <p>有效的統計資訊 : Sum</p>
PendingTasks	<p>以 1 分鐘為間隔的待處理任務特定任務清單計數。</p> <p>CloudWatch 單位 : Count</p> <p>維度 : Domain, TaskListName</p> <p>有效的統計資訊 : Sum</p>
StartedDecisionTasksTimedOutOnClose	<p>已開始但於關閉時已逾時的決策作業計數。</p> <p>CloudWatch 單位 : Count</p> <p>維度 : Domain, WorkflowTypeName, WorkflowTypeVersion</p> <p>有效的統計資訊 : Sum</p>
WorkflowStartToCloseTime	<p>工作流程從開始到其關閉之間的時間 (毫秒)。</p> <p>CloudWatch 單位 : Time</p> <p>維度 : Domain, WorkflowTypeName, WorkflowTypeVersion</p> <p>有效的統計資訊 : Average, Minimum, Maximum</p>

指標	描述
WorkflowsCanceled	<p>已取消的工作流程的計數。</p> <p>CloudWatch 單位 : Count</p> <p>維度 : Domain, WorkflowTypeName, WorkflowTypeVersion</p> <p>有效的統計資訊 : Sum</p>
WorkflowsCompleted	<p>已完成的工作流程的計數。</p> <p>CloudWatch 單位 : Count</p> <p>維度 : Domain, WorkflowTypeName, WorkflowTypeVersion</p> <p>有效的統計資訊 : Sum</p>
WorkflowsContinuedAsNew	<p>繼續做為新工作流程的計數。</p> <p>CloudWatch 單位 : Count</p> <p>維度 : Domain, WorkflowTypeName, WorkflowTypeVersion</p> <p>有效的統計資訊 : Sum</p>
WorkflowsFailed	<p>失敗工作流程的計數。</p> <p>CloudWatch 單位 : Count</p> <p>維度 : Domain, WorkflowTypeName, WorkflowTypeVersion</p> <p>有效的統計資訊 : Sum</p>

指標	描述
WorkflowsTerminated	<p>已終止之工作流程的計數。</p> <p>CloudWatch 單位 : Count</p> <p>維度 : Cause, Domain, WorkflowTypeName, WorkflowTypeVersion</p> <p>有效的統計資訊 : Sum</p>
WorkflowsTimedOut	<p>因任何原因逾時的工作流程計數。</p> <p>CloudWatch 單位 : Count</p> <p>維度 : Domain, WorkflowTypeName, WorkflowTypeVersion</p> <p>有效的統計資訊 : Sum</p>
ActivityTaskScheduleToCloseTime	<p>活動自排程後到其關閉之間的時間間隔 (毫秒)。</p> <p>CloudWatch 單位 : Time</p> <p>維度 : Domain, ActivityTypeName, ActivityTypeVersion</p> <p>有效的統計資訊 : Average, Minimum, Maximum</p>
ActivityTaskScheduleStartTime	<p>活動作業建立排程的時間與其開始的時間之間的時間間隔 (毫秒)。</p> <p>CloudWatch 單位 : Time</p> <p>維度 : Domain, ActivityTypeName, ActivityTypeVersion</p> <p>有效的統計資訊 : Average, Minimum, Maximum</p>

指標	描述
ActivityTaskStartToCloseTime	<p>活動任務從開始到其關閉之間的時間間隔 (毫秒)。</p> <p>CloudWatch 單位 : Time</p> <p>維度 : Domain, ActivityTypeName, ActivityTypeVersion</p> <p>有效的統計資訊 : Average, Minimum, Maximum</p>
ActivityTasksCancelled	<p>已取消的活動作業的計數。</p> <p>CloudWatch 單位 : Count</p> <p>維度 : Domain, ActivityTypeName, ActivityTypeVersion</p> <p>有效的統計資訊 : Sum</p>
ActivityTasksCompleted	<p>已完成的活動作業的計數。</p> <p>CloudWatch 單位 : Count</p> <p>維度 : Domain, ActivityTypeName, ActivityTypeVersion</p> <p>有效的統計資訊 : Sum</p>
ActivityTasksFailed	<p>失敗的活動作業的計數。</p> <p>CloudWatch 單位 : Count</p> <p>維度 : Domain, ActivityTypeName, ActivityTypeVersion</p> <p>有效的統計資訊 : Sum</p>

指標	描述
ScheduledActivityTasksTimedOutOnClose	<p>已排程但於關閉時已逾時的活動作業計數。</p> <p>CloudWatch 單位 : Count</p> <p>維度 : Domain, ActivityTypeName, ActivityTypeVersion</p> <p>有效的統計資訊 : Sum</p>
ScheduledActivityTasksTimedOutOnStart	<p>已排程但於開始時已逾時的活動作業計數。</p> <p>CloudWatch 單位 : Count</p> <p>維度 : Domain, ActivityTypeName, ActivityTypeVersion</p> <p>有效的統計資訊 : Sum</p>
StartedActivityTasksTimedOutOnClose	<p>已開始但於關閉時已逾時的活動作業計數。</p> <p>CloudWatch 單位 : Count</p> <p>維度 : Domain, ActivityTypeName, ActivityTypeVersion</p> <p>有效的統計資訊 : Sum</p>
StartedActivityTasksTimedOutOnHeartbeat	<p>已開始但因為活動訊號逾時而逾時的活動作業計數。</p> <p>CloudWatch 單位 : Count</p> <p>維度 : Domain, ActivityTypeName, ActivityTypeVersion</p> <p>有效的統計資訊 : Sum</p>

指標	描述
ThrottledEvents	<p>已調節之請求的計數。</p> <p>CloudWatch 單位 : Count</p> <p>維度 : APIName, DecisionName, ThrottlingScope</p> <p>有效的統計資訊 : Sum</p>
ProvisionedBucketsize	<p>每秒的可用請求計數。</p> <p>維度 : APIName, DecisionName</p> <p>有效的統計資訊 : Minimum</p>
ConsumedCapacity	<p>每秒的請求計數。</p> <p>CloudWatch 單位 : Count</p> <p>維度 : APIName, DecisionName</p> <p>有效的統計資訊 : Sum</p>
ConsumedLimit	<p>已使用的一般限制數量。</p> <p>維度 : GeneralLimitType</p>
ProvisionedRefillRate	<p>每秒允許進入儲存貯體的請求計數。</p> <p>維度 : APIName, DecisionName</p> <p>有效的統計資訊 : Minimum</p>
ProvisionedLimit	<p>佈建給帳戶的一般限制數量。</p> <p>維度 : GeneralLimitType</p>
維度	描述
Domain	將資料篩選至執行工作流程或活動的 Amazon SWF 網域。

維度	描述
ActivityTypeName	將資料篩選為活動類型的名稱。
ActivityTypeVersion	將資料篩選為活動類型的版本。
WorkflowTypeName	將資料篩選為此工作流程執行的工作流程類型名稱。
WorkflowTypeVersion	將資料篩選為此工作流程執行的工作流程類型版本。
APIName	將資料篩選為指定之 API 名稱的 API。
DecisionName	將資料篩選為所指定的決策名稱。
TaskListName	將資料篩選為所指定的任務清單名稱。
TaskListClassification	將資料篩選為工作清單的分類。決策任務清單的值為「D」，而活動任務清單則為「A」。
ThrottlingScope	將資料篩選至指定的節流範圍。超出帳戶層級配額時，值為「帳戶」，或超出工作流程層級配額時為「工作流程」。

## Amazon SWF 非 ASCII 資源名稱和 CloudWatch 維度

Amazon SWF 允許在資源名稱中使用非 ASCII 字符，例如 TaskList 和 DomainName。但是，量度的維 CloudWatch 度值只能包含可列印的 ASCII 字元。為確保 Amazon SWF 使用與 [CloudWatch 需求](#) 相容的維度值，會轉換不符合這些需求的 Amazon SWF 資源名稱，並附加如下的總和檢查碼：

- 任何非 ASCII 字元都會被取代為「？」。
- 如有必要，輸入字符串或轉換後的字符串將被截斷。這樣可以確保在附加校驗和時，新的字符串長度不會超過 CloudWatch 最大值。
- 由於任何非 ASCII 字元都會轉換為「？」，因此轉換前不同的某些 CloudWatch 量度維度值可能會在轉換後看起來相同。為了幫助區分它們，在資源名稱後面附加一個底線（\_），後跟原始資源名稱的 SHA256 校驗和的前 16 個字符。

轉換範例：

- test apple 將被轉換為 test ?pple\_82cc5b8e3a771d12

- 名稱將會轉換為 `2fec5edbb2c05c22`。
- 名稱 `TaskList` 稱 `app1` 和 `app2` 將被轉換為 `pp1`，並且將是相同的。附加總和檢查碼會傳回不同的值，`pp1_f39a36df9d85a69d` 與 `pp1_da3efb4f11dd0f7f`

#### Tip

您可以生成自己的 SHA256 校驗和。例如，若要使用 `shasum` 命令行工具：

```
echo -n "<the original resource name>" | shasum -a 256 | cut -c1-16
```

## 使用 CloudWatch 的 Amazon SWF 指標 AWS Management Console

Amazon CloudWatch 提供 Amazon SWF 工作流程和活動的數個可檢視指標。您可以查看您的 Amazon SWF 工作流程執行的指標並使用 [AWS Management Console](#)。「您必須登入主控台才能繼續進行」。

如需每個可用指標的說明，請參閱「[Amazon SWF 指標 CloudWatch](#)」。

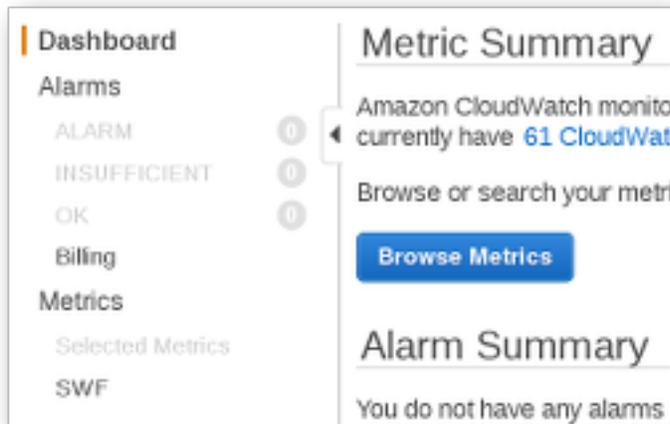
### 主題

- [檢視 指標](#)
- [設定警示](#)

## 檢視 指標

### 檢視 Amazon SWF 指標

1. 登入 AWS Management Console 並開啟位於 <https://console.aws.amazon.com/cloudwatch/> 的 CloudWatch 主控台。
2. 在導覽窗格中，於 Metrics (指標) 下選擇 SWF。



如果您最近執行過任何工作流程執行，即會呈現兩份指標清單：工作流程類型指標和活動類型指標。

Showing all results (61) for SWF Metrics.

Select All | Clear

**SWF > Workflow Type Metrics**

Domain	WorkflowTypeName	WorkflowTypeVersion	Metric Name
HelloWorld	HelloWorldWorkflow.hello_workflow	1.0	WorkflowStartToCloseTime
HelloWorld	HelloWorldWorkflow.hello_workflow	1.0	WorkflowsCompleted

**SWF > Activity Type Metrics**

Domain	ActivityTypeName	ActivityTypeVersion	Metric Name
Booking	BookingActivity.reserve_airline	1.0	ActivityTaskScheduleToStartTime
Booking	BookingActivity.reserve_airline	1.0	ActivityTaskStartToCloseTime

### Note

一開始，您可能只會看到 Workflow Type Metrics (工作流程類型指標)；Activity Type Metrics (活動類型指標) 會呈現在相同的檢視中，但需要您向下捲動才能看到。

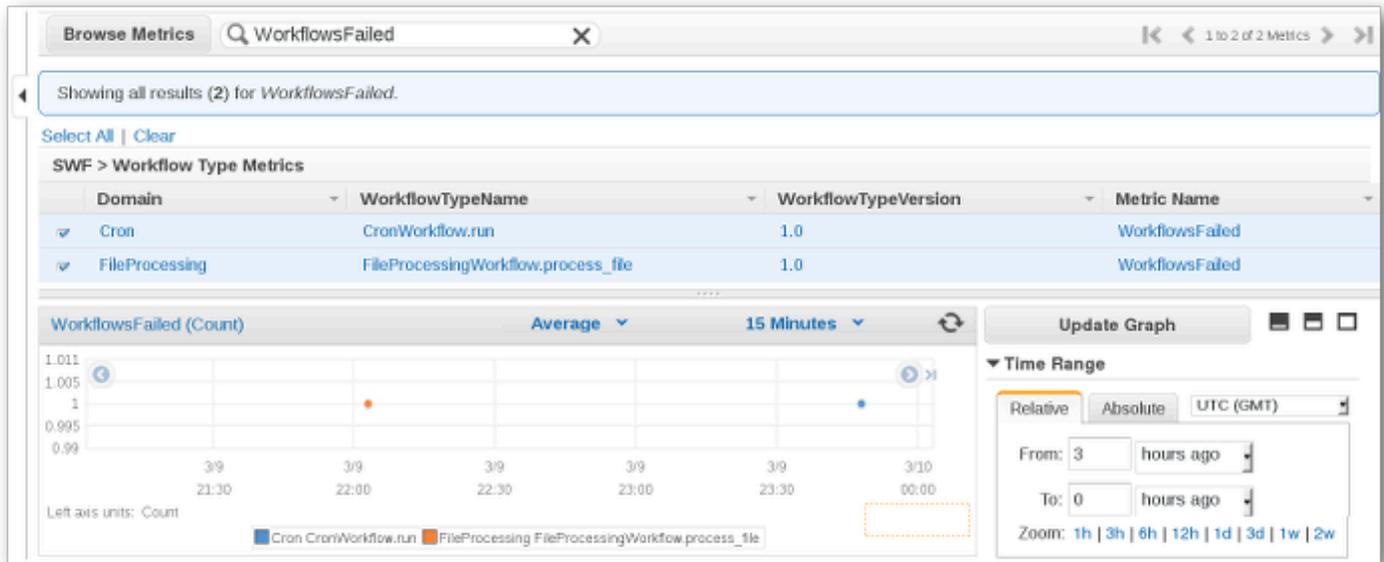
一次最多顯示 50 個最新指標 (區分工作流程和活動指標)。

您可以使用清單中每欄上方的互動式標題，透過任何提供的維度來排序指標。針對工作流程，維度是 Domain (網域)、WorkflowTypeName (工作流程類型名稱)、WorkflowTypeVersion (工作流程類型版

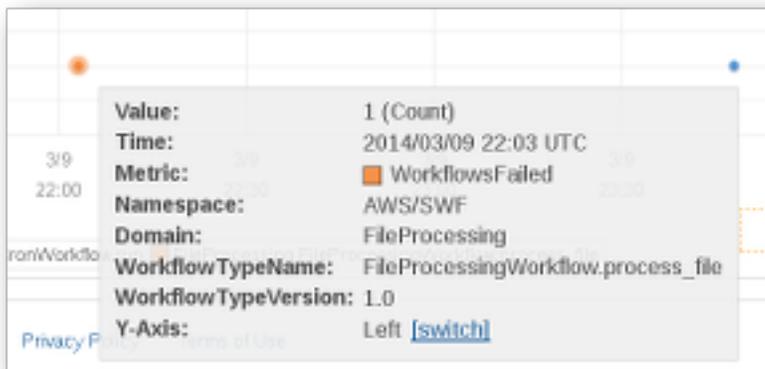
本) 和 Metric Name (指標名稱)。針對活動，維度是 Domain (網域)、ActivityTypeName (活動類型名稱)、ActivityTypeVersion (活動類型版本) 和 Metric Name (指標名稱)。

「[Amazon SWF 指標 CloudWatch](#)」中說明各種類型的指標。

您可以選擇清單中指標列旁邊的方塊來檢視指標圖形，以及使用圖形檢視右側的 Time Range (時間範圍) 控制項來變更圖形參數。



如需圖形上任何點的詳細資訊，請將游標放在圖形點上方。即會顯示點維度的詳細資訊。



如需使用 CloudWatch 指標的詳細資訊，請參[查看、繪製圖形和發佈度量](#)中的 Amazon CloudWatch 使用者指南。

## 設定警示

您可以使用 CloudWatch 警示來執行動作，例如在達到警示閾值時通知您。例如，您可以設定警示，在 WorkflowsFailed 指標超過特定閾值時傳送通知給 SNS 主題或傳送電子郵件。

### 設定任何指標的警示

1. 選擇單一指標的方塊，來選擇單一指標。
2. 在圖形右側的 Tools (工具) 控制項中，選擇 Create Alarm (建立警示)。
3. 在 Define Alarm (定義警示) 畫面上，輸入警示閾值、期間參數以及要採取的動作。

The screenshot shows the 'Define Alarm' console interface. On the left, there are navigation buttons: 'Back', 'Next', and 'Cancel'. A blue box contains the instruction: 'Please set the alarm threshold, actions and click Create Alarm below.' Below this is a 'Create Alarm' button. The main area is titled 'Alarm Threshold' and contains the following fields:

- Name:** [Text input field]
- Description:** [Text input field]
- Whenever:** WorkflowsFailed
- is:** >= [Dropdown menu]
- for:** 2 [Text input field] consecutive period(s)

Below the threshold settings is the 'Actions' section, titled 'Define what actions are taken when your alarm changes state.' It contains a 'Notification' section with the following fields:

- Whenever this alarm:** State is ALARM [Dropdown menu]
- Send notification to:** SWF\_Sample\_Topic [Dropdown menu] with a 'New list' link
- Email list:** me@example.com [Text input field]

At the bottom of the 'Actions' section are three buttons: '+ Notification', '+ AutoScaling Action', and '+ EC2 Action'.

