



開發人員指南

Amazon Simple Workflow Service



API 版本 2012-01-25

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

Amazon Simple Workflow Service: 開發人員指南

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

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

Table of Contents

什麼是 Amazon SWF ?	1
工作流程元件	2
工作流程元件	2
執行您的工作流程	3
設定開發環境	4
使用 AWS SDKs 開發	4
考慮 AWS Flow Framework	4
開始使用	6
關於工作流程	7
先決條件	7
教學步驟	8
第 1 部分：搭配適用於 Ruby 的 SDK 使用 Amazon SWF	8
包含 適用於 Ruby 的 AWS SDK	8
設定 AWS 工作階段	8
註冊 Amazon SWF 網域	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	22
定義 SubscribeTopicActivity	24
定義 WaitForConfirmationActivity	27
定義 SendResultActivity	29
後續步驟	31
第 4 部分：實作活動任務輪詢器	31
執行工作流程	34
接下來做些什麼？	37
在 主控台中工作	38

註冊網域	38
註冊工作流程類型	38
註冊活動類型	39
啟動工作流程	40
使用主控台啟動工作流程執行	40
管理工作流程執行	40
基本概念	43
建立工作流程	44
為工作流程和其活動建模	45
執行中的工作流程	45
工作流程歷史記錄	46
物件識別符	50
網域	51
動作者	51
什麼是 Amazon SWF 中的演員？	52
工作流程啟動者	53
決策者	53
活動工作者	54
動作者之間的資料交換	54
任務	54
任務清單	56
決策任務清單	56
活動任務清單	56
任務路由	57
工作流程執行關閉	57
工作流程執行生命週期	58
工作流程執行生命週期	58
輪詢任務	63
進階概念	64
版本控制	64
訊號	65
子工作流程	66
標記	68
標籤	69
管理標籤	69
標記工作流程執行	69

使用標籤控制對網域的存取	71
獨家選擇	71
計時器	74
取消活動任務	75
安全	78
資料保護	78
加密	79
身分和存取權管理	79
目標對象	80
使用身分驗證	81
使用政策管理存取權	82
存取控制	83
政策動作	83
政策資源	84
政策條件索引鍵	85
ACL	85
ABAC	85
臨時憑證	86
主體許可	86
服務角色	86
服務連結角色	86
身分型政策	87
資源型政策	87
Amazon Simple Workflow Service 如何與 IAM 搭配使用	87
身分型政策範例	88
基本原則	90
Amazon SWF IAM 政策	91
API 摘要	97
標籤類型政策	105
Amazon VPC 端點	105
疑難排解	107
記錄和監控	108
適用於 CloudWatch 的 Amazon SWF 指標	109
檢視 Amazon SWF 指標	118
錄製至 CloudTrail	122
Amazon SWF 的 EventBridge	129

AWS 使用者通知 搭配 Amazon SWF 使用	137
合規驗證	137
恢復能力	138
基礎設施安全性	138
組態與漏洞分析	139
使用 AWS CLI	140
使用 API	142
提出 HTTP 請求	142
HTTP 標頭內容	142
HTTP 內文內容	144
範例 JSON 請求和回應	145
計算 HMAC-SHA 簽章	146
Amazon SWF 動作的清單	148
與活動相關的動作	148
與決策者相關的動作	149
與工作流程執行相關的動作	149
與管理相關的動作	149
可見度動作	150
註冊網域	151
另請參閱	152
設定逾時值	152
逾時值的配額	152
工作流程執行和決策任務逾時	152
活動任務逾時	153
另請參閱	154
註冊工作流程類型	154
另請參閱	154
註冊活動類型	154
另請參閱	155
Lambda 任務	155
關於 AWS Lambda	155
使用 Lambda 任務的優點和限制	156
在工作流程中使用 Lambda 任務	156
開發活動工作者	160
輪詢活動任務	161
執行活動任務	162

報告活動任務活動訊號	162
完成或讓活動任務失敗	163
啟動活動工作者	164
開發決策者	164
定義協調性邏輯	165
輪詢決策任務	166
套用協調性邏輯	168
回應決策	168
關閉工作流程執行	169
啟動決策者	170
啟動工作流程	171
設定任務優先順序	172
為工作流程設定任務優先順序	173
為活動設定任務優先順序	175
傳回任務優先順序資訊的動作	176
處理錯誤	176
驗證錯誤	176
制定動作或決策時發生錯誤	177
逾時	177
使用者代碼所引發的錯誤	177
與關閉工作流程執行相關的錯誤	178
配額	179
Amazon SWF 的一般帳戶配額	179
工作流程執行的配額	180
任務執行的配額	180
Amazon SWF 限流配額	182
調節所有區域的配額	182
所有區域的決策配額	184
工作流程層級配額	184
請求提高配額	185
其他資源	186
逾時類型	186
工作流程和決策任務的逾時	186
活動任務的逾時	187
端點	189
其他 文件	189

Amazon Simple Workflow Service API 參考	189
AWS Flow Framework 文件	189
AWS SDK 文件	190
AWS CLI 文件	191
Web 資源	191
Amazon SWF 論壇	192
Amazon SWF 常見問答集	192
Amazon SWF 影片	192
Ruby Flow 選擇	192
繼續使用 Ruby Flow Framework	193
遷移至 Java Flow Framework	193
遷移至 Step Functions	193
直接使用 Amazon SWF API	194
文件歷史紀錄	195
.....	cxcix

什麼是 Amazon Simple Workflow Service ？

透過 Amazon Simple Workflow Service (Amazon SWF) ，您可以建置、執行和擴展具有平行或循序步驟的背景任務。您可以跨分散式元件協調工作，並追蹤任務的狀態。

在 Amazon SWF 中，任務代表由應用程式元件執行的邏輯工作單位。跨協調任務包括管理應用程式流程中的任務間相依性、排程和並行。使用 Amazon SWF ，您可以控制和協調任務，而無需擔心基礎複雜性，例如追蹤進度和維護任務狀態。

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) 或您自己的內部部署。您可以建立長時間執行的任務，或者可能失敗、逾時或需要重新啟動的任務，或者可能以不同的輸送量和延遲完成的任務。Amazon SWF 會存放任務，並在任務準備就緒時將其指派給工作者、追蹤進度和維護狀態，包括任務完成的詳細資訊。

為了協調任務，您撰寫的程式會從 Amazon SWF 取得最新的任務狀態，並使用該狀態來啟動後續任務。Amazon SWF 會長期維護應用程式的執行狀態，因此您的應用程式可應對個別元件故障。使用 Amazon SWF ，您可以獨立建置、部署、擴展和修改應用程式元件。

其他 AWS 工作流程服務

對於大多數使用案例，我們建議您 AWS Step Functions 考慮工作流程和協同運作需求。透過 Step Functions ，您可以建立工作流程，也稱為狀態機器，以建置分散式應用程式、自動化程序、協調微服務，以及建立資料和機器學習管道。在 VS Code 中的 Step Functions 主控台或 AWS 工具組中，您可以使用圖形工作流程 Studio 來視覺化、編輯、測試和偵錯應用程式的工作流程。

如需詳細資訊，請參閱 [AWS Step Functions 開發人員指南](#)。

使用 Amazon SWF 開發工作流程元件

開發分散式應用程式需要協調許多元件，並處理遠端通訊固有的延遲和不可靠性。

透過 Amazon Simple Workflow Service (Amazon SWF)，您可以提供程式設計模型和基礎設施來協調分散式元件，並以可靠的方式維護其執行狀態，藉此開發非同步和分散式應用程式。透過依賴 Amazon SWF，您可以自由地專注於建立應用程式的各個方面來區分它。

工作流程的元件

[工作流程的元件](#) 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，工作流程可能會收到客戶訂單，並採取履行訂單所需的任何動作。

每個工作流程都會在稱為網域的資源中執行，以控制工作流程的範圍。帳戶 AWS 可以有多個網域，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，您可以定義每個必要的活動。然後，您可以將每個活動註冊到 Amazon SWF 作為活動類型。您將提供名稱、版本和逾時值。例如，客戶可能預期訂單會在 24 小時內出貨。

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。如果客戶購買多個項目，則此活動必須執行多次。Amazon SWF 具有活動任務的概念，代表一個活動叫用。在我們的範例中，每個項目的處理會以單一活動任務表示。

活動工作者是接收活動任務、執行它們並提供結果的程式。任務實際上可能由人員執行。例如，統計分析師可能會收到資料集、分析資料，然後傳回其分析。

活動任務和執行它們的活動工作者可以同步或非同步執行。工作者可以在一個位置執行或分散在多部電腦，可能位於不同的地理區域。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伺服器上執行，而另一個活動工作者可能在北美洲的行動裝置上執行。

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、向活動工作者提供輸入、處理工作流程進行時抵達的事件，並在達到目標後結束（或關閉）工作流程。

Amazon SWF 服務的角色是在決策者、活動工作者和其他相關實體之間交換資料的可靠中樞，例如管理工作流程的人員。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耐久的方式存放狀態。

決策者透過從 Amazon SWF 接收決策任務，並以決策回應 Amazon SWF 來引導工作流程。決策代表動作或一組動作，這是工作流程中的後續步驟。一個典型的決策是排程活動任務。決策也可以用來延遲具有計時器的任務、請求取消進行中的任務，以及完成工作流程。

活動工作者和決策者接收其任務（分別為活動任務和決策任務）的機制是透過輪詢 Amazon SWF 服務。

Amazon SWF 透過在每個決策任務中包含目前工作流程執行歷史記錄的副本，通知決策者工作流程的狀態。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。事件的範例包括任務完成、任務逾時或計時器過期。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)，因此您可以控制對 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。

執行您的工作流程

以下概述在 Amazon SWF 中開發和執行工作流程所需的步驟：

1. 撰寫活動工作者以執行工作流程中的處理步驟。
2. 撰寫決策者來處理工作流程的協調邏輯。
3. 向 Amazon SWF 註冊您的活動和工作流程。

您可以編寫程式或使用執行此步驟 AWS 管理主控台。

4. 啟動您的活動工作者和決策者。

這些演員可以在可存取 Amazon SWF 端點的任何運算裝置上執行。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)；資料中心的伺服器；甚至是行動裝置，來託管決策者或活動工作者。啟動後，決策者和活動工作者應該開始輪詢 Amazon SWF 任務。

5. 啟動您工作流程的一或多個執行。

您可以透過程式設計方式或透過啟動工作流程 AWS 管理主控台。

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。為了回應，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 管理主控台。

您可以篩選和檢視執行中和已完成執行的完整詳細資訊。例如，您可以選取開啟的執行，以查看哪些任務已完成及其結果。

設定開發環境

您可以選擇使用支援的任何程式設計語言來開發 Amazon SWF AWS。對於 Java AWS Flow Framework 開發人員，也可以使用。如需詳細資訊，請參閱 [AWS Flow Framework](#) 網站，請參閱 [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 在不同區域中提供端點。

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資料或屬性共用。

當您註冊 Amazon SWF 網域、工作流程或活動時，它只會存在於您註冊它的區域內。例如，您可以在 SWF-Flows-1 兩個不同的區域中註冊名為 `example.com` 的網域，但它們不會彼此共用資料或屬性，每個都充當完全獨立的網域。

如需 Amazon SWF 端點的清單，請參閱 [區域和端點](#)。

使用 AWS SDKs 開發

Amazon SWF 受適用於 Java、.NET、Node.js、PHP、Python 和 Ruby AWS SDKs 支援，提供以您選擇的程式設計語言使用 Amazon SWF HTTP API 的便利方式。

您可以使用這些程式庫公開的 API 來開發決策者、活動工作者或工作流程啟動者。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工具 AWS，包括 SDKs，請前往 [開發人員中心](#)。

如需每個 SDK 中 Amazon SWF 操作的詳細資訊，請參閱 SDK 的語言特定參考文件。

考慮 AWS Flow Framework

AWS Flow Framework 是增強型 SDK，用於撰寫在 Amazon SWF 上作為工作流程執行的分散式非同步程式。此架構適用於 Java 程式設計語言，並提供撰寫複雜分散式程式的類別。

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映射到程式中的方法。AWS Flow Framework 支援標準物件導向概念，例如例外型錯誤處理。使用編寫的程式 AWS

Flow Framework 可以完全在您偏好的編輯器或 IDE 內建立、執行和偵錯。如需詳細資訊，請參閱 [AWS Flow Framework](#) 網站，請參閱 [AWS Flow Framework Java 開發人員指南](#)。

Amazon SWF 入門

您可以開始使用下列 Amazon Simple Workflow Service 工作流程應用程式，其中包含一組四個依序運作的活動。本教學課程也涵蓋下列主題：

- 設定 default 和 execution-time 工作流程與活動選項。
- 輪詢 Amazon SWF 以進行決策和活動任務。
- 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、訂閱使用者，以及發佈訊息至訂閱的端點。

您可以同時使用 Amazon SWF 和 Amazon Simple Notification Service (Amazon SNS) 來模擬「人力任務」工作流程，其中需要人力工作者來執行一些動作，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 是雲端型 Web 服務，因此與 Amazon SWF 的通訊可能來自網際網路連線可用的任何位置。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 [適用於 Ruby 的 AWS SDK](#) 存取 Amazon SWF 和 Amazon SNS，但有許多可用的開發選項，包括 AWS Flow Framework 適用於 Ruby 的，可讓您更輕鬆地與 Amazon SWF 進行協調和通訊。

Note

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

主題

- [關於工作流程](#)
- [先決條件](#)
- [教學步驟](#)
- [訂閱工作流程教學課程第 1 部分：搭配使用 Amazon SWF 適用於 Ruby 的 AWS SDK](#)
- [訂閱工作流程教學第 2 部分：實作工作流程](#)

- [訂閱工作流程教學第 3 部分：實作活動](#)
- [訂閱工作流程教學第 4 部分：實作活動任務輪詢器](#)
- [訂閱工作流程教學：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)。本教學課程也將說明如何在完成時提供活動結果，以及如何將輸入傳遞給正在排程的任務。Amazon SWF 會處理活動與工作流程之間資訊的協調和交付，反之亦然。

我們也使用鍵盤輸入和 Amazon SNS 來處理 Amazon SWF 與提供資料給工作流程的人類之間的通訊。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 提供非常簡單的方法，使用電子郵件或文字訊息來通知使用者工作流程中的事件。

先決條件

若要遵循本教學，您將需要下列項目：

- [Amazon Web Services 帳戶](#)
- [Ruby 解譯器](#)
- [適用於 Ruby 的 AWS SDK](#)

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，您仍然可以遵循教學課程，無論您選擇的開發選項為何，本教學課程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. [訂閱工作流程教學課程第 1 部分：搭配使用 Amazon SWF 適用於 Ruby 的 AWS SDK](#)
2. [訂閱工作流程教學第 2 部分：實作工作流程](#)
3. [訂閱工作流程教學第 3 部分：實作活動](#)
4. [訂閱工作流程教學第 4 部分：實作活動任務輪詢器](#)
5. [訂閱工作流程教學：執行工作流程](#)

訂閱工作流程教學課程第 1 部分：搭配使用 Amazon SWF 適用於 Ruby 的 AWS SDK

主題

- [包含適用於 Ruby 的 AWS SDK](#)
- [設定 AWS 工作階段](#)
- [註冊 Amazon SWF 網域](#)
- [後續步驟](#)

包含適用於 Ruby 的 AWS SDK

從建立名為 `utils.rb` 的檔案開始。此檔案中的程式碼會取得或視需要建立工作流程和活動程式碼所使用的 Amazon SWF 網域，並提供放置所有類別通用程式碼的位置。

首先，我們需要在程式碼中包含程式碼 `aws-sdk-v1` 庫，以便使用適用於 Ruby 的 SDK 所提供的功能。

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

這可讓我們存取 AWS 命名空間，這可讓您設定全域工作階段相關值，例如您的 AWS 登入資料和區域，也可讓您存取 AWS 服務 APIs。

設定 AWS 工作階段

我們將設定登入 AWS 資料（存取 AWS 服務時需要）和要使用 AWS 的區域，以設定 AWS 工作階段。

有多種方式可以在適用於 Ruby 的 AWS SDK 中設定 AWS 登入資料：在環境變數 (AWS_ACCESS_KEY_ID 和 AWS_SECRET_ACCESS_KEY) 中設定登入資料，或使用設定登入資料 [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
```

立即建立此檔案，以您的 AWS 存取金鑰 ID 和私密存取金鑰取代以 REPLACE_WITH_ 開頭的字串。如需 AWS 存取金鑰的相關資訊，請參閱《Amazon Web Services 一般參考》中的 [如何取得安全登入資料？](#)。

我們也需要設定要使用 AWS 的區域。由於我們將使用 [簡訊服務 \(SMS\)](#) 透過 Amazon SNS 將文字訊息傳送到使用者的電話，因此我們需要確定我們使用 Amazon SNS 支援的區域。請參閱《Amazon Simple Notification Service 開發人員指南》中的 [支援的區域和國家/地區](#)。

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。您可以從範例中移除 SMS 功能，並使用電子郵件做為訂閱 Amazon SNS 主題的唯一端點。如需傳送簡訊的詳細資訊，請參閱《[Amazon Simple Notification Service 開發人員指南](#)》中的 [使用 Amazon SNS 傳送和接收簡訊通知](#)。

我們現在會將一些程式碼新增至 `utils.rb` 以載入組態檔案、取得使用者的登入資料，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 網域

若要使用 Amazon SWF，您需要設定網域：可保存工作流程和活動的具名實體。您可以註冊許多 Amazon SWF 網域，但它們在 AWS 您的帳戶中都必須有唯一的名稱，而且工作流程無法跨網域互動：應用程式的所有工作流程和活動都必須位於相同的網域中，才能彼此互動。

由於我們將在整個應用程式中使用相同的網域，因此我們會在 `utils.rb` 名為 `init_domain` 的中建立函數 `init_domain`，該函數會擷取名為 `SWFSampleDomain` 的 Amazon SWF 網域。

當您一註冊網域後，就可以重複使用它處理許多工作流程執行。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。

若要在適用於 Ruby 的 SDK 中使用 Amazon SWF 網域，請使用 [AWS::SimpleWorkflow.domains](#)，這會傳回可用於列舉和註冊網域的 [DomainCollection](#)：

- 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「[訂閱工作流程教學第 2 部分：實作工作流程](#)」中建立工作流程和啟動者程式碼。

訂閱工作流程教學第 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來源檔案中，但這與 Amazon SWF 的設計方式相反。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。由於 Amazon SWF 會追蹤工作流程和活動的詳細資訊，因此您的工作流程可以協調活動的排程和資料傳輸，無論活動在何處執行。

設定工作流程式碼

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。

網域名稱、活動名稱和活動版本足以讓 Amazon SWF 正面識別活動類型，因此這是我們為了排定活動而需要保留的所有資料。

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。

如果我們找不到相符項目，則會以名稱 'swf-sns-workflow'、版本 '1' 和下列選項註冊新的工作流程類型（在搜尋工作流程的相同 `workflow_types` 集合上呼叫[註冊](#)）。

```
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工作流程類型的「預設行為」，因此不需要在每次開始新的工作流程執行時設定這些值。

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)，以及工作流程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，則任務或工作流程將會逾時。

如需逾時值的詳細資訊，請參閱「[Amazon SWF 逾時類型](#)」。

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，請在網域的 [decision_tasks](#) 集合上呼叫輪詢，以循環切換可用的決策任務。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說明，請參閱《Amazon Simple Workflow Service API 參考》中的 [HistoryEvent](#)。

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 會向您的決策者發出訊號，表示應該開始工作流程執行。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」只是由名稱和版本編號組成的地圖，但這是 Amazon SWF 識別排程活動所需的一切，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" } )
```

當我們排程活動時，Amazon SWF 會將活動任務傳送至我們在排程活動任務時傳入的活動任務清單，發出開始任務的訊號。我們將在「[訂閱工作流程教學第 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 應該排程。

我們需要處理的下一個活動是 ActivityTaskCompleted 事件，會在 Amazon SWF 從活動任務收到活動完成的回應時發生。

```
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
```

由於我們以線性方式執行任務，而且一次只執行一個活動，我們將藉此機會從 `activity_list` 堆疊中彈出已完成的任務。如果此結果為空白清單，則表示我們的工作流程已完成。在此情況下，我們會呼叫任務的 [complete_workflow_execution](#)，向 Amazon SWF 發出我們的工作流程已完成的訊號。

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，這次我們會查看先前的活動是否在完成時傳回任何結果資料至 Amazon SWF，該資料會在事件的屬性中，以選用的 `result` 金鑰提供給工作流程。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。我們會為 `ActivityTaskTimedOut` 和 `ActivityTaskFailed` 事件提供示範處理常式程式碼，並為 `WorkflowExecutionCompleted` 事件提供示範處理常式程式碼，該事件會在 Amazon SWF 處理我們用完要執行的活動時發出的 `complete_workflow_execution` 呼叫時產生。

```
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。

若要啟動工作流程執行，請呼叫已註冊工作流程類型的 [start_execution](#) (`AWS::SimpleWorkflow::WorkflowType`)。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。

由於我們希望工作流程在執行檔案時開始執行，請新增一些執行個體化類別的程式碼，然後呼叫我們剛定義的 `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「[訂閱工作流程教學第 3 部分：實作活動](#)」中定義活動和啟動者程式碼。

訂閱工作流程教學第 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。由於我們的活動將共用一些功能，讓我們做一些基礎，並建立一些可以共用的常見程式碼。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
```

這些屬性會存取類別 `initialize` 方法中定義的執行個體資料，其會採用活動名稱，以及在向 Amazon SWF 註冊活動時要使用的選用版本和選項映射。

```
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

  # 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執行的第一個活動是 `get_contact_activity`，這會擷取使用者的 Amazon SNS 主題訂閱資訊。

建立一個名為 `get_contact_activity.rb` 的新檔案，並且需要兩個 `yaml`，我們將使用它來準備字串以傳遞至 Amazon SWF，以及 `basic_activity.rb`，我們將使用它作為此 `GetContactActivity` 類別的基礎。

```
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
```