如需設定和使用 CloudWatch 警示的詳細資訊，請參[建立 Amazon CloudWatch 警示](#)中的 Amazon CloudWatch 使用者指南。

## 使用記錄 API 呼叫 AWS CloudTrail

Amazon 簡單工作流程服務與服務整合 [AWS CloudTrail](#)，該服務可提供使用者、角色或角色所採取的動作記錄 AWS 服務。CloudTrail 以事件形式擷取 Amazon SWF 的所有 API 呼叫。擷取的呼叫包

括來自 Amazon SWF 主控台的呼叫，以及對 Amazon SWF API 操作的程式碼呼叫。使用收集的資訊 CloudTrail，您可以判斷向 Amazon SWF 發出的請求、提出請求的來源 IP 地址、提出請求的時間以及其他詳細資訊。

每一筆事件或日誌專案都會包含產生請求者的資訊。身分資訊可協助您判斷下列事項：

- 該請求是否使用根使用者還是使用者憑證提出。
- 請求是否代表 IAM 身分中心使用者提出。
- 提出該請求時，是否使用了特定角色或聯合身分使用者的暫時安全憑證。
- 該請求是否由另一項 AWS 服務服務提出。

CloudTrail 在您創建帳戶 AWS 帳戶 時處於活動狀態，並且您自動可以訪問 CloudTrail 事件歷史記錄。CloudTrail 事件歷史記錄提供了過去 90 天的記錄管理事件的可查看，可搜索，可下載和不可變的記錄。AWS 區域若要取得更多資訊，請參閱 [《使用指南》中的〈AWS CloudTrail 使用 CloudTrail 事件歷程〉](#)。查看活動歷史記錄不 CloudTrail 收取任何費用。

如需過 AWS 帳戶 去 90 天內持續的事件記錄，請建立追蹤或 [CloudTrailLake](#) 事件資料存放區。

## CloudTrail 小徑

追蹤可 CloudTrail 將日誌檔交付到 Amazon S3 儲存貯體。使用建立的所有系統線 AWS Management Console 都是多區域。您可以使用建立單一區域或多區域系統線。AWS CLI 建議您建立多區域追蹤，因為您會擷取帳戶 AWS 區域 中的所有活動。如果您建立單一區域追蹤，則只能檢視追蹤記錄中的 AWS 區域事件。如需有關 [追蹤的詳細資訊](#)，請參閱 [《AWS CloudTrail 使用指南》中的「為您的建立追蹤」AWS 帳戶和「為組織建立追蹤」](#)。

您可以透 CloudTrail 過建立追蹤，免費將一份正在進行的管理事件副本傳遞到 Amazon S3 儲存貯體，但是需要支付 Amazon S3 儲存費用。如需有關 CloudTrail 定價的詳細資訊，請參閱 [AWS CloudTrail 定價](#)。如需 Amazon S3 定價的相關資訊，請參閱 [Amazon S3 定價](#)。

## CloudTrail 湖泊事件資料存放區

CloudTrail Lake 可讓您針對事件執行 SQL 型查詢。CloudTrail 湖將基於行的 JSON 格式的現有事件轉換為 [Apache ORC](#) 格式。ORC 是一種單欄式儲存格式，針對快速擷取資料進行了最佳化。系統會將事件彙總到事件資料存放區中，事件資料存放區是事件的不可變集合，其依據為您透過套用 [進階事件選取器](#) 選取的條件。套用於事件資料存放區的選取器控制哪些事件持續存在並可供您查詢。若要取得有關 CloudTrail Lake 的更多資訊，請參閱 [使用指南中的〈AWS CloudTrail 使用 AWS CloudTrail Lake〉](#)。

CloudTrail Lake 事件資料存放區和查詢會產生費用。建立事件資料存放區時，您可以選擇要用於事件資料存放區的[定價選項](#)。此定價選項將決定擷取和儲存事件的成本，以及事件資料存放區的預設和最長保留期。如需有關 CloudTrail 定價的詳細資訊，請參閱[AWS CloudTrail 定價](#)。

## 資料事件 CloudTrail

[資料事件](#)提供在資源上或在資源中執行的資源操作的相關資訊 (例如，讀取或寫入 Amazon S3 物件)。這些也稱為資料平面操作。資料事件通常是大量資料的活動。依預設，CloudTrail 不會記錄資料事件。CloudTrail 事件歷史記錄不會記錄數據事件。

資料事件需支付額外的費用。如需有關 CloudTrail 定價的詳細資訊，請參閱[AWS CloudTrail 定價](#)。

您可以使用 CloudTrail 主控台或 CloudTrail API 操作記錄 Amazon SWF 資源類型的資料事件。AWS CLI [有關如何記錄資料事件的詳細資訊](#)，請參閱AWS CloudTrail 使用《使用指南》AWS Command Line Interface中的[記錄資料事件 AWS Management Console](#)和[記錄資料事件](#)。

下表列出您可以記錄其資料事件的 Amazon SWF 資源類型。「資料事件類型」欄會顯示可從 CloudTrail 主控台的「資料」事件類型清單中選擇的值。resource .type 值欄會顯示 **resources.type** 值，您可以在使用或 API 設定進階事件選取器時指定這個值。AWS CLI CloudTrail 記錄到資料 CloudTrail欄中的資料 API 會顯示 CloudTrail 針對資源類型記錄的 API 呼叫。

您可以設定進階事件選取器來篩選eventNamereadOnly、和resources.ARN欄位，以僅記錄對您很重要的事件。如需這些欄位的詳細資訊，請參閱 AWS CloudTrail API 參考[AdvancedFieldSelector](#)中的。

資料事件類型	resources.type 值	記錄到的資料 API CloudTrail
SWF 網域名稱	AWS::SWF::Domain	工作流事件 <ul style="list-style-type: none"> <li>• <a href="#">CountClosedWorkflowExecutions</a></li> <li>• <a href="#">CountOpenWorkflowExecutions</a></li> <li>• <a href="#">DescribeWorkflowExecution</a></li> <li>• <a href="#">ListClosedWorkflowExecutions</a></li> <li>• <a href="#">ListOpenWorkflowExecutions</a></li> </ul>

資料事件類型	resources.type 值	記錄到的資料 API CloudTrail
		<ul style="list-style-type: none"> <li>• <a href="#">GetWorkflowExecutionHistory</a></li> <li>• <a href="#">RequestCancelWorkflowExecution</a></li> <li>• <a href="#">SignalWorkflowExecution</a></li> <li>• <a href="#">StartWorkflowExecution</a></li> <li>• <a href="#">TerminateWorkflowExecution</a></li> </ul> <p>任務事件</p> <ul style="list-style-type: none"> <li>• <a href="#">CountPendingActivityTasks</a></li> <li>• <a href="#">PollForDecisionTask</a></li> <li>• <a href="#">PollForActivityTask</a></li> <li>• <a href="#">RecordActivityTaskHeartbeat</a></li> <li>• <a href="#">RespondActivityTaskCanceled</a></li> <li>• <a href="#">RespondActivityTaskCompleted</a></li> <li>• <a href="#">RespondActivityTaskFailed</a></li> <li>• <a href="#">RespondDecisionTaskCompleted</a></li> </ul> <p>決策事件</p> <ul style="list-style-type: none"> <li>• <a href="#">CancelTimer</a></li> <li>• <a href="#">CancelWorkflowExecution</a></li> <li>• <a href="#">CompleteWorkflowExecution</a></li> <li>• <a href="#">ContinueAsNewWorkflowExecution</a></li> </ul>

資料事件類型	resources.type 值	記錄到的資料 API CloudTrail
		<ul style="list-style-type: none"> <li>• <a href="#">FailWorkflowExecution</a></li> <li>• <a href="#">RecordMarker</a></li> <li>• <a href="#">RequestCancelActivityTask</a></li> <li>• <a href="#">RequestCancelExternalWorkflowExecution</a></li> <li>• <a href="#">ScheduleActivityTask</a></li> <li>• <a href="#">ScheduleLambdaFunction</a></li> <li>• <a href="#">SignalExternalWorkflowExecution</a></li> <li>• <a href="#">StartChildWorkflowExecution</a></li> <li>• <a href="#">StartTimer</a></li> </ul>

### CloudTrail 事件和 RespondDecisionTaskCompleted

[RespondDecisionTaskCompleted](#) 動作會在要求有效負載中取得決策清單。完成的調用將發出 N+1 CloudTrail 數據事件，每個決策一個加一個 API 調用本身。資料事件和 API 事件都會有相同的要求識別碼。

## 管理事件 CloudTrail

[管理事件](#) 提供有關在您的資源上執行的管理作業的資訊 AWS 帳戶。這些也稱為控制平面操作。依預設，會 CloudTrail 記錄管理事件。

Amazon 簡單工作流程服務將下列控制平面操作記錄 CloudTrail 為管理事件。

### 網域活動

- [RegisterDomain](#)
- [DescribeDomain](#)
- [ListDomains](#)
- [DeprecateDomain](#)

- [UndeprecateDomain](#)

## 活動活動

- [RegisterActivityType](#)
- [DescribeActivityType](#)
- [ListActivityTypes](#)
- [DeprecateActivityType](#)
- [UndeprecateActivityType](#)
- [DeleteActivityType](#)

## WorkflowType 活動

- [RegisterWorkflowType](#)
- [DescribeWorkflowType](#)
- [ListWorkflowTypes](#)
- [DeprecateWorkflowType](#)
- [UndeprecateWorkflowType](#)
- [DeleteWorkflowType](#)

## 標記事件

- [TagResource](#)
- [UntagResource](#)
- [ListTagsforResource](#)

## 範例事件

事件代表來自任何來源的單一請求，並包括有關請求的 API 操作，操作的日期和時間，請求參數等信息。CloudTrail 日誌文件不是公共 API 調用的有序堆棧跟踪，因此事件不會以任何特定順序顯示。

下列範例顯示示範CountClosedWorkflowExecutions作業的 CloudTrail 事件。

```
{
  "eventVersion": "1.09",
```

```
"userIdentity": {
  "type": "AssumedRole",
  "principalId": "1234567890abcdef02345:admin",
  "arn": "arn:aws:sts::111122223333:assumed-role/Admin/admin",
  "accountId": "111122223333",
  "accessKeyId": "abcdef01234567890abc",
  "sessionContext": {
    "sessionIssuer": {
      "type": "Role",
      "principalId": "1234567890abcdef02345",
      "arn": "arn:aws:iam::111122223333:role/Admin",
      "accountId": "111122223333",
      "userName": "Admin"
    },
    "attributes": {
      "creationDate": "2023-11-23T16:37:38Z",
      "mfaAuthenticated": "false"
    }
  }
},
"eventTime": "2023-11-23T17:52:46Z",
"eventSource": "swf.amazonaws.com",
"eventName": "CountClosedWorkflowExecutions",
"awsRegion": "us-east-1",
"sourceIPAddress": "198.51.100.42",
"userAgent": "aws-internal/3 aws-sdk-java/1.11.42",
"requestParameters": {
  "domain": "nsg-domain",
  "closeTimeFilter": {
    "oldestDate": "Nov 23, 2023 5:52:46 PM",
    "latestDate": "Nov 23, 2023 5:52:46 PM"
  }
},
"responseElements": null,
"requestID": "a1b2c3d4-5678-90ab-cdef-EXAMPLEaaaaa",
"eventID": "a1b2c3d4-5678-90ab-cdef-EXAMPLEbbbbbb",
"readOnly": true,
"resources": [
  {
    "accountId": "111122223333",
    "type": "AWS::SWF::Domain",
    "ARN": "arn:aws:swf:us-east-1:111122223333:/domain/nsg-domain"
  }
],
```

```
"eventType": "AwsApiCall",
"managementEvent": false,
"recipientAccountId": "111122223333",
"eventCategory": "Data",
"tlsDetails": {
  "clientProvidedHostHeader": "swf.example.amazondomains.com"
}
}
```

若要取得有關 CloudTrail 記錄內容的資訊，請參閱AWS CloudTrail 使用指南中的[CloudTrail記錄內容](#)。

## Amazon SWF EventBridge 執行狀態更改

您可以使用 Amazon EventBridge 來回應AWS資源。當 Amazon SWF 發出事件時，一律會前往您帳戶的默認 EventBridge 事件匯流排。您可以為事件創建規則，將其與默認事件總線相關聯，並指定當 EventBridge 收到與規則匹配的事件時要執行的目標操作。這樣，您就可以監控工作流程，而不需使用[GetWorkflowExecutionHistory](#) API。根據工作流程執行中的更改，您可以使用 EventBridge 目標調用AWS Lambda功能、傳送消息至 Amazon Simple Notification Service (Amazon SNS) 主題以及更多信息。

您可以查看執行狀態更改事件的完整內容，使用[DescribeWorkflowExecution](#)。

如需詳細資訊，請參閱《[Amazon EventBridge 使用者指南](#)》。

### EventBridge 事件

歷史事件類型包含執行狀態更改。所以此detail部分至少包含以下參數：

- `eventId`：獲取 workflow 執行歷史記錄顯示的事件 ID。
- `workflowExecutionDetail`：發出事件時 workflow 的狀態。
- `eventType`：歷史事件類型，以下其中一個選項：
  - `ActivityTaskCanceled`
  - `ActivityTaskFailed`
  - `ActivityTaskTimedOut`
  - `WorkflowExecutionCanceled`
  - `WorkflowExecutionCompleted`
  - `WorkflowExecutionFailed`
  - `WorkflowExecutionStarted`

- WorkflowExecutionTerminated
- WorkflowExecutionTimedOut
- WorkflowExecutionContinuedAsNew
- CancelTimerFailed
- CancelWorkflowExecutionFailed
- ChildWorkflowExecutionFailed
- ChildWorkflowExecutionTimedOut
- CompleteWorkflowExecutionFailed
- ContinueAsNewWorkflowExecutionFailed
- DecisionTaskTimedOut
- FailWorkflowExecutionFailed
- RecordMarkerFailed
- RequestCancelActivityTaskFailed
- RequestCancelExternalWorkflowExecutionFailed
- ScheduleActivityTaskFailed
- SignalExternalWorkflowExecutionFailed
- StartActivityTaskFailed
- StartChildWorkflowExecutionFailed
- StartTimerFailed
- TimerCanceled
- LambdaFunctionFailed
- LambdaFunctionTimedOut
- StartLambdaFunctionFailed
- ScheduleLambdaFunctionFailed

## Amazon SWF 事件範例

以下是 Amazon SWF 傳送事件至 EventBridge 的範例：

### 主題

- [Amazon SWF EventBridge](#)
  - [執行已開始](#)

- [執行完成](#)
- [執行已失敗](#)
- [執行超時](#)
- [執行已終止](#)

在每個案例中，在事件資料中的 detail 區段會提供與 [DescribeWorkflowExecution](#) API 相同的資訊。所以此 executionStatus 字段指出事件傳送時的執行狀態，OPEN 或者 CLOSED。

### 執行已開始

```
{
  "version": "0",
  "id": "4444444444444444",
  "detail-type": "Simple Workflow Execution State Change",
  "source": "aws.swf",
  "account": "4444444444444444",
  "time": "2020-05-08T15:57:38Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:swf:us-east-1:4444444444444444:/domain/SimpleWorkflowUserSimulator"
  ],
  "detail": {
    "eventId": 1,
    "eventType": "WorkflowExecutionStarted",
    "workflowExecutionDetail": {
      "executionInfo": {
        "execution": {
          "workflowId": "123456789012",
          "runId": "AKIAIOSFODNN7EXAMPLE"
        },
        "workflowType": {
          "name": "SimpleWorkflowUserSimulator",
          "version": "myWorkflow"
        },
        "startTimestamp": 1588953458484,
        "closeTimestamp": null,
        "executionStatus": "OPEN",
        "closeStatus": null,
        "parent": null,
        "parentExecutionArn": null,
        "tagList": null,

```

```

    "cancelRequested": false
  },
  "executionConfiguration": {
    "taskStartToCloseTimeout": "60",
    "executionStartToCloseTimeout": "1000",
    "taskList": {
      "name": "4444444444444444"
    },
    "taskPriority": null,
    "childPolicy": "ABANDON",
    "lambdaRole": "arn:aws:iam::4444444444444444:role/BasicSWFLambdaExecution"
  },
  "openCounts": {
    "openActivityTasks": 0,
    "openDecisionTasks": 1,
    "openTimers": 0,
    "openChildWorkflowExecutions": 0,
    "openLambdaFunctions": 0
  },
  "latestActivityTaskTimestamp": null,
}
}
}

```

## 執行完成

```

{
  "version": "0",
  "id": "1111-2222-3333",
  "detail-type": "Simple Workflow Execution State Change",
  "source": "aws.swf",
  "account": "444455556666",
  "time": "2020-05-08T15:57:39Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:swf:us-east-1:444455556666:/domain/SimpleWorkflowUserSimulator"
  ],
  "detail": {
    "eventId": 35,
    "eventType": "WorkflowExecutionCompleted",
    "workflowExecutionDetail": {
      "executionInfo": {
        "execution": {

```

```

    "workflowId": "1234-5678-9012",
    "runId": "777788889999"
  },
  "workflowType": {
    "name": "SimpleWorkflowUserSimulator",
    "version": "myWorkflow"
  },
  "startTimestamp": 1588953458820,
  "closeTimestamp": 1588953459448,
  "executionStatus": "CLOSED",
  "closeStatus": "COMPLETED",
  "parent": null,
  "parentExecutionArn": null,
  "tagList": null,
  "cancelRequested": false
},
"executionConfiguration": {
  "taskStartToCloseTimeout": "60",
  "executionStartToCloseTimeout": "1000",
  "taskList": {
    "name": "1111-1111-1111"
  },
  "taskPriority": null,
  "childPolicy": "ABANDON",
  "lambdaRole": "arn:aws:iam::444455556666:role/BasicSWFLambdaExecution"
},
"openCounts": {
  "openActivityTasks": 0,
  "openDecisionTasks": 0,
  "openTimers": 0,
  "openChildWorkflowExecutions": 0,
  "openLambdaFunctions": 0
},
"latestActivityTaskTimestamp": 1588953459402,
}
}
}

```

## 執行已失敗

```

{
  "version": "0",
  "id": "1111-2222-3333",

```

```
"detail-type": "Simple Workflow Execution State Change",
"source": "aws.swf",
"account": "444455556666",
"time": "2020-05-08T15:57:38Z",
"region": "us-east-1",
"resources": [
  "arn:aws:swf:us-east-1:444455556666:/domain/SimpleWorkflowUserSimulator"
],
"detail": {
  "eventId": 11,
  "eventType": "WorkflowExecutionFailed",
  "workflowExecutionDetail": {
    "executionInfo": {
      "execution": {
        "workflowId": "1234-5678-9012",
        "runId": "777788889999"
      },
      "workflowType": {
        "name": "SimpleWorkflowUserSimulator",
        "version": "myWorkflow"
      },
      "startTimestamp": 1588953158481,
      "closeTimestamp": 1588953458560,
      "executionStatus": "CLOSED",
      "closeStatus": "FAILED",
      "parent": null,
      "parentExecutionArn": null,
      "tagList": null,
      "cancelRequested": false
    },
    "executionConfiguration": {
      "taskStartToCloseTimeout": "60",
      "executionStartToCloseTimeout": "1000",
      "taskList": {
        "name": "1111-1111-1111"
      },
      "taskPriority": null,
      "childPolicy": "ABANDON",
      "lambdaRole": "arn:aws:iam::444455556666:role/BasicSWFLambdaExecution"
    },
    "openCounts": {
      "openActivityTasks": 0,
      "openDecisionTasks": 0,
      "openTimers": 0,
```

```
    "openChildWorkflowExecutions": 0,  
    "openLambdaFunctions": 0  
  },  
  "latestActivityTaskTimestamp": null,  
}  
}  
}
```