因為我們將活動註冊碼放在 `BasicActivity` 中，所以 `GetContactActivity initialize` 的方法非常簡單。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)."
```

`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。原因很重要：您在完成活動時傳遞給 Amazon SWF 的任何結果都只能是字串資料。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`

我們現在將深入探索 Amazon SNS，並建立使用產生的資訊 `get_contact_activity` 來訂閱使用者 Amazon SNS 主題的活動。

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
```

現在我們已備妥程式碼來設定和註冊活動，我們將新增一些程式碼來建立 Amazon SNS 主題。為此，我們將使用 [AWS::SNS::Client](#) 物件的 [create_topic](#) 方法。

將 `create_topic` 方法新增至您的類別，這會需要傳入的 Amazon SNS 用戶端物件。

```
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
end
```

```
end

return topic_arn
end
```

取得主題的 Amazon Resource Name (ARN) 後，我們可以將其與 Amazon SNS 用戶端的 [set_topic_attributes](#) 方法搭配使用，以設定主題的 `DisplayName`，這是使用 Amazon SNS 傳送簡訊時的必要項目。

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))
```

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。

假設一切都沒問題，但我們將繼續填寫我們的 `do_activity` 方法，使用取得 Amazon SNS 用戶端適用於 Ruby 的 AWS SDK，並將其傳遞至我們的 `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](#) 來設定 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，此活動會使用傳入端點（電子郵件和簡訊）來訂閱使用者 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 格式重新封裝下一個活動的資訊，以便將其傳回 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
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`

使用者訂閱 Amazon SNS 主題後，仍需要確認訂閱請求。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
end
```

```

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 主題會維護該主題的[訂閱](#)清單，我們可以檢查使用者是否已確認特定訂閱，方法是查看訂閱的 ARN 是否設定為 以外的任何項目 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。

由於等待訂閱確認可能需要一些時間，我們偶爾會 `record_heartbeat` 呼叫活動任務。這會向 Amazon SWF 發出活動仍在處理的訊號，也可以用來提供活動進度的更新（如果您正在處理您可以報告進度的檔案等動作）。

```
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`

如果工作流程目前有所進展，我們已成功將使用者訂閱 Amazon SNS 主題，而且使用者已確認訂閱。

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
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 主題會將您提供的訊息傳送至該主題存在的所有訂閱和已確認端點。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「[訂閱工作流程教學第 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。

訂閱工作流程教學第 4 部分：實作活動任務輪詢器

在 Amazon SWF 中，執行中工作流程執行的活動任務會出現在活動任務清單中，該任務會在您排程工作流程中的活動時提供。

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以啟動活動時，使用它來啟動我們的活動。

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 Amazon SWF 註冊每個活動。
- 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

```

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。由於每個類別都會註冊其相關聯的活動 (`basic_activity.rb` 如果您需要檢閱該程式碼，請參閱)，這足以讓 Amazon SWF 知道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 `poll_for_activities` 的新方法，並在網域持有的 [activity_tasks](#) 上呼叫 [輪詢](#)，以取得活動任務。

```

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
  end
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
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

此程式碼會在最終活動啟動後從輪詢器結束，因為它已完成其任務並啟動所有活動。在您自己的 Amazon SWF 程式碼中，如果您的活動可能再次執行，您可能想要讓活動輪詢器無限期地執行。

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。您只需要執程式碼，並在「[訂閱工作流程教學：執行工作流程](#)」中查看其運作方式。

訂閱工作流程教學：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。

如果您尚未這麼做，則需要在 `aws-config.txt` 檔案中提供 AWS 存取金鑰，例如教學課程第 1 [設定 AWS 工作階段](#) 部分的。

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
```

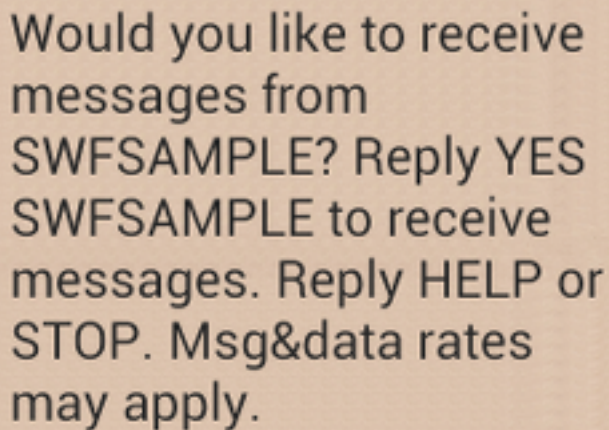
```
You entered:
  email: me@example.com
  phone: 12065550101
```

```
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。

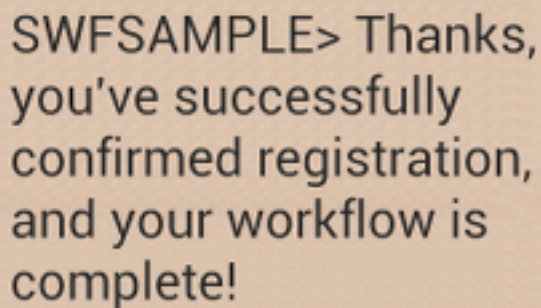
輸入此資訊後不久，您應該會收到來自 Amazon SNS 的電子郵件或文字訊息，要求您確認主題訂閱。如曾輸入簡訊號碼，您會在電話中看到類似以下的內容。

A screenshot of an SMS message in a light brown bubble. The text reads: "Would you like to receive messages from SWFSAMPLE? Reply YES SWFSAMPLE to receive messages. Reply HELP or STOP. Msg&data rates may apply." The time "3:39 PM" is shown at the bottom left.

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。

A screenshot of an SMS message in a light brown bubble. The text reads: "SWFSAMPLE> Thanks, you've successfully confirmed registration, and your workflow is complete!" The time "3:39 PM" is shown at the bottom left.

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」。在取消訂閱主題之前重新執行工作流程可能會導致自動成功，因為 `wait_for_confirmation_activity` 會看到您的訂閱已確認。

若要取消訂閱 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？

本教學課程涵蓋了許多領域，但您仍然可以進一步了解適用於 Ruby 的 AWS SDK、Amazon SWF 或 Amazon SNS。如需詳細資訊及更多範例，請參閱各官方文件：

- [適用於 Ruby 的 AWS SDK 文件](#)
- [Amazon Simple Notification Service 文件](#)
- [Amazon Simple Workflow Service 文件](#)

在 Amazon SWF 主控台中工作

Amazon SWF 主控台提供設定、啟動和管理工作流程執行的選項。

透過 Amazon SWF 主控台，您可以：

- 註冊工作流程網域。
- 註冊工作流程類型和活動類型。
- 啟動、檢視、發出訊號、取消、終止和重新啟動工作流程執行。

註冊網域

工作流程會在稱為網域的資源中執行，該 AWS 資源控制工作流程的範圍。帳戶 AWS 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。

網域註冊是主控台最初唯一可用的功能。至少註冊一個網域之後，您可以對該網域執行下列動作：

- 註冊工作流程和活動類型。
- 啟動工作流程執行。
- 取消、終止及傳送訊號給執行中的工作流程執行。
- 重新啟動已關閉的工作流程執行。

您也可以執行網域管理動作，例如棄用和取消棄用網域。

棄用網域之後，您就無法使用它來建立新的工作流程執行或註冊新的工作流程。棄用網域也會棄用網域中註冊的所有活動和工作流程。在網域棄用之前啟動的執行會繼續執行。

在取消棄用之前已棄用的網域之後，您可以繼續使用網域來註冊工作流程類型並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](#)。

註冊工作流程類型

您可以在至少註冊一個網域之後，在 Amazon SWF 主控台中註冊工作流程類型。

工作流程類型是一組活動類型，可執行目標並包含協調活動的邏輯。工作流程類型會協調和管理可跨多個運算裝置非同步執行的活動執行，並同時採用循序和平行處理方法。

使用主控台註冊 Amazon SWF 工作流程類型

1. 開啟您要註冊工作流程的網域。
2. 選擇註冊，然後選擇註冊工作流程。
3. 在註冊工作流程頁面上，輸入工作流程名稱和工作流程版本。您也可以選擇性地指定[預設任務清單](#)，用來排程執行此工作流程的決策任務。
4. （選用）選擇進階選項，為您的工作流程指定下列詳細資訊：
 - [預設任務優先順序](#) – 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。
 - [預設任務開始關閉逾時](#) – 此工作流程決策任務的預設最長持續時間。
 - [預設子政策](#) – 用於子工作流程執行的預設政策。
 - [預設 Lambda 角色](#) – 連接到此工作流程的預設 IAM 角色。
5. 選擇註冊工作流程。

註冊活動類型

活動是您希望工作流程類型協調和執行的任務（例如：驗證客戶的訂單、使用信用卡等）。執行活動的順序取決於工作流程類型的協調邏輯。

您可以在至少註冊一個網域之後註冊活動類型。

使用主控台註冊 Amazon SWF 活動類型

1. 開啟您要註冊活動的網域。
2. 選擇註冊，然後選擇註冊活動。
3. 在註冊活動頁面上，輸入[活動名稱](#)和[活動版本](#)。您也可以選擇性地指定[預設任務清單](#)，用於排程此活動的任務。
4. （選用）選擇進階選項，為您的活動指定下列詳細資訊：
 - [預設任務優先順序](#) – 要指派給活動的預設任務優先順序。
 - [啟動逾時的預設任務排程](#) – 此活動的任務在指派給工作者之前可等待的預設最長持續時間。
 - [預設任務開始關閉逾時](#) – 工作者處理此活動任務所需的預設最長持續時間。

- [預設任務排程關閉逾時](#) – 此活動任務的預設最長持續時間。
- [預設任務活動訊號逾時](#) – 工作者處理此類型任務之前的預設最長時間，必須透過呼叫 [RecordActivityTaskHeartbeat](#) 來報告進度。

5. 選擇註冊活動。

啟動工作流程

您可以從 Amazon SWF 主控台啟動工作流程執行。您必須先註冊至少一個工作流程，才能開始工作流程執行。

使用主控台啟動工作流程執行

1. 開啟 Amazon SWF 主控台，然後在左側導覽窗格中選擇網域。
2. 在網域名稱下方，選擇工作流程。
3. 在工作流程頁面上，選擇您要執行的工作流程。
4. 選擇 Start execution (開始執行)。
5. 在開始執行頁面上，輸入 [工作流程名稱](#) 和執行 ID，以名稱識別您的執行。您也可以選擇性地指定 [任務清單](#)，用於為此工作流程執行產生的決策任務。
6. (選用) 選擇進階選項，為您的工作流程執行指定下列詳細資訊：
 - [任務優先順序](#) – 用於此工作流程執行的任務優先順序。
 - [執行開始關閉逾時](#) – 此工作流程執行的總持續時間。
 - [任務開始關閉逾時](#) – 此工作流程執行的決策任務最長持續時間。
 - [子政策](#) – 用於此工作流程執行的子工作流程執行的政策，如果終止，請明確或由於過期逾時而呼叫 [TerminateWorkflowExecution](#) 動作。
 - [Lambda 角色](#) – 要連接到此工作流程執行的 IAM 角色。
7. 選擇 Start execution (開始執行)。

管理工作流程執行

您可以依名稱、狀態、ID 和標籤篩選工作流程執行。您可以傳送輸入為作用中工作流程執行的訊號。如果您需要取消或終止工作流程，您可以使用 Try-cancel 選項。取消優於終止工作流程執行，因為取消可讓工作流程有機會執行任何清除任務，然後正確關閉。

在主控台中，您可以管理目前正在執行和/或已關閉的工作流程執行。

管理工作流程執行

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，選擇依屬性篩選執行，然後在屬性下選擇下列其中一個篩選條件：

選擇	若要套用此篩選條件
工作流程	<p>選擇此篩選條件以列出特定工作流程的執行。例如，若要檢視的執行fiction-books-order-workflow，請執行下列動作：</p> <ol style="list-style-type: none"> 1. 選擇工作流程。 2. 在運算子下，選擇等於。 3. 在工作流程下，選擇fiction-books-order-workflow。 4. (選用) 選擇清除篩選條件以移除篩選條件，並開始新的執行搜尋。
狀態	<p>選擇此篩選條件以列出具有特定狀態的執行。例如，若要檢視狀態為失敗的執行，請執行下列動作：</p> <ol style="list-style-type: none"> 1. 選擇狀態。 2. 在運算子下，選擇等於。 3. 在狀態下，選擇失敗。 4. (選用) 選擇清除篩選條件以移除篩選條件，並開始新的執行搜尋。
執行 ID	<p>選擇此篩選條件，根據工作流程的 ID 檢視工作流程執行。例如，若要檢視 ID 為的執行fiction-books-order-category1，請執行下列動作：</p> <ol style="list-style-type: none"> 1. 選擇執行 ID。 2. 在運算子下，選擇等於。 3. 在執行 IDs 下，選擇 fiction-books-order-category1。

選擇	若要套用此篩選條件
	4. (選用) 選擇清除篩選條件以移除篩選條件，並開始新的執行搜尋。
標籤	<p>選擇此篩選條件以列出具有特定標籤的執行。例如，若要檢視狀態為的執行purchaseOrder，請執行下列動作：</p> <ol style="list-style-type: none"> 1. 選擇標籤。 2. 在運算子下，選擇等於。 3. 在標籤下，選擇 purchaseOrder。 4. (選用) 選擇清除篩選條件以移除篩選條件，並開始新的執行搜尋。

4. (選用) 套用所需的篩選條件以列出工作流程執行後，您可以對作用中執行執行下列操作：
 - Signal – 使用此選項傳送執行中的工作流程執行額外資料。若要執行此作業：
 1. 選擇您要傳送其他資料的執行。
 2. 選擇訊號，然後在訊號執行對話方塊中指定資料。
 3. 選擇訊號。
 - Try-cancel – 使用此選項來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 Try-cancel。
 - 終止 – 使用此選項來終止工作流程執行。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. 對於子政策，請確定已選取終止。
 3. (選用) 指定終止執行的原因和詳細資訊。
 4. 選擇終止。
5. (選用) 重新執行 – 使用此選項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，重新執行按鈕會變成啟用。選擇重新執行。
 2. 在重新執行執行頁面上，指定工作流程執行的詳細資訊，如中所述[啟動工作流程](#)。

Amazon SWF 中的基本工作流程概念

Note

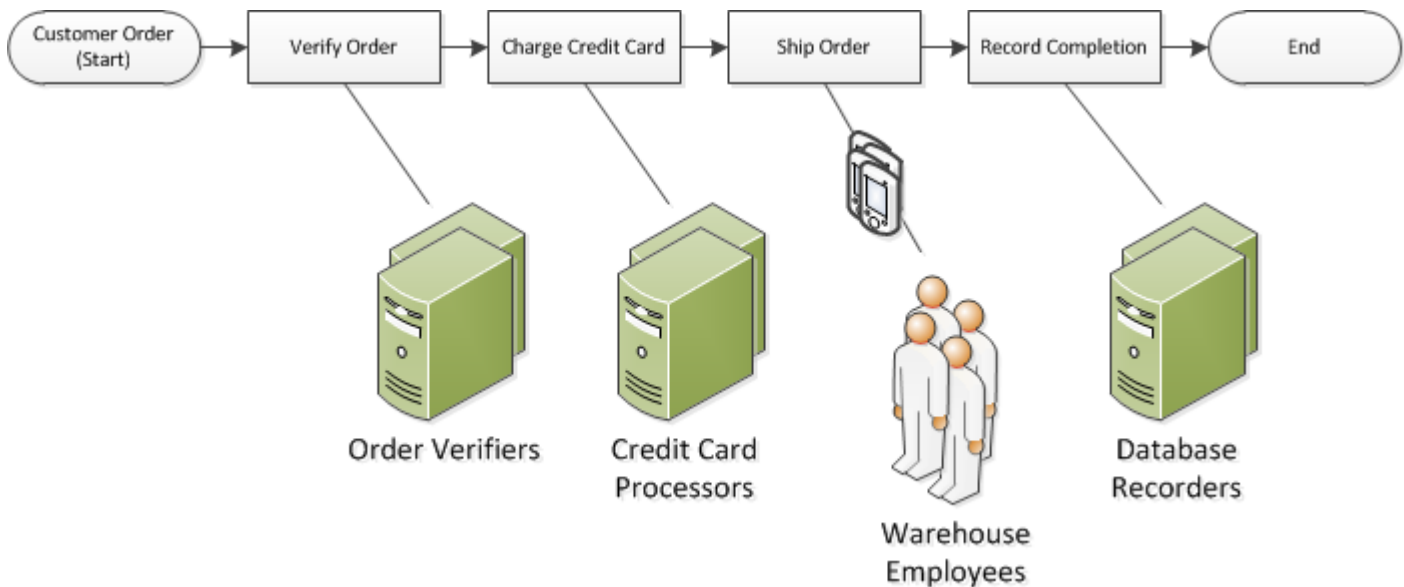
本章中的概念提供 Amazon Simple Workflow Service 的概觀，並說明其主要功能。如果您要尋找範例，請參閱 [使用 Amazon SWF APIs](#)。

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，這些任務會以活動表示。活動的執行順序是根據工作流程的協調性邏輯所決定。

電子商務應用程式的範例工作流程

下圖顯示涉及人員和自動化程序的電子商務訂單處理工作流程：



電子商務應用程式工作流程會在客戶下訂單時開始，並包含四個任務：

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，因為 Amazon SWF 支援分散式程序，所以這些任務可以在不同的位置執行。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。

除了循序處理任務之外，Amazon SWF 也支援具有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。

其他概念

- [在 Amazon SWF 中建立工作流程](#)
- [在 Amazon SWF 中執行工作流程](#)
- [Amazon SWF 中的工作流程歷史記錄](#)
- [Amazon SWF 中的物件識別符](#)
- [Amazon SWF 中的網域](#)
- [Amazon SWF 中的演員](#)
- [Amazon SWF 中的任務](#)
- [Amazon SWF 中的任務清單](#)
- [Amazon SWF 中的工作流程執行關閉](#)
- [Amazon SWF 工作流程的生命週期](#)
- [在 Amazon SWF 中輪詢任務](#)

在 Amazon SWF 中建立工作流程

建立基本循序工作流程包含下列階段。

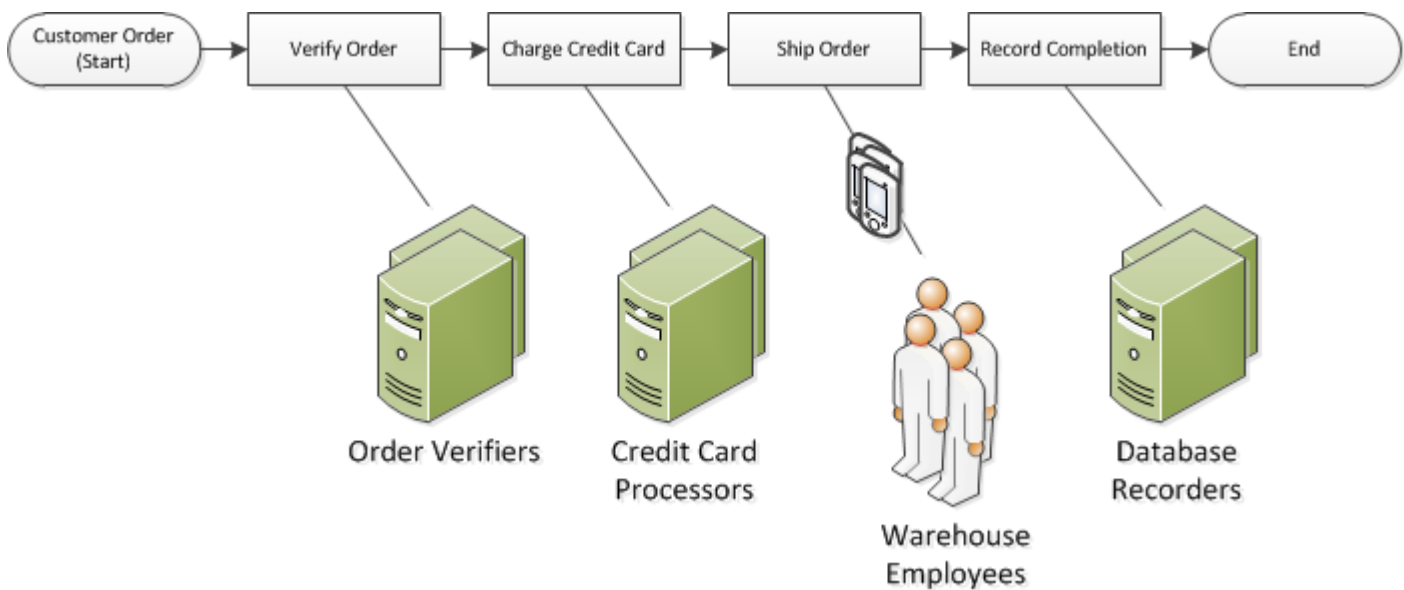
- 為工作流程建模、註冊其類型，以及註冊其活動類型
- 開發與啟動可執行活動任務的活動工作者
- 開發與啟動決策者，以使用工作流程歷史記錄來判斷後續步驟
- 開發與啟動工作流程啟動者 (即啟動工作流程執行的應用程式)。

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。



在 Amazon SWF 中執行工作流程

在設計協調邏輯和活動之後，您會向 Amazon SWF 將這些元件註冊為工作流程和活動類型。在註冊期間，您可以指定每種類型的名稱、版本和預設組態值。

只有已註冊的工作流程和活動類型才能與 Amazon SWF 搭配使用。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。

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，每個客戶訂單會啟動新的工作流程執行。啟動工作流程的應用程式類型取決於客戶下訂單的方式。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，您可以將稱為 `workflowId` 的識別符與工作流程執行建立關聯，以便將現有的業務識別符整合到工作流程中。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。

除了您提供的識別符之外，Amazon SWF 還會將系統產生的唯一識別符 `runId` 與每個工作流程執行建立關聯。Amazon SWF 僅允許一個具有此識別符的工作流程執行在任何指定時間執行；雖然您可以有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不同的 `runId`。

Amazon SWF 中的工作流程歷史記錄

Amazon SWF 會在工作流程歷史記錄中記錄每個工作流程執行的進度 - 詳細、完整且一致的記錄自工作流程執行開始以來發生的每個事件。

事件代表工作流程執行狀態中的離散變更，例如正在排程的新活動或正在完成的執行中活動。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：

- 應用程式可以是無狀態的，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，作為 Amazon SWF 與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副本。

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"
}
]
```

如需工作流程執行歷史記錄中可顯示之不同類型事件的詳細清單，請參閱《Amazon Simple Workflow Service API 參考》中的 [HistoryEvent](#) 資料類型。

Amazon SWF 會在執行關閉後的可設定天數內存放所有工作流程執行的完整歷史記錄。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 中的物件識別符

以下清單說明如何唯一識別 Amazon SWF 物件，例如工作流程執行。

- 工作流程類型 – 已註冊的工作流程類型由其網域、名稱和版本識別。工作流程類型指定於對 `RegisterWorkflowType` 的呼叫中。
- 活動類型 – 已註冊的活動類型會依其網域、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 – 每個決策任務和活動任務都由唯一的任務字符識別。任務字符是由 Amazon SWF 產生，並與來自 `PollForDecisionTask` 或 `PollForActivityTask` 的回應中任務的其他資訊一起傳回 `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 – 透過網域、工作流程 ID 和執行 ID 來識別工作流程的單一執行。前兩項是傳遞至 [StartWorkflowExecution](#) 的參數。執行 ID 則由 `StartWorkflowExecution` 傳回。

Amazon SWF 中的網域

工作流程會在稱為網域的 AWS 資源中執行，該網域提供擴展您 AWS 帳戶中 Amazon SWF 資源的方法。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。

帳戶 AWS 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」。保留期間是工作流程執行完成後，Amazon SWF 將繼續保留工作流程執行相關資訊的時間長度。

網域註冊是主控台最初唯一可用的功能。在至少註冊一個網域之後，您可以對該網域執行下列動作：

- 註冊工作流程和活動類型。
- 啟動工作流程執行。
- 取消、終止及傳送訊號給執行中的工作流程執行。
- 重新啟動已關閉的工作流程執行。

您也可以執行網域管理動作，例如棄用和取消棄用網域。

棄用網域之後，您就無法使用它來建立新的工作流程執行或註冊新的工作流程。棄用網域也會棄用網域中註冊的所有活動和工作流程。在網域棄用之前啟動的執行會繼續執行。

在取消棄用先前已棄用網域之後，您可以繼續使用網域來註冊工作流程類型並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](#)。

Amazon SWF 中的演員

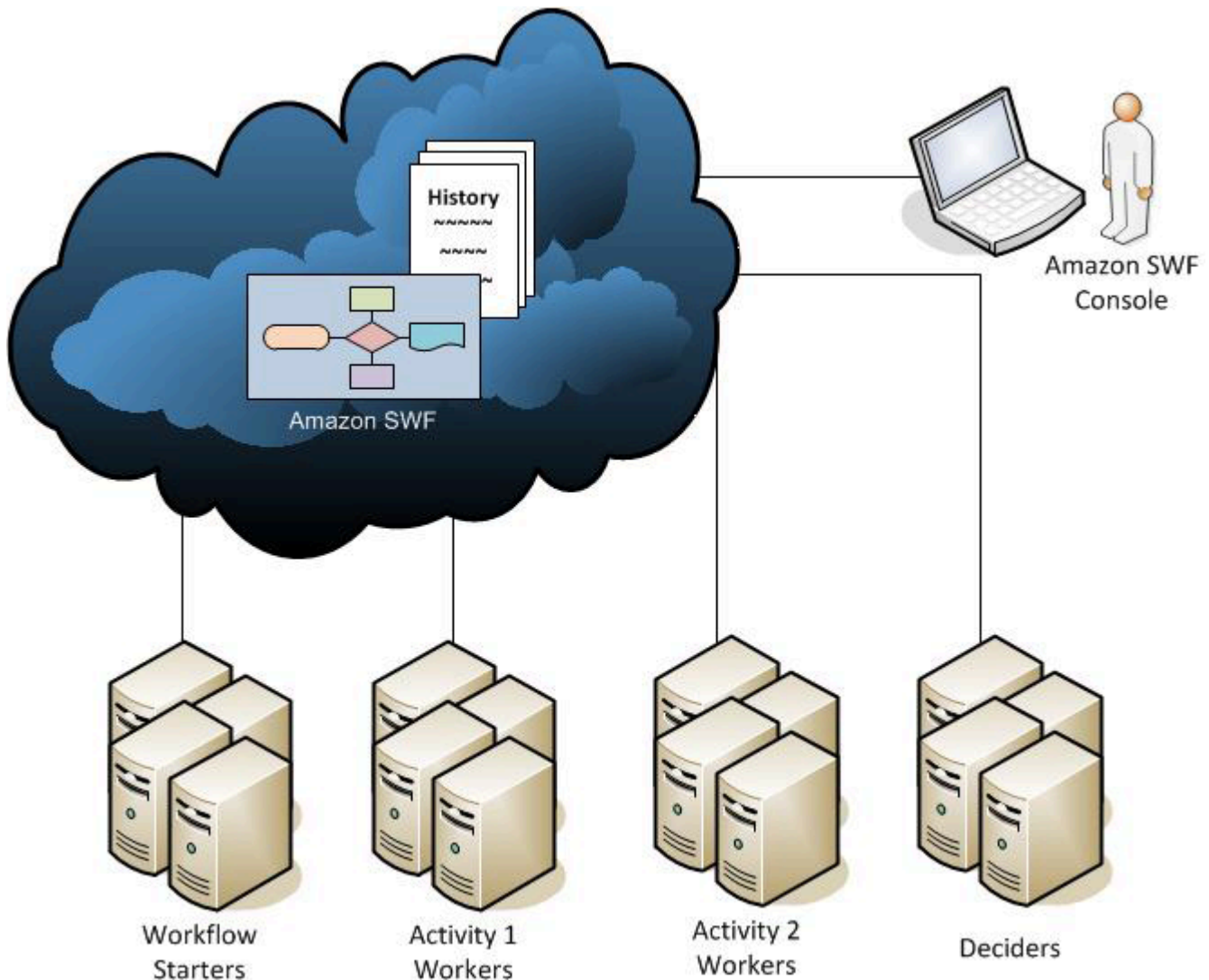
主題

- [什麼是 Amazon SWF 中的演員？](#)
- [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 中的演員？

在操作過程中，Amazon SWF 會與多種不同類型的程式設計演員互動。動作者可以是[工作流程啟動者](#)、[決策者](#)或[活動工作者](#)。這些演員會透過其 API 與 Amazon SWF 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，包括 Amazon SWF 及其演員。



工作流程啟動者

工作流程啟動者是什麼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。當決策者從 Amazon SWF 收到決策任務時，它會分析工作流程執行歷史記錄，以確定工作流程執行中的下一個適當步驟。決策者會使用決策將這些步驟傳回 Amazon SWF。決策是一種 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《Amazon Simple Workflow Service API 參考》中的[決策](#)。

以下是 JSON 格式的決策範例，這是傳輸到 Amazon SWF 的格式。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"
  }
}
```

在工作流程執行啟動時，及每次工作流程執行內的狀態變更時，決策者會接收到決策任務。決策者透過接收決策任務並以更多決策回應 Amazon SWF，繼續推動工作流程執行，直到決策者判斷工作流程執行完成為止。直到最後，決策者會回應關閉工作流程執行的決策。工作流程執行關閉後，Amazon SWF 將不會排程該執行的其他任務。

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工作流程類型的任務。

活動工作者

活動工作者是一種程序或執行緒，可執行屬於工作流程一部分的「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。

若要在工作流程中使用活動任務，您必須使用 Amazon SWF 主控台或 [RegisterActivityType](#) 動作來註冊。

每個活動工作者都會輪詢 Amazon SWF 是否有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執行。收到任務後，活動工作者會處理任務以完成，然後向 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。當活動任務完成時，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。每個演員都可以透過使用者定義的字串形式，向 Amazon SWF 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，或將指標傳遞至儲存在另一個系統或服務（例如 Amazon S3 或 DynamoDB）上的資料。直接傳遞的資料和其他資料存放區的指標都會記錄在工作流程執行歷史記錄中；不過，Amazon SWF 不會從外部存放區複製或快取任何資料做為歷史記錄的一部分。

由於 Amazon SWF 會維護每個工作流程執行的完整執行狀態，包括輸入和任務的結果，因此所有演員都可以是無狀態。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 中的任務

Amazon SWF 透過提供稱為任務的工作指派，與活動工作者和決策者互動。Amazon SWF 中有三種類型的任務：

- 活動任務 – 活動任務會告知活動工作者執行其功能，例如檢查庫存或向信用卡收費。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 – Lambda 任務類似於活動任務，但會執行 Lambda 函數，而非傳統的 Amazon SWF 活動。如需如何定義 Lambda 任務的詳細資訊，請參閱「[AWS Lambda Amazon SWF 中的任務](#)」。
- 決策任務 – 決策任務會告知決策者工作流程執行的狀態已變更，以便決策者可以判斷需要執行的下一個活動。決策任務包含目前工作流程歷史記錄。

Amazon SWF 會在工作流程開始時以及工作流程狀態變更時排程決策任務，例如活動任務完成時。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，並使用一組指定工作流程執行中後續應發生事項的決策來回應 Amazon SWF。基本上，每個決策任務都讓決策者有機會評估工作流程，並向 Amazon SWF 提供方向。

為了確保不會處理任何衝突決策，Amazon SWF 會將每個決策任務只指派給一個決策者，並一次僅允許一個決策任務在工作流程執行中處於作用中狀態。

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

當活動工作者完成活動任務時，它會向 Amazon SWF 報告任務已完成，並包含產生的任何相關結果。Amazon SWF 會使用指出任務已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，將更新的歷史記錄傳輸給決策者。

Amazon SWF 會將每個活動任務只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 中的任務清單

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。在 Amazon SWF 中排程任務時，您可以指定要放入其中的佇列（任務清單）。同樣地，當您輪詢 Amazon SWF 以取得任務時，您會說出要從哪個佇列（任務清單）取得任務。

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，與活動和工作流程一樣，任務清單的範圍是特定 AWS 區域和 Amazon SWF 網域。

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。任務會依序排程在任務清單上。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。

Amazon SWF 中的工作流程執行關閉

啟動工作流程執行後，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 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 會監控工作流程執行，以確保工作流程執行不會超過任何使用者指定的逾時設定。如果工作流程執行逾時，Amazon SWF 會自動將其關閉。如需逾時值的詳細資訊，請參閱「[Amazon SWF 逾時類型](#)」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。

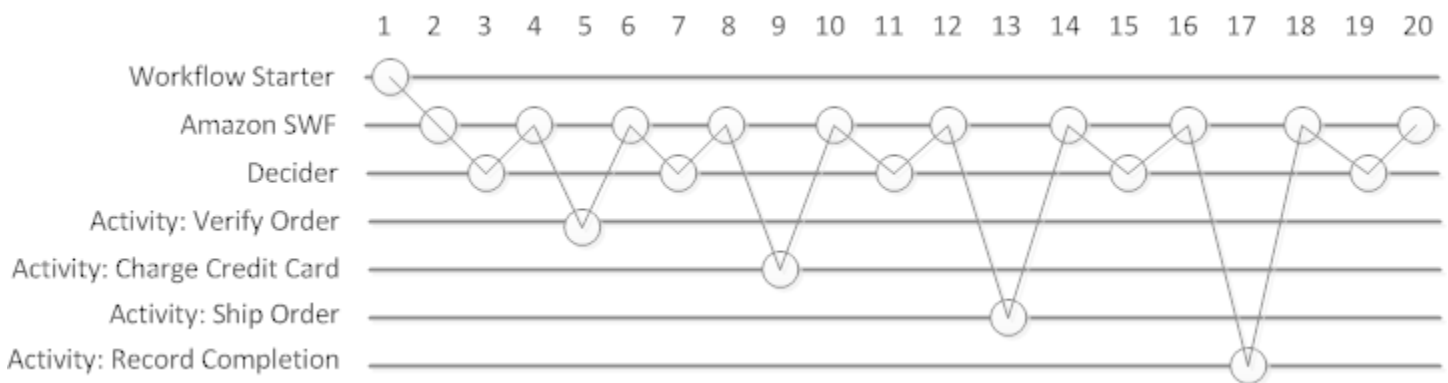
最後，您可以直接從 Amazon SWF 主控台或以程式設計方式使用 [TerminateWorkflowExecution](#) API 終止工作流程執行。終止會強制關閉工作流程執行。取消優於終止，因為您的決策者可管理工作流程執行的關閉。

如果執行超過特定服務定義的限制，Amazon SWF 會終止工作流程執行。如果父工作流程已終止，且適用的子政策指出也應終止子工作流程，則 Amazon SWF 會終止子工作流程。

Amazon SWF 工作流程的生命週期

從工作流程執行開始到完成，Amazon SWF 會透過指派適當的任務來與演員互動，包括活動任務或決策任務。

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 動作來啟動訂單的工作流程執行，並提供訂單資訊。	StartWorkflowExecution 動作。
2. Amazon SWF 會收到啟動工作流程執行請求，然後排程第一個決策任務。	WorkflowExecutionStarted 事件和 DecisionTaskScheduled 事件。

描述	動作、決策或事件
3. 決策者從 Amazon SWF 接收任務、檢閱歷史記錄、套用協調邏輯以判斷先前未發生的活動、決定使用活動工作者處理任務所需的資訊來排程驗證訂單活動，並將決策傳回給 Amazon SWF。	PollForDecisionTask 動作。 RespondDecisionTaskCompleted 動作和 ScheduleActivityTask 決策。
4. Amazon SWF 會收到決策、排程驗證訂單活動任務，並等待活動任務完成或逾時。	ActivityTaskScheduled 事件
5. 可執行驗證訂單活動的活動工作者會收到任務、執行任務，並將結果傳回給 Amazon SWF。	PollForActivityTask 動作和 RespondActivityTaskCompleted 動作。
6. Amazon SWF 會收到驗證訂單活動的結果，將它們新增至工作流程歷史記錄，並排程決策任務。	ActivityTaskCompleted 事件和 DecisionTaskScheduled 事件。

描述	動作、決策或事件
7. 決策者從 Amazon SWF 接收任務、檢閱歷史記錄、套用協調邏輯、決定使用活動工作者處理任務所需的資訊來排程 ChargeCreditCard 活動任務，並將決策傳回給 Amazon SWF。	PollForDecisionTask 動作。 RespondDecisionTaskCompleted 動作和 ScheduleActivityTask 決策。
8. Amazon SWF 會收到決策、排程 ChargeCreditCard 活動任務，並等待它完成或逾時。	DecisionTaskCompleted 事件和 ActivityTaskScheduled 事件。
9. 可執行 ChargeCreditCard 活動的活動工作者會收到任務、執行任務，並將結果傳回給 Amazon SWF。	PollForActivityTask 和 RespondActivityTaskCompleted 動作。
10. Amazon SWF 會收到 ChargeCreditCard 活動任務的結果，將其新增至工作流程歷史記錄，並排程決策任務。	ActivityTaskCompleted 事件和 DecisionTaskScheduled 事件。

描述	動作、決策或事件
11. 決策者從 Amazon SWF 接收任務、檢閱歷史記錄、套用協調邏輯、決定使用活動工作者執行任務所需的資訊來排程 ShipOrder 活動任務，並將決策傳回給 Amazon SWF。	PollForDecisionTask 動作。 RespondDecisionTaskCompleted 與 ScheduleActivityTask 決策。
12. Amazon SWF 會收到決策、排程 ShipOrder 活動任務，並等待它完成或逾時。	DecisionTaskCompleted 事件和 ActivityTaskScheduled 事件。
13. 可執行 ShipOrder 活動的活動工作者會收到任務、執行任務，並將結果傳回給 Amazon SWF。	PollForActivityTask 動作和 RespondActivityTaskCompleted 動作。
14. Amazon SWF 會收到 ShipOrder 活動任務的結果，將其新增至工作流程歷史記錄，並排程決策任務。	ActivityTaskCompleted 事件和 DecisionTaskScheduled 事件。

描述	動作、決策或事件
<p>15. 決策者從 Amazon SWF 接收任務、檢閱歷史記錄、套用協調邏輯、決定使用活動工作者執行任務所需的資訊來排程 RecordCompletion 活動任務，並將決策傳回給 Amazon SWF。</p>	<p>PollForDecisionTask 動作。RespondDecisionTaskCompleted 動作和 ScheduleActivityTask 決策。</p>
<p>16. Amazon SWF 會收到決策、排程 RecordCompletion 活動任務，並等待它完成或逾時。</p>	<p>DecisionTaskCompleted 事件和 ActivityTaskScheduled 事件。</p>
<p>17. 可執行 RecordCompletion 活動的活動工作者會收到任務、執行任務，並將結果傳回給 Amazon SWF。</p>	<p>PollForActivityTask 動作和 RespondActivityTaskCompleted 動作。</p>
<p>18. Amazon SWF 會收到 RecordCompletion 活動任務的結果、將其新增至工作流程歷史記錄，以及排程決策任務。</p>	<p>ActivityTaskCompleted 事件和 DecisionTaskScheduled 事件。</p>

描述	動作、決策或事件
19. 決策者從 Amazon SWF 接收任務、檢閱歷史記錄、套用協調邏輯、決定關閉工作流程執行，並將決策連同任何結果傳回給 Amazon SWF。	PollForDecisionTask 動作。 RespondDecisionTaskCompleted 動作和 CompleteWorkflowExecution 決策。
20. Amazon SWF 會關閉工作流程執行，並封存歷史記錄以供日後參考。	WorkflowExecutionCompleted 事件。

在 Amazon SWF 中輪詢任務

決策者和活動工作者使用長輪詢與 Amazon SWF 通訊。決策者或活動工作者會定期啟動與 Amazon SWF 的通訊，通知 Amazon SWF 接受任務的可用性，然後指定任務清單以從中取得任務。

如果指定任務清單中有可用的任務，Amazon SWF 會立即在回應中傳回該任務。如果沒有可用的任務，Amazon SWF 會將 TCP 連線保持開啟長達 60 秒，因此，如果任務在此期間變成可用，則可以在相同的連線中傳回。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](#)及[輪詢活動任務](#)。

Amazon SWF 中的進階工作流程概念

「[???](#)」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。本節說明可用於建構工作流程的進階工作流程概念。

進階概念

- [版本控制](#)
- [訊號](#)
- [Amazon SWF 中的子工作流程](#)
- [Amazon SWF 中的標記](#)
- [Amazon SWF 中的標籤](#)
- [使用 Amazon SWF 實作獨佔選擇](#)
- [Amazon SWF 中的計時器](#)
- [在 Amazon SWF 中取消活動任務](#)

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 中的任務清單](#)」(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`。

若要使用訊號，請定義要傳遞給訊號的訊號名稱和資料，如果有的話。然後，以程式設計決策者辨識歷史記錄中的訊號事件 ([WorkflowExecutionSignaled](#))，並適當地加以處理。當程序想要發出工作流程執行的訊號時，它會呼叫 Amazon SWF（使用 [SignalWorkflowExecution](#) 動作，或在決策者的情況下使用 [SignalExternalWorkflowExecution](#) 決策），以指定目標工作流程執行的識別符、訊號名稱和訊號資料。然後，Amazon SWF 會收到訊號，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，使用者可以傳送訊號來取消訂單，但只能在下訂單後的一小時內取消。Amazon SWF 沒有基本的，可讓決策者等待來自服務的訊號。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。

對於應該取消工作流程的情況，例如，客戶取消訂單本身，應使用 `RequestCancelWorkflowExecution` 動作，而不是傳送訊號到工作流程。

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訊號以取消訂單。

```
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 會傳回如下的成功 HTTP 回應。Amazon SWF 將產生決策任務，通知決策者處理訊號。

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

Amazon SWF 中的子工作流程

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` 決策的屬性也會指定子政策，也就是 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 將終止子執行。
- `REQUEST_CANCEL`：Amazon SWF 會在子工作流程執行歷史記錄中放置 `WorkflowExecutionCancelRequested` 事件，嘗試取消子系執行。

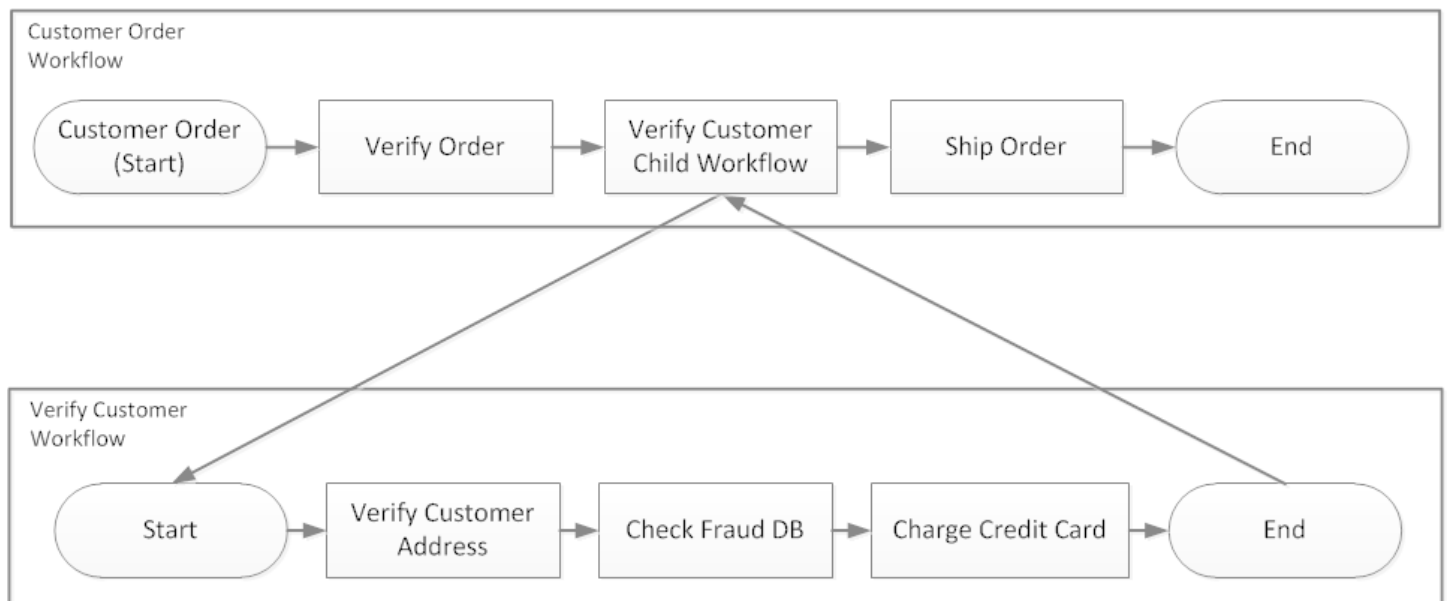
- ABANDON：Amazon SWF 不會採取任何動作；子執行將繼續執行。

子工作流程執行在啟動之後，會像一般執行一樣運作。完成後，Amazon SWF 會在父工作流程執行的工作流程歷史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。

Amazon SWF 中的標記

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，決策者會使用 `RecordMarker` 決策、命名標記、將所需資料附加至決策，並使用 `RespondDecisionTaskCompleted` 動作通知 Amazon SWF。Amazon SWF 會接收請求、在工作流程歷史記錄中記錄標記，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，則會傳回類似以下內容的成功 HTTP 回應。

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

Amazon SWF 中的標籤

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 SDKs 或直接與 Amazon SWF API 互動來管理 Amazon Simple Workflow Service 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 APIs](#)，請參閱 [和 Amazon Simple Workflow Service 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建立關聯，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操作時篩選 listi。透過仔細選取您指派給執行的標籤，您可以使用它們來提供有意義的清單。

例如，假設您執行數個履行中心。透過標籤，您可以列出特定履行中心發生的程序。或者，如果客戶轉換不同類型的媒體檔案，標籤可能會在轉換視訊、音訊和影像檔案時指出不同的程序。

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。當您使用可見性動作來列出或計數工作流程執行時，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. 透過使用 `ListOpenWorkflowExecutions`、`ListClosedWorkflowExecutions` 和 `CountClosedWorkflowExecutions` 動作指定 `tagFilter` 參數 `CountOpenWorkflowExecutions`，列出或計算以特定標籤標記的執行。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"
}
}
```