## 執行超時

```
{  
  "version": "0",  
  "id": "1111-2222-3333",  
  "detail-type": "Simple Workflow Execution State Change",  
  "source": "aws.swf",  
  "account": "444455556666",  
  "time": "2020-05-05T17:26:30Z",  
  "region": "us-east-1",  
  "resources": [  
    "arn:aws:swf:us-east-1:444455556666:/domain/SimpleWorkflowUserSimulator"  
  ],  
  "detail": {  
    "eventId": 6,  
    "eventType": "WorkflowExecutionTimedOut",  
    "workflowExecutionDetail": {  
      "executionInfo": {  
        "execution": {  
          "workflowId": "1234-5678-9012",  
          "runId": "777788889999"  
        },  
        "workflowType": {  
          "name": "SimpleWorkflowUserSimulator",  
          "version": "myWorkflow"  
        },  
        "startTimestamp": 1588698073748,  
        "closeTimestamp": 1588699590745,  
        "executionStatus": "CLOSED",  
        "closeStatus": "TIMED_OUT",  
        "parent": null,  
        "parentExecutionArn": null,  
        "tagList": null,  
        "cancelRequested": false  
      },  
    },  
  },  
}
```

```

    "executionConfiguration": {
      "taskStartToCloseTimeout": "60",
      "executionStartToCloseTimeout": "1000",
      "taskList": {
        "name": "1111-1111-1111"
      },
      "taskPriority": null,
      "childPolicy": "ABANDON",
      "lambdaRole": "arn:aws:iam::444455556666:role/BasicSWFLambdaExecution"
    },
    "openCounts": {
      "openActivityTasks": 1,
      "openDecisionTasks": 0,
      "openTimers": 0,
      "openChildWorkflowExecutions": 0,
      "openLambdaFunctions": 0
    },
    "latestActivityTaskTimestamp": 1588699585802,
  }
}
}

```

## 執行已終止

```

{
  "version": "0",
  "id": "1111-2222-3333",
  "detail-type": "Simple Workflow Execution State Change",
  "source": "aws.swf",
  "account": "444455556666",
  "time": "2020-05-08T22:37:26Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:swf:us-east-1:444455556666:/domain/canary"
  ],
  "detail": {
    "eventId": 48,
    "eventType": "WorkflowExecutionTerminated",
    "workflowExecutionDetail": {
      "executionInfo": {
        "execution": {
          "workflowId": "1234-5678-9012",
          "runId": "777788889999"
        }
      }
    }
  }
}

```

```
    },
    "workflowType": {
      "name": "1111-1111-1111",
      "version": "1.3"
    },
  },
  "startTimestamp": 1588977445279,
  "closeTimestamp": 1588977446062,
  "executionStatus": "CLOSED",
  "closeStatus": "TERMINATED",
  "parent": null,
  "parentExecutionArn": null,
  "tagList": null,
  "cancelRequested": false
},
"executionConfiguration": {
  "taskStartToCloseTimeout": "60",
  "executionStartToCloseTimeout": "120",
  "taskList": {
    "name": "1111-1111-1111-2222-2222-2222"
  },
  "taskPriority": null,
  "childPolicy": "TERMINATE",
  "lambdaRole": null
},
"openCounts": {
  "openActivityTasks": 0,
  "openDecisionTasks": 1,
  "openTimers": 0,
  "openChildWorkflowExecutions": 0,
  "openLambdaFunctions": 0
},
"latestActivityTaskTimestamp": 1588977445882,
}
}
```

## 使AWS 使用者通知用 Amazon 簡單 workflow 服務

您可以使用設 [AWS 使用者通知](#) 定交付管道，以接收有關 Amazon 簡單 workflow 服務事件的通知。當事件符合您指定的規則時，便會收到通知。您可以透過多個管道接收事件通知，包括電子郵件、[AWS Chatbot](#) 聊天通知或 [AWS Console Mobile Application](#) 推送通知。您也可在 [Console Notifications Center](#) 中看到通知。使用者通知 支援彙總，可減少您在特定事件期間收到的通知數目。

## Amazon 簡易工作流程服務的合規驗證

第三方稽核員會評估 Amazon 簡單工作流程服務的安全性和合規性，作為多個 AWS 合規計劃的一部分。這些計劃包括 SOC、PCI、FedRAMP、HIPAA 等等。

如需特定規範計劃範圍內的 AWS 服務清單，請參閱[合規計劃AWS 服務範圍](#)方案)。如需一般資訊，請參閱[AWS 規範計劃AWS](#)。

您可以使用下載第三方稽核報告 AWS Artifact。如需詳細資訊，請參閱[下載 AWS Artifact](#) 的報表。

使用 Amazon SWF 時的合規責任取決於資料的敏感度、公司的合規目標以及適用的法律和法規。

AWS 提供下列資源以協助遵循法規：

- [安全性與合規快速入門指南](#) — 這些部署指南討論架構考量，並提供在上部署以安全性和法規遵循為重點的基準環境的步驟。AWS
- [建構 HIPAA 安全性與合規性白皮書 — 本白皮書](#)說明公司如何使用建立符合 HIPAA 標準的應用 AWS 程式。
- [AWS 合規資源AWS](#) — 此工作簿和指南集合可能適用於您的產業和所在地。
- [使用AWS Config 開發人員指南中的規則評估資源](#) — 此 AWS Config 服務會評估您的資源組態符合內部實務、產業準則和法規的程度。
- [AWS Security Hub](#)— 此 AWS 服務提供安全狀態的全面檢視，協助您檢查您 AWS 是否符合安全性產業標準和最佳做法。

## Amazon Simple Workflow Service 中的彈性

AWS 全球基礎設施是以 AWS 區域與可用區域為中心建置的。AWS 區域提供多個分開且隔離的實際可用區域，並以低延遲、高輸送量和高度備援聯網功能相互連結。透過可用區域，您可以設計與操作的應用程式和資料庫，在可用區域之間自動容錯移轉而不會發生中斷。可用區域的可用性、容錯能力和擴充能力，均較單一或多個資料中心的傳統基礎設施還高。

如需有關 AWS 區域與可用區域的詳細資訊，請參閱 [AWS 全域基礎設施](#)。

除了AWS全球基礎設施，Amazon SWF 提供數種支援資料復原和備份需求的功能。

## Amazon Simple Workflow Service 的基礎設施安全

作為一種受管服務，Amazon Simple Workflow Service 受AWS全局網絡安全過程，詳情請參閱[Amazon Web Services：安全流程概觀](#)白皮書。

您使用AWS發佈的 API 呼叫，透過網路存取 Amazon SWF。用戶端必須支援 Transport Layer Security (TLS) 1.0 或更新版本。建議使用 TLS 1.2 或更新版本。用戶端也必須支援具備完美轉送私密 (PFS) 的密碼套件，例如臨時 Diffie-Hellman (DHE) 或橢圓曲線臨時 Diffie-Hellman (ECDHE)。現代系統 (如 Java 7 和更新版本) 大多會支援這些模式。

此外，請求必須使用存取金鑰 ID 和與 IAM 委託人相關聯的私密存取金鑰來簽署。或者，您可以使用 [AWS Security Token Service](#) (AWS STS) 來產生暫時安全憑證來簽署請求。

您可從任何網路位置呼叫這些 API 操作，而 Amazon SWF 確實可支援基於資源的存取政策，以納入依據來源 IP 地址的限制。您也可以使用 Amazon SWF 政策，以便從特定 Amazon Virtual Private Cloud (Amazon VPC) 端點或特定 VPC 中控制存取權。實際上，這只 VPC 離AWS網路存取。

## Amazon Simple Workflow Service 中的組態與漏洞分析

組態和 IT 控制是 AWS 與身為我們客戶的您共同的責任。如需詳細資訊，請參閱 AWS [共同的責任模型](#)。

# 使用AWS CLI使用 Amazon Simple Workflow Service

Amazon Simple Workflow Service 的許多功能可以從AWS CLI。所以此AWS CLI提供了一種替代方法，而不是將 Amazon SWF 與AWS Management Console或在某些情況下，以使用 Amazon SWF API 和AWS Flow Framework。

例如，您可以使用 AWS CLI 註冊新的工作流程類型：

```
aws swf register-workflow-type --domain MyDomain --name "MySimpleWorkflow" --workflow-version "v1"
```

您也可以列出您已註冊的工作流程類型：

```
aws swf list-workflow-types --domain MyDomain --registration-status REGISTERED
```

下列示範 JSON 格式的預設輸出範例：

```
{
  "typeInfos": [
    {
      "status": "REGISTERED",
      "creationDate": 1377471607.752,
      "workflowType": {
        "version": "v1",
        "name": "MySimpleWorkflow"
      }
    },
    {
      "status": "REGISTERED",
      "creationDate": 1371454149.598,
      "description": "MyDomain subscribe workflow",
      "workflowType": {
        "version": "v3",
        "name": "subscribe"
      }
    }
  ]
}
```

中的 Amazon SWF 命令AWS CLI可讓您啟動和管理工作流程執行、輪詢活動任務、記錄任務心跳等！若要獲取 Amazon SWF 命令的完整清單，以及可用引數描述以及顯示其使用方式的範例，請參[Amazon SWF](#)命令AWS CLI命令參考。

所以此AWS CLI命令與 Amazon SWF API 非常相似，所以您可使用AWS CLI，瞭解基礎的 Amazon SWF API。您也可以使用您現有的 API 知識在命令列上建立程式碼原型或執行 Amazon SWF 動作。

若要進一步了解AWS CLI，請參[AWS Command Line Interface使用者指南](#)。

# 使用 Amazon SWF API

除了使用中所述的 AWS 開發套件之外[使用 AWS SDK 進行開發](#)，您還可以直接使用 HTTP API。

若要使用 API，請將 HTTP 請求傳送給 [SWF 端點](#)，其符合您想要用於網域、工作流程和活動的區域。如需針對 Amazon SWF 發出 HTTP 要求的詳細資訊，請參閱[向 Amazon SWF 提出 HTTP 請求](#)。

本節提供有關使用 HTTP API 透過 Amazon SWF 開發工作流程的基本資訊。本節提供了更多進階功能，例如使用計時器、記錄工作流程 CloudTrail 和標記工作流程。[Amazon SWF 運作方式](#)

## 主題

- [向 Amazon SWF 提出 HTTP 請求](#)
- [Amazon SWF 動作按類別列表](#)
- [在 Amazon SWF 中建立基本工作流程](#)
- [向註冊網域](#)
- [在 Amazon SWF 中設定逾時值](#)
- [使用 Amazon SWF 註冊工作流程類型](#)
- [使用 Amazon SWF 註冊活動類型](#)
- [AWS Lambda 任務](#)
- [在 Amazon SWF 中開發活動工作者](#)
- [在 Amazon SWF 中開發決策者](#)
- [使用 Amazon SWF 啟動工作流程執行](#)
- [設定任務優先順序](#)
- [Amazon SWF 中處理錯誤](#)

## 向 Amazon SWF 提出 HTTP 請求

如果您未使用其中一個 AWS SDK，您可以使用 POST 請求方法透過 HTTP 執行 Amazon Simple Workflow Service (Amazon SWF) 操作。POST 方法需要您在請求標頭中指定操作，並在請求內文中提供 JSON 格式的操作資料。

## HTTP 標頭內容

Amazon SWF 需要 HTTP 請求標頭中的下列資訊：

- hostAmazon SWF 終端節點。
- x-amz-date 您必須在 HTTP 提供時間戳記 Date 標頭或 AWS x-amz-date header (有些 HTTP 用戶端程式庫不讓您設定 Date 標頭)。當 x-amz-date 標題存在時，系統會在驗證請求時略過任何 Date 標題。

日期必須使用 HTTP/1.1 RFC 所指定之下列三種格式中的其中一種來指定：

- Sun, 06 Nov 1994 08:49:37 GMT (RFC 822 , 已於 RFC 1123 更新)
- Sunday, 06-Nov-94 08:49:37 GMT (RFC 850 , 已於 RFC 1036 淘汰)
- Sun Nov 6 08:49:37 1994 (ANSI C 的 asctime() 格式)
- x-amzn-authorization 下列格式的簽署請求參數：

```
AWS3 AWSAccessKeyId=####,Algorithm=HmacSHA256, [,SignedHeaders=Header1;Header2;...]
Signature=S(StringToSign)
```

AWS3—這是一個AWS實作專屬標籤表示用來簽署請求的身分驗證版本 (目前, Amazon SWF), 此值始終為AWS3。

AWSAccessKeyId—您的AWS存取金鑰 ID。

Algorithm—用來建立登入字串之 HMAC-SHA 值的演算法 (例如HmacSHA256或者HmacSHA1。

Signature—Base64 (算法 (StringToSign, 簽署金鑰))。詳細資訊, 請參閱「[計算 Amazon SWF 的 HMAC-SHA 簽章](#)」

SignedHeaders—(選用) 存在時, 必須包含標準化 HttpHeaders 計算中所使用的所有 HTTP 標頭列表。單一分號字元 (;) (ASCII 字元 59) 必須用作清單值的分隔符號。

- x-amz-target—請求的目標服務和資料的操作, 格式如下：

```
com.amazonaws.swf.service.model.SimpleWorkflowService. + <action>
```

例如 com.amazonaws.swf.service.model.SimpleWorkflowService.RegisterDomain

- content-type—需要指定 JSON 和字元集的類型, 如 application/json; charset=UTF-8

以下是建立網域之 HTTP 請求的範例標頭。

```
POST http://swf.us-east-1.amazonaws.com/ HTTP/1.1
Host: swf.us-east-1.amazonaws.com
```

```
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; rv:1.9.2.25) Gecko/20111212
  Firefox/3.6.25 ( .NET CLR 3.5.30729; .NET4.0E)
Accept: application/json, text/javascript, */*
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 115
Connection: keep-alive
Content-Type: application/json; charset=UTF-8
X-Requested-With: XMLHttpRequest
X-Amz-Date: Fri, 13 Jan 2012 18:42:12 GMT
X-Amz-Target: com.amazonaws.swf.service.model.SimpleWorkflowService.RegisterDomain
Content-Encoding: amz-1.0
X-Amzn-Authorization: AWS3
  AWSSignature=AKIAIOSFODNN7EXAMPLE,Algorithm=HmacSHA256,SignedHeaders=Host;X-Amz-
Date;X-Amz-Target;Content-Encoding,Signature=tzjkF551xAxPhzp/BRGFYQRQRq6CqrM254dTDE/
EncI=
Referer: http://swf.us-east-1.amazonaws.com/explorer/index.html
Content-Length: 91
Pragma: no-cache
Cache-Control: no-cache

{"name": "867530902",
  "description": "music",
  "workflowExecutionRetentionPeriodInDays": "60"}
```

以下是對應的 HTTP 回應範例。

```
HTTP/1.1 200 OK
Content-Length: 0
Content-Type: application/json
x-amzn-RequestId: 4ec4ac3f-3e16-11e1-9b11-7182192d0b57
```

## HTTP 內文內容

HTTP 請求的內文包含 HTTP 請求標頭中所指定之操作的資料。使用 JSON 資料格式，同時傳遞資料值和資料結構。使用括號符號，可以將元素巢套於其他元素內。例如，下面顯示了一個請求，該請求列出在兩個指定時間點之間啟動的所有工作流程執行（使用 Unix Time 表示法）。

```
{
  "domain": "867530901",
  "startTimeFilter":
```

```
{
  "oldestDate": 1325376070,
  "latestDate": 1356998399
},
"tagFilter":
{
  "tag": "music purchase"
}
}
```

## 範例 Amazon SWF JSON 請求和回應

下列範例顯示向 Amazon SWF 提出請求，取得先前建立之網域的描述。接着會顯示 Amazon SWF 回應。

### HTTP POST 請求

```
POST http://swf.us-east-1.amazonaws.com/ HTTP/1.1
Host: swf.us-east-1.amazonaws.com
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; rv:1.9.2.25) Gecko/20111212
  Firefox/3.6.25 ( .NET CLR 3.5.30729; .NET4.0E)
Accept: application/json, text/javascript, */*
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 115
Connection: keep-alive
Content-Type: application/json; charset=UTF-8
X-Requested-With: XMLHttpRequest
X-Amz-Date: Sun, 15 Jan 2012 03:13:33 GMT
X-Amz-Target: com.amazonaws.swf.service.model.SimpleWorkflowService.DescribeDomain
Content-Encoding: amz-1.0
X-Amzn-Authorization: AWS3
  AWSSecretAccessKey=AKIAIOSFODNN7EXAMPLE,Algorithm=HmacSHA256,SignedHeaders=Host;X-Amz-
  Date;X-Amz-Target;Content-
  Encoding,Signature=IFJtq3M366CHqM1TpyqYqd9z0ChCoKDC5SCJBsLifu4=
Referer: http://swf.us-east-1.amazonaws.com/explorer/index.html
Content-Length: 21
Pragma: no-cache
Cache-Control: no-cache

{"name": "867530901"}
```

## Amazon SWF 回應

```
HTTP/1.1 200 OK
Content-Length: 137
Content-Type: application/json
x-amzn-RequestId: e86a6779-3f26-11e1-9a27-0760db01a4a8

{"configuration":
  {"workflowExecutionRetentionPeriodInDays": "60"},
  "domainInfo":
  {"description": "music",
    "name": "867530901",
    "status": "REGISTERED"}
}
```

請注意，協定 (HTTP/1.1) 後面會接著狀態碼 (200)。代碼值 200 表示操作成功。

Amazon SWF 不序列化 null 值。如果您的 JSON 解析器設定為序列化請求的 null 值，Amazon SWF 會忽略這些值。

## 計算 Amazon SWF 的 HMAC-SHA 簽章

每個向 Amazon SWF 發出的請求都必須進行身分驗證。所以此 AWSSDK 會自動簽署您的請求並管理您的字符型身份驗證。但若您想要撰寫自己的 HTTP POST 請求，您需要為 HTTP POST Header 內容建立 x-amzn-authorization 值，做為請求身分驗證的一部分。

如需格式化這些標頭的詳細資訊，請參閱「[HTTP 標頭內容](#)」。對於 AWS SDK for Java 的實作 AWS 版本 3 簽名，請參閱 [AWSSigner.java](#) 類別。

### 建立請求簽章

在您建立 HMAC-SHA 請求簽章之前，您必須取得您的 AWS 登入資料 (存取金鑰 ID 和私密金鑰)。

#### Important

您可以使用 SHA1 或 SHA256 簽署您的請求。但整個簽署過程中請務必使用同一個方法。您選擇的方法必須符合 HTTP 標頭中 Algorithm 名稱的值。

### 建立請求簽章

1. 建立 HTTP 請求標頭的正式格式。HTTP 標頭的正式格式包含以下內容：

- host
- 任何以 x-amz- 開頭的標頭元素

如需這些內含標頭的詳細資訊，請參閱「[HTTP 標頭內容](#)」。

- 將每一個標頭名稱值對的標頭名稱 (不是標頭值) 轉換成小寫。
- 建立逗號分隔標頭值的標頭名稱映射。

```
x-amz-example: value1
x-amz-example: value2 => x-amz-example:value1,value2
```