使用標籤控制對網域的存取

您可以在 IAM 中參考與 Amazon SWF 網域相關聯的標籤，以控制對 Amazon Simple Workflow Service 網域的存取。

例如，您可以限制 Amazon SWF 網域，其中包含具有 金鑰environment和 值production的標籤，條件如下：

```
"Condition": {
  "StringEquals": {"aws:ResourceTag/environment": "production"}
}
```

如需詳細資訊，請參閱：

- [使用 IAM 標籤控制存取](#)
- [標籤類型政策](#)

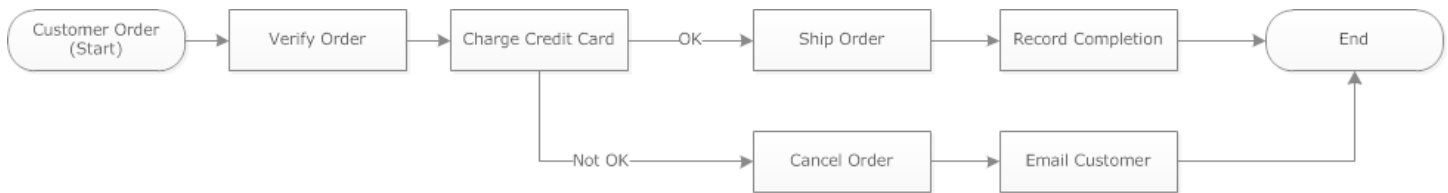
使用 Amazon SWF 實作獨佔選擇

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。透過獨家選擇模式，您可以建立符合應用程式複雜需求的彈性工作流程。

Amazon SWF 沒有特定的獨佔選擇動作。若要實作獨佔選擇，您必須撰寫決策者邏輯，以根據先前活動的結果做出決策。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分驗證活動

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 能夠驗證 RespondDecisionTaskCompleted 動作中的資料，Amazon SWF 會傳回如下的成功 HTTP 回應。

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

Amazon SWF 中的計時器

透過計時器，您可以在經過特定時間後通知決策者。

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。經過指定的時間後，Amazon SWF 會將 TimerFired 事件新增至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。

在 Amazon SWF 中取消活動任務

活動任務取消會通知決策者結束不再需要執行的活動。Amazon SWF 使用合作取消機制，不會強制中斷執行中的活動任務。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 會傳回成功的 HTTP 回應，如下所示。Amazon SWF 將產生決策任務，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，則會傳回類似以下內容的成功 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` 事件所示，請設定決策者採取適當的步驟，以遵循任務取消，例如關閉工作流程執行。

如果活動任務無法取消，例如，如果任務完成、失敗或逾時而不是取消，您的決策者應該接受活動的結果，或執行使用案例所需的任何清除或緩解。

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。活動工作者可以定期使用 `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，Amazon SWF 會將 `cancelRequest` 物件的值設定為 `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 不會阻止成功完成已請求取消的活動任務；取決於活動以決定如何處理取消請求。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，它會更新工作流程執行歷史記錄，並排程決策任務以通知決策者。

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 中的安全性

本節提供有關 Amazon Simple Workflow Service 安全性和身分驗證的資訊。

主題

- [Amazon Simple Workflow Service 中的資料保護](#)
- [Amazon Simple Workflow Service 中的 Identity and Access Management](#)
- [記錄和監控](#)
- [Amazon Simple Workflow Service 的合規驗證](#)
- [Amazon Simple Workflow Service 中的彈性](#)
- [Amazon Simple Workflow Service 中的基礎設施安全](#)
- [Amazon Simple Workflow Service 中的組態和漏洞分析](#)

Amazon SWF 使用 IAM 控制對其他服務 AWS 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](#)。如需安全憑證的概觀，請參閱《Amazon Web Services 一般參考》中的[AWS 安全憑證](#)。

Amazon Simple Workflow Service 中的資料保護

AWS [共同責任模型](#)適用於 Amazon Simple Workflow Service 中的資料保護。如此模型所述，AWS 負責保護執行所有的全球基礎設施 AWS 雲端。您負責維護在此基礎設施上託管內容的控制權。您也同時負責所使用 AWS 服務的安全組態和管理任務。如需資料隱私權的詳細資訊，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，我們建議您保護 AWS 帳戶登入資料，並使用 AWS IAM Identity Center 或 AWS Identity and Access Management (IAM) 設定個別使用者。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。
- 使用設定 API 和使用者活動記錄 AWS CloudTrail。如需有關使用 CloudTrail 追蹤擷取 AWS 活動的資訊，請參閱AWS CloudTrail 《使用者指南》中的[使用 CloudTrail 追蹤](#)。

- 使用 AWS 加密解決方案，以及其中的所有預設安全控制 AWS 服務。
- 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 AWS 透過命令列界面或 API 存取時需要 FIPS 140-3 驗證的密碼編譯模組，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 \(FIPS\) 140-3](#)。

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，例如名稱欄位。這包括當您使用 Amazon SWF 或使用主控台 AWS CLI、API 或 AWS SDKs 的其他 AWS 服務時。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 Simple Workflow Service 中的加密

靜態加密

Amazon SWF 一律會加密靜態資料。Amazon Simple Workflow Service 中的資料會使用透明的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 Transport Layer Security (TLS) 加密。

Amazon Simple Workflow Service 中的 Identity and Access Management

存取 Amazon SWF 需要 AWS 可用於驗證請求的登入資料。這些登入資料必須具有存取 AWS 資源的許可，例如從其他 AWS 資源擷取事件資料。下列各節提供如何使用 [AWS Identity and Access Management \(IAM\)](#) 和 Amazon SWF 的詳細資訊，透過控制資源的存取來協助保護您的資源。

AWS Identity and Access Management (IAM) 是一種 AWS 服務，可協助管理員安全地控制對 AWS 資源的存取。IAM 管理員可控制誰可以進行身分驗證（登入）和授權（具有許可）來使用 Amazon SWF 資源。IAM 是您可以免費使用 AWS 服務的。

主題

- [目標對象](#)

- [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 的政策條件索引鍵](#)
- [Amazon SWF 中的 ACLs](#)
- [ABAC 搭配 Amazon SWF](#)
- [搭配 Amazon SWF 使用臨時憑證](#)
- [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 的身分型政策](#)
- [Amazon SWF 內的資源型政策](#)
- [Amazon Simple Workflow Service 如何與 IAM 搭配使用](#)
- [Amazon Simple Workflow Service 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 Amazon VPC 端點](#)
- [對 Amazon Simple Workflow Service 身分和存取進行故障診斷](#)

目標對象

您的使用方式 AWS Identity and Access Management (IAM) 會根據您的角色而有所不同：

- 服務使用者 — 若無法存取某些功能，請向管理員申請所需許可 (請參閱 [對 Amazon Simple Workflow Service 身分和存取進行故障診斷](#))
- 服務管理員 — 負責設定使用者存取權並提交相關許可請求 (請參閱 [Amazon Simple Workflow Service 如何與 IAM 搭配使用](#))

- IAM 管理員 — 撰寫政策以管理存取控制 (請參閱 [Amazon Simple Workflow Service 的身分型政策範例](#))

使用身分驗證

身分驗證是您 AWS 使用身分憑證登入的方式。您必須以 AWS 帳戶根使用者、IAM 使用者或擔任 IAM 角色身分進行身分驗證。

您可以使用身分來源的登入資料，例如 AWS IAM Identity Center (IAM Identity Center)、單一登入身分驗證或 Google/Facebook 登入資料，以聯合身分的形式登入。如需有關登入的詳細資訊，請參閱《AWS 登入 使用者指南》中的 [如何登入您的 AWS 帳戶](#)。

對於程式設計存取，AWS 提供 SDK 和 CLI 以密碼編譯方式簽署請求。如需詳細資訊，請參閱《IAM 使用者指南》中的 [API 請求的 AWS 第 4 版簽署程序](#)。

AWS 帳戶 根使用者

當您建立時 AWS 帳戶，您會從一個名為 AWS 帳戶 theroot 使用者的登入身分開始，該身分具有對所有 AWS 服務和資源的完整存取權。強烈建議不要使用根使用者來執行日常任務。有關需要根使用者憑證的任務，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

最佳實務是要求人類使用者使用聯合身分提供者，以 AWS 服務使用臨時憑證存取。

聯合身分是來自您的企業目錄、Web 身分提供者的使用者，或使用來自身分來源的 AWS 服務憑證存取 Directory Service 的使用者。聯合身分會擔任角色，而該角色會提供臨時憑證。

若需集中化管理存取權限，建議使用 AWS IAM Identity Center。如需詳細資訊，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 https://docs.aws.amazon.com/IAM/latest/UserGuide/id_users.html 是一種身分具備單人或應用程式的特定許可權。建議以臨時憑證取代具備長期憑證的 IAM 使用者。如需詳細資訊，請參閱《IAM 使用者指南》中的 [要求人類使用者使用聯合身分提供者，以 AWS 使用臨時憑證存取](#)。

[IAM 群組](#) 會指定 IAM 使用者集合，使管理大量使用者的許可權更加輕鬆。如需詳細資訊，請參閱《IAM 使用者指南》中的 [IAM 使用者的使用案例](#)。

IAM 角色

IAM 角色 https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles.html 的身分具有特定許可權，其可以提供臨時憑證。您可以透過 [從使用者切換到 IAM 角色（主控台）](#) 或呼叫 AWS CLI 或 AWS API 操作來擔任角色。如需詳細資訊，請參閱《IAM 使用者指南》中的 [擔任角色的方法](#)。

IAM 角色適用於聯合身分使用者存取、臨時 IAM 使用者許可、跨帳戶存取權與跨服務存取，以及在 Amazon EC2 執行的應用程式。如需詳細資訊，請參閱《IAM 使用者指南》中的 [IAM 中的快帳戶資源存取](#)。

使用政策管理存取權

您可以透過建立政策並將其連接到身分或資源 AWS 來控制 AWS 中的存取。政策定義與身分或資源相關聯的許可。當委託人提出請求時 AWS，會評估這些政策。大多數政策會以 JSON 文件 AWS 形式存放在中。如需進一步了解 JSON 政策文件，請參閱《IAM 使用者指南》中的 [JSON 政策概觀](#)。

管理員會使用政策，透過定義哪些主體可在哪些條件下對哪些資源執行動作，以指定可存取的範圍。

預設情況下，使用者和角色沒有許可。IAM 管理員會建立 IAM 政策並將其新增至角色，供使用者後續擔任。IAM 政策定義動作的許可，無論採用何種方式執行。

身分型政策

身分型政策是附加至身分 (使用者、使用者群組或角色) 的 JSON 許可政策文件。這類政策控制身分可對哪些資源執行哪些動作，以及適用的條件。如需了解如何建立身分型政策，請參閱《IAM 使用者指南》中的 [透過客戶管理政策定義自訂 IAM 許可](#)。

身分型政策可分為內嵌政策 (直接內嵌於單一身分) 與受管政策 (可附加至多個身分的獨立政策)。如需了解如何在受管政策及內嵌政策之間做選擇，請參閱《IAM 使用者指南》中的 [在受管政策與內嵌政策之間選擇](#)。

資源型政策

資源型政策是附加到資源的 JSON 政策文件。範例包括 IAM 角色信任政策與 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。您必須在資源型政策中 [指定主體](#)。

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來自 IAM 的 AWS 受管政策。

其他政策類型

AWS 支援其他政策類型，可設定更多常見政策類型授予的最大許可：

- 許可界限 — 設定身分型政策可授與 IAM 實體的最大許可。如需詳細資訊，請參閱《IAM 使用者指南》中的 [IAM 實體許可界限](#)。
- 服務控制政策 (SCP) — 為 AWS Organizations 中的組織或組織單位指定最大許可。如需詳細資訊，請參閱《AWS Organizations 使用者指南》中的 [服務控制政策](#)。
- 資源控制政策 (RCP) — 設定您帳戶中資源可用許可的上限。如需詳細資訊，請參閱《AWS Organizations 使用者指南》中的 [資源控制政策 \(RCP\)](#)。
- 工作階段政策 — 在以程式設計方式為角色或聯合身分使用者建立臨時工作階段時，以參數形式傳遞的進階政策。如需詳細資訊，請參《IAM 使用者指南》中的 [工作階段政策](#)。

多種政策類型

當多種類型的政策適用於請求時，產生的許可會更複雜而無法理解。若要了解如何在涉及多個政策類型時 AWS 決定是否允許請求，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，但除非您具有許可，否則無法建立或存取 Amazon SWF 資源。例如，您必須具有叫用 AWS Lambda、Amazon Simple Notification Service (Amazon SNS) 和與 Amazon SWF 規則相關聯的 Amazon Simple Queue Service (Amazon SQS) 目標的許可。

下列各節說明如何管理 Amazon SWF 的許可。我們建議您先閱讀概觀。

- [基本原則](#)
- [Amazon SWF IAM 政策](#)
- [撰寫 Amazon SWF 的政策](#)

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。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 動作的清單，請參閱《服務授權參考》中的 [Amazon Simple Workflow Service 定義的資源](#)。

Amazon SWF 中的政策動作在動作之前使用下列字首：

```
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 身分型政策的範例，請參閱 [Amazon Simple Workflow Service 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。最佳實務是使用其 [Amazon Resource Name \(ARN\)](#) 來指定資源。若動作不支援資源層級許可，使用萬用字元 (*) 表示該陳述式適用於所有資源。

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

若要查看 Amazon SWF 資源類型及其 ARNs，請參閱《服務授權參考》中的 [Amazon Simple Workflow Service 定義的動作](#)。若要了解您可以使用哪些動作指定每個資源的 ARN，請參閱 [Amazon Simple Workflow Service 定義的資源](#)。

若要檢視 Amazon SWF 身分型政策的範例，請參閱 [Amazon Simple Workflow Service 的身分型政策範例](#)。

Amazon SWF 的政策條件索引鍵

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 元素會根據定義的條件，指定陳述式的執行時機。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。若要查看所有 AWS 全域條件索引鍵，請參閱《IAM 使用者指南》中的[AWS 全域條件內容索引鍵](#)。

若要查看 Amazon SWF 條件金鑰清單，請參閱《服務授權參考》中的 [Amazon Simple Workflow Service 的條件金鑰](#)。若要了解您可以使用條件金鑰的動作和資源，請參閱 [Amazon Simple Workflow Service 定義的資源](#)。

若要檢視 Amazon SWF 身分型政策的範例，請參閱 [Amazon Simple Workflow Service 的身分型政策範例](#)。

Amazon SWF 中的 ACLs

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。

ABAC 搭配 Amazon SWF

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，根據稱為標籤的屬性定義許可權。您可以將標籤連接至 IAM 實體 AWS 和資源，然後設計 ABAC 政策，以便在委託人的標籤符合資源上的標籤時允許操作。

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[使用 ABAC 授權定義許可](#)。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 使用臨時憑證

支援臨時憑證：是

臨時登入資料提供 AWS 資源的短期存取權，當您使用聯合或切換角色時，會自動建立。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 使用者指南》中的[IAM 中的臨時安全憑證與可與 IAM 搭配運作的 AWS 服務](#)。

Amazon SWF 的跨服務主體許可

支援轉寄存取工作階段 (FAS)：是

轉送存取工作階段 (FAS) 使用呼叫的委託人許可 AWS 服務，結合 AWS 服務請求向下游服務提出請求。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](#)。

Warning

變更服務角色的許可可能會中斷 Amazon SWF 功能。只有在 Amazon SWF 提供指引時，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

服務連結角色是連結至的一種服務角色 AWS 服務。服務可以擔任代表您執行動作的角色。服務連結角色會出現在您的 [中 AWS 帳戶](#)，並由服務擁有。IAM 管理員可以檢視，但不能編輯服務連結角色的許可。

如需建立或管理服务連結角色的詳細資訊，請參閱[可搭配 IAM 運作的 AWS 服務](#)。在資料表中尋找服務，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 的身分型政策

支援身分型政策：是

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。如需了解如何建立身分型政策，請參閱《IAM 使用者指南》中的[透過客戶管理政策定義自訂 IAM 許可](#)。

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。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 的身分型政策範例

若要檢視 Amazon SWF 身分型政策的範例，請參閱[Amazon Simple Workflow Service 的身分型政策範例](#)。

Amazon SWF 內的資源型政策

支援資源型政策：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](#)。委託人可以包括帳戶、使用者、角色、聯合身分使用者或 AWS 服務。

如需啟用跨帳戶存取權，您可以在其他帳戶內指定所有帳戶或 IAM 實體作為資源型政策的主體。如需詳細資訊，請參閱《IAM 使用者指南》中的[IAM 中的快帳戶資源存取](#)。

Amazon Simple Workflow Service 如何與 IAM 搭配使用

在您使用 IAM 管理對 Amazon SWF 的存取之前，請先了解哪些 IAM 功能可與 Amazon SWF 搭配使用。

您可以搭配 Amazon Simple Workflow Service 使用的 IAM 功能

IAM 功能	Amazon SWF 支援
身分型政策	是

IAM 功能	Amazon SWF 支援
資源型政策	否
政策動作	是
政策資源	是
政策條件索引鍵 (服務特定)	是
ACL	否
ABAC(政策中的標籤)	部分
臨時憑證	是
主體許可	是
服務角色	是
服務連結角色	否

若要全面了解 Amazon SWF 和其他 AWS 服務如何與大多數 IAM 功能搭配使用，請參閱 [《AWS IAM 使用者指南》](#) 中的 [與 IAM 搭配使用的服務](#)。

Amazon Simple Workflow Service 的身分型政策範例

根據預設，使用者和角色沒有建立或修改 Amazon SWF 資源的許可。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。

如需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 \(主控台\)](#)。

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 ARNs 格式，請參閱 [《服務授權參考》](#) 中的 [Amazon Simple Workflow Service 的動作、資源和條件金鑰](#)。

主題

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

政策最佳實務

身分型政策會判斷您帳戶中的某個人員是否可以建立、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 受管政策並邁向最低權限許可 – 若要開始將許可授予您的使用者和工作負載，請使用將許可授予許多常見使用案例的 AWS 受管政策。它們可在您的 中使用 AWS 帳戶。我們建議您定義特定於使用案例 AWS 的客戶受管政策，以進一步減少許可。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。如果透過特定 等使用服務動作 AWS 服務，您也可以使用條件來授予其存取權 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，可協助您撰寫安全且實用的政策。如需詳細資訊，請參閱《IAM 使用者指南》中的 [使用 IAM Access Analyzer 驗證政策](#)。
- 需要多重要素驗證 (MFA) – 如果您的案例需要 IAM 使用者或 中的根使用者 AWS 帳戶，請開啟 MFA 以提高安全性。如需在呼叫 API 操作時請求 MFA，請將 MFA 條件新增至您的政策。如需詳細資訊，請參閱《IAM 使用者指南》中的 [透過 MFA 的安全 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 主控台

若要存取 Amazon Simple Workflow Service 主控台，您必須擁有一組最低許可。這些許可必須允許您列出和檢視 中 Amazon SWF 資源的詳細資訊 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。

對於僅呼叫 AWS CLI 或 AWS API 的使用者，您不需要允許最低主控台許可。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。

為了確保使用者和角色仍然可以使用 Amazon SWF 主控台，請將 Amazon SWF *ConsoleAccess* 或 *ReadOnly* AWS 受管政策連接到實體。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。此政策包含在主控台或使用或 AWS CLI AWS API 以程式設計方式完成此動作的許可。

```
{
  "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 存取控制主要基於兩種類型的許可：

- 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 許可：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，或呼叫任何網域中的任何 Amazon SWF 動作，或完全拒絕存取權。不過，IAM 支援更精細的存取控制方法，通常更實用。例如，您可以：

- 允許使用者在沒有限制的情況下呼叫任何 Amazon SWF 動作，但僅限於指定的網域。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。
- 存取控制模型會使用預設拒絕政策；任何未明確允許的存取權皆予以拒絕。
- 您可以將適當的 IAM 政策連接至工作流程的演員，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。
- 如果您授予 [RespondDecisionTaskCompleted](#) 的使用許可，則可以表達該動作中所含決策清單的許可。

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。如需元素的完整清單，以及如何建構政策的一般討論，請參閱 [存取政策語言](#)。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 預設拒絕存取。