如需詳細資訊，請參閱「[RFC 2616 的 4.2 節](#)」。

- 將每一個標頭名稱值對的名稱值對轉換成格式為 headerName:headerValue 的字串。裁剪 headerName 和 headerValue 開頭和結尾的任何空格，冒號兩側不留空格。

```
x-amz-example1:value1,value2
x-amz-example2:value3
```

- 在每個轉換的字串後面插入新的一行 (U+000A)，包括最後的字串。
  - 依標頭名稱按字母順序排序轉換的字串集合。
- 建立登入字串值，包含下列項目：

- 行1：HTTP 方法 (POST)，後面接新行。
- 行2：請求 URI (/)，後面接新行。
- 行3：空字串，後面接新行。

#### Note

查詢字串一般會出現在此，但 Amazon SWF 不使用查詢字串。

- LINES4-n：代表您在步驟 1 中計算之正式請求標頭的字串，後面接新行。此新行會在標頭和 HTTP 請求內文之間建立空白行。如需詳細資訊，請參閱 [RFC 2616](#)。
  - 請求內文後面「不」接新行。
- 運算登入字串值的 SHA256 或 SHA1 摘要。整個過程使用同一個 SHA 方法。

4. 使用上一個步驟所得值的 SHA256 或 SHA1 摘要 (視您使用的方法而定)，以及暫時私密存取金鑰來計算 HMAC-SHA 並進行 Base64 編碼。AWS 使用 [GetSessionToken](#) API 動作。

**Note**

Amazon SWF 期望一個等號 (=) 在 Base64 編碼的 HMAC-SHA 值結尾處。如果您的 Base64 編碼例行作業不包含附加的等號，請在此值結尾處加上等號。

如需將暫時安全憑證結合使用 Amazon SWF 和其他 AWS 服務，請參閱 [AWS 可搭配 IAM 運作的服務](#) 中的 IAM User Guide。

5. 將結果值放在 Signature 名稱中 x-amzn-authorization 發送給 Amazon SWF 的 HTTP 請求標頭。
6. Amazon SWF 會驗證請求並執行指定的操作。

## Amazon SWF 動作按類別列表

本節列出 Amazon SWF 應用程式設計介面 (API) 中 Amazon SWF 動作的參考主題。其依「功能類別」列出。

如需字母順序的動作清單，請參閱 [Amazon 簡單工作流程服務 API 參考](#)。

### 主題

- [與活動相關的動作](#)
- [與決策者相關的動作](#)
- [與工作流程執行相關的動作](#)
- [與管理相關的動作](#)
- [可見度動作](#)

## 與活動相關的動作

活動工作者使用 `PollForActivityTask` 來取得新的活動工作。工作者接收來自 Amazon SWF 的活動工作後，將執行該任務，若成功會回應 `RespondActivityTaskCompleted`，若失敗則回應 `RespondActivityTaskFailed`。

下列為活動工作者執行的動作。

- [PollForActivityTask](#)
- [RespondActivityTaskCompleted](#)
- [RespondActivityTaskFailed](#)
- [RespondActivityTaskCanceled](#)
- [RecordActivityTaskHeartbeat](#)

## 與決策者相關的動作

決策者使用 `PollForDecisionTask` 來取得決策工作。決策者接收來自 Amazon SWF 的決策任務後，命令會檢查其工作流程執行歷史記錄，並決定要執行的作業。其會呼叫 `RespondDecisionTaskCompleted` 來完成決策工作，並提供零個或多個後續決策。

下列為決策者執行的動作。

- [PollForDecisionTask](#)
- [RespondDecisionTaskCompleted](#)

## 與工作流程執行相關的動作

在工作流程執行方面可使用下列動作。

- [RequestCancelWorkflowExecution](#)
- [StartWorkflowExecution](#)
- [SignalWorkflowExecution](#)
- [TerminateWorkflowExecution](#)

## 與管理相關的動作

雖然您可以從 Amazon SWF 主控台執行管理任務，但您可以使用本節中的動作來自動化功能或建立自己的管理工具。

### 活動管理

- [RegisterActivityType](#)
- [DeprecateActivityType](#)

- [UndeprecateActivityType](#)
- [DeleteActivityType](#)

## 工作流程管理

- [RegisterWorkflowType](#)
- [DeprecateWorkflowType](#)
- [UndeprecateWorkflowType](#)
- [DeleteWorkflowType](#)

## 網域管理

這些動作可讓您註冊和取代 Amazon SWF 網域。

- [RegisterDomain](#)
- [DeprecateDomain](#)
- [UndeprecateDomain](#)

如需這些網域管理動作的詳細資訊和範例，請參閱「[向註冊網域](#)」。

## 工作流程執行管理

- [RequestCancelWorkflowExecution](#)
- [TerminateWorkflowExecution](#)

## 可見度動作

雖然您可以從 Amazon SWF 主控台執行可見性動作，但您可以使用本節中的動作來建立自己的主控台或管理工具。

## 活動可見度

- [ListActivityTypes](#)
- [DescribeActivityType](#)

## 工作流程可見度

- [ListWorkflow类型](#)
- [DescribeWorkflow类型](#)

## 工作流程執行可見度

- [DescribeWorkflowExecution](#)
- [ListOpenWorkflowExecutions](#)
- [ListClosedWorkflowExecutions](#)
- [CountOpenWorkflowExecutions](#)
- [CountClosedWorkflowExecutions](#)
- [GetWorkflowExecutionHistory](#)

## 網域可見度

- [ListDomains](#)
- [DescribeDomain](#)

## 工作清單可見度

- [CountPendingActivityTasks](#)
- [CountPendingDecisionTasks](#)

## 在 Amazon SWF 中建立基本工作流程

建立基本循序工作流程包含下列階段。

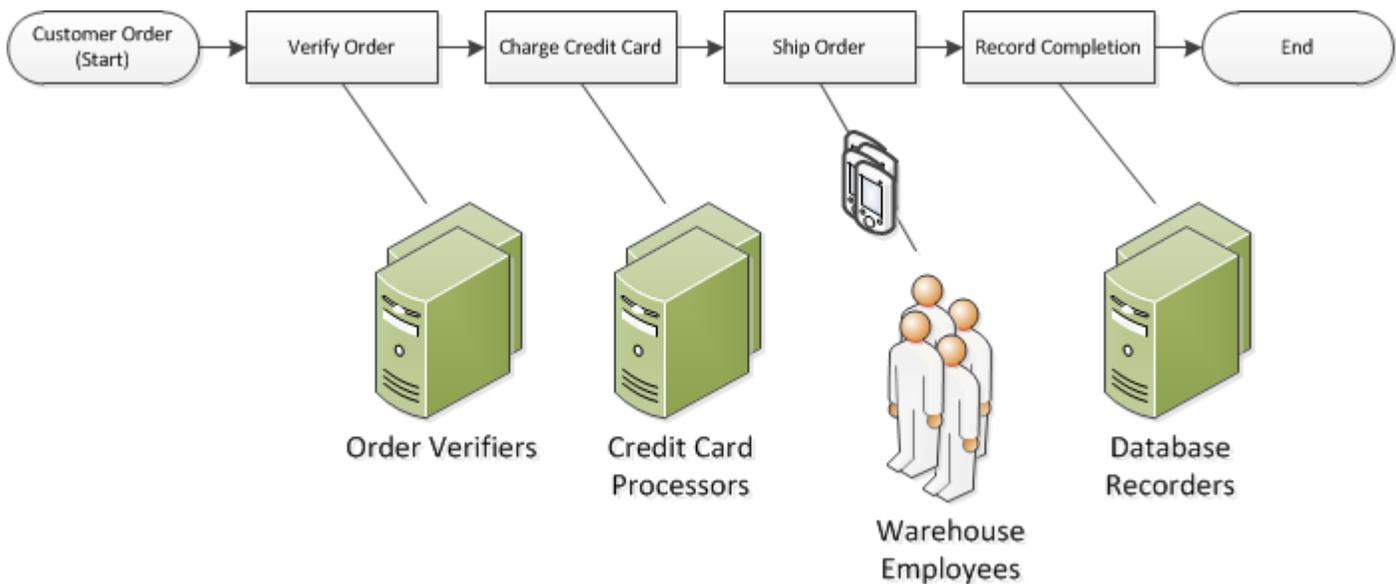
- 為工作流程建模、註冊其類型，以及註冊其活動類型
- 開發與啟動可執行活動任務的活動工作者
- 開發與啟動決策者，以使用工作流程歷史記錄來判斷後續步驟
- 開發與啟動工作流程啟動者 (即啟動工作流程執行的應用程式)。

## 為工作流程和其活動建模

若要使用 Amazon SWF，請將應用程式中的邏輯步驟建模為活動。活動代表工作流程中的單一邏輯步驟或任務。例如，授權信用卡是一種活動，包含提供信用卡卡號和其他資訊，以及接收核准碼或信用卡遭拒的訊息。

除了定義活動之外，您也需要定義可處理決策點的協調性邏輯。例如，協調性邏輯可能會根據信用卡獲准還是遭拒來排定不同的後續活動。

下圖顯示具有四個活動（「確認訂單」、「使用信用卡付費」、「送出訂單」和「記錄完成」）的循序客戶訂單工作流程範例。



## 向註冊網域

您工作流程和活動類型，以及工作流程執行本身皆屬於「網域」的範疇內。網域會自同個帳戶中隔離出一組類型、執行和任務清單。

您可以通過使用AWS Management Console或通過使用RegisterDomain操作。下列為使用 API 的範例。

```
https://swf.us-east-1.amazonaws.com
RegisterDomain
{
  "name" : "867530901",
  "description" : "music",
  "workflowExecutionRetentionPeriodInDays" : "60"
```

```
}
```

參數以 JavaScript 物件標記法 (JSON) 格式表示。保留期間在此設定為 60 天。在保留期間，使用 AWS Management Console 或 Amazon SWF API。

註冊網域之後，您應該註冊工作流程所使用的工作流程類型和活動類型。因為註冊工作流程和活動類型的必要資訊中需有已註冊的網域名稱，所以您需要先註冊網域。

## 另請參閱

[RegisterDomain](#) 中的 Amazon Simple Workflow Service API 參考

## 在 Amazon SWF 中設定逾時值

### 主題

- [逾時值限額](#)
- [工作流程執行和決策任務逾時](#)
- [活動任務逾時](#)
- [另請參閱](#)

## 逾時值限額

超時值始終以秒為單位聲明，並且可以設置為一年 ( 31536000 秒 ) 的任意秒數——任何 workflow 或活動的最大執行限制。特殊值 NONE 用來設定逾時參數「無逾時」或無限，但仍適用一年的上限。

## 工作流程執行和決策任務逾時

您可在註冊 workflow 類型時，為您的 workflow 和決策任務設定逾時值。例如：

```
https://swf.us-east-1.amazonaws.com
RegisterWorkflowType
{
  "domain": "867530901",
  "name": "customerOrderWorkflow",
  "version": "1.0",
  "description": "Handle customer orders",
  "defaultTaskStartToCloseTimeout": "600",
  "defaultExecutionStartToCloseTimeout": "3600",
```

```
"defaultTaskList": { "name": "mainTaskList" },
"defaultChildPolicy": "TERMINATE"
}
```

此工作流程類型註冊會將 [defaultTaskStartToCloseTimeout](#) 設成 600 秒 (10 分鐘)，將 [defaultExecutionStartToCloseTimeout](#) 設成 3600 秒 (1 小時)。

如需工作流程類型註冊的詳細資訊，請參[使用 Amazon SWF 註冊工作流程類型](#)，和[RegisterWorkflowType](#)中的Amazon Simple Workflow Service API 參考。

您可以透過指定 [executionStartToCloseTimeout](#)，覆寫為 `defaultExecutionStartToCloseTimeout` 設定的值。

## 活動任務逾時

您可在註冊活動類型時，為您的活動任務設定逾時值。例如：

```
https://swf.us-east-1.amazonaws.com
RegisterActivityType
{
  "domain": "867530901",
  "name": "activityVerify",
  "version": "1.0",
  "description": "Verify the customer credit",
  "defaultTaskStartToCloseTimeout": "600",
  "defaultTaskHeartbeatTimeout": "120",
  "defaultTaskList": { "name": "mainTaskList" },
  "defaultTaskScheduleToStartTimeout": "1800",
  "defaultTaskScheduleToCloseTimeout": "5400"
}
```

此活動類型註冊將 [defaultTaskStartToCloseTimeout](#) 設成 600 秒 (10 分鐘)、將 [defaultTaskHeartbeatTimeout](#) 設成 120 秒 (2 分鐘)、將 [defaultTaskScheduleToStartTimeout](#) 設成 1800 秒 (30 分鐘)，以及將 [defaultTaskScheduleToCloseTimeout](#) 設成 5400 秒 (1.5 小時)。

如需活動類型註冊的詳細資訊，請參[使用 Amazon SWF 註冊活動類型](#)，和[RegisterActivityType](#)中的Amazon Simple Workflow Service API 參考。

排定活動任務時，您可以透過指定 [taskStartToCloseTimeout](#)，覆寫為 `defaultTaskStartToCloseTimeout` 設定的值。

## 另請參閱

### [Amazon SWF 超時類型](#)

## 使用 Amazon SWF 註冊工作流程類型

本節討論的範例使用 Amazon SWF API 註冊工作流程類型。您在註冊期間指定的名稱和版本會形成工作流程類型的唯一識別符。指定的網域必須已使用 [RegisterDomain](#) API 動作來註冊。

以下範例中的逾時參數是以秒指定的期間值。您可以為 `defaultTaskStartToCloseTimeout` 參數使用期間指標 `NONE` 來表示沒有逾時。但是您不能為 `defaultExecutionStartToCloseTimeout` 指定 `NONE` 值，工作流程執行可執行的時間上限為一年。超過此限制一律會導致工作流程執行逾時。如果您指定大於一年的 `defaultExecutionStartToCloseTimeout` 值，註冊就會失敗。

```
https://swf.us-east-1.amazonaws.com
RegisterWorkflowType
{
  "domain" : "867530901",
  "name" : "customerOrderWorkflow",
  "version" : "1.0",
  "description" : "Handle customer orders",
  "defaultTaskStartToCloseTimeout" : "600",
  "defaultExecutionStartToCloseTimeout" : "3600",
  "defaultTaskList" : { "name": "mainTaskList" },
  "defaultChildPolicy" : "TERMINATE"
}
```

## 另請參閱

[RegisterWorkflowType](#) 中的 Amazon Simple Workflow Service API 參考

## 使用 Amazon SWF 註冊活動類型

下列範例使用 Amazon SWF API 註冊活動類型。您在註冊期間指定的名稱和版本會形成網域內活動類型的唯一識別符。指定的網域必須已使用 `RegisterDomain` 動作註冊。

本範例中的逾時參數是以秒指定的期間值。您可以使用期間指標 `NONE` 來表示沒有逾時。

```
https://swf.us-east-1.amazonaws.com
```

```
RegisterActivityType
{
  "domain" : "867530901",
  "name" : "activityVerify",
  "version" : "1.0",
  "description" : "Verify the customer credit",
  "defaultTaskStartToCloseTimeout" : "600",
  "defaultTaskHeartbeatTimeout" : "120",
  "defaultTaskList" : { "name" : "mainTaskList" },
  "defaultTaskScheduleToStartTimeout" : "1800",
  "defaultTaskScheduleToCloseTimeout" : "5400"
}
```

## 另請參閱

[RegisterActivityType](#)中的Amazon Simple Workflow Service API 參考

## AWS Lambda 任務

### 主題

- [關於 AWS Lambda](#)
- [使用 Lambda 任務的優點與限制](#)
- [在工作流程中使用 Lambda 任務](#)

## 關於 AWS Lambda

AWS Lambda是一種完整受管的運算服務，可執行程式碼回應自訂程式碼產生的事件，或各種AWS服務 (例如 Amazon S3、DynamoDB、Amazon Kinesis、Amazon SNS 和 Amazon Cognito)。如需有關 Lambda 的詳細資訊，請參閱 [AWS Lambda 開發人員指南](#)。

Amazon Simple Workflow Services 提供 Lambda 任務，以讓您執行 Lambda 函數來取代傳統 Amazon SWF 活動，或與之並存。

### Important

您的AWS帳戶將會收取 Amazon SWF 代您執行 Lambda 執行 (請求) 的費用。如需 Lambda 定價的詳細資訊，請參<https://aws.amazon.com/lambda/pricing/>。

## 使用 Lambda 任務的優點與限制

使用 Lambda 任務取代傳統 Amazon SWF 活動有多項優點：

- Lambda 任務不用像 Amazon SWF 活動類型一樣需要註冊或版本控制。
- 您可以使用任何已於工作流程中定義的現有 Lambda 函數。
- 會直接呼叫 Lambda 函數；有別於傳統活動的執行所需，您不需要實作工作者程式即可予以執行。
- Lambda 為您提供用於跟蹤與分析函數執行的指標和日誌。

您還需要知道數個 Lambda 任務的相關限制：

- Lambda 任務只能在 AWS 區域，為 Lambda 提供支援。請參閱 [Lambda 區域與端點](#) 中的 Amazon Web Services 一般參考，瞭解有關 Lambda 當前支持的區域的詳細信息。
- 目前只有基本 SWF HTTP API 和適用於 Java 的 AWS Flow Framework 支援 Lambda 任務。適用於 Ruby 的 AWS Flow Framework 目前不支援 Lambda 任務。

## 在工作流程中使用 Lambda 任務

若要在 Amazon SWF 工作流程中使用 Lambda 任務，您需要：

1. 設定 IAM 角色以向 Amazon SWF 提供呼叫 Lambda 函數的許可。
2. 將 IAM 角色連接至工作流程。
3. 在工作流程執行期間，呼叫 Lambda 函數。

### 設定 IAM 角色

在您可以從 Amazon SWF 呼叫 Lambda 函數之前，您必須先提供 IAM 角色，以提供從 Amazon SWF 存取 Lambda 的存取權。您可擇一方法：

- 選擇一個預定義的角色，AWSLambdaRole，以授予您的工作流程，以呼叫與您帳戶相關聯的任何 Lambda 函數。
- 定義您自己的政策和相關聯的角色將許可授予工作流程，以呼叫其 Amazon Lambda 函數 (ARN) 指定的特定 Lambda 函數。

## 限制 IAM 角色的許可

您可以通過使用SourceArn和SourceAccount資源信任策略中的上下文鍵。這些密鑰限制 IAM 策略的使用，以便僅從屬於指定域 ARN 的 Amazon 簡單 workflow 服務執行中使用。如果您同時使用兩個全局條件上下文鍵，aws:SourceAccount值和aws:SourceArn在同一政策陳述式中使用時，值必須使用相同的帳戶 ID。

在以下信任政策範例中，我們使用SourceArn上下文鍵限制 IAM 服務角色僅用於屬於someDomain在帳戶中，123456789012。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": "swf.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "ArnLike": {
          "aws:SourceArn": "arn:aws:swf:*:123456789012:/domain/someDomain"
        }
      }
    }
  ]
}
```

在以下信任政策範例中，我們使用SourceAccount上下文鍵將 IAM 服務角色限制在帳戶中的 Amazon 簡單 workflow 服務執行中使用，123456789012。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": "swf.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
```

```
    "Condition": {
      "StringLike": {
        "aws:SourceAccount": "123456789012"
      }
    }
  ]
}
```

向 Amazon SWF 提供呼叫任何 Lambda 角色的存取權

您可以使用預先定義的角色 `AWSLambdaRole`，讓您的 Amazon SWF 工作流程能夠呼叫與您賬戶相關聯的任何 Lambda 函數。

使用 `AWSLambdaRole` 授予 Amazon SWF 訪問權，以呼叫 Lambda 函數

1. 開啟 [Amazon IAM 主控台](#)。
2. 選擇 Roles (角色)，然後選擇 Create New Role (建立新角色)。
3. 為您的角色提供名稱，例如 `swf-lambda`，然後選擇 Next Step (下一步)。
4. UNDERAWS 服務角色，選擇 Amazon SWF，然後選擇後續步驟。
5. 在連接政策屏幕上，選擇 `AWSLambdaRole` 從清單中。
6. 當您檢閱好角色後，請選擇 Next Step (下一步)，然後選擇 Create Role (建立角色)。

定義 IAM 角色以提供呼叫特定 Lambda 函數的存取權

如果您想要提供從您的工作流程呼叫特定 Lambda 函數的存取權，您需要定義您自己的 IAM 政策。

建立 IAM 政策以提供特定 Lambda 函數的存取權

1. 開啟 [Amazon IAM 主控台](#)。
2. 選擇 Policies (政策)，然後選擇 Create Policy (建立政策)。
3. 選擇複製 AWS 托管策略，然後選擇 `AWSLambdaRole` 從清單中。即會為您產生政策。您可選擇性編輯其名稱及描述，以符合您的需求。
4. 在中資源欄位 Policy Document (政策文件)，請添加 Lambda 函數的 ARN。例如：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
```

```
"Effect": "Allow",
"Action": [
  "lambda:InvokeFunction"
],
"Resource": [
  "arn:aws:lambda:us-east-1:111111000000:function:hello_lambda_function"
]
}
]
}
```

### Note

有關如何在 IAM 角色中指定資源的完整說明，請參閱[IAM 政策概觀](#)在使用 IAM。

5. 選擇 Create Policy (建立政策) 來完成您的政策建立程序。

然後您可以在建立新 IAM 角色時選擇此政策，並使用該角色授予您 Amazon SWF 工作流程的呼叫存取權。此程序與使用AWSLambdaRole政策。相反，請在建立角色時選擇您自己的政策。

使用您的 Lambda 策略創建 Amazon SWF 角色

1. 開啟 [Amazon IAM 主控台](#)。
2. 選擇 Roles (角色)，然後選擇 Create New Role (建立新角色)。
3. 為您的角色提供名稱，例如 swf-lambda-function，然後選擇 Next Step (下一步)。
4. UNDERAWS服務角色，選擇Amazon SWF，然後選擇後續步驟。
5. 在連接政策窗口中，從清單中選擇 Lambda 函數專用政策。
6. 當您檢閱好角色後，請選擇 Next Step (下一步)，然後選擇 Create Role (建立角色)。

## 將 IAM 角色連接至工作流程

在您定義 IAM 角色之後，需要將角色連接至工作流程，讓工作流程用以呼叫您供 Amazon SWF 取的 Lambda 函數。

有兩個地方可讓您將角色連接至工作流程：

- 工作流程類型註冊期間。接著可能會使用此角色做為每次執行該工作流程類型的預設 Lambda 角色。

- 重新啟動工作流程執行時。只有在此工作流程執行期間 (和整個執行期間) 才會使用此角色。

提供工作流程類型的預設 Lambda 角色

- 調用時RegisterWorkflow鍵入，設置defaultLambdaRole欄位設定為所定義角色的 ARN。

提供要在工作流程執行期間使用的 Lambda 角色

- 調用時StartWorkflow執行時，請將 LambdaRole 欄位設定為所定義角色的 ARN。

#### Note

如果帳戶呼叫RegisterWorkflow類型或StartWorkflowExecution 沒有使用指定角色的許可，則呼叫將會因OperationNotPermittedFault。

## 從 Amazon SWF 工作流程呼叫 Lambda 函數

您可以使用ScheduleLambdaFunctionDecision將數據類型歸類型，識別要在工作流程執行期間呼叫的 Lambda 函數。

在調用RespondDecisionTaskCompleted，請提供ScheduleLambdaFunctionDecision決策列表的屬性。例如：

```
{
  "decisions": [{
    "ScheduleLambdaFunctionDecisionAttributes": {
      "id": "lambdaTaskId",
      "name": "myLambdaFunctionName",
      "input": "inputToLambdaFunction",
      "startToCloseTimeout": "30"
    },
  ]
}
```

設定下列參數：

- 具有 Lambda 任務識別符的 id。這必須是 1-256 個字元的字串，而且不得包含：(冒號)、/(斜線)、| (垂直線) 字元、任何控制字元 (\u0000 - \u001f 和 \u007f - \u009f)，以及文字字串 arn。

- 具有 Lambda 函數名稱的 name。您的 Amazon SWF 工作流程必須擁有 IAM 角色，讓其能夠呼叫 Lambda 函數。提供的名稱必須遵循 FunctionName 參數，例如在 Lambda 呼叫動作中。
- 具有函數之選用輸入資料的 input。如果設定，則該觸發必須遵循 ClientContext 參數，例如在 Lambda 呼叫動作中。
- startToCloseTimeout (逾時)，其中包括函數在任務失敗和超時異常之前執行的可選最長期間 (秒)。NONE 值可以用來指定持續時間無限制。

如需詳細資訊，請參閱「[實作AWS Lambda工作](#)」

## 在 Amazon SWF 中開發活動工作者

活動工作者提供一或多個活動類型的實作。活動工作者會與 Amazon SWF 進行通訊，以接收並執行活動任務。您可以有多個活動工作者執行相同活動類型的活動任務。

決策者排程活動任務時，Amazon SWF 會讓活動任務供活動工作者使用。決策者排程活動任務時，會提供活動工作者執行活動任務所需的資料 (由您決定)。Amazon SWF 會將此資料插入活動任務，再將之傳送至活動工作者。

活動工作者將由您管理。活動工作者可以使用任何語言進行撰寫。工作者只要可以透過 API 與 Amazon SWF 進行通訊，即可於任何位置執行。因為 Amazon SWF 提供執行活動任務所需的所有資訊，所以所有活動工作者都可以是無狀態。無狀態可讓工作流程具高度擴充性；若要處理增加的容量需求，只需要新增更多活動工作者即可。

本節將說明如何實作活動工作者。活動工作者應該重複執行下列作業。

1. 向輪詢 Amazon SWF 執行活動任務。
2. 開始執行任務。
3. 如果任務為長期執行，則會定期向報告活動訊號給 Amazon SWF。
4. 報告任務已完成或失敗，並將結果傳回給 Amazon SWF。

### 主題

- [輪詢活動任務](#)
- [執行活動任務](#)
- [報告活動任務活動訊號](#)
- [完成或讓活動任務失敗](#)
- [啟動活動工作者](#)

## 輪詢活動任務

若要執行活動任務，每個活動工作者都必須定期呼叫PollForActivityTask動作。

在下列範例中，活動工作者 ChargeCreditCardWorker01 會輪詢任務清單 ChargeCreditCard-v0.1 上的任務。如果沒有活動任務，會在 60 秒後送回空回應。空回應是 taskToken 值為空白字串的 Task 結構。

```
https://swf.us-east-1.amazonaws.com
PollForActivityTask
{
  "domain" : "867530901",
  "taskList" : { "name": "ChargeCreditCard-v0.1" },
  "identity" : "ChargeCreditCardWorker01"
}
```

如果有活動任務，則 Amazon SWF 會將之傳回給活動工作者。任務包含決策者在排程活動時指定的資料。

活動工作者收到活動任務之後，即準備好要執行工作。下節將提供執行活動任務的相關資訊。

## 執行活動任務

收到活動任務之後，活動工作者即準備好執行之。

### 執行活動任務

1. 程式設計活動工作者，解譯任務中輸入欄位的內容。此欄位包含決策者在排程任務時指定的資料。
2. 程式設計活動工作者，開始處理資料和執行您的邏輯。

下節將說明如何程式設計活動工作者，將狀態更新提供給 Amazon SWF，讓您執行長時間執行的活動。

## 報告活動任務活動訊號

如果已向活動類型註冊活動訊號逾時，則活動工作者必須在超過活動訊號逾時之前記錄活動訊號。如果活動任務未在逾時內提供活動訊號，則任務會逾時，則 Amazon SWF 會予以關閉並排程新的決策任務，以通知決策者此逾時情形。決策者接著可以重新排程活動任務，或採取另一個動作。

逾時之後，如果活動工作者嘗試連絡 Amazon SWF，例如呼叫RespondActivityTaskCompleted，Amazon SWF 將返回UnknownResource錯誤能力

本節將說明如何提供活動的活動訊號。

若要記錄活動任務活動訊號，請程式設計活動工作者，呼叫 `RecordActivityTaskHeartbeat` 動作。此動作也提供字串欄位，可用來存放任意形式的資料，便於以適合您應用程式的方式來量化進度。

在此範例中，活動工作者會向回報 Amazon SWF，並使用詳細資訊欄位回報活動任務已完成 40%。若要回報活動訊號，活動工作者必須指定活動任務的任務字符。

```
https://swf.us-east-1.amazonaws.com
RecordActivityTaskHeartbeat
{
  "taskToken" : "12342e17-80f6-FAKE-TASK-TOKEN32f0223",
  "details" : "40"
}
```

此動作本身不會在工作流程執行歷史記錄中建立事件；不過，如果任務逾時，則工作流程執行歷史記錄將會包含 `ActivityTaskTimedOut` 事件，以包含來自活動工作者所產生的最後一個活動訊號中的資訊。

## 完成或讓活動任務失敗

執行任務之後，活動工作者應該回報活動任務已完成還是失敗。

### 完成活動任務

若要完成活動任務，請程式設計活動工作者，在成功完成活動任務之後呼叫 `RespondActivityTaskCompleted` 動作，指定任務字符。

在此範例中，活動工作者指出任務已順利完成。

```
https://swf.us-east-1.amazonaws.com
RespondActivityTaskCompleted
{
  "taskToken": "12342e17-80f6-FAKE-TASK-TOKEN32f0223",
  "results": "40"
}
```

活動完成時，Amazon SWF 會排程與活動相關聯之工作流程執行的新決策任務。

程式設計活動工作者，在即將完成任務之後輪詢另一個活動任務。這會建立活動工作者持續輪詢並完成任務的迴圈。

如果活動在開始關閉超時週期，或者如果計劃關閉超時，則 Amazon SWF 會逾時活動任務，並排程決策任務。這可讓決策者採取適當的動作 (例如重新排程任務)。

例如，如果 Amazon EC2 實例執行活動任務，而執行個體在任務完成之前失敗，則決策者會在工作流程執行歷史記錄中收到逾時事件。如果活動任務使用活動訊號，則任務無法在 Amazon EC2 實例失敗之後交付下個活動訊號時，決策者會收到事件。否則，活動任務無法在到達其中一個整體逾時之前完成時，決策者最終會收到事件。決策者接著可決定要重新指派任務，或採取其他動作。

## 讓活動任務失敗

如果活動工作者因某個原因而無法執行活動任務，但仍能與 Amazon SWF 進行通訊，請程式設計活動工作者讓任務失敗。

若要程式設計活動工作者讓活動任務失敗，請程式設計活動工作者，呼叫可指定任務之任務字符的 `RespondActivityTaskFailed` 動作。

```
https://swf.us-east-1.amazonaws.com
RespondActivityTaskFailed
{
  "taskToken" : "12342e17-80f6-FAKE-TASK-TOKEN32f0223",
  "reason" : "CC-Invalid",
  "details" : "Credit Card Number Checksum Failed"
}
```

身為開發人員，您可以定義存放在原因和詳細資訊欄位中的值。這些是任意形式的字串；您可以使用您應用程式適用的任何錯誤代碼慣例。Amazon SWF 不處理這些值。不過，Amazon SWF 可能會在主控台中顯示這些值。

活動任務失敗時，Amazon SWF 會排程與活動任務相關聯之工作流程執行的決策任務，來通知決策者此失敗情形。程式設計決策者，根據失敗的因素來處理失敗的活動，例如重新排程活動，或讓工作流程執行失敗。

## 啟動活動工作者

若要啟動活動工作者，請將邏輯封裝至可用於活動工作者平台的執行檔。例如，您可以將活動程式碼封裝為可在 Linux 和 Windows 伺服器上執行的 Java 執行檔。

啟動之後，工作者會開始輪詢任務。決策者排程活動任務之前，這些輪詢會逾時且沒有任務，而工作者會繼續輪詢。

因為輪詢是對外請求，所以活動工作者可以在任何可存取 Amazon SWF 端點的網路上執行。

您可以視需要啟動任意數目的活動工作者。決策者排程活動任務時，Amazon SWF 會將活動任務自動分配給輪詢活動工作者。

## 在 Amazon SWF 中開發決策者

決策者是在工作流程執行期間執行之工作流程類型的協調性邏輯實作。您可以為單一工作流程類型執行多個決策者。

因為工作流程執行的執行狀態存放在其工作流程歷史記錄中，所以決策者可以是無狀態的。Amazon SWF 會維護工作流程執行歷史記錄，並使用每個決策任務將其提供給決策者。這可讓您視需要動態新增和移除決策者，以高度擴展工作流程的處理。隨著系統負載的成長，您只要新增其他決策者即可處理增加的容量。不過，請注意，針對指定的工作流程執行，一次只會開啟一個決策任務。

每次工作流程執行的狀態變更時，Amazon SWF 都會排定決策任務。每次決策者收到決策任務時，都會執行下列作業：

- 解譯決策任務所提供的工作流程執行歷史記錄
- 根據工作流程執行歷史記錄來套用協調性邏輯，並做出後續步驟的決策。每個決策都是以「決策」結構表示
- 完成決策任務，並將決策清單提供給 Amazon SWF。

本節說明如何開發決策者，包含：

- 以程式設計您的決策者輪詢決策任務
- 以程式設計您的決策者解譯工作流程執行歷史記錄，並做出決策
- 以程式設計您的決策者回應決策任務。

本節中的範例顯示如何以程式設計您的決策者進行電子商務範例工作流程。

您可以使用任何您喜歡的語言來實作決策者，並在任何位置執行，只要決策者可以透過其服務 API 與 Amazon SWF 進行通訊。

### 主題

- [定義協調性邏輯](#)
- [輪詢決策任務](#)
- [套用協調性邏輯](#)
- [回應決策](#)

- [關閉工作流程執行](#)
- [啟動決策者](#)

## 定義協調性邏輯

開發決策者的第一件事是定義協調性邏輯。在電子商務範例中，於前一個活動完成之後排定每個活動的協調性邏輯可能會與下述類似：

```
IF lastEvent = "StartWorkflowInstance"
  addToDecisions ScheduleVerifyOrderActivity

ELSIF lastEvent = "CompleteVerifyOrderActivity"
  addToDecisions ScheduleChargeCreditCardActivity

ELSIF lastEvent = "CompleteChargeCreditCardActivity"
  addToDecisions ScheduleCompleteShipOrderActivity

ELSIF lastEvent = "CompleteShipOrderActivity"
  addToDecisions ScheduleRecordOrderCompletion

ELSIF lastEvent = "CompleteRecordOrderCompletion"
  addToDecisions CloseWorkflow

ENDIF
```

決策者會將協調性邏輯套用至工作流程執行歷史記錄，並在使用 RespondDecisionTaskCompleted 動作完成決策任務時建立決策清單。

## 輪詢決策任務

每個決策者都會輪詢決策任務。決策任務包含決策者用來產生決策的資訊 (例如，排定活動任務)。若要輪詢決策任務，決策者會使用 PollForDecisionTask 動作。

在此範例中，決策者會輪詢決策任務，並指定 customerOrderWorkflow-0.1 任務清單。

```
https://swf.us-east-1.amazonaws.com
PollForDecisionTask
{
  "domain": "867530901",
  "taskList": {"name": "customerOrderWorkflow-v0.1"},
  "identity": "Decider01",
```

```
"maximumPageSize": 50,  
"reverseOrder": true  
}
```

如果可從指定的任務清單中取得決策任務，則 Amazon SWF 會立即傳回它。如果沒有決策任務可取得，Amazon SWF 會持續開啟連線最多 60 秒，並在任務時立即傳回任務。如果沒有任務可取得，Amazon SWF 會傳回空回應。空回應是 taskToken 值為空白字串的 Task 結構。確定以程式設計您的決策者要在收到空回應時輪詢另一個任務。

如果有決策任務可取得，Amazon SWF 會傳回回應，其中包含決策任務以及工作流程執行歷史記錄的分頁檢視。

在此範例中，最新事件的類型指出已啟動的工作流程執行，而且輸入元素包含執行第一個任務所需的資訊。

```
{  
  "events": [  
    {  
      "decisionTaskStartedEventAttributes": {  
        "identity": "Decider01",  
        "scheduledEventId": 2  
      },  
      "eventId": 3,  
      "eventTimestamp": 1326593394.566,  
      "eventType": "DecisionTaskStarted"  
    }, {  
      "decisionTaskScheduledEventAttributes": {  
        "startToCloseTimeout": "600",  
        "taskList": { "name": "specialTaskList" }  
      },  
      "eventId": 2,  
      "eventTimestamp": 1326592619.474,  
      "eventType": "DecisionTaskScheduled"  
    }, {  
      "eventId": 1,  
      "eventTimestamp": 1326592619.474,  
      "eventType": "WorkflowExecutionStarted",  
      "workflowExecutionStartedEventAttributes": {  
        "childPolicy" : "TERMINATE",  
        "executionStartToCloseTimeout" : "3600",  
        "input" : "data-used-decider-for-first-task",  
        "parentInitiatedEventId": 0,  
        "tagList" : ["music purchase", "digital", "ricoh-the-dog"],  
      }  
    }  
  ]  
}
```