資源

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

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。若要表達網域的許可，請將 Resource 設定為網域的 Amazon Resource Name (ARN)，其格式為 "arn:aws:swf:Region:AccountID:/domain/DomainName"。## 是 AWS 區域，AccountID 是沒有破折號的帳戶 ID，而 DomainName 是網域名稱。

Action

(必要) 套用陳述式的動作，您可以使用下列格式參照：*serviceId:action*。對於 Amazon SWF，將 *serviceID* 設定為 swf。例如，swf:StartWorkflowExecution 參照 [StartWorkflowExecution](#) 動作，並且用來控制允許啟動工作流程的使用者。

如果您授予 [RespondDecisionTaskCompleted](#) 的使用許可，則也可以控制所含決策清單的存取權，方法是使用 Action 來表達虛擬 API 的許可。由於 IAM 預設拒絕存取，因此必須明確允許決策者的決策，否則將不會接受決策者決策。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 動作通常範圍廣泛，您可以使用 IAM 條件來減少。例如，若要限制允許 [PollForActivityTask](#) 動作存取的任務清單，請包含 Condition 並使用 swf:taskList.name 索引鍵指定允許清單。

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 `StringEquals`，例如，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所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，您可能想要允許決策者指定一組 [RespondDecisionTaskCompleted](#) 決策，但也允許決策者只指定其中一個決策來進行任何特定呼叫。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](#)。

撰寫 Amazon SWF 的政策

工作流程由多個演員組成：活動、決策者等。您可以連接適當的 IAM 政策來控制每個演員的存取。

透過下列動作，將授予演員所有區域的完整帳戶存取權：

- 動作：`swf:*`
- 資源：`arn:aws:swf:*:123456789012:/domain/*`

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 (*) 表示資源許可適用於所有區域。

若要限制單一區域的許可，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](#)

網域許可

若要將部門的工作流程限制在特定網域，您可以授予許可，允許演員呼叫任何動作，但僅限特定部門。

若要授予演員對多個網域的存取權，請將每個網域的許可表達為陳述式清單：

- 動作：swf:*
- 資源：arn:aws:swf:*:123456789012:/domain/department1
- 資源：arn:aws:swf:*:123456789012:/domain/department2

您可以允許演員在 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

您可以在 Action 元素中指定動作，以控制演員可以使用的動作。

透過下列動作，演員只能呼叫 StartWorkflowExecution 來啟動工作流程。它無法使用任何其他動作。

- 動作：swf:StartWorkflowExecution

條件

您可以使用 Condition 元素，選擇性地限制動作的允許參數值。

若要限制演員可以啟動哪些工作流程，請限制一或多個 StartWorkflowExecution 參數值，如下所示：

```
"Condition" : {
  "StringEquals" : {
    "swf:workflowType.name" : "workflow1",
    "swf:workflowType.version" : "version2"
  }
}
```

具有先前限制條件的演員只能執行 version2，workflow1 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：

```
"Condition" : {
  "StringEqualsIfExists" : { "swf:taskList.name" : "task_list_name" }
}
```

具有先前政策的演員可以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。每個標籤都有單獨的索引鍵，因此您可以使用 `swf:tagList.member.0` 來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，上限為 5。

您必須小心限制標籤清單的方式。例如，不建議下列條件。

不建議下列條件，因為它可讓您選擇性地指定 `some_ok_tag` 或 `another_ok_tag`。不過，條件只會限制標籤清單的第一個元素。清單可能會有其他元素，其中包含允許的所有任意值，因為條件不會將任何條件套用至 `swf:tagList.member.1`、`swf:tagList.member.2` 等。

```
// Example to illustrate an insecure Condition
"Condition" : {
  "StringEqualsIfExists" : {
    "swf:tagList.member.0" : "some_ok_tag", "another_ok_tag"
  }
}
```

解決上一個問題的一種方法是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`。

```
"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

若要限制 可用的決策 `RespondDecisionTaskCompleted`，您必須先允許演員呼叫 `RespondDecisionTaskCompleted`。然後，您可以使用與一般 API 相同的語法來表達適當虛擬 API 成員的許可，如下所示：

- 陳述式 1

資源 : `arn:aws:swf:*:123456789012:/domain/*`

動作 : `swf:RespondDecisionTaskCompleted`

- 陳述式 2

資源 : *

動作 : `swf:ScheduleActivityTask`

條件 : `"StringEquals" : { "swf:activityType.name" : "SomeActivityType" }`

第一個Statement允許演員呼叫 `RespondDecisionTaskCompleted`。第二個陳述式允許演員使用 `ScheduleActivityTask` 決定指示 Amazon SWF 排程活動任務。若要允許所有決策，請將 `"swf:ScheduleActivityTask"` 取代為 `"swf:*"`。

您可以使用 `Condition` 運算子限制參數，如同一般 API。上一個範例中的 `StringEquals` 運算子 `Condition` 允許 `RespondDecisionTaskCompleted` 排程活動的活動任務 `SomeActivityType`，而且必須排程該任務。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 受管政策 : `SimpleWorkflowFullAccess`

您可將 `SimpleWorkflowFullAccess` 政策連接到 IAM 身分。

此政策提供 Amazon SWF 組態服務的完整存取權。

IAM 政策的服務模型限制

建立 IAM 政策時，您必須考慮服務模型限制條件。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然會失敗，因為它是無效的請求。

例如，Amazon SWF 服務模型不允許在相同的 [ListOpenWorkflowExecutions](#) 請求中使用 `typeFilter` 和 `tagFilter` 參數。下列條件允許服務透過擲回拒絕的呼叫 `ValidationException` 做為無效的請求：

```
"Condition" : {
  "StringEquals" : {
```

```
"swf:typeFilter.name" : "workflow_name",
"swf:typeFilter.version" : "workflow_version",
"swf:tagFilter.tag" : "some_tag"
}
}
```

API 摘要

本節簡短說明如何使用 IAM 政策來控制演員如何使用每個 API 和虛擬 API 來存取 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 – 字串限制條件。索引鍵為 swf:tagFilter.tag
- typeFilter.name – 字串限制條件。金鑰為 swf:typeFilter.name。
- typeFilter.version – 字串限制條件。金鑰為 swf:typeFilter.version。

Note

CountClosedWorkflowExecutions 需要 typeFilter 和 tagFilter 互斥。

[CountOpenWorkflowExecutions](#)

- `tagFilter.tag` – 字串限制條件。索引鍵為 `swf:tagFilter.tag`
- `typeFilter.name` – 字串限制條件。金鑰為 `swf:typeFilter.name`。
- `typeFilter.version` – 字串限制條件。金鑰為 `swf:typeFilter.version`。

 Note

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

[CountPendingActivityTasks](#)

- `taskList.name` – 字串限制條件。金鑰為 `swf:taskList.name`。

[CountPendingDecisionTasks](#)

- `taskList.name` – 字串限制條件。金鑰為 `swf:taskList.name`。

[DeleteActivityType](#)

- `activityType.name` – 字串限制條件。金鑰為 `swf:activityType.name`。
- `activityType.version` – 字串限制條件。金鑰為 `swf:activityType.version`。

[DeprecateActivityType](#)

- `activityType.name` – 字串限制條件。金鑰為 `swf:activityType.name`。
- `activityType.version` – 字串限制條件。金鑰為 `swf:activityType.version`。

[DeprecateDomain](#)

- 您無法限制此動作的參數。

[DeleteWorkflowType](#)

- `workflowType.name` – 字串限制條件。金鑰為 `swf:workflowType.name`。
- `workflowType.version` – 字串限制條件。金鑰為 `swf:workflowType.version`。

[DeprecateWorkflowType](#)

- `workflowType.name` – 字串限制條件。金鑰為 `swf:workflowType.name`。
- `workflowType.version` – 字串限制條件。金鑰為 `swf:workflowType.version`。

[DescribeActivityType](#)

- `activityType.name` – 字串限制條件。金鑰為 `swf:activityType.name`。
- `activityType.version` – 字串限制條件。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` – 字串限制條件。金鑰為 `swf:workflowType.name`。
- `workflowType.version` – 字串限制條件。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` – 字串限制條件。索引鍵為 `swf:tagFilter.tag`
- `typeFilter.name` – 字串限制條件。金鑰為 `swf:typeFilter.name`。
- `typeFilter.version` – 字串限制條件。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 – 字串限制條件。索引鍵為 swf:tagFilter.tag
- typeFilter.name – 字串限制條件。金鑰為 swf:typeFilter.name。
- typeFilter.version – 字串限制條件。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 – 字串限制條件。金鑰為 swf:taskList.name。

PollForDecisionTask

- taskList.name – 字串限制條件。金鑰為 swf:taskList.name。

RecordActivityTaskHeartbeat

- 您無法限制此動作的參數。

RegisterActivityType

- `defaultTaskList.name` – 字串限制條件。金鑰為 `swf:defaultTaskList.name`。
- `name` – 字串限制條件。金鑰為 `swf:name`。
- `version` – 字串限制條件。金鑰為 `swf:version`。

RegisterDomain

- `name` – 要註冊的網域名稱可作為此動作的資源。

RegisterWorkflowType

- `defaultTaskList.name` – 字串限制條件。金鑰為 `swf:defaultTaskList.name`。
- `name` – 字串限制條件。金鑰為 `swf:name`。
- `version` – 字串限制條件。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` – 透過虛擬 API 許可間接限制。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` – 字串限制條件。索引鍵為 `swf:tagList.member.0`
- `tagList.member.1` – 字串限制條件。索引鍵為 `swf:tagList.member.1`
- `tagList.member.2` – 字串限制條件。索引鍵為 `swf:tagList.member.2`
- `tagList.member.3` – 字串限制條件。索引鍵為 `swf:tagList.member.3`
- `tagList.member.4` – 字串限制條件。索引鍵為 `swf:tagList.member.4`
- `taskList.name` – 字串限制條件。金鑰為 `swf:taskList.name`。
- `workflowType.name` – 字串限制條件。金鑰為 `swf:workflowType.name`。
- `workflowType.version` – 字串限制條件。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` – 字串限制條件。索引鍵為 `swf:tagList.member.0`
- `tagList.member.1` – 字串限制條件。索引鍵為 `swf:tagList.member.1`
- `tagList.member.2` – 字串限制條件。索引鍵為 `swf:tagList.member.2`
- `tagList.member.3` – 字串限制條件。索引鍵為 `swf:tagList.member.3`
- `tagList.member.4` – 字串限制條件。索引鍵為 `swf:tagList.member.4`
- `taskList.name` – 字串限制條件。金鑰為 `swf:taskList.name`。
- `workflowTypeVersion` – 字串限制條件。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` – 字串限制條件。金鑰為 `swf:activityType.name`。
- `activityType.version` – 字串限制條件。金鑰為 `swf:activityType.version`。
- `taskList.name` – 字串限制條件。金鑰為 `swf:taskList.name`。

SignalExternalWorkflowExecution

- 您無法限制此動作的參數。

StartChildWorkflowExecution

- `tagList.member.0` – 字串限制條件。索引鍵為 `swf:tagList.member.0`
- `tagList.member.1` – 字串限制條件。索引鍵為 `swf:tagList.member.1`
- `tagList.member.2` – 字串限制條件。索引鍵為 `swf:tagList.member.2`
- `tagList.member.3` – 字串限制條件。索引鍵為 `swf:tagList.member.3`
- `tagList.member.4` – 字串限制條件。索引鍵為 `swf:tagList.member.4`
- `taskList.name` – 字串限制條件。金鑰為 `swf:taskList.name`。
- `workflowType.name` – 字串限制條件。金鑰為 `swf:workflowType.name`。
- `workflowType.version` – 字串限制條件。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 Amazon SWF 網域，其中包含具有金鑰 environment 和 值 production 的標籤，條件如下：

```
"Condition": {
  "StringEquals": {"aws:ResourceTag/environment": "production"}
}
```

如需標記的詳細資訊，請參閱：

- [Amazon SWF 中的標籤](#)
- [使用 IAM 標籤控制存取](#)

Amazon SWF 的 Amazon VPC 端點

Note

AWS PrivateLink 目前僅在 AWS 最高機密 - 東部、AWS 機密區域和中國區域提供支援。

如果您使用 Amazon Virtual Private Cloud (Amazon VPC) 託管 AWS 資源，您可以在 Amazon VPC 和 Amazon Simple Workflow Service 工作流程之間建立連線。您可以將此連線與 Amazon SWF 工作流程搭配使用，而無需跨公有網際網路。

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。如需有關 Amazon VPC 的詳細資訊，請參閱 [《Amazon VPC 使用者指南》](#)。

若要將 Amazon VPC 連線至 Amazon SWF，您必須先定義介面 VPC 端點，以將您的 VPC 連線至其他端點 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 `awscli` (AWS CLI) AWS 管理主控台、AWS SDK、Amazon SWF API 或在 AWS Command Line Interface VPC 中建立 Amazon SWF 端點 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

當您建立端點時，請將 Amazon SWF 指定為您希望 VPC 連線的服務。在 Amazon VPC 主控台中 AWS，服務名稱會根據區域而有所不同。例如，在 AWS 最高機密 - 東部區域中，Amazon SWF 的服務名稱為 `com.amazonaws.us-iso-east-1.swf`。

如需有關使用 建立和設定端點的資訊 CloudFormation，請參閱 CloudFormation 《使用者指南》中的[AWS::EC2::VPCEndpoint](#) 資源。

Amazon VPC 端點政策

若要控制 Amazon SWF 的連線存取，您可以在建立 Amazon VPC 端點時連接 AWS Identity and Access Management (IAM) 端點政策。您可以連接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 Amazon Virtual Private Cloud 端點政策 Amazon SWF](#)
- [使用 VPC 端點控制對服務的存取](#)

Amazon SWF 的 Amazon Virtual Private Cloud 端點政策 Amazon SWF

您可以為 Amazon SWF 建立 Amazon VPC 端點政策，並在其中指定下列項目：

- 可執行動作的委託人。
- 可執行的動作。
- 可供執行動作的資源。

下列範例會將特定 IAM 角色新增至政策：

```
"Principal": {
  "AWS": "arn:aws:iam::123456789012:role/MyRole"
}
```

- 如需建立端點政策的詳細資訊，請參閱[使用 VPC 端點控制對服務的存取](#)。
- 如需有關如何使用 IAM 控制對 AWS 和 Amazon SWF 資源的存取的資訊，請參閱[Amazon Simple Workflow Service 中的 Identity and Access Management](#)。

對 Amazon Simple Workflow Service 身分和存取進行故障診斷

使用以下資訊來協助您診斷和修正使用 Amazon SWF 和 IAM 時可能遇到的常見問題。

主題

- [我無權在 Amazon SWF 中執行動作](#)
- [我未獲得執行 iam:PassRole 的授權](#)
- [我想要允許以外的人員 AWS 帳戶存取我的 Amazon SWF 資源](#)

我無權在 Amazon SWF 中執行動作

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 管理員。您的管理員提供您的簽署憑證。

我未獲得執行 iam:PassRole 的授權

如果您收到錯誤，告知您無權執行 `iam:PassRole` 動作，您的政策必須更新，以允許您將角色傳遞給 Amazon SWF。

有些 AWS 服務 可讓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 marymajor 使用者嘗試使用主控台在 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 沒有將角色傳遞給服務的許可。

```
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 管理員。您的管理員提供您的簽署憑證。

我想要允許以外的人員 AWS 帳戶 存取我的 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 Simple Workflow Service 如何與 IAM 搭配使用](#)。
- 若要了解如何 AWS 帳戶 在您擁有的 資源之間提供存取權，請參閱 [《IAM 使用者指南》中的在您擁有 AWS 帳戶 的另一個 IAM 使用者中提供存取權](#)。
- 若要了解如何將資源的存取權提供給第三方 AWS 帳戶，請參閱 [《IAM 使用者指南》中的將存取權提供給第三方 AWS 帳戶 擁有](#)。
- 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 \(聯合身分\)](#)。
- 如需了解使用角色和資源型政策進行跨帳戶存取之間的差異，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

- [適用於 CloudWatch 的 Amazon SWF 指標](#)

- [使用 檢視 CloudWatch 的 Amazon SWF 指標 AWS 管理主控台](#)
- [使用 記錄 API 呼叫 AWS CloudTrail](#)
- [Amazon SWF 執行狀態變更的 EventBridge](#)
- [AWS 使用者通知 搭配 Amazon Simple Workflow Service 使用](#)

適用於 CloudWatch 的 Amazon SWF 指標

Amazon SWF 現在提供 CloudWatch 的指標，可用來追蹤工作流程和活動，並針對您選擇的閾值設定警示。您可以使用 [檢視指標 AWS 管理主控台](#)。如需詳細資訊，請參閱 [使用 檢視 CloudWatch 的 Amazon SWF 指標 AWS 管理主控台](#)。

主題

- [Amazon SWF 指標的報告單位](#)
- [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 指標的報告單位

報告時間間隔的指標

CloudWatch 的部分 Amazon SWF 指標是時間間隔，一律以毫秒為單位。CloudWatch 單位報告為 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 逾時類型](#)」。

報告計數的指標

CloudWatch 報告結果的一些 Amazon SWF 指標以計數表示。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。

在 CloudWatch Count 中報告的一些 CloudWatch Amazon SWF 指標是每秒計數。例如，在 CloudWatch Count 中報告為 ProvisionedRefillRate 的代表每秒請求 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 CloudWatch 中同時監控 API 和決策事件，以深入了解您的用量和容量。請參閱 [Amazon SWF 中的基本工作流程概念](#) 一節中的 [決策者](#)，以及 [Amazon Simple Workflow Service 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 限流限制時發出警示。如需這些限制和其預設設定的描述，請參閱「[Amazon SWF 限流配額](#)」。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「[???](#)」。

最佳實務是，您應該將 CloudWatch 警示設定為大約 60% 的 API 或決策事件容量。這可讓您在啟用 Amazon SWF 限流之前調整工作流程或請求提高服務限制。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

下列指標適用於 Amazon SWF：

指標	Description
DecisionTaskScheduleToStartTime	<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>

指標	Description
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> <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>

指標	Description
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>
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>

指標	Description
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>
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>

指標	Description
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>
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>

指標	Description
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>
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>

指標	Description
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>
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>已耗用的一般限制數量。</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>

指標	Description
ProvisionedLimit	佈建到帳戶的一般限制量。 維度：GeneralLimitType
維度	Description
Domain	篩選資料至工作流程或活動執行所在的 Amazon SWF 網域。
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	篩選資料至任務清單的分類。決策任務清單的值為「D」，活動任務清單的值為「A」。
ThrottlingScope	將資料篩選至指定的限流範圍。超過帳戶層級配額時的值為「帳戶」，超過工作流程層級配額時的值為「工作流程」。

Amazon SWF 非 ASCII 資源名稱和 CloudWatch 維度

Amazon SWF 允許資源名稱中的非 ASCII 字元，例如 TaskList 和 DomainName。不過，CloudWatch 指標的維度值只能包含可列印的 ASCII 字元。為了確保 Amazon SWF 使用與 [CloudWatch 要求](#) 相容的維度值，會轉換不符合這些要求的 Amazon SWF 資源名稱，並附加檢查總和，如下所示：

- 任何非 ASCII 字元都會取代為 ?。

- 輸入字串或轉換後的字串將視需要截斷。這可確保附加檢查總和時，新的字串長度不會超過 CloudWatch 上限。
- 由於任何非 ASCII 字元都會轉換為？，因此一些轉換前不同的 CloudWatch 指標維度值在轉換後可能看起來相同。為了協助區分它們，原始資源名稱 SHA256 檢查總和的前 16 個字元後面加上底線 () 會附加至資源名稱。

轉換範例：

- test apple 會轉換為 test ?pple_82cc5b8e3a771d12
- ààà 會轉換為 ???_2fec5edbb2c05c22。
- TaskList 名稱 àpplé 和 âpplè 都會轉換為 ?pp1?，並且會是相同的。附加檢查總和會傳回不同的值，?pp1?_f39a36df9d85a69d 以及 ?pp1?_da3efb4f11dd0f7f。

Tip

您可以產生自己的 SHA256 檢查總和。例如，若要使用 shasum 命令列工具：
echo -n "<the original resource name>" | shasum -a 256 | cut -c1-16

使用 檢視 CloudWatch 的 Amazon SWF 指標 AWS 管理主控台

Amazon CloudWatch 為 Amazon SWF 工作流程和活動提供許多可檢視的指標。您可以使用 檢視指標，並設定 Amazon SWF 工作流程執行的警示 [AWS 管理主控台](#)。「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「[適用於 CloudWatch 的 Amazon SWF 指標](#)」。

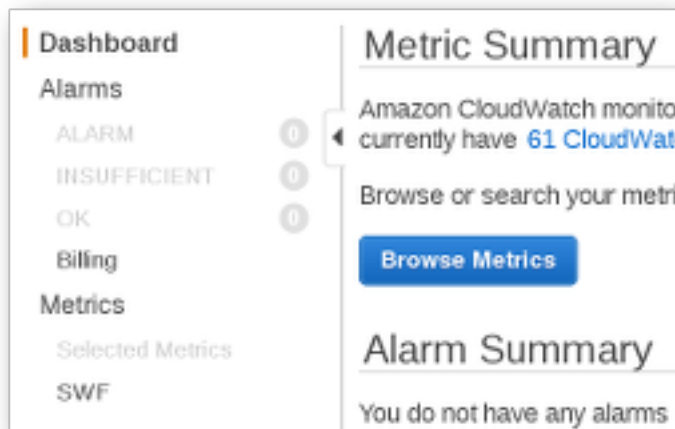
主題

- [檢視 指標](#)
- [設定警示](#)