```
    "taskList": { "name": "specialTaskList" },
    "taskStartToCloseTimeout": "600",
    "workflowType": {
      "name": "customerOrderWorkflow",
      "version": "1.0"
    }
  }
},
...
}
```

收到工作流程執行歷史記錄之後，決策者會解譯歷史記錄，並根據協調性邏輯做出決策。

因為單一工作流程執行的工作流程歷史記錄事件數目可能很大，所以可能會將傳回的結果分割為數個頁面。要檢索後續頁面，請對PollForDecisionTask使用下一頁令牌由初始調用返回。請注意，您會不呼叫GetWorkflowExecutionHistory用這個下一頁令牌。相反地，請重新呼叫PollForDecisionTask。

## 套用協調性邏輯

決策者收到決策任務之後，請以程式設計您的決策者解譯工作流程執行歷史記錄，以判斷目前為止所發生的情況。據此，決策者應該會產生決策清單。

在電子商務範例中，我們只關心工作流程歷史記錄中的最後一個事件，因此邏輯定義如下。

```
IF lastEvent = "StartWorkflowInstance"
  addToDecisions ScheduleVerifyOrderActivity

ELSIF lastEvent = "CompleteVerifyOrderActivity"
  addToDecisions ScheduleChargeCreditCardActivity

ELSIF lastEvent = "CompleteChargeCreditCardActivity"
  addToDecisions ScheduleCompleteShipOrderActivity

ELSIF lastEvent = "CompleteShipOrderActivity"
  addToDecisions ScheduleRecordOrderCompletion

ELSIF lastEvent = "CompleteRecordOrderCompletion"
  addToDecisions CloseWorkflow

ENDIF
```

如果 `lastEvent` 是 `CompleteVerifyOrderActivity`，則 `ScheduleChargeCreditCardActivity` 活動將新增至決策清單。

決策者決定要做出的決策之後，可以使用適當的決策來回應 Amazon SWF。

## 回應決策

解譯工作流程歷史記錄並產生決策清單之後，決策者就準備好使用這些決策來回應 Amazon SWF。

以程式設計您的決策者從工作流程執行歷史記錄擷取所需的資料，然後建立可指定工作流程之下一個適當動作的決策。決策者會使用 `RespondDecisionTaskCompleted` 動作。請參 Amazon Simple Workflow Service API 參考查看可用 [決策類型](#)。

在電子商務範例中，決策者回應所產生的一組決策時，也會包含工作流程執行歷史記錄中的信用卡輸入。活動工作者接著會具有執行活動任務所需的資訊。

工作流程執行中的所有活動都完成時，決策者會關閉工作流程執行。

```
https://swf.us-east-1.amazonaws.com
RespondDecisionTaskCompleted
{
  "taskToken" : "12342e17-80f6-FAKE-TASK-TOKEN32f0223",
  "decisions" : [
    {
      "decisionType" : "ScheduleActivityTask",
      "scheduleActivityTaskDecisionAttributes" : {
        "control" : "OPTIONAL_DATA_FOR_DECIDER",
        "activityType" : {
          "name" : "ScheduleChargeCreditCardActivity",
          "version" : "1.1"
        },
        "activityId" : "3e2e6e55-e7c4-beef-feed-aa815722b7be",
        "scheduleToCloseTimeout" : "360",
        "taskList" : { "name" : "CC_TASKS" },
        "scheduleToStartTimeout" : "60",
        "startToCloseTimeout" : "300",
        "heartbeatTimeout" : "60",
        "input" : "4321-0001-0002-1234: 0212 : 234"
      }
    }
  ]
}
```

## 關閉工作流程執行

決策者判斷商務流程完成 (亦即，沒有其他可執行的活動) 時，決策者會產生決策來關閉工作流程執行。

若要關閉工作流程執行，請以程式設計您的決策者解譯工作流程歷史記錄中的事件，以判斷目前為止執行中所發生的情況，並查看是否應該關閉工作流程執行。

如果工作流程已順利完成，則會使用 `CompleteWorkflowExecution` 決策呼叫 `RespondDecisionTaskCompleted` 來關閉工作流程執行。或者，您可以使用 `FailWorkflowExecution` 決策讓錯誤執行失敗。

在電子商務範例中，決策者會檢閱歷史記錄、根據協調性邏輯將關閉工作流程執行的決策新增至其決策清單，並使用關閉工作流程決策來啟動 `RespondDecisionTaskCompleted` 動作。

### Note

在一些情況下，關閉工作流程執行會失敗。例如，如果在決策者關閉工作流程執行時收到訊號，則關閉決策將會失敗。若要處理此可能性，請確定決策者繼續輪詢決策任務。此外，確保接收下一個決策任務的決策者響應阻止執行關閉的事件 (在本例中為信號)。

您也可能支援取消工作流程執行。這特別適用於長時間執行的工作流程。若要支援取消，您的決策者應該處理歷史記錄中的 `WorkflowExecutionCancelRequested` 事件。此事件指出已請求取消執行。您的決策者應該執行適當的清除動作，例如，取消進行中活動任務，以及關閉使用 `CancelWorkflowExecution` 決策呼叫 `RespondDecisionTaskCompleted` 動作的工作流程。

下列範例呼叫 `RespondDecisionTaskCompleted` 以指定取消目前工作流程執行。

```
https://swf.us-east-1.amazonaws.com
RespondDecisionTaskCompleted
{
  "taskToken" : "12342e17-80f6-FAKE-TASK-TOKEN32f0223",
  "decisions" : [
    {
      "decisionType": "CancelWorkflowExecution",
      "CancelWorkflowExecutionAttributes": {
        "Details": "Customer canceled order"
      }
    }
  ]
}
```

```
}
```

Amazon SWF 會檢查以確認關閉或取消工作流程執行的決策是決策者所傳送的最後一個決策。亦即，一組決策不允許在關閉工作流程的決策後面還有決策。

## 啟動決策者

完成決策者開發之後，您即可準備啟動一或多個決策者。

若要啟動決策者，請將協調性邏輯封裝至可用於決策者平台的執行檔。例如，您可以將決策者程式碼封裝為可在 Linux 和 Windows 電腦上執行的 Java 執行檔。

啟動之後，決策者應該會開始向 Amazon SWF 輪詢任務。除非您啟動工作流程執行，而且 Amazon SWF 會排定決策任務，否則這些輪詢將會超時，並取得空回應。空回應是 `taskToken` 值為空白字串的 Task 結構。決策者只應該繼續輪詢。

Amazon SWF 會確保無論何時，都只會有一個決策任務作用於一個工作流程執行。這能防止決策發生衝突的問題。而且，Amazon SWF 會確保將單一決策任務指派給單一決策者，無論執行中的決策者數目為何。

在決策者處理另一個決策任務時，如果產生決策任務有問題，Amazon SWF 在目前任務完成之後才將新任務放入隊列中。目前任務完成之後，Amazon SWF 會將新的決策任務設為可用。而且，在下列情況下，會批次處理決策任務：如果多個活動在決策者處理決策任務時完成，則 Amazon SWF 只會建立單一新的決策任務來考量多個任務完成。不過，每個任務完成都會在工作流程執行歷史記錄中收到個別事件。

因為輪詢是對外請求，所以決策者可以在任何可存取 Amazon SWF 終端節點的網路上執行。

若要工作流程執行有所進展，必須執行一或多個決策者。您可以視需要啟動任意數目的決策者。Amazon SWF 支援相同任務清單上的多個決策者輪詢。

## 使用 Amazon SWF 啟動工作流程執行

您可以使用 `StartWorkflowExecution` 動作，從任何應用程式啟動已註冊之工作流程類型的工作流程執行。當您啟動執行時，就會與識別符 (稱為 `workflowId`) 建立關聯。`workflowId` 可以是您應用程式適用的任何字串 (例如訂單處理應用程式中的訂單號碼)。您無法將相同的 `workflowId` 用於相同網域內多個開啟的工作流程執行。例如，如果您使用 `workflowId Customer Order 01` 來啟動兩個工作流程執行，則不會啟動第二個工作流程執行，而且請求將會失敗。但是您可以使用 `workflowId` 一個封閉的執行。Amazon SWF 還會關聯一個系統生成的唯一標識符，稱為 `runId` 與每個工作流程執行。

註冊工作流程和活動類型之後，呼叫 `StartWorkflowExecution` 動作來啟動工作流程。 `input` 參數的值可以是啟動工作流程之應用程式所指定的任何字串。 `executionStartToCloseTimeout` 是工作流程執行從開始到結束之間可使用的時間長度 (秒)。超過此限制會導致工作流程執行逾時。與 Amazon SWF 中的其他一些超時參數不同，您不能指定 `NONE` 工作流程執行可執行的時間上限為一年。同樣地， `taskStartToCloseTimeout` 是與此工作流程執行相關聯的決策任務在逾時前所花的時間長度 (秒)。

```
https://swf.us-east-1.amazonaws.com
StartWorkflowExecution
{
  "domain" : "867530901",
  "workflowId" : "20110927-T-1",
  "workflowType" : {
    "name" : "customerOrderWorkflow", "version" : "1.1"
  },
  "taskList" : { "name" : "specialTaskList" },
  "input" : "arbitrary-string-that-is-meaningful-to-the-workflow",
  "executionStartToCloseTimeout" : "1800",
  "tagList" : [ "music purchase", "digital", "ricoh-the-dog" ],
  "taskStartToCloseTimeout" : "1800",
  "childPolicy" : "TERMINATE"
}
```

如果 `StartWorkflowExecution` 操作成功，Amazon SWF 將返回 `runId` 工作流程執行。工作流程執行的 `runId` 在特定區域內是唯一的。Save (儲存) `runId` 如果您稍後需要在呼叫中指定此工作流程執行，則請先 Amazon SWF。例如，如果您稍後需要將訊號傳送給工作流程執行，則會使用 `runId`。

```
{"runId": "9ba33198-4b18-4792-9c15-7181fb3a8852"}
```

## 設定任務優先順序

根據預設，任務清單上的任務根據它們的「到達時間」來交付：先排程的任務一般盡可能先執行。通過設置一個可選的任務優先順序，您可以優先考慮某些任務：Amazon SWF 會嘗試先交付優先順序較高的任務，再交付優先順序較高的任務，再交付優先順序較高的任務

### Note

先排定的任務一般會先執行，但不保證一定如此。

您可以為工作流程和活動都設定任務優先順序。工作流程的任務優先順序並不影響其排程的任何活動任務優先順序，也不影響其啟動的任何子工作流程。活動或工作流程的預設優先順序會在註冊期間 (由您或) 設定，且系統一律使用已註冊的任務優先順序，除非在排程活動或啟動工作流程執行時該順序被覆寫。

任務優先順序值的範圍介於 "-2147483648" 到 "2147483647"，數字愈大表示優先順序愈高。如果您未設定活動或工作流程的任務優先順序，則會為其指派優先順序零 ("0")。

## 主題

- [為工作流程設定任務優先順序](#)
- [為活動設定任務優先順序](#)
- [傳回任務優先順序資訊的動作](#)

## 為工作流程設定任務優先順序

當您註冊或啟動工作流程時，您可為它設定任務優先順序。除非在啟動工作流程執行時被覆寫，否則在註冊工作流程類型時設定的任務優先順序，會用為該類型任何工作流程執行的預設值。

要使用默認任務優先級註冊工作流類型，請將defaultTaskPriority選項時使用[RegisterWorkflowType](#)動作：

```
{
  "domain": "867530901",
  "name": "expeditedOrderWorkflow",
  "version": "1.0",
  "description": "Expedited customer orders workflow",
  "defaultTaskStartToCloseTimeout": "600",
  "defaultExecutionStartToCloseTimeout": "3600",
  "defaultTaskList": {"name": "mainTaskList"},
  "defaultTaskPriority": "10",
  "defaultChildPolicy": "TERMINATE"
}
```

當您使用 [StartWorkflowExecution](#) 啟動工作流程執行時，可以覆寫該工作流程類型已註冊的任務優先順序：

```
{
  "childPolicy": "TERMINATE",
  "domain": "867530901",
  "executionStartToCloseTimeout": "1800",
```

```

tagList": ["music purchase", "digital", "ricoh-the-dog"],
>taskList": {"name": "specialTaskList"},
>taskPriority": "-20",
>taskStartToCloseTimeout": "600",
>workflowId": "20110927-T-1",
>workflowType": {"name": "customerOrderWorkflow", "version": "1.0"},
}

```

啟動子工作流程或以新項目繼續執行工作流程時 (例如，以 [RespondDecisionTaskCompleted](#) 回應決策)，也可覆寫已註冊的任務優先順序。

若要設定子工作流程的任務優先順序，請在 `startChildWorkflowExecutionDecisionAttributes` 中提供值：

```

{
  "taskToken": "AAAAKgAAAAEAAAAAAAAAAAA...",
  "decisions": [
    {
      "decisionType": "StartChildWorkflowExecution",
      "startChildWorkflowExecutionDecisionAttributes": {
        "childPolicy": "TERMINATE",
        "control": "digital music",
        "executionStartToCloseTimeout": "900",
        "input": "201412-Smith-011x",
        "taskList": {"name": "specialTaskList"},
        "taskPriority": "5",
        "taskStartToCloseTimeout": "600",
        "workflowId": "verification-workflow",
        "workflowType": {
          "name": "MyChildWorkflow",
          "version": "1.0"
        }
      }
    }
  ]
}

```

以新項目繼續執行工作流程時，請在 `continueAsNewWorkflowExecutionDecisionAttributes` 中設定任務優先順序：

```
{
```

```

"taskToken": "AAAAKgAAAAEAAAAAAAAAAAA...",
"decisions": [
  {
    "decisionType": "ContinueAsNewWorkflowExecution",
    "continueAsNewWorkflowExecutionDecisionAttributes": {
      "childPolicy": "TERMINATE",
      "executionStartToCloseTimeout": "1800",
      "input": "5634-0056-4367-0923,12/12,437",
      "taskList": {"name": "specialTaskList"},
      "taskStartToCloseTimeout": "600",
      "taskPriority": "100",
      "workflowTypeVersion": "1.0"
    }
  }
]
}

```

## 為活動設定任務優先順序

您可在註冊或排程活動時，為它設定任務優先順序。除非在排程活動時被覆寫，否則在註冊活動類型時設定的任務優先順序，會用為活動執行的預設值。

要在註冊活動類型時設置任務優先級，請將defaultTaskPriority選項時使用[RegisterActivityType](#)動作：

```

{
  "defaultTaskHeartbeatTimeout": "120",
  "defaultTaskList": {"name": "mainTaskList"},
  "defaultTaskPriority": "10",
  "defaultTaskScheduleToCloseTimeout": "900",
  "defaultTaskScheduleToStartTimeout": "300",
  "defaultTaskStartToCloseTimeout": "600",
  "description": "Verify the customer credit card",
  "domain": "867530901",
  "name": "activityVerify",
  "version": "1.0"
}

```

要計劃具有任務優先級的任務，請使用taskPriority選項調度活動時使用[RespondDecisionTaskCompleted](#)動作：

```

{
  "taskToken": "AAAAKgAAAAEAAAAAAAAAAAA...",

```

```
"decisions": [
  {
    "decisionType": "ScheduleActivityTask",
    "scheduleActivityTaskDecisionAttributes": {
      "activityId": "verify-account",
      "activityType": {
        "name": "activityVerify",
        "version": "1.0"
      },
      "control": "digital music",
      "input": "abab-101",
      "taskList": {"name": "mainTaskList"},
      "taskPriority": "15"
    }
  }
]
```

## 傳回任務優先順序資訊的動作

您可以從下列 Amazon SWF 動作取得所設定的任務優先順序 (或所設定的預設任務優先順序) 的相關資訊：

- [DescribeActivityType](#) 傳回 defaultTaskPriority 中活動類型的 configuration 部分。
- [DescribeWorkflowExecution](#) 傳回 taskPriority 工作流程執行的 executionConfiguration 部分。
- [DescribeWorkflowType](#) 傳回 defaultTaskPriority 中工作流程類型的 configuration 部分。
- [GetWorkflowExecutionHistory](#) 和 [PollForDecisionTask](#) 會在回應的 activityTaskScheduledEventAttributes、decisionTaskScheduledEventAttributes、workflowExecutionStartedEventAttributes 區段內提供任務優先順序資訊。

## Amazon SWF 中處理錯誤

工作流程執行過程中可能發生數種不同類型的錯誤。

### 主題

- [驗證錯誤](#)
- [制定動作或決策時發生錯誤](#)
- [逾時](#)

- [使用者代碼所引發的錯誤](#)
- [與關閉工作流程執行相關的錯誤](#)

## 驗證錯誤

Amazon SWF 請求失敗時發生驗證錯誤，因為格式錯誤或包含無效數據。在此內容中，請求可以是 DescribeDomain 這類動作，或可能是 StartTimer 這類決策。如果請求是動作，Amazon SWF 會於回應中傳回錯誤代碼。請檢查此錯誤代碼，因其有可能針對請求中導致失敗的面相提供相關資訊。例如，與請求一起傳遞的一或多個引數可能無效。如需常見錯誤代碼列表，請前往 Amazon Simple Workflow Service API 參考。

如果失敗的請求是決策，則會在工作流程執行歷史記錄中列出適當的事件。例如，如果 StartTimer 決策失敗，您會在歷史記錄中看到 StartTimerFailed 事件。決策者應該在收到歷史記錄以回應 PollForDecisionTask 或 GetWorkflowExecutionHistory 時檢查這些事件。以下是在決策的格式不正確或包含無效資料時，可能發生的決策失敗事件清單。

## 制定動作或決策時發生錯誤

Amazon SWF 嘗試執行請求時，即使請求格式正確，也可能發生錯誤。在這些情況下，歷史記錄中的下列其中一個事件將指出發生錯誤。請查看事件的 reason 欄位，判定失敗原因。

- [CancelTimerFailed](#)
- [RequestCancelActivityTaskFailed](#)
- [RequestCancelExternalWorkflowExecutionFailed](#)
- [ScheduleActivityTaskFailed](#)
- [SignalExternalWorkflowExecutionFailed](#)
- [StartChildWorkflowExecutionFailed](#)
- [StartTimerFailed](#)

## 逾時

[決策者](#)、[活動工作者](#)和[工作流程執行](#)都是在逾時期間的限制條件下運作。在這類型的錯誤中，任務或子工作流程逾時。說明逾時的事件將會出現在歷史記錄中。例如，決策者應該重新排程任務或重新啟動子工作流程來處理此事件。如需逾時的詳細資訊，請參閱「[Amazon SWF 超時類型](#)」。

- [ActivityTaskTimedOut](#)

- [ChildWorkflowExecutionTimedOut](#)
- [DecisionTaskTimedOut](#)
- [WorkflowExecutionTimedOut](#)

## 使用者代碼所引發的錯誤

這類型的錯誤條件範例為活動任務失敗和子工作流程失敗。與超時錯誤相同，Amazon SWF 會將適當的事件添加到工作流程執行歷史記錄中。例如，決策者可能會重新排程任務或重新啟動子工作流程來處理此事件。

- [ActivityTaskFailed](#)
- [ChildWorkflowExecutionFailed](#)

## 與關閉工作流程執行相關的錯誤

決策者若嘗試關閉具有待定決策任務的工作流程，則也可能會看到下列事件。

- [FailWorkflowExecutionFailed](#)
- [CompleteWorkFlowExecutionFailed](#)
- [ContinueAsNewWorkflowExecutionFailed](#)
- [CancelWorkflowExecutionFailed](#)

有關上面列出的任何事件的詳細信息，請參閱[歷史記錄事件](#)在 Amazon SWF API 參考中處理。

# Amazon SWF 配額

Amazon SWF 會針對特定工作流程參數的大小設定配額，例如每個帳戶的網域數量以及工作流程執行歷程記錄的大小。這些配額旨在防止錯誤的工作流程消耗系統的所有資源，但不是硬性限制。如果您發現您的應用程式經常超過這些配額，您可以[要求提高服務配額](#)。

## 目錄

- [Amazon SWF 的一般帳戶配額](#)
- [工作流程執行配額](#)
- [工作執行配額](#)
- [Amazon SWF 節流配額](#)
  - [所有區域的節流配額](#)
  - [所有區域的決策配額](#)
  - [工作流程層級配額](#)
- [請求提高配額](#)

## Amazon SWF 的一般帳戶配額

- 最大註冊域名 — 100

此配額包括已註冊和已淘汰的網域。

- 工作流程和活動類型上限 — 每個網域各 10,000 個

此配額包括已註冊和已淘汰的類型。

- API 呼叫配額 — 除了罕見的尖峰之外，如果應用程式在很短的時間內進行大量 API 呼叫，則可能會限制應用程式。
- 最大請求大小 — 每個請求 1 MB

這是每個 Amazon SWF API 請求的總資料大小，包括請求標頭和所有其他相關聯的請求資料。

- Count API 的截斷回應 — 表示已達到內部配額，且回應不是完整計數。

在返回完整響應之前，某些查詢將在內部達到上述 1 MB 的配額。以下會傳回截斷的回應，不會傳回完整的計數。

- [CountClosedWorkflowExecutions](#)

- [CountOpenWorkflowExecutions](#)
- [CountPendingActivityTasks](#)
- [CountPendingDecisionTasks](#)

如果 `truncated` 回應設為 `true`，則其每個計數都會少於完整數量。此內部配額無法增加。

- 標籤數目上限 — 每個資源 50 個標籤。

嘗試新增超過 50 個標籤將會產生 400 錯誤：`TooManyTagsFault`。

## 工作流程執行配額

- 開啟工作流程執行次數上限 — 每個網域 100,000 次

此計數包含子工作流程執行。

- 最長工作流程執行時間 — 1 年。這是無法變更的硬配額。
- 最大工作流執行歷史記錄大小 — 25,000 個事件。這是無法變更的硬配額。

最佳實務是建構每個工作流程，讓它的歷史記錄不要成長超過 10,000 個事件。因為決策者必須擷取工作流程歷史記錄，所以較小的歷史記錄可讓決策者更快完成工作。如果使用「[流程架構](#)」，您可以使用 `ContinueAsNew` 繼續具有新歷程記錄的工作流程。

- 開啟子工作流程執行次數上限 — 每個工作流程執行 1,000
- 工作流程執行閒置時間配額 — 1 年 (受工作流程執行時間上限限制)

如果您工作流程的某個特定階段耗時過長，您可設定[工作流程逾時](#)來造成逾時事件。

- 工作流程保留時間配額 — 90 天

在這個時間之後，就不能再擷取或檢視工作流程歷史記錄。Amazon SWF 保留的已關閉工作流程執行次數沒有進一步限制。

如果您的使用案例要求您超出這些配額，您可以使用 Amazon SWF 提供的功能來繼續執行，並使用[子工作流程執行](#)來建構應用程式。如果您發現仍需要增加配額，請參閱[請求提高配額](#)。

## 工作執行配額

- 每個工作清單的最大輪詢者數量 — 每個工作清單 1,000

可同時輪詢特定任務清單的輪詢器上限為 1,000 個。如果超過 1,000 個，您就會收到 `LimitExceededException`。

#### Note

雖然最大值為 1,000，但您可能會在此配額之前遇到 `LimitExceededException` 錯誤。此錯誤並不意味著您的任務正在延遲。相反，這意味著您在任務列表上有最大的閒置輪詢器數量。Amazon SWF 設定此限制以節省用戶端和伺服器端的資源。設定限制可防止過多的輪詢者不必要地等待。您可以使用多個工作清單來散發輪詢，以減少 `LimitExceededException` 錯誤。

- 每秒排程的工作數上限 — 每個工作清單 2,000

您可以在特定任務列表上每秒安排最多 2,000 個任務。如果您超過 2,000，您的 `ScheduleActivityTask` 決定將失敗並顯示 `ACTIVITY_CREATION_RATE_EXCEEDED` 錯誤。

#### Note

雖然最大值為 2,000，但您可能會在此配額之前遇到 `ACTIVITY_CREATION_RATE_EXCEEDED` 錯誤。若要減少這些錯誤，請使用多個工作清單來分配負載。

- 工作執行時間上限：1 年 (受工作流程執行時間上限限制)

您可以配置 [活動超時](#)，以便在 [活動任務](#) 執行的特定階段花費太長時間時導致超時事件發生。

- SWF 將工作保留在佇列中的最長時間 — 1 年 (受工作流程執行時間配額的限制)

您可以在 [活動註冊期間](#) 配置默認活動超時，如果 [活動任務](#) 執行的特定階段花費太長時間，將導致發生超時事件。當您在決策者程式碼中排程活動任務時，您也可以覆寫預設的活動逾時。

- 開啟活動任務上限 — 每個工作流程執行 1,000 個。

此配額包括已排程的活動任務與 Worker 正在處理的活動任務。

- 最大開啟計時器 — 每個工作流程執行 1,000
- 最大輸入/結果資料大小 — 32,768 個字元

此配額會影響活動或工作流程執行結果資料、排程活動任務或工作流程執行時的輸入資料，以及隨 [工作流程執行訊號](#) 傳送的輸入。

- 決策任務回應中的最大決策 — 各不相同

由於 [API 要求大小](#) 上限為 1 MB 的配額，因此單次呼叫中傳回的決定數量 `RespondDecisionTaskCompleted` 將根據每個決策所使用的資料大小 (包括提供給排定活動任務或工作流程執行的任何輸入資料的大小) 來限制。

## Amazon SWF 節流配額

除了先前描述的服務配額之外，使用 [權杖儲存貯體](#) 配置，還會限制特定 Amazon SWF API 呼叫和決策事件以維持服務頻寬。如果您的要求速率持續超過此處列出的費率，您可以 [要求提高節流配額](#)。

所有區域的節流和決策配額都相同。

### 所有區域的節流配額

下列配額適用於個別帳戶層級。您也可以要求增加以下配額。如需執行此作業的詳細資訊，請參閱 [請求提高配額](#)。

API 名稱	儲存貯體大小	每秒重新填滿速率
<code>CountClosedWorkflowExecutions</code>	2000	6
<code>CountOpenWorkflowExecutions</code>	2000	6
<code>CountPendingActivityTasks</code>	200	6
<code>CountPendingDecisionTasks</code>	200	6
<code>DeleteActivityType</code>	200	6
<code>DeleteWorkflowType</code>	200	6
<code>DeprecateActivityType</code>	200	6
<code>DeprecateDomain</code>	100	6
<code>DeprecateWorkflowType</code>	200	6
<code>DescribeActivityType</code>	2000	6

API 名稱	儲存貯體大小	每秒重新填滿速率
DescribeDomain	200	6
DescribeWorkflowExecution	2000	6
DescribeWorkflowType	2000	6
GetWorkflowExecutionHistory	2000	60
ListActivityTypes	200	6
ListClosedWorkflowExecutions	200	6
ListDomains	100	6
ListOpenWorkflowExecutions	200	48
ListTagsForResource	50	30
ListWorkflowTypes	200	6
PollForActivityTask	2000	200
PollForDecisionTask	2000	200
RecordActivityTaskHeartbeat	2000	160
RegisterActivityType	200	60
RegisterDomain	100	6
RegisterWorkflowType	200	60
RequestCancelWorkflowExecution	2000	30
RespondActivityTaskCanceled	2000	200
RespondActivityTaskCompleted	2000	200
RespondActivityTaskFailed	2000	200

API 名稱	儲存貯體大小	每秒重新填滿速率
RespondDecisionTaskCompleted	2000	200
SignalWorkflowExecution	2000	30
StartWorkflowExecution	2000	200
TagResource	50	30
TerminateWorkflowExecution	2000	60
UndeprecateActivityType	200	6
UndeprecateDomain	100	6
UndeprecateWorkflowType	200	6
UntagResource	50	30

## 所有區域的決策配額

下列配額適用於個別帳戶層級。您也可以要求增加以下配額。如需執行此作業的詳細資訊，請參閱[請求提高配額](#)。

API 名稱	儲存貯體大小	每秒重新填滿速率
RequestCancelExternalWorkflowExecution	1200	120
ScheduleActivityTask	500	200
SignalExternalWorkflowExecution	1200	120
StartChildWorkflowExecution	500	12
StartTimer	2000	200

## 工作流程層級配額

下列配額適用於工作流程層級，無法增加。

API 名稱	儲存貯體大小	每秒重新填滿速率
GetWorkflowExecutionHistory	400	200
SignalWorkflowExecution	1000	1000
RecordActivityTaskHeartbeat	1000	1000
RequestCancelWorkflowExecution	200	200

## 請求提高配額

使用中的「Support 中心」頁面 AWS Management Console 來要求[所有區域的節流配額](#)和的配額增加[所有區域的決策配額](#)。如需詳細資訊，請參閱[如果要求提高配額的 AWS 一般參考](#)。

# Amazon 簡單的工作流服務

本章提供使用 Amazon SWF 開發工作流程時非常有用的其他資源和參考資訊。

## 主題

- [Amazon SWF 超時類型](#)
- [Amazon 簡單工作流程服務端點](#)
- [Amazon 簡易工作流程服務的其他文件](#)
- [Amazon 簡單工作流程服務的 Web 資源](#)
- [Ruby 流程的移轉選項](#)

## Amazon SWF 超時類型

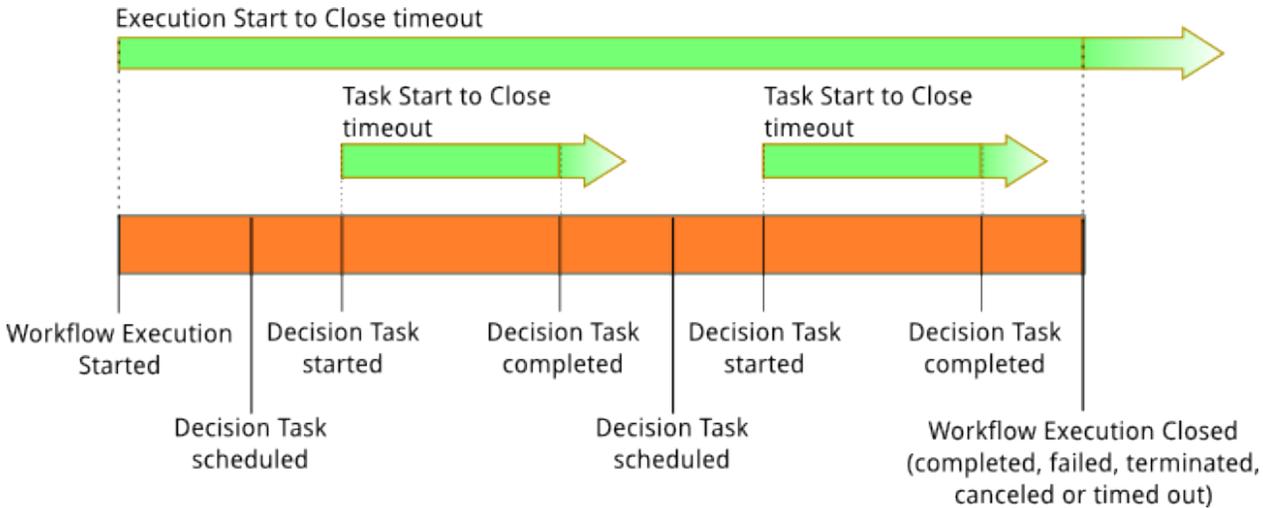
為確保工作流程執行正確執行，Amazon SWF 可讓您設定不同類型的逾時。有些逾時會指定工作流程總共可以執行多久。有些逾時則指定活動任務要多久才能指派給工作者，以及從排程開始要多久才可以完成。Amazon SWF API 中的所有逾時時間都會以秒為單位指定。Amazon SWF 也支援該字串NONE做為逾時值，表示沒有逾時。

對於與決策任務和活動任務相關的逾時，Amazon SWF 會在工作流程執行歷程記錄中新增事件。事件的屬性提供有關發生的逾時類型，以及哪些決策任務或活動任務受到影響的資訊。Amazon SWF 也會排程決策任務。當決策程序接收到新的決策任務時，它會看到歷史記錄中的超時事件，並通過調用操作採取適當的[RespondDecisionTaskCompleted](#)操作。

任務從排程起到結束的這段時間，視為開啟。因此，任務在工作者處理它時會回報為開啟。當工作者回報任務為[完成](#)、[取消](#)或[失敗](#)時，任務即結束。Amazon SWF 也可能因為逾時而關閉任務。

## 工作流程和決策任務的逾時

下圖示範工作流程和決策逾時與工作流程生命週期的關係：

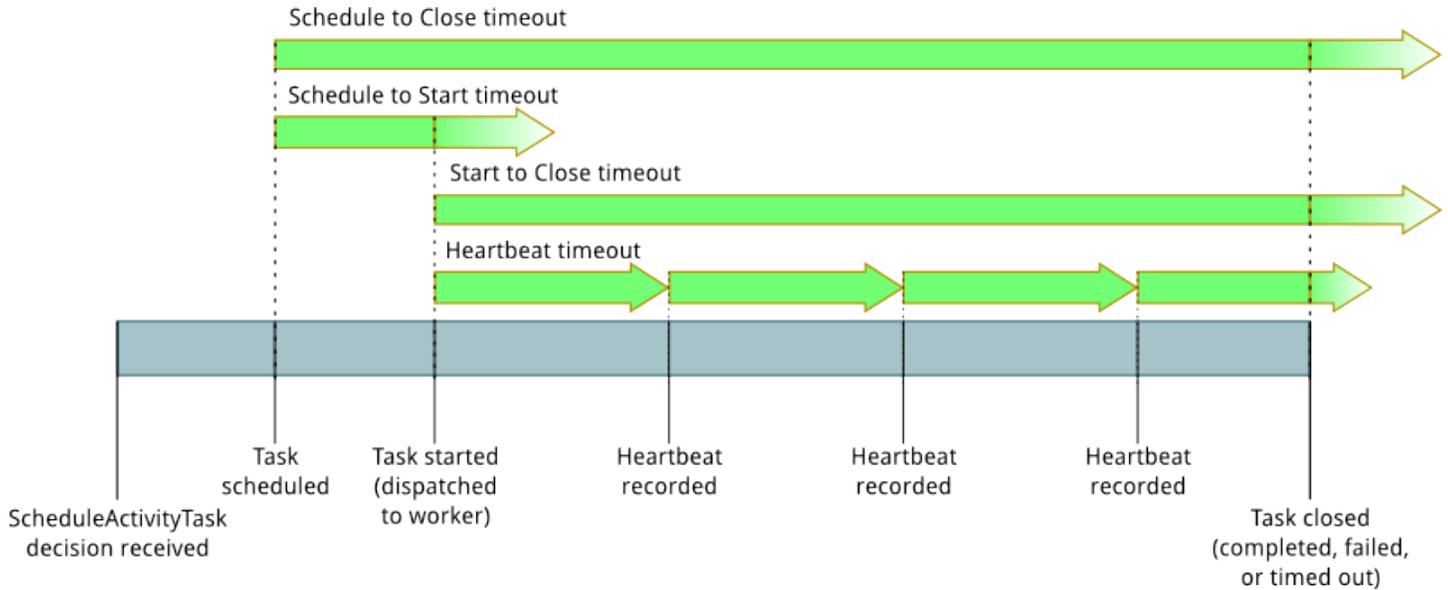


有兩種逾時類型與工作流程和決策任務相關：

- 工作流程開始至關閉 (**timeoutType: START\_TO\_CLOSE**) — 此逾時指定完成工作流程執行所需的時間上限。它在工作流程註冊期間設為預設值，但可在工作流程開始後用不同的值覆寫。如果超過此逾時，Amazon SWF 會關閉工作流程執行，並將類型 [WorkflowExecutionTimedOut](#) 為的事件新增至工作流程執行歷程記錄。除 `timeoutType` 之外，事件屬性還會指定對此工作流程執行生效的 `childPolicy`。子政策指定如果父工作流程執行逾時或終止，子工作流程執行的處理方式。例如，如果 `childPolicy` 設為 `TERMINATE`，則子工作流程執行會予以終止。一旦工作流程執行逾時，除可見度呼叫外，您無法對它採取任何動作。
- 決策任務開始至關閉 (**timeoutType: START\_TO\_CLOSE**) — 此逾時指定對應決策程式完成決策任務所需的時間上限。它是在工作流程型註冊期間設定。如果超過此逾時，任務會在工作流程執行歷程記錄中標示為逾時，而 Amazon SWF 會在工作流程歷程記錄中新增類型 [DecisionTaskTimedOut](#) 事件。這些事件屬性會包含當此決策任務排程 (`scheduledEventId`) 及啟動時 (`startedEventId`) 所對應的事件 ID。除了新增事件之外，Amazon SWF 還會排程新的決策任務，以提醒決策程式此決策任務逾時。此逾時發生後，使用 `RespondDecisionTaskCompleted` 完成逾時決策任務的嘗試會失敗。

## 活動任務的逾時

下圖示範逾時與活動任務生命週期的關係：



有四種逾時類型與活動任務相關：

- 活動任務開始至關閉 (**timeoutType: START\_TO\_CLOSE**) — 此逾時指定活動 Worker 收到任務後處理任務所需的時間上限。嘗試使用 [RespondActivityTaskCanceled](#)、[RespondActivityTaskCompleted](#) 和關閉逾時的活動工作 [RespondActivityTaskFailed](#) 將會失敗。
- 活動工作活動訊號 (**timeoutType: HEARTBEAT**) — 此逾時指定 `RecordActivityTaskHeartbeat` 在透過動作提供進度之前，可執行的時間上限。
- 啟動的活動任務排程 (**timeoutType: SCHEDULE\_TO\_START**) — 此逾時指定如果沒有工作者可用來執行任務，Amazon SWF 會在逾時活動任務之前等待多久。一旦逾時，過期的任務就會不指派給其他工作者。
- 要關閉的活動任務排程 (**timeoutType: SCHEDULE\_TO\_CLOSE**) — 此逾時指定任務從排定到完成的時間可能需要多長時間。最佳作法是，此值不應大於工作逾時和作業 `schedule-to-start` 逾時的總和。 `start-to-close`