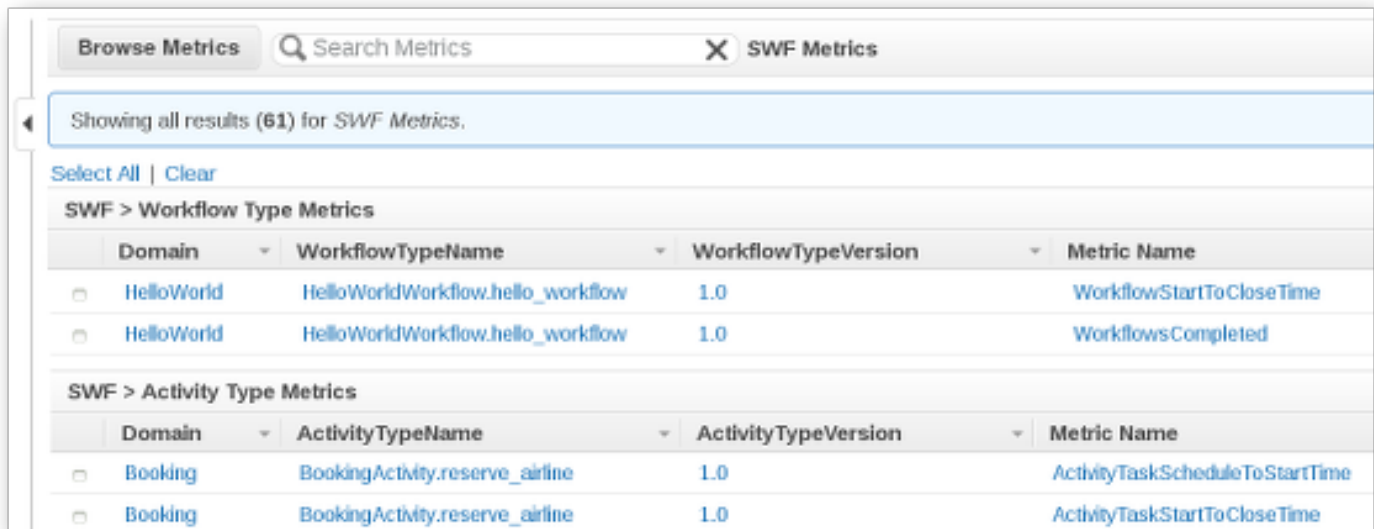
檢視 指標

檢視 Amazon SWF 的指標

1. 登入 AWS 管理主控台，並在 <https://console.aws.amazon.com/cloudwatch/> 開啟 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：Workflow Type Metrics (工作流程類型指標) 和 Activity Type Metrics (活動類型指標)。



Browse Metrics Search Metrics SWF Metrics

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	ActivityTask.ScheduleToStartTime
Booking	BookingActivity.reserve_airline	1.0	ActivityTask.StartToCloseTime

Note

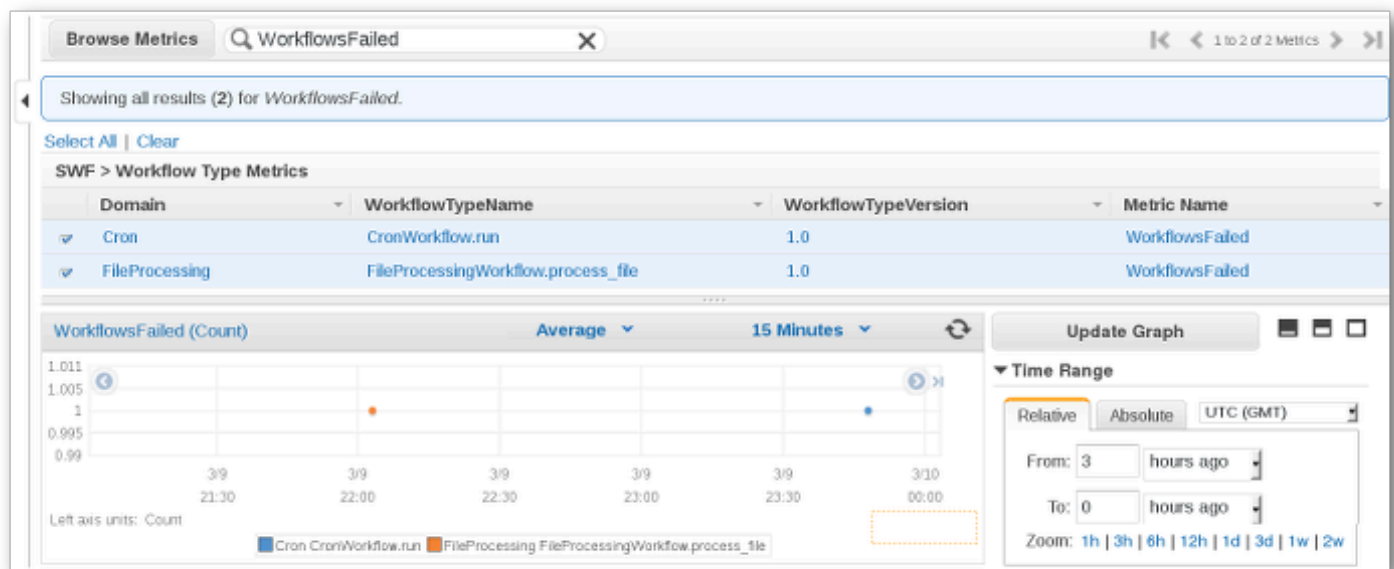
一開始，您可能只會看到 Workflow Type Metrics (工作流程類型指標)；Activity Type Metrics (活動類型指標) 會呈現在相同的檢視中，但需要您向下捲動才能看到。

一次最多顯示 50 個最新指標 (區分工作流程和活動指標)。

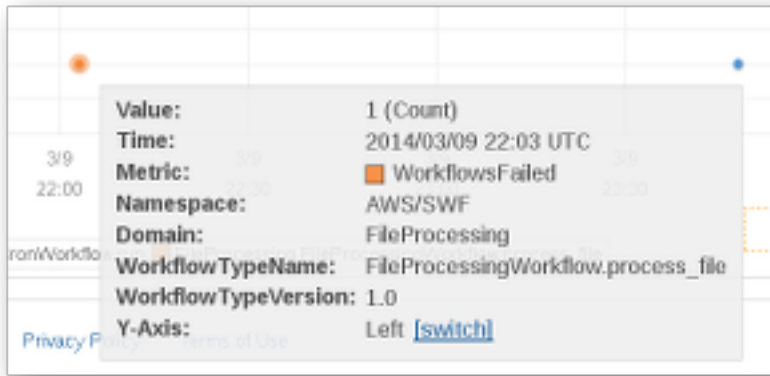
您可以使用清單中每欄上方的互動式標題，透過任何提供的維度來排序指標。針對工作流程，維度是 Domain (網域)、WorkflowTypeName (工作流程類型名稱)、WorkflowTypeVersion (工作流程類型版本) 和 Metric Name (指標名稱)。針對活動，維度是 Domain (網域)、ActivityTypeName (活動類型名稱)、ActivityTypeVersion (活動類型版本) 和 Metric Name (指標名稱)。

「[適用於 CloudWatch 的 Amazon SWF 指標](#)」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，請參閱《Amazon CloudWatch 使用者指南》中的[檢視、繪製和發佈指標](#)。

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。

1. Select Metric
2. Define Alarm

Back Next
Cancel

Please set the alarm threshold, actions and click **Create Alarm** below.

Create Alarm

Alarm Threshold

Provide the details and threshold for your alarm. Use the graph on the right to help set the appropriate threshold.

Name:
Description:

Whenever: WorkflowsFailed
is:
for: consecutive period(s)

Actions

Define what actions are taken when your alarm changes state.

Notification Delete

Whenever this alarm:
Send notification to: [New list](#)
Email list:

+ Notification + AutoScaling Action + EC2 Action

如需設定和使用 CloudWatch 警示的詳細資訊，請參閱 [《Amazon CloudWatch 使用者指南》](#) 中的 [建立 Amazon CloudWatch 警示](#)。Amazon CloudWatch

使用 記錄 API 呼叫 AWS CloudTrail

Amazon Simple Workflow Service 已與 整合 [AWS CloudTrail](#)，這項服務可提供使用者、角色或所採取動作的記錄 AWS 服務。CloudTrail 會將 Amazon SWF 的所有 API 呼叫擷取為事件。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。您可以使用 CloudTrail 所收集的資訊，判斷對 Amazon SWF 提出的請求、提出請求的 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 Identity Center 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。

當您建立帳戶 AWS 帳戶 時 CloudTrail 會在中處於作用中狀態，而且您會自動存取 CloudTrail 事件歷史記錄。CloudTrail 事件歷史記錄為 AWS 區域中過去 90 天記錄的管理事件，提供可檢視、可搜尋、可下載且不可變的記錄。如需詳細資訊，請參閱「AWS CloudTrail 使用者指南」中的[使用 CloudTrail 事件歷史記錄](#)。檢視事件歷史記錄不會產生 CloudTrail 費用。

如需 AWS 帳戶 過去 90 天內持續記錄的事件，請建立線索或 [CloudTrail Lake](#) 事件資料存放區。

CloudTrail 追蹤

線索能讓 CloudTrail 將日誌檔案交付至 Amazon S3 儲存貯體。使用 建立的所有線索 AWS 管理主控台 都是多區域。您可以使用 AWS CLI 建立單一或多區域追蹤。建議您建立多區域追蹤，因為您擷取 AWS 區域 帳戶中所有的活動。如果您建立單一區域追蹤，您只能檢視追蹤 AWS 區域中記錄的事件。如需追蹤的詳細資訊，請參閱《AWS CloudTrail 使用者指南》中的[為您的 AWS 帳戶建立追蹤](#)和[為組織建立追蹤](#)。

您可以透過建立追蹤，免費將持續管理事件的一個複本從 CloudTrail 傳遞至您的 Amazon S3 儲存貯體，但這樣做會產生 Amazon S3 儲存費用。如需 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 Lake 事件資料存放區

CloudTrail Lake 讓您能夠對事件執行 SQL 型查詢。CloudTrail Lake 會將分列式 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。如需 CloudTrail Lake 的詳細資訊，請參閱 AWS CloudTrail 《使用者指南》中的[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 CloudTrail 定價的詳細資訊，請參閱 [AWS CloudTrail 定價](#)。

CloudTrail 中的資料事件

[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。根據預設，CloudTrail 不會記錄資料事件。CloudTrail 事件歷史記錄不會記錄資料事件。

資料事件需支付額外的費用。如需 CloudTrail 定價的詳細資訊，請參閱 [AWS CloudTrail 定價](#)。

您可以使用 CloudTrail 主控台 AWS CLI 或 CloudTrail API 操作，記錄 Amazon SWF 資源類型的資料事件。如需如何記錄資料事件的詳細資訊，請參閱《AWS CloudTrail 使用者指南》中的 [使用 AWS 管理主控台記錄資料事件](#) 和 [使用 AWS Command Line Interface 記錄資料事件](#)。

下表列出您可以記錄資料事件的 Amazon SWF 資源類型。資料事件類型欄顯示從 CloudTrail 主控台上的資料事件類型清單中選擇的值。resources.type 值欄會顯示值，您會在使用 AWS CLI 或 CloudTrail APIs 設定進階事件選取器時指定此 resources.type 值。記錄到 CloudTrail 的資料 API 資料行會針對資源類型顯示記錄到 CloudTrail 的 API 呼叫。

您可以設定進階事件選取器來篩選 eventName、readOnly 和 resources.ARN 欄位，以僅記錄對您重要的事件。如需這些欄位的詳細資訊，請參閱 AWS CloudTrail API 參考中的 [AdvancedFieldSelector](#)。

資料事件類型	resources.type 值	記錄到 CloudTrail 的資料 API
SWF 網域	AWS::SWF::Domain	工作流程事件 <ul style="list-style-type: none"> • CountClosedWorkflowExecutions • CountOpenWorkflowExecutions • DescribeWorkflowExecution • ListClosedWorkflowExecutions • ListOpenWorkflowExecutions • GetWorkflowExecutionHistory • RequestCancelWorkflowExecution • SignalWorkflowExecution • StartWorkflowExecution • TerminateWorkflowExecution 任務事件

資料事件類型	resources.type 值	記錄到 CloudTrail 的資料 API
		<ul style="list-style-type: none"> • CountPendingActivityTasks • PollForDecisionTask • PollForActivityTask • RecordActivityTaskHeartbeat • RespondActivityTaskCanceled • RespondActivityTaskCompleted • RespondActivityTaskFailed • RespondDecisionTaskCompleted <p>決策事件</p> <ul style="list-style-type: none"> • CancelTimer • CancelWorkflowExecution • CompleteWorkflowExecution • ContinueAsNewWorkflowExecution • FailWorkflowExecution • RecordMarker • RequestCancelActivityTask • RequestCancelExternalWorkflowExecution • ScheduleActivityTask • ScheduleLambdaFunction • SignalExternalWorkflowExecution

資料事件類型	resources.type 值	記錄到 CloudTrail 的資料 API
		<ul style="list-style-type: none"> • StartChildWorkflow Execution • StartTimer

CloudTrail 事件和 RespondDecisionTaskCompleted

[RespondDecisionTaskCompleted](#) 動作會取得請求承載中的決策清單。已完成的呼叫會發出 N +1 CloudTrail 資料事件，每個決策一個，API 呼叫本身一個。資料事件和 API 事件都會有相同的請求 ID。

CloudTrail 中的管理事件

[管理事件](#)提供有關在資源上執行的管理操作的資訊 AWS 帳戶。這些也稱為控制平面操作。根據預設，CloudTrail 記錄管理事件。

Amazon Simple Workflow Service 會將下列控制平面操作記錄到 CloudTrail 做為管理事件。

網域事件

- [RegisterDomain](#)
- [DescribeDomain](#)
- [ListDomains](#)
- [DeprecateDomain](#)
- [UndeprecateDomain](#)

活動事件

- [RegisterActivityType](#)
- [DescribeActivityType](#)
- [ListActivityTypes](#)
- [DeprecateActivityType](#)
- [UndeprecateActivityType](#)
- [DeleteActivityType](#)

WorkflowType 事件

- [RegisterWorkflowType](#)
- [DescribeWorkflowType](#)
- [ListWorkflowTypes](#)
- [DeprecateWorkflowType](#)
- [UndeprecateWorkflowType](#)
- [DeleteWorkflowType](#)

標籤事件

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

範例事件

一個事件代表任何來源提出的單一請求，並包含請求 API 操作的相關資訊、操作的日期和時間、請求參數等。CloudTrail 日誌檔案不是公有 API 呼叫的已排序堆疊追蹤，因此事件不會以任何特定順序顯示。

以下範例顯示的 CloudTrail 事件會示範 CountClosedWorkflowExecutions 操作。

```
{
  "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"
  }
}
```

如需有關 CloudTrail 記錄內容的資訊，請參閱《AWS CloudTrail 使用者指南》中的 [CloudTrail record contents](#)。

Amazon SWF 執行狀態變更的 EventBridge

您可以使用 Amazon EventBridge 來回應 AWS 資源中的狀態變更或事件。當 Amazon SWF 發出事件時，一律會前往您帳戶的預設 EventBridge 事件匯流排。您可以建立事件的規則，將其與預設事件匯流排建立關聯，並指定 EventBridge 收到符合規則的事件時要採取的目標動作。如此一來，您就可以監控工作流程，而不必持續使用 [GetWorkflowExecutionHistory](#) API 輪詢。根據工作流程執行中的變更，您可以使用 EventBridge 目標來呼叫 AWS Lambda 函數、將訊息發佈至 Amazon Simple Notification Service (Amazon SNS) 主題等。

您可以使用 [查看執行狀態變更事件的完整內容](#) [DescribeWorkflowExecution](#)。

如需詳細資訊，請參閱「[Amazon EventBridge 使用者指南](#)」。

EventBridge 事件

歷史記錄事件類型包含執行狀態變更。每個事件的 detail 區段至少包含下列參數：

- `eventId` : `GetWorkflowExecutionHistory` 顯示的事件 ID。
- `workflowExecutionDetail` : 發出事件時的工作流程狀態。
- `eventType` : 歷史記錄事件類型，下列其中一項：
 - `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 將事件傳送至 EventBridge 的範例：

主題

- [執行已開始](#)
- [執行完成](#)
- [執行失敗](#)
- [執行逾時](#)
- [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。executionStatus 欄位指出傳送事件時執行的狀態，可以是 OPEN 或 CLOSED。

執行已開始

```
{
  "version": "0",
  "id": "44444444444444",
  "detail-type": "Simple Workflow Execution State Change",
  "source": "aws.swf",
  "account": "44444444444444",
  "time": "2020-05-08T15:57:38Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:swf:us-east-1:44444444444444:/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": "44444444444444"
        },
        "taskPriority": null,
        "childPolicy": "ABANDON",
        "lambdaRole": "arn:aws:iam::44444444444444: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,
}
}
```

執行逾時

```
{
  "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,
}
}
```

AWS 使用者通知 搭配 Amazon Simple Workflow Service 使用

您可以使用 [AWS 使用者通知](#) 設定交付管道，以收到有關 Amazon Simple Workflow Service 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的通知，包括電子郵件、[聊天應用程式中的 Amazon Q Developer](#) 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 Simple Workflow Service 的合規驗證

在多個合規計畫中，第三方稽核人員會評估 Amazon Simple Workflow Service 的安全性和 AWS 合規性。這些計畫包括 SOC、PCI、FedRAMP、HIPAA 等等。

如需特定合規計畫範圍內 AWS 的服務清單，請參閱 [AWS 合規計畫範圍內的服務](#)。如需一般資訊，請參閱 [AWS Compliance Programs](#)。

您可以使用 下載第三方稽核報告 AWS Artifact。如需詳細資訊，請參閱在 [AWS Artifact 中下載報告](#) 在。

您使用 Amazon SWF 時的合規責任取決於資料的機密性、您公司的合規目標，以及適用的法律和法規。AWS 提供下列資源來協助合規：

- [安全與合規快速入門指南](#) – 這些部署指南討論架構考量，並提供部署以安全與合規為中心之基準環境的步驟 AWS。
- [HIPAA 安全與合規架構白皮書](#) – 此白皮書說明公司如何使用 AWS 來建立符合 HIPAA 規範的應用程式。
- [AWS 合規資源](#) – 此工作手冊和指南集合可能適用於您的產業和位置。
- 《AWS Config 開發人員指南》中的 [使用規則評估資源](#) – AWS Config 服務會評估資源組態符合內部實務、產業準則和法規的程度。
- [AWS Security Hub CSPM](#) – AWS 此服務提供 內安全狀態的完整檢視 AWS，協助您檢查是否符合安全產業標準和最佳實務。

Amazon Simple Workflow Service 中的彈性

AWS 全球基礎設施是以 AWS 區域和可用區域為基礎建置。AWS 區域提供多個實體分隔和隔離的可用區域，這些可用區域與低延遲、高輸送量和高備援聯網連接。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、容錯能力和擴展能力，均較單一或多個資料中心的傳統基礎設施還高。

如需 AWS 區域和可用區域的詳細資訊，請參閱 [AWS 全球基礎設施](#)。

除了 AWS 全球基礎設施之外，Amazon SWF 還提供數種功能，以協助支援您的資料彈性和備份需求。

Amazon Simple Workflow Service 中的基礎設施安全

作為受管服務，受到 AWS 全球網路安全的保護。如需 AWS 安全服務以及 如何 AWS 保護基礎設施的相關資訊，請參閱 [AWS 雲端安全](#)。若要使用基礎設施安全的最佳實務設計您的 AWS 環境，請參閱安全支柱 AWS Well-Architected Framework 中的 [基礎設施保護](#)。

您可以使用 AWS 發佈的 API 呼叫，透過網路存取。使用者端必須支援下列專案：

- Transport Layer Security (TLS)。我們需要 TLS 1.2 並建議使用 TLS 1.3。

- 具備完美轉送私密(PFS)的密碼套件，例如 DHE (Ephemeral Diffie-Hellman)或 ECDHE (Elliptic Curve Ephemeral Diffie-Hellman)。現代系統(如 Java 7 和更新版本)大多會支援這些模式。

您可以從任何網路位置呼叫這些 API 操作，但 Amazon SWF 確實支援以資源為基礎的存取政策，其中可能包含根據來源 IP 地址的限制。您也可以使用 Amazon SWF 政策來控制來自特定 Amazon Virtual Private Cloud (Amazon VPC) 端點或特定 VPCs存取。實際上，這只會隔離網路中特定 VPC 對指定 Amazon SWF 資源 AWS 的網路存取。

Amazon Simple Workflow Service 中的組態和漏洞分析

組態和 IT 控制是客戶 AWS 與您之間共同責任。如需詳細資訊，請參閱 AWS [共同的責任模型](#)。

AWS CLI 搭配 Amazon Simple Workflow Service 使用

您可以從存取 Amazon Simple Workflow Service 的許多功能 AWS CLI。AWS CLI 提供使用 Amazon SWF 搭配 AWS 管理主控台 或 的替代方案，在某些情況下提供使用 Amazon SWF API 和 進程式設計的替代方案 AWS Flow Framework。

例如，您可以使用 AWS CLI 來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、記錄任務活動訊號等！如需 Amazon SWF 命令的完整清單，以及可用引數的說明和顯示其使用方式的範例，請參閱《命令參考》中的 [Amazon SWF](#) 命令。AWS CLI

這些 AWS CLI 命令緊密遵循 Amazon SWF API，因此您可以使用 AWS CLI 來了解基礎 Amazon SWF API。您也可以使用現有的 API 知識來原型程式碼，或在命令列上執行 Amazon SWF 動作。

若要進一步了解 AWS CLI，請參閱 [AWS Command Line Interface](#) 《使用者指南》。

使用 Amazon SWF APIs

除了使用中所述 AWS SDKs 之外[使用 AWS SDKs 開發](#)，您也可以直接使用 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。如需對 Amazon SWF 提出 HTTP 請求的詳細資訊，請參閱 [向 Amazon SWF 發出 HTTP 請求](#)。

本節提供使用 HTTP API 透過 Amazon SWF 開發工作流程的基本資訊。本節提供更進階的功能，例如使用計時器、使用 CloudTrail 記錄和標記您的工作流程[Amazon SWF 中的基本工作流程概念](#)。

主題

- [向 Amazon SWF 發出 HTTP 請求](#)
- [依類別的 Amazon SWF 動作清單](#)
- [向 Amazon SWF 註冊網域](#)
- [在 Amazon SWF 中設定逾時值](#)
- [向 Amazon SWF 註冊工作流程類型](#)
- [向 Amazon SWF 註冊活動類型](#)
- [AWS Lambda Amazon SWF 中的 任務](#)
- [在 Amazon SWF 中開發活動工作者](#)
- [在 Amazon SWF 中開發決策者](#)
- [在 Amazon SWF 中啟動工作流程](#)
- [在 Amazon SWF 中設定任務優先順序](#)
- [處理 Amazon SWF 中的錯誤](#)