#### **Note**

每種逾時類型都有預設值，一般設為 NONE (無限)。但是任何活動執行的時間上限都限制在一年。

您是在活動類型註冊期間設定這些預設值，但在 [排程](#) 活動任務時還可使用新值加以覆寫。當其中一項逾時發生時，Amazon SWF 會在工作流程歷程記錄中新 [ActivityTaskTimedOut](#) 增類型事件。此事

件的 `timeoutType` 值屬性會指定這些逾時中的哪一個會發生。這些逾時每一個的 `timeoutType` 值都會出現在括號中。這些事件屬性也會包含當活動任務排程 (`scheduledEventId`) 及啟動時 (`startedEventId`) 所對應的事件 ID。除了新增事件之外，Amazon SWF 還會排程新的決策任務，以警示決策程式發生逾時。

## Amazon 簡單工作流程服務端點

中提供了目前 [Amazon SWF 區域和端點](#) 的清單 Amazon Web Services 一般參考，以及其他服務的端點。

Amazon SWF 網域及所有相關工作流程和活動必須存在於同一個區域內，才能彼此通訊。此外，一個區域中註冊的任何網域、工作流程和活動，不會存在於其他區域中。例如，如果您在 `us-east-1` 和 `us-west-2` 中建立名為「MySampleDomain」的網域，它們會以個別的網域形式存在：與您的網域相關聯的任何工作流程、工作清單、活動或資料都不會跨區域共用。

如果您在工作流程中使用其他 AWS 資源 (例如 Amazon EC2 執行個體)，這些資源也必須與 Amazon SWF 資源位於相同的區域中。唯一的例外是跨區域的服務，例如 Amazon S3 和 IAM。您可以從存在於任何支援區域中的工作流程存取這些服務。

## Amazon 簡易工作流程服務的其他文件

除了本《開發人員指南》之外，下列文件也十分有幫助。

### Amazon 簡單工作流服務 API 參考

[Amazon 簡易工作流程服務 API 參考](#) 提供有關 Amazon SWF HTTP API 的詳細資訊，包括動作、請求和回應結構以及錯誤代碼。

### AWS Flow Framework 文件

這 [AWS Flow Framework](#) 是一種程式設計架構，可簡化使用 Amazon SWF 來管理其工作流程和活動的分散式非同步應用程式的實作程序，因此您可以專注於實作工作流程邏輯。

每種設計都 AWS Flow Framework 是以慣用設計的語言運作，因此您可以自然地使用自己選擇的語言來實作工作流程，並享有 Amazon SWF 的所有優點。

有一個適用於 Java 的 AWS 流程框架。[Java 開發人員指南](#) 提供有關如何取得、設定和使用 AWS Flow Framework for Java 的相關資訊。AWS Flow Framework

## AWS SDK 文件集

AWS 軟體開發套件 (SDK) 可讓您以多種不同的程式設計語言存取 Amazon SWF。這些開發套件會緊密遵循 HTTP API，但也為某些 Amazon SWF 功能提供特定語言的程式設計界面。您可以瀏覽下列連結來深入了解每個開發套件的相關資訊。

### Note

此處僅列出在撰寫本文時支援 Amazon SWF 的開發套件。如需可用 AWS 開發套件的完整清單，請造訪 [Amazon Web Services 工具](#) 頁面。

### Java

提 AWS SDK for Java 供 AWS 基礎架構服務的 Java API。

若要檢視現已提供的文件，請參閱 [AWS SDK for Java 文件](#) 頁面。您也可以透過以下連結直接前往 SDK 參考資料中的 Amazon SWF 區段：

- [Class: AmazonSimpleWorkflowClient](#)
- [Class: AmazonSimpleWorkflowAsyncClient](#)
- [Interface: AmazonSimpleWorkflow](#)
- [Interface: AmazonSimpleWorkflowAsync](#)

### JavaScript

AWS SDK for JavaScript 允許開發人員構建庫或應用程序，使用簡單的和 easy-to-use API 可在瀏覽器或 AWS 服務器上的 Node.js 應用程序內部使用服務。

若要檢視現已提供的文件，請參閱 [AWS SDK for JavaScript 文件](#) 頁面。您也可以透過以下連結直接前往 SDK 參考資料中的 Amazon SWF 區段：

- [Class: AWS.SimpleWorkflow](#)

### .NET

AWS SDK for .NET 是單一、可下載的套件，其中包含 Visual Studio 專案範本、AWS .NET 程式庫、C# 程式碼範例和文件。這 AWS SDK for .NET 使得 Windows 開發人員可以更輕鬆地為 Amazon SWF 和其他服務構建 .NET 應用程序。

若要檢視現已提供的文件，請參閱 [AWS SDK for .NET 文件](#) 頁面。您也可以透過以下連結直接前往 SDK 參考資料中的 Amazon SWF 區段：

- [Namespace: Amazon.SimpleWorkflow](#)
- [Namespace: Amazon.SimpleWorkflow.Model](#)

## PHP

該 AWS SDK for PHP 提供了一個 PHP 編程接口 Amazon SWF。

若要檢視現已提供的文件，請參閱[AWS SDK for PHP 文件](#)頁面。您也可以透過以下連結直接前往 SDK 參考資料中的 Amazon SWF 區段：

- [Class: SwfClient](#)

## Python

該 AWS SDK for Python (Boto) 提供了一個 Python 編程接口 Amazon SWF。

要查看可用的文檔，請參閱[boto : Amazon Web Services 的 Python 界面](#)頁面。您也可以透過以下連結直接前往文件中的 Amazon SWF 區段：

- [Amazon SWF 教程](#)
- [Amazon SWF 參考](#)

## Ruby

該 AWS SDK for Ruby 提供了一個紅寶石編程接口 Amazon SWF。

若要檢視現已提供的文件，請參閱[AWS SDK for Ruby 文件](#)頁面。您也可以透過以下連結直接前往 SDK 參考資料中的 Amazon SWF 區段：

- [類別：AWS::SimpleWorkflow](#)

## AWS CLI 文件

AWS Command Line Interface (AWS CLI) 是管理 AWS 服務的統一工具。只要下載和設定一個工具，您就可以從命令列控制多個 AWS 服務，並透過指令碼將它們自動化。

如需有關的詳細資訊 AWS CLI，請參閱[AWS Command Line Interface](#)頁面。

如需 Amazon SWF 可用命令的概觀，請參閱AWS CLI 命令參考中的 [swf](#)。

## Amazon 簡單工作流程服務的 Web 資源

您可以使用許多 Web 資源進一步了解 Amazon SWF，或取得有關使用服務和開發工作流程的協助。

## Amazon SWF 論壇

Amazon SWF 論壇為您提供了一個與 Amazon SWF 開發人員和 Amazon SWF 開發團隊成員進行交流的地方，以提出問題並獲得答案。

您可以訪問論壇：[論壇：Amazon 簡單工作流服務](#)。

## Amazon SWF 常見問

Amazon SWF 常見問答集提供有關 Amazon SWF 的常見問題解答，包括常見使用案例概觀、Amazon SWF 與其他服務之間的差異等。

您可以在此處存取常見問題集：[Amazon SWF 常見問題集](#)。

## Amazon SWF 視頻

[Amazon Web Services](#) 渠道上 YouTube 提供了所有的亞馬遜網絡服務的視頻培訓，包括 Amazon SWF。有關亞馬遜 SWF 相關視頻的完整列表，請使用以下查詢：[Amazon Web Services 中的簡單工作流程](#)

## Ruby 流程的移轉選項

Ruby 不再處 AWS Flow Framework 於積極開發狀態。雖然現有的程式碼會無限期地繼續運作，但不會有新功能或版本。本主題將介紹繼續使用 Amazon SWF 的用法和移轉選項，以及如何遷移至 Step Functions 的相關資訊。

選項	描述
<a href="#">繼續使用 Ruby Flow Framework</a>	目前，Ruby Flow Framework 會繼續運作。如果您不採取任何動作，您的程式碼會依現狀繼續運作。計劃在不久 AWS Flow Framework 的 future 從 Ruby 遷移。
<a href="#">遷移至 Java Flow Framework</a>	Java Flow Framework 會繼續積極開發，並繼續收到新功能和更新。
<a href="#">遷移至 Step Functions</a>	Step Functions 可讓您使用由狀態機器控制的視覺化工作流程，來協調分散式應用程式的元件。
<a href="#">直接使用 SWF API，而不使用 Flow Framework</a>	您可以繼續在 Ruby 中工作，並直接使用 SWF API 而不是 Ruby Flow Framework。

Ruby 或 Java Flow Framework 提供的優點是可讓您專注於工作流程邏輯。框架會處理大部分的通訊和協調詳細資訊，並將一些複雜性抽象化。您可以移轉至 Java 流程架構，繼續擁有相同層級的抽象概念，或直接與 Amazon SWF 開發套件進行互動。

## 繼續使用 Ruby Flow Framework

在短期內，AWS Flow Framework 對於 Ruby 將繼續像現在一樣運作。如果您有以 Ruby 撰寫的工作流程，這些工作流程將會繼續運作。AWS Flow Framework 如果沒有更新、支援或安全性修正，最好有一個堅定的計劃，在不久的 future 從 Ruby 遷移。

## 遷移至 Java Flow Framework

對 AWS Flow Framework 於 Java 將保持積極開發。從概念上講，AWS Flow Framework for Java 類似 AWS Flow Framework 於 for Ruby：您仍然可以專注於工作流程邏輯，並且框架將有助於管理決策程序邏輯，並使 Amazon SWF 的其他方面更易於管理。

- [AWS Flow Framework 對於爪哇](#)
- [AWS Flow Framework 應用程式介面參考](#)

## 遷移至 Step Functions

AWS Step Functions 提供類似於 Amazon SWF 的服務，但工作流程邏輯由狀態機器控制。Step Functions 可讓您使用視覺化工作流程來協調分散式應用程式和微服務的元件。您可以從個別元件建立應用程式，這些元件會各自執行不同的功能或任務，讓您快速擴展及變更應用程式。Step Functions 提供了一種可靠的方法來協調組件並逐步執行應用程式的功能。圖形化主控台可讓您將應用程式的元件視覺化成一連串的步驟。它會自動觸發和追蹤每個步驟，並在發生錯誤時重試，因此您的應用程式每次都能按預期順序執行。Step Functions 記錄每個步驟的狀態，因此當事情出錯時，您可以快速診斷和調試問題。

在 Step Functions 中，您可以使用使用 [Amazon 州語言](#) 定義的宣告式 JSON 撰寫的狀態機器來管理任務的協調。通過使用狀態機，您不必編寫和維護決策程序來控制應用程式邏輯。Step Functions 提供直覺式、生產力且敏捷的方法，使用視覺化工作流程協調應用程式元件。您應該考慮使 AWS Step Functions 用所有新的應用程式，而 Step Functions 為您目前在 for Ruby 中實作的工作流程提供了一個絕佳的 AWS Flow Framework 平台。

為了協助您將任務遷移到 Step Functions，同時繼續利用您的 Ruby 語言技能，Step Functions 提供了一個示例 Ruby 活動工作者。此範例使用實作活動 Worker 的最佳作法，並可作為範本來將任務邏輯移轉至 Step Functions。如需詳細資訊，請參閱開發 [人 AWS Step Functions 員指南中的 Ruby 中的範例活動工作者](#) 主題。

**Note**

對於許多客戶而言，從 Ruby 遷移到 Step Functions 是最佳選擇。AWS Flow Framework 但是，如果您要求該訊號介入您的程序，或者您需要啟動將結果傳回給父系的子程序，請考慮直接使用 Amazon SWF API，或移轉至 AWS Flow Framework for Java。

如需詳細資訊 AWS Step Functions，請參閱：

- [AWS Step Functions 開發人員指南](#)
- [AWS Step Functions API 參考](#)
- [AWS Step Functions 指令行參考](#)

## 直接使用 Amazon SWF API

雖然紅寶石可以管理 Amazon SWF 的一些複雜性，但您也可以直接使用 Amazon SWF API。AWS Flow Framework 直接使用 API 可讓您建立能夠完全控制任務實作與協調的工作流程，而無需擔心基本的複雜性，例如追蹤其進度及維護其狀態。

- [Amazon 簡單工作流程服務開發人員](#)
- [Amazon 簡單工作流程服務 API 參考](#)

# Amazon Simple Work開發人員指南

下表說明自上次發行 Amazon Simple Work開發人員指南》文件的重要變更。

- API 版本：2012-01-25
- 最近的文件更新時間：2021 年 5 月

變更	描述	變更日期
僅文件更新	Amazon SWF 現在包含一個關於AWS使用者通知的區段，AWS 服務該區段可做為您在中AWS通知的中央位置 AWS Management Console。如需詳細資訊，請參閱 <a href="#">使用 AWS 使用者通知用 Amazon 簡單工作流服務</a> 。	2023 年 5 月 4 日
更新	Amazon SWF 現在提供全新的主控台體驗來管理 SWF 工作流程及其執行相關動作。如需詳細資訊，請參閱 <a href="#">Amazon SWF 主控台教學課程</a> 。	2022 年 9 月 12 日
更新	已更新要包含的 <a href="#">工作執行配額</a> 區段Maximum tasks scheduled per second，以及包含 <a href="#">使用非 ASCII 資源名稱</a> 相關資訊的 <a href="#">Amazon SWF 指標 CloudWatch</a> CloudWatch 頁面。	2021 年 5 月 12 日
新功能	Amazon Simple Workflow Service 現已支援 Amazon Simple EventBridge 的 Work。如需詳細資訊，請參閱： <ul style="list-style-type: none"> <li>• <a href="#">Amazon SWF EventBridge</a></li> <li>• 《EventBridge 使用者指南》<a href="https://docs.aws.amazon.com/eventbridge/latest/userguide/">https://docs.aws.amazon.com/eventbridge/latest/userguide/</a></li> </ul>	2020 年 12 月 18 日
新功能	Amazon 簡單工作流服務支援使用標籤的 IAM 許可。如需詳細資訊，請參閱下列內容。 <ul style="list-style-type: none"> <li>• <a href="#">標籤</a> <ul style="list-style-type: none"> <li>• <a href="#">管理標籤</a></li> <li>• <a href="#">標記工作流程執行</a></li> </ul> </li> </ul>	2019 年 6 月 20 日

變更	描述	變更日期
	<ul style="list-style-type: none"> <li>• <a href="#">使用標籤控制網域的存取</a></li> <li>• <a href="#">TagResource</a></li> <li>• <a href="#">UntagResource</a></li> <li>• <a href="#">ListTagsForResource</a></li> <li>• <a href="#">RegisterDomain</a></li> </ul>	
新功能	Amazon Simple Workflow Service 現已在歐洲 (斯德哥爾摩) 區域提供。	2018 年 12 月 12 日
更新	改進了有關CloudTrail集成的 Amazon 簡單工作流程服務主題。請參閱 <a href="#">使用記錄 API 呼叫 AWS CloudTrail</a> 。	2018 年 8 月 7 日
更新	新增 CloudWatch 的新 PendingTasks 指標資訊。如需詳細資訊，請參閱 <a href="#">Amazon SWF 指標</a> 。	2018 年 6 月 18 日
更新	改善程式碼範本中的語法醒目提示。	2018 年 3 月 29 日
更新	新增描述 Ruby Flow 使用者移出該平台之選項的主題。如需詳細資訊，請參閱 <a href="#">Ruby 流程的移轉選項</a> 。	2018 年 3 月 9 日
更新	改善進階概念主題的導覽。請參閱 <a href="#">Amazon SWF 中的高級概念</a> 。	2018 年 2 月 19 日
更新	透過新增有效的統計資訊，改善CloudWatch量度文件。請參閱 <a href="#">Amazon SWF 指標 CloudWatch</a> 。	2017 年 12 月 4 日
更新	變更 TOC 以改善文件結構。新增「 <a href="#">API 和決策事件指標</a> 」的相關資訊。	2017 年 11 月 9 日
更新	更新「 <a href="#">Amazon SWF 配額</a> 」一節，以包括所有區域的調節限制。	2017 年 10 月 18 日
更新	將「 <a href="#">Amazon SWF 和 Amazon SNS 的訂閱工作流程教程</a> 」中的 task_list 變更為 workflowId ，以免與 activity_list 混淆。	2017 年 7 月 25 日

變更	描述	變更日期
更新	清除整本指南的程式碼範例。	2017 年 6 月 5 日
更新	簡化並改善本指南的組織和內容。	2017 年 5 月 19 日
更新	更新和連結修正。	2017 年 5 月 16 日
更新	更新和連結修正。	2016 年 10 月 1 日
Lambda 任務支援	除了工作流程中的傳統活動任務外，您還可以指定 Lambda 任務。如需詳細資訊，請參閱 <a href="#">AWS Lambda 任務</a> 。	2015 年 7 月 21 日
設定任務優先順序的支援	Amazon SWF 現在支援在任務清單上設定任務的優先順序，並會嘗試在優先順序較低的任務之前交付優先順序較高的任務。「 <a href="#">設定任務優先順序</a> 」中提供如何設定工作流程和活動之任務優先順序的相關資訊。	2014 年 12 月 17 日
更新	已新增一個新主題，說明如何使用以下方式記錄 Amazon SWF API 呼叫 CloudTrail： <a href="#">使用記錄 API 呼叫 AWS CloudTrail</a> 。	2014 年 5 月 8 日
更新	已新增兩個與 Amazon SWF CloudWatch 指標相關的新主題： <a href="#">Amazon SWF 指標 CloudWatch</a> ，其中提供支援指標的清單和說明 <a href="#">使用 CloudWatch 的 Amazon SWF 指標</a> <a href="#">AWS Management Console</a> ，並提供如何使用 AWS Management Console。	2014 年 4 月 28 日
更新	新增章節：「 <a href="#">Amazon 簡單的工作流服務</a> 」。本節提供一些服務參考資訊，並為 Amazon SWF 開發人員提供其他文件、範例、程式碼和其他 Web 資源的相關資訊。	2014 年 3 月 19 日
更新	新增工作流程教學。請參閱 <a href="#">Amazon SWF 和 Amazon SNS 的訂閱工作流程教程</a> 。	2013 年 10 月 25 日

變更	描述	變更日期
更新	新增 <a href="#">AWS CLI 資訊和範例</a> 。	2013 年 8 月 26 日
更新	更新和修正。	2013 年 8 月 1 日
更新	已更新文件，說明如何使用 IAM 進行存取控制。	2013 年 2 月 22 日
初始版本	這是 Amazon Simple Work開發人員指南》第一個發行版本。	2012 年 10 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。