向 Amazon SWF 發出 HTTP 請求

如果您不使用其中一個 AWS SDKs，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 在 HTTP 請求的 標頭中需要以下資訊：

- host Amazon SWF 端點。
- x-amz-date 您必須在 HTTP Date 標頭或 中提供時間戳記 AWS x-amz-date header (某些 HTTP 用戶端程式庫不允許您設定 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 AWS 實作特定的標籤，表示用來簽署請求的身分驗證版本 (目前，對於 Amazon SWF，此值一律為 AWS3)。

AWSAccessKeyId – 您的 AWS 存取金鑰 ID。

Algorithm – 用來建立 string-to-sign 的 HMAC-SHA 值的演算法，例如 HmacSHA256 或 HmacSHA1。

Signature – Base64(Algorithm(StringToSign、SigningKey))。詳細資訊，請參閱「[計算 Amazon SWF 的 HMAC-SHA 簽章](#)」

SignedHeaders – (選用) 如果存在，必須包含 Canonicalized HttpHeaders 計算中使用的所有 HTTP 標頭清單。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 – 請求的目的地服務，以及資料的操作，格式為

```
com.amazonaws.swf.service.model.SimpleWorkflowService. + <action>
```

例如 com.amazonaws.swf.service.model.SimpleWorkflowService.RegisterDomain

- content-type – 類型需要將 JSON 和字元集指定為 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
  AWSAccessKeyId=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，以下顯示使用 Unix Time 標記法，列出在兩個指定時間點之間啟動的所有工作流程執行的請求。

```
{
  "domain": "867530901",
  "startTimeFilter":
```

```
{
  "oldestDate": 1325376070,
  "latestDate": 1356998399
},
"tagFilter":
{
  "tag": "music purchase"
}
}
```

Amazon SWF JSON 請求和回應範例

下列範例顯示對 Amazon SWF 提出的請求，以取得我們先前建立的網域描述。然後顯示 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 不會序列化 null 值。如果您的 JSON 剖析器設定為序列化請求的 null 值，Amazon SWF 會忽略它們。

計算 Amazon SWF 的 HMAC-SHA 簽章

對 Amazon SWF 提出的每個請求都必須經過身分驗證。AWS SDKs 會自動簽署您的請求，並管理您的字符型身分驗證。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](#)」。如需第 3 AWS 版簽署的適用於 Java 的 AWS SDK 實作，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

一般而言，查詢字串會顯示在此處，但 Amazon SWF 不會使用查詢字串。

- 行 4-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. 使用上一個步驟產生的值的 SHA256 或 SHA1 摘要（取決於您使用的方法），以及使用 [GetSessionToken](#) API 動作來自 AWS Security Token Service 的臨時私密存取金鑰，來計算和 Base64-encode HMAC-SHA。

 Note

Amazon SWF 預期 Base64-encoded HMAC-SHA 值結尾會有等於符號 (=)。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。

如需搭配 Amazon SWF 和其他 AWS 服務使用臨時安全登入資料的詳細資訊，請參閱《IAM 使用者指南》中的與 [AWS IAM 搭配使用的服務](#)。

5. 將產生的值作為對 Amazon SWF 的 HTTP 請求 x-amzn-authorization 標頭中 Signature 名稱的值。
6. Amazon SWF 會驗證請求並執行指定的操作。

依類別的 Amazon SWF 動作清單

本節列出 Amazon SWF 應用程式程式設計界面 (API) 中 Amazon SWF 動作的參考主題。其依「功能類別」列出。

如需動作的字母清單，請參閱 [Amazon Simple Workflow Service 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函數或建置您自己的管理工具。

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 Amazon SWF 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置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

- [ListWorkflowTypes](#)
- [DescribeWorkflowType](#)

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](#)

向 Amazon SWF 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。

您可以使用 AWS 管理主控台 或使用 Amazon SWF API 中的 RegisterDomain 動作來註冊網域。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，所有工作流程執行的相關資訊都可以透過使用 AWS 管理主控台 或 Amazon SWF API 的可見性操作取得。

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](#) Amazon Simple Workflow Service API 參考中的

在 Amazon SWF 中設定逾時值

主題

- [逾時值的配額](#)
- [工作流程執行和決策任務逾時](#)
- [活動任務逾時](#)
- [另請參閱](#)

逾時值的配額

逾時值一律以秒為單位宣告，並可設定為任何秒數，最長可達一年 (31536000 秒)：任何工作流程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

您可在註冊工作流程類型時，為您的工作流程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，請參閱《Amazon Simple Workflow Service API 參考 [RegisterWorkflowType](#)》中的 [向 Amazon SWF 註冊工作流程類型](#) 和 。

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，請參閱《Amazon Simple Workflow Service API 參考 [RegisterActivityType](#)》中的 [向 Amazon SWF 註冊活動類型](#) 和 。

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 逾時類型](#)

向 Amazon SWF 註冊工作流程類型

本節中討論的範例會使用 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

Amazon Simple Workflow Service API 參考中的 [RegisterWorkflowType](#)

向 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

Amazon Simple Workflow Service API 參考中的 [RegisterActivityType](#)

AWS Lambda Amazon SWF 中的 任務

主題

- [關於 AWS Lambda](#)
- [使用 Lambda 任務的優點和限制](#)
- [在工作流程中使用 Lambda 任務](#)

關於 AWS Lambda

AWS Lambda 是一種全受管的運算服務，可執行您的程式碼，以回應自訂程式碼產生的事件，或來自 Amazon S3、DynamoDB、Amazon Kinesis、Amazon SNS 和 Amazon Cognito 等各種 AWS 服務的事件。如需有關 Lambda 的詳細資訊，請參閱 [AWS Lambda 開發人員指南](#)。

Amazon Simple Workflow Service 提供 Lambda 任務，讓您可以執行 Lambda 函數來取代或搭配傳統 Amazon SWF 活動。

Important

AWS 您的帳戶將針對 Amazon SWF 代表您執行的 Lambda 執行（請求）付費。如需 Lambda 定價的詳細資訊，請參閱 <https://aws.amazon.com/lambda/pricing/>。

使用 Lambda 任務的優點和限制

使用 Lambda 任務取代傳統 Amazon SWF 活動有許多好處：

- Lambda 任務不需要像 Amazon SWF 活動類型一樣進行註冊或版本控制。
- 您可以使用已在工作流程中定義的任何現有 Lambda 函數。
- Lambda 函數由 Amazon SWF 直接呼叫；您不需要實作工作者程式來執行它們，就像傳統活動一樣。
- Lambda 為您提供用於追蹤和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 任務只能在支援 Lambda AWS 的區域執行。如需 [Lambda 目前支援區域的詳細資訊](#)，請參閱 [Amazon Web Services 一般參考中的 Lambda 區域和端點](#)。
- Lambda 任務目前僅受基礎 SWF HTTP API 和 AWS Flow Framework 適用於 Java 的支援。目前不支援 AWS Flow Framework 適用於 Ruby 的 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 角色為 Amazon SWF 提供叫用 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 叫用 Lambda 函數之前，您必須提供 IAM 角色，以從 Amazon SWF 存取 Lambda。您可擇一方法：

- 選擇預先定義的角色 `AWSLambdaRole`，讓您的工作流程有權叫用與您的帳戶相關聯的任何 Lambda 函數。
- 定義您自己的政策和相關聯的角色，以授予工作流程叫用特定 Lambda 函數的許可，這些函數由其 Amazon Resource Name (ARNs) 指定。

限制 IAM 角色的許可

您可以使用 資源信任政策中的 `SourceArn` 和 `SourceAccount` 內容索引鍵，限制您提供給 Amazon SWF 之 IAM 角色的許可。這些金鑰會限制 IAM 政策的使用，使其僅用於屬於指定網域 ARN 的 Amazon Simple Workflow Service 執行。如果您同時使用兩個全域條件內容索引鍵，則 `aws:SourceAccount` 值中參考 `aws:SourceArn` 的值和帳戶在相同政策陳述式中使用時，必須使用相同的帳戶 ID。

在下列範例中，`SourceArn` 內容索引鍵限制 IAM 服務角色只能用於屬於帳戶 `someDomain` 的 Amazon Simple Workflow Service 執行 `123456789012`。

- 陳述式 1

委託人：`"Service": "swf.amazonaws.com"`

動作：`sts:AssumeRole`

```
"Condition": {
  "ArnLike": {
    "aws:SourceArn": "arn:aws:swf:*:123456789012:/domain/someDomain"
  }
}
```

在下列範例中，`SourceAccount` 內容索引鍵限制 IAM 服務角色只能用於帳戶中的 Amazon Simple Workflow Service 執行 `123456789012`

```
"Condition": {
  "StringLike": {
    "aws:SourceAccount": "123456789012"
  }
}
```

為 Amazon SWF 提供叫用任何 Lambda 角色的存取權

您可以使用預先定義的角色 `AWSLambdaRole`，讓 Amazon SWF 工作流程能夠叫用與您的帳戶相關聯的任何 Lambda 函數。

使用 `AWSLambdaRole` 讓 Amazon SWF 能夠叫用 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. 在AWS 服務角色下，選擇 Amazon SWF，然後選擇下一步。
5. 在 Attach Policy (附加政策) 畫面上，從清單選擇 AWSLambdaRole。
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 角色以提供叫用特定 Lambda 函數的存取權

如果您想要提供從工作流程叫用特定 Lambda 函數的存取權，您將需要定義自己的 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 受管政策，然後從清單中選擇 AWSLambdaRole。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. 在政策文件的資源欄位中，新增 Lambda 函數的 ARN (Lambda)。例如：

- 資源：arn:aws:lambda:us-east-1:111122223333:function:hello_lambda_function

Note

如需如何在 IAM 角色中指定資源的完整描述，請參閱使用 [IAM 中的 IAM 政策概觀](#)。

5. 選擇 Create Policy (建立政策) 來完成您的政策建立程序。

然後，您可以在建立新的 IAM 角色時選取此政策，並使用該角色來授予叫用 Amazon SWF 工作流程的存取權。此程序與使用 AWSLambdaRole 政策建立角色非常類似。差別在於您要在建立角色時選擇您自己的政策。

使用 Lambda 政策建立 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. 在AWS 服務角色下，選擇 Amazon SWF，然後選擇下一步。
5. 在連接政策畫面上，從清單中選擇您的 Lambda 函數特定政策。
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

定義 IAM 角色後，您需要將其連接至將使用該角色的工作流程，以呼叫您為 Amazon SWF 提供存取權的 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

- 呼叫 RegisterWorkflowType 時，請將 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

- 呼叫 StartWorkflowExecution 時，請將 lambdaRole 欄位設定為所定義角色的 ARN。

Note

如果呼叫 RegisterWorkflowType 或 StartWorkflowExecution 的帳戶不具使用指定角色的許可，則呼叫會因 OperationNotPermittedFault 而失敗。

從 Amazon SWF 工作流程呼叫 Lambda 函數

您可以使用 ScheduleLambdaFunctionDecisionAttributes 資料類型，識別要在工作流程執行期間呼叫的 Lambda 函數。

在呼叫 RespondDecisionTaskCompleted 期間，將 ScheduleLambdaFunctionDecisionAttributes 提供給決策清單。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 工作流程必須具備 IAM 角色，讓它能夠呼叫 Lambda 函數。提供的名稱必須遵循 FunctionName 參數的限制條件 (例如在「Lambda 呼叫」動作中)。
- 具有函數之選用輸入資料的 input。設定時，這必須遵循 ClientContext 參數的限制條件 (例如在「Lambda 呼叫」動作中)。
- 在任務因逾時例外狀況而失敗之前，函數能夠執行之選用最長期間 (秒) 的 startToCloseTimeout。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 任務](#)

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 通訊，以接收活動任務並進行執行。您可以有多個活動工作者執行相同活動類型的活動任務。

當決策者排程活動任務時，Amazon SWF 可讓活動工作者使用活動任務。當決策者排程活動任務時，它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，再將其傳送給活動工作者。

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。只要工作者可以透過 API 與 Amazon SWF 通訊，就可以在任何地方執行。由於 Amazon SWF 提供執行活動任務所需的所有資訊，因此所有活動工作者都可以無狀態。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. 輪詢活動任務的 Amazon SWF。
2. 開始執行任務。
3. 如果任務長期存在，請定期向 Amazon SWF 報告活動訊號。
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` 動作來輪詢 Amazon SWF。

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。如果沒有可用的活動任務，在 60 秒後，Amazon SWF 會傳回空的回應。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"
}
```

如果活動任務可用，Amazon SWF 會將其傳回給活動工作者。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。

下一節說明如何編寫活動工作者的程式，為 Amazon SWF 提供長時間執行活動的狀態更新。

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，Amazon SWF 會將其關閉並排程新的決策任務，以通知決策者逾時。決策者接著可以重新排程活動任務，或採取另一個動作。

如果在逾時後，活動工作者嘗試聯絡 Amazon SWF，例如透過呼叫 `RespondActivityTaskCompleted`，Amazon SWF 將傳回 `UnknownResource` 錯誤。

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，活動工作者向 Amazon SWF 報告活動訊號，並使用詳細資訊欄位報告活動任務已完成 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"
}
```

當活動完成時，Amazon SWF 會為與活動相關聯的工作流程執行排程新的決策任務。

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。

如果活動未在 `StartToCloseTimeout` 期間內回應，或超過 `ScheduleToCloseTimeout`，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 執行個體正在執行活動任務，且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，決策者會在任務在 Amazon EC2 執行個體失敗後無法交付下一個活動訊號時收到事件。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

如果活動工作者因為某些原因而無法執行活動任務，但仍然可以與 Amazon SWF 通訊，您可以對其進行程式設計以讓任務失敗。

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。這些是自由格式字串；您可以使用任何為您的應用程式提供服務的錯誤碼慣例。Amazon SWF 不會處理這些值。不過，Amazon SWF 可能會在主控台中顯示這些值。

當活動任務失敗時，Amazon SWF 會為與活動任務相關聯的工作流程執行排程決策任務，以通知決策者失敗。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。

由於輪詢是傳出請求，活動工作者可以在可存取 Amazon SWF 端點的任何網路上執行。

您可以視需要啟動任意數目的活動工作者。當決策者排程活動任務時，Amazon SWF 會自動將活動任務分發給輪詢活動工作者。

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。

由於工作流程執行的執行狀態會存放在其工作流程歷史記錄中，因此決策者可以是無狀態。Amazon SWF 會維護工作流程執行歷史記錄，並隨每個決策任務提供給決策者。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。

每次工作流程執行發生狀態變更時，Amazon SWF 都會排程決策任務。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，並向 Amazon SWF 提供決策清單。

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。

您可以使用您喜歡的任何語言實作決策者，並在任何地方執行它，只要它可以透過其服務 API 與 Amazon SWF 通訊即可。

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  
}
```

如果決策任務可從指定的任務清單中取得，Amazon SWF 會立即傳回它。如果沒有可用的決策任務，Amazon SWF 會保持連線開啟長達 60 秒，並在任務可用時立即傳回任務。如果沒有可用的任務，Amazon SWF 會傳回空的回應。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。

如果決策任務可用，Amazon SWF 會傳回包含決策任務的回應，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。若要擷取後續頁面，PollForDecisionTask請使用初始呼叫傳回的 nextPageToken 對 進行額外呼叫。請注意，您不會GetWorkflowExecutionHistory使用此 nextPageToken 呼叫。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。

在決策者決定要做出的決定後，可以使用適當的決策來回應 Amazon SWF (Amazon SWF)。

回應決策

解譯工作流程歷史記錄並產生決策清單後，決策者已準備好以這些決策回應 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` 動作將這些決策傳輸回 Amazon SWF。如需可用 [決策類型的](#) 清單，請參閱 Amazon Simple Workflow Service API 參考。

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回應事件，在此情況下為訊號，以防止執行關閉。

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 會檢查以確保關閉或取消工作流程執行的決定是決策者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。

啟動後，您的決策者應該開始輪詢 Amazon SWF 任務。在您開始工作流程執行和 Amazon SWF 排程決策任務之前，這些輪詢將會逾時並取得空的回應。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 可確保隨時只有一個決策任務可用於工作流程執行。這能防止決策發生衝突的問題。此外，Amazon SWF 可確保將單一決策任務指派給單一決策者，無論執行的決策者數目為何。

如果發生決策者正在處理另一個決策任務時產生決策任務的情況，Amazon SWF 會將新任務排入佇列，直到目前的任務完成為止。目前任務完成後，Amazon SWF 就會提供新的決策任務。此外，決策任務的批次處理概念是，如果在決策者處理決策任務時完成多個活動，Amazon SWF 只會建立單一新決策任務，以考慮完成多個任務。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。

由於輪詢是傳出請求，決策者可以在可存取 Amazon SWF 端點的任何網路上執行。

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量的決策者。Amazon SWF 支援在同一任務清單上輪詢多個決策者。

在 Amazon SWF 中啟動工作流程

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。不過，您可以重複使用已關閉執行 `workflowId` 的。Amazon SWF 也會將稱為 `runId` 的唯一系統產生識別符與每個工作流程執行建立關聯。

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 中的一些其他逾時參數不同，您無法 `NONE` 為此逾時指定的值；工作流程執行可執行的時間有一年上限。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` 動作成功，Amazon SWF 會傳回 `runId` 工作流程執行的。工作流程執行的 `runId` 在特定區域內是唯一的。`runId` 如果您稍後需要在對 Amazon SWF 的呼叫中指定此工作流程執行，請儲存。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"}
```

在 Amazon SWF 中設定任務優先順序

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。透過設定選用任務優先順序，您可以優先處理某些任務：Amazon SWF 會嘗試在任務清單上交付優先順序較高的任務，然後再處理優先順序較低的任務。

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是在註冊期間設定（由您或 Amazon SWF 設定），除非在排程活動或啟動工作流程執行時覆寫，否則一律會使用已註冊的任務優先順序。

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。

若要以預設任務優先順序註冊工作流程類型，請在使用 [RegisterWorkflowType](#) 動作時設定 `defaultTaskPriority` 選項：

```
{
  "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",
  "input": "arbitrary-string-that-is-meaningful-to-the-workflow",
  "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。

若要在註冊活動類型時設定任務優先順序，請在使用 [RegisterActivityType](#) 動作時設定 `defaultTaskPriority` 選項：

```
{
  "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"
}
```

若要以任務優先順序排程任務，請在使用 [RespondDecisionTaskCompleted](#) 動作排程活動時使用 `taskPriority` 選項：

```
{
  "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 動作取得有關設定任務優先順序（或設定預設任務優先順序）的資訊：

- [DescribeActivityType](#) 傳回回應 configuration 區段中活動類型的 defaultTaskPriority。
- [DescribeWorkflowExecution](#) 傳回回應 executionConfiguration 區段中工作流程執行的 taskPriority。
- [DescribeWorkflowType](#) 傳回回應 configuration 區段中工作流程類型的 defaultTaskPriority。
- [GetWorkflowExecutionHistory](#) 和 [PollForDecisionTask](#) 會在回應的 activityTaskScheduledEventAttributes、decisionTaskScheduledEventAttributes、workflowExecutionStartedEventAttributes 區段內提供任務優先順序資訊。

處理 Amazon SWF 中的錯誤

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

當對 Amazon SWF 的請求因為格式不正確或包含無效資料而失敗時，就會發生驗證錯誤。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 會在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。如需常見錯誤代碼的清單，請前往《Amazon Simple Workflow Service API 參考》中的 動作主題。

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

即使請求格式正確，Amazon SWF 嘗試執行請求時仍可能發生錯誤。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 逾時類型](#)」。

- [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。如同逾時錯誤，Amazon SWF 會將適當的事件新增至工作流程執行歷史記錄。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](#)

如需上述任何事件的詳細資訊，請參閱《Amazon SWF API 參考》中的[歷史記錄事件](#)。

Amazon SWF 配額

Amazon SWF 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和 workflows 執行歷史記錄的大小。這些配額旨在防止錯誤的工作流程耗用系統的所有資源，但並非硬性限制。如果您發現應用程式經常超過這些配額，您可以[請求提高服務配額](#)。

目錄

- [Amazon SWF 的一般帳戶配額](#)
- [工作流執行的配額](#)
- [任務執行的配額](#)
- [Amazon SWF 限流配額](#)
 - [調節所有區域的配額](#)
 - [所有區域的決策配額](#)
 - [工作流層級配額](#)
- [請求提高配額](#)

Amazon SWF 的一般帳戶配額

- 註冊網域上限 – 100

此配額包括已註冊和已棄用的網域。

- 工作流和活動類型上限 – 每個網域 10,000 個

此配額包括已註冊和已棄用的類型。

- API 呼叫配額 – 除了不常峰值之外，如果應用程式在非常短的時間內進行大量 API 呼叫，則可能會調節應用程式。
- 請求大小上限 – 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。

- 計數 APIs 的截斷回應 – 表示已達到內部配額，且回應不是完整計數。

在傳回完整回應之前，某些查詢將在內部達到上述的 1 MB 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。

- 標籤數量上限 – 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`。

工作流程執行的配額

- 開啟的工作流程執行上限 – 每個網域 100,000 個

此計數包含子工作流程執行。

- 工作流程執行時間上限 – 1 年。這是無法變更的硬性配額。
- 工作流程執行歷史記錄大小上限 – 25,000 個事件。這是無法變更的硬性配額。

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。由於決策者必須擷取工作流程歷史記錄，較小的歷史記錄可讓決策者更快完成。如果使用 [Flow Framework](#)，您可以使用 `ContinueAsNew` 以全新的歷史記錄繼續工作流程。

- 開啟的子工作流程執行上限 – 每個工作流程執行 1,000 個

如果您的使用案例要求您超過這些配額，您可以使用 Amazon SWF 提供的功能，繼續使用[子工作流程執行](#)來繼續執行和建構應用程式。如果您發現仍然需要增加配額，請參閱 [請求提高配額](#)。

任務執行的配額

- 每個任務清單的輪詢器上限 – 每個任務清單 1,000 個

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

雖然上限為 1,000，但在此配額之前，您可能會遇到 `LimitExceededException` 錯誤。此錯誤並不表示您的任務正在延遲。相反地，這表示您在任務清單上有閒置輪詢

器的最大數量。Amazon SWF 會設定此限制，以在用戶端和伺服器端儲存資源。設定限制可防止過多的輪詢器不必要的等待。您可以使用多個任務清單來分配輪詢，以減少LimitExceededException錯誤。

- 每秒排程的任務上限 – 每個任務清單 2,000 個

您可以在特定任務清單上排程每秒最多 2,000 個任務。如果您超過 2,000，您的ScheduleActivityTask決策將會失敗並顯示ACTIVITY_CREATION_RATE_EXCEEDED錯誤。

Note

雖然上限為 2,000，但在此配額之前，您可能會遇到ACTIVITY_CREATION_RATE_EXCEEDED錯誤。若要減少這些錯誤，請使用多個任務清單來分配負載。

- 任務執行時間上限 – 1 年（受工作流程執行時間上限限制）

您可以設定[活動逾時](#)，以便在[活動任務](#)執行的特定階段花費太長的時間時，導致發生逾時事件。

- SWF 將任務保留在佇列中的時間上限 – 1 年（受工作流程執行時間配額限制）

您可以在活動註冊期間設定預設[活動逾時](#)，如果[活動任務](#)執行的特定階段時間過長，將導致發生逾時事件。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和工作者正在處理的活動任務。

- 開啟計時器上限 – 每個工作流程執行 1,000 個
- 輸入/結果資料大小上限 – 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，以及使用[工作流程執行訊號](#)傳送的輸入。

- 決策任務回應中的決策上限 – 有所不同

由於 [API 請求大小上限](#) 的 1 MB 配額，在單一呼叫中傳回的決策數量RespondDecisionTaskCompleted會根據每個決策所使用的資料大小來限制，包括提供給排程活動任務或工作流程執行的任何輸入資料大小。

Amazon SWF 限流配額

除了上述的服務配額之外，某些 Amazon SWF API 呼叫和決策事件也會受到調節，以使用[字符儲存貯體](#)方案來維持服務頻寬。如果您的請求速率持續超過此處列出的速率，您可以[請求提高限流配額](#)。

調節和決策配額在所有區域都相同。

調節所有區域的配額

下列配額適用於個別帳戶層級。您也可以請求提高下列配額。如需執行此操作的詳細資訊，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
CountClosedWorkflowExecutions	2000	6
CountOpenWorkflowExecutions	2000	6
CountPendingActivityTasks	200	6
CountPendingDecisionTasks	200	6
DeleteActivityType	200	6
DeleteWorkflowType	200	6
DeprecateActivityType	200	6
DeprecateDomain	100	6
DeprecateWorkflowType	200	6
DescribeActivityType	2000	6
DescribeDomain	200	6
DescribeWorkflowExecution	2000	6
DescribeWorkflowType	2000	6
GetWorkflowExecutionHistory	2000	60

API 名稱	儲存貯體大小	每秒重新填滿速率
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
RespondDecisionTaskCompleted	2000	200
SignalWorkflowExecution	2000	30
StartWorkflowExecution	2000	200
TagResource	50	30

API 名稱	儲存貯體大小	每秒重新填滿速率
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。您也可以請求提高下列配額。如需執行此操作的詳細資訊，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	1000	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

API 名稱	儲存貯體大小	每秒重新填滿速率
SignalWorkflowExecution	1000	1000
RecordActivityTaskHeartbeat	1000	1000
RequestCancelWorkflowExecution	200	200

請求提高配額

如需詳細資訊，請參閱《AWS》中的 [AWS 一般參考服務配額](#)。

Amazon SWF 的其他資源和參考資訊

本章提供使用 Amazon SWF 開發工作流程時有用的其他資源和參考資訊。

主題

- [Amazon SWF 逾時類型](#)
- [Amazon Simple Workflow Service 端點](#)
- [Amazon Simple Workflow Service 的其他文件](#)
- [Amazon Simple Workflow Service 的 Web 資源](#)
- [Ruby Flow 的遷移選項](#)

Amazon SWF 逾時類型

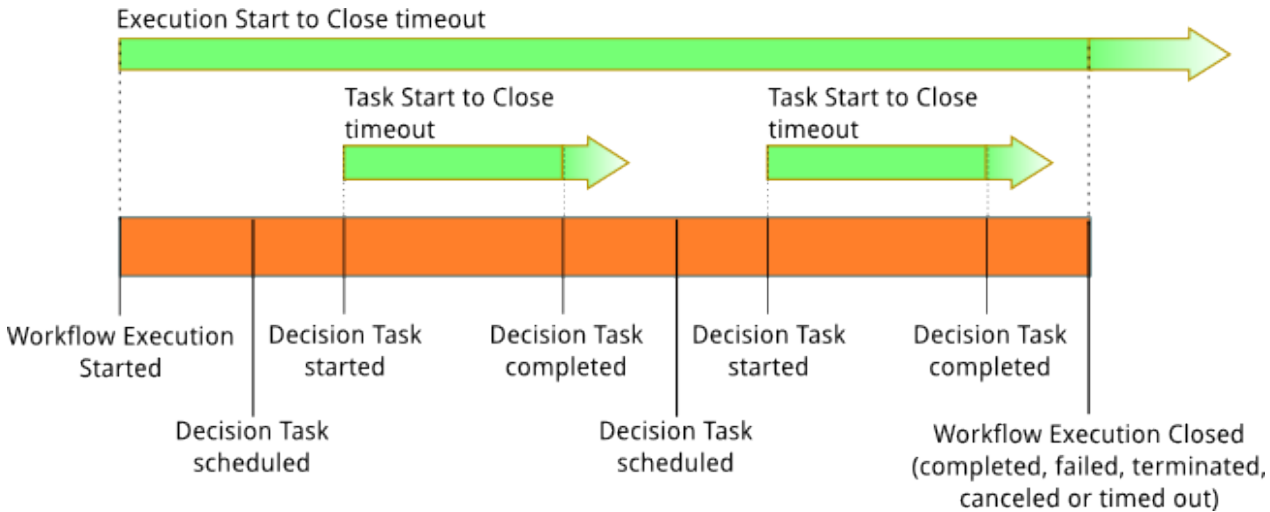
為了確保工作流程執行正確執行，您可以使用 Amazon SWF 設定不同類型的逾時。有些逾時會指定工作流程總共可以執行多久。有些逾時則指定活動任務要多久才能指派給工作者，以及從排程開始要多久才可以完成。Amazon SWF API 中的所有逾時都會以秒為單位指定。Amazon SWF 也支援字串NONE做為逾時值，表示沒有逾時。

對於與決策任務和活動任務相關的逾時，Amazon SWF 會將事件新增至工作流程執行歷史記錄。事件的屬性提供有關發生哪種類型的逾時，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。當決策者接到新的決策任務時，會在歷史記錄中看到逾時事件，並呼叫 [RespondDecisionTaskCompleted](#) 動作以採取適當的動作。

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。由於逾時，Amazon SWF 也可能會關閉任務。

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：

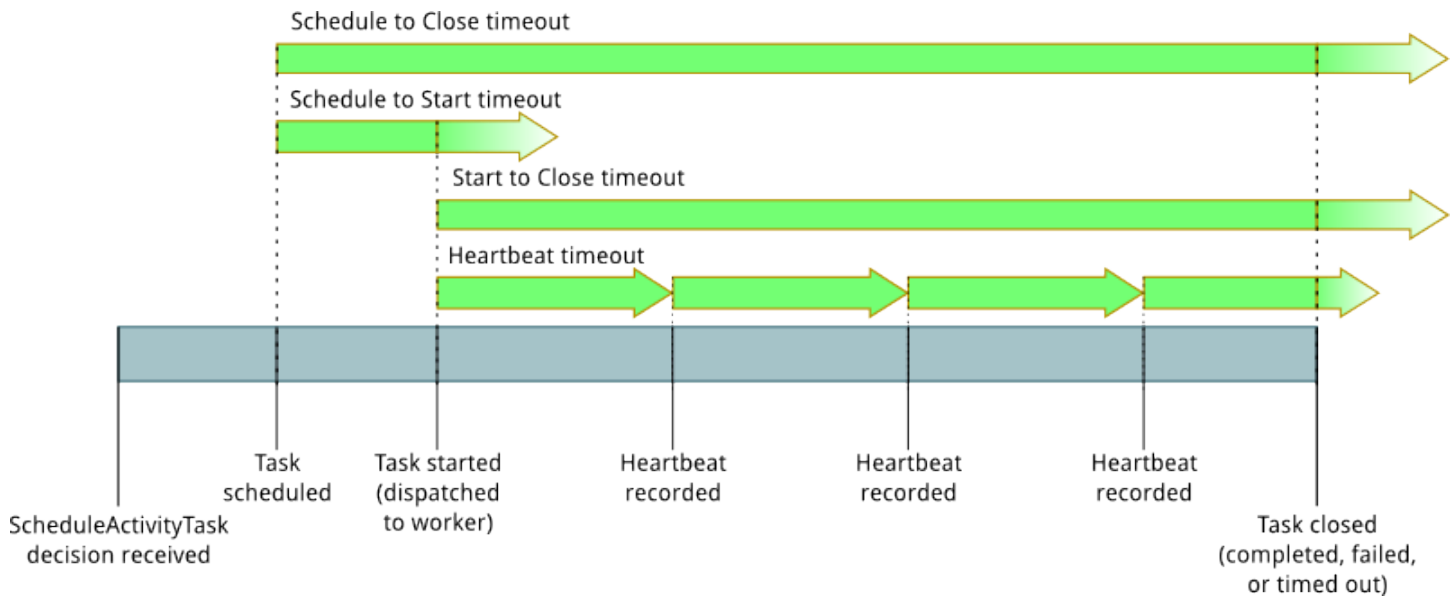


有兩種逾時類型與工作流程和決策任務相關：

- 工作流程開始關閉 (**timeoutType: START_TO_CLOSE**) – 此逾時會指定工作流程執行完成所需的時間上限。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，並將類型為 [WorkflowExecutionTimedOut](#) 的事件新增至工作流程執行歷史記錄。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。
- 決策任務開始關閉 (**timeoutType: START_TO_CLOSE**) – 此逾時會指定對應決策者完成決策任務所需的最長時間。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史記錄中標記為逾時，Amazon SWF 會將類型為 [DecisionTaskTimedOut](#) 的事件新增至工作流程歷史記錄。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，以提醒決策者此決策任務已逾時。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：

- 活動任務開始關閉 (**timeoutType: START_TO_CLOSE**) – 此逾時會指定活動工作者在收到任務後處理任務所需的時間上限。嘗試使用 [RespondActivityTaskCanceled](#)、[RespondActivityTaskCompleted](#) 和 [RespondActivityTaskFailed](#) 結束逾時的活動任務都將失敗。
- 活動任務活動訊號 (**timeoutType: HEARTBEAT**) – 此逾時會指定任務在透過 `RecordActivityTaskHeartbeat` 動作提供進度之前可執行的時間上限。
- 活動任務排程開始 (**timeoutType: SCHEDULE_TO_START**) – 此逾時會指定如果沒有工作者可執行任務，Amazon SWF 會等待多久才逾時活動任務。一旦逾時，過期的任務就會不指派給其他工作者。
- 活動任務排程關閉 (**timeoutType: SCHEDULE_TO_CLOSE**) – 此逾時會指定任務從排程到完成所需的時間。就最佳實務而言，此值不應大於任務從排程到開始逾時和任務從開始到結束逾時的總和。

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。當其中一個逾時發生時，Amazon SWF 會將 [ActivityTaskTimedOut](#) 類型的事件新增至工作流程歷史記錄。此事件的时间Type 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，以提醒決策者發生逾時。

Amazon Simple Workflow Service 端點

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 網域和所有相關工作流程和活動必須位於相同區域內，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，而且這兩個網域以「個別網域」存在，則無法跨區域共享與您網域相關聯的工作流程、任務清單、活動或資料。

如果您在工作流程中使用其他 AWS 資源，例如 Amazon EC2 執行個體，這些資源也必須存在於與 Amazon SWF 資源相同的區域中。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 Simple Workflow Service 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 Simple Workflow Service API 參考

[Amazon Simple Workflow Service 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

是一種 [AWS Flow Framework](#) 程式設計架構，可簡化實作分散式非同步應用程式的程序，這些應用程式使用 Amazon SWF 來管理其工作流程和活動，因此您可以專注於實作工作流程邏輯。

每個 AWS Flow Framework 都旨在以其設計的語言以特殊方式運作，因此您可以自然地使用您選擇的語言來實作具有 Amazon SWF 所有優點的工作流程。

有適用於 Java 的 AWS 流程架構。[AWS Flow Framework 適用於 Java 的開發人員指南](#) 提供如何取得、設定和使用 AWS Flow Framework 適用於 Java 的的相關資訊。

AWS SDK 文件

AWS 軟體開發套件 (SDKs) 提供許多不同程式設計語言的 Amazon SWF 存取權。SDKs 密切遵循 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

此處僅列出寫入時支援 Amazon SWF 的 SDKs。如需可用 AWS SDKs 的完整清單，請造訪 [Tools for Amazon Web Services](#) 頁面。

Java

為 AWS 基礎設施服務 適用於 Java 的 AWS SDK 提供 Java API。

若要檢視現已提供的文件，請參閱 [適用於 Java 的 AWS SDK 文件](#) 頁面。您也可以依照下列連結，直接前往 SDK 參考中的 Amazon SWF 區段：

- [Class: AmazonSimpleWorkflowClient](#)
- [Class: AmazonSimpleWorkflowAsyncClient](#)
- [Interface: AmazonSimpleWorkflow](#)
- [Interface: AmazonSimpleWorkflowAsync](#)

JavaScript

適用於 JavaScript 的 AWS SDK 可讓開發人員使用瀏覽器或伺服器上 Node.js 應用程式內部提供的簡單且 easy-to-use API，建置利用 AWS 服務的程式庫或應用程式。

若要檢視現已提供的文件，請參閱 [適用於 JavaScript 的 AWS SDK 文件](#) 頁面。您也可以依照此連結直接前往 SDK 參考中的 Amazon SWF 區段：

- [Class: AWS.SimpleWorkflow](#)

.NET

適用於 .NET 的 AWS SDK 是可下載的單一套件，其中包含 Visual Studio 專案範本、AWS .NET 程式庫、C# 程式碼範例和文件。適用於 .NET 的 AWS SDK 可讓 Windows 開發人員更輕鬆地為 Amazon SWF 和其他服務建置 .NET 應用程式。

若要檢視現已提供的文件，請參閱 [適用於 .NET 的 AWS SDK 文件](#) 頁面。您也可以依照下列連結，直接前往 SDK 參考中的 Amazon SWF 區段：

- [Namespace: Amazon.SimpleWorkflow](#)
- [Namespace: Amazon.SimpleWorkflow.Model](#)

PHP

為 Amazon SWF AWS SDK for PHP 提供 PHP 程式設計界面。

若要檢視現已提供的文件，請參閱[AWS SDK for PHP 文件](#)頁面。您也可以依照此連結直接前往 SDK 參考中的 Amazon SWF 區段：

- [Class: SwfClient](#)

Python

為 Amazon SWF 適用於 Python (Boto) 的 AWS SDK 提供 Python 程式設計界面。

若要檢視可用的文件，請參閱 [boto : Amazon Web Services 的 Python 界面](#) 頁面。您也可以依照下列連結，直接前往 文件中的 Amazon SWF 區段：

- [Amazon SWF 教學課程](#)
- [Amazon SWF 參考](#)

Ruby

為 Amazon SWF 適用於 Ruby 的 AWS SDK 提供 Ruby 程式設計界面。

若要檢視現已提供的文件，請參閱[適用於 Ruby 的 AWS SDK 文件](#)頁面。您也可以依照此連結直接前往 SDK 參考中的 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。只需一個工具即可下載和設定，您可以從命令列控制多個 AWS 服務，並透過指令碼將其自動化。

如需的詳細資訊 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 Simple Workflow Service 的 Web 資源

您可以使用多種 Web 資源來進一步了解 Amazon SWF，或取得使用服務和開發工作流程的協助。

Amazon SWF 論壇

Amazon SWF 論壇可讓您與其他 Amazon SWF 開發人員和 Amazon SWF 開發團隊的成員進行通訊，以提出問題並獲得解答。

您可以造訪論壇：[論壇：Amazon Simple Workflow Service](#)。

Amazon SWF 常見問答集

Amazon SWF 常見問答集提供有關 Amazon SWF 常見問題的解答，包括常見使用案例的概觀、Amazon SWF 與其他服務之間的差異等等。

您可以在此存取常見問答集：[Amazon SWF 常見問答集](#)。

Amazon SWF 影片

YouTube 上的 [Amazon Web Services](#) 頻道為所有 Amazon Web Services 提供影片訓練，包括 Amazon SWF。如需 Amazon SWF 相關影片的完整清單，請使用下列查詢：[Amazon Web Services 中的簡易工作流程](#)

Ruby Flow 的遷移選項

AWS Flow Framework 適用於 Ruby 的不再處於主動開發狀態。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。本主題將涵蓋繼續使用 Amazon SWF 的用量和遷移選項，以及如何遷移至 Step Functions 的資訊。

選項	描述
繼續使用 Ruby Flow Framework	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。計劃在不久的將來移出 AWS Flow Framework for Ruby。
遷移至 Java Flow Framework	Java Flow Framework 會繼續積極開發，並繼續收到新功能和更新。
遷移至 Step Functions	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。

選項	描述
直接使用 SWF API ，而不使用 Flow Framework	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。您可以遷移至 Java Flow Framework 來繼續具有相同層級的抽象，也可以直接與 Amazon SWF SDK 互動。

繼續使用 Ruby Flow Framework

AWS Flow Framework 適用於 Ruby 的 會在短期內繼續運作。如果您有在 AWS Flow Framework for Ruby 中寫入的工作流程，這些工作流程將繼續運作。如果沒有更新、支援或安全性修正，最好有堅實的計劃在不久的將來遷移出 AWS Flow Framework 適用於 Ruby 的。

遷移至 Java Flow Framework

AWS Flow Framework 適用於 Java 的 將保留在作用中的開發中。概念上，AWS Flow Framework 適用於 Java 的 類似於 AWS Flow Framework for Ruby：您仍然可以專注於您的工作流程邏輯，架構將有助於管理您的決策者邏輯，並使 Amazon SWF 的其他方面更容易管理。

- [AWS Flow Framework 適用於 Java 的](#)
- [AWS Flow Framework for Java API 參考](#)

遷移至 Step Functions

AWS Step Functions 提供的服務類似於 Amazon SWF，但您的工作流程邏輯是由狀態機器控制。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 提供可靠的方法來協調元件，並逐步完成應用程式的函數。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會按順序和預期執行。Step Functions 會記錄每個步驟的狀態，因此當發生問題時，您可以快速診斷和偵錯問題。

在 Step Functions 中，您可以使用使用 [Amazon States Language](#) 定義的宣告式 JSON 撰寫的狀態機器來管理任務協調。透過使用狀態機器，您不需要撰寫和維護決策者程式來控制應用程式邏輯。Step Functions 提供直覺式、具生產力且靈活的方法來使用視覺化工作流程協調應用程式元件。您應該考慮

將 AWS Step Functions 用於所有新的應用程式，而 Step Functions 提供了一個絕佳的平台，可讓您針對目前在 AWS Flow Framework for Ruby 中實作的工作流程遷移至。

為了協助您將任務遷移至 Step Functions，同時繼續利用 Ruby 語言技能，Step Functions 提供 Ruby 活動工作者範例。此範例使用實作活動工作者的最佳實務，並可做為範本，將任務邏輯遷移至 Step Functions。如需詳細資訊，請參閱《[AWS Step Functions 開發人員指南](#)》中的 [Ruby 中的範例活動工作者](#) 主題。

Note

對於許多客戶而言，從 AWS Flow Framework for Ruby 遷移至 Step Functions 是最佳選項。但是，如果您要求訊號介入您的程序，或者如果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，或遷移至 AWS Flow Framework 適用於 Java 的。

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 命令列參考](#)

直接使用 Amazon SWF API

雖然 AWS Flow Framework for Ruby 管理 Amazon SWF 的一些複雜性，您也可以直接使用 Amazon SWF API。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 Simple Workflow Service 開發人員指南](#)
- [Amazon Simple Workflow Service API 參考](#)

文件歷史記錄

下表說明自上次發行 Amazon Simple Workflow Service 開發人員指南以來文件的重要變更。

變更	描述	變更日期
僅限文件的更新	Amazon SWF 現在包含有關 AWS 使用者通知的區段，AWS 服務 這是 作為 中 AWS 通知的中心位置 AWS 管理主控台。如需詳細資訊，請參閱 AWS 使用者通知 搭配 Amazon Simple Workflow Service 使用 。	2023 年 5 月 4 日
更新	Amazon SWF 現在提供新的主控台體驗，以管理 SWF 工作流程及其執行相關動作。如需詳細資訊，請參閱 Amazon SWF 主控台教學課程 。	2022 年 9 月 12 日
更新	更新 任務執行的配額 章節以包含 Maximum tasks scheduled per second，並更新 適用於 CloudWatch 的 Amazon SWF 指標 頁面以包含有關搭配 CloudWatch 使用非 ASCII 資源名稱 的資訊。	2021 年 5 月 12 日
新功能	Amazon Simple Workflow Service 現在支援 Amazon EventBridge。如需詳細資訊，請參閱： <ul style="list-style-type: none"> Amazon SWF 的 EventBridge EventBridge 使用者指南 	2020 年 12 月 18 日
新功能	Amazon Simple Workflow Service 支援使用標籤的 IAM 許可。如需更多資訊，請參閱下列內容。 <ul style="list-style-type: none"> Amazon SWF 中的標籤 <ul style="list-style-type: none"> 管理標籤 標記工作流程執行 使用標籤控制對網域的存取 TagResource UntagResource ListTagsForResource 	2019 年 6 月 20 日

變更	描述	變更日期
	<ul style="list-style-type: none"> RegisterDomain 	
新功能	Amazon Simple Workflow Service 現已在歐洲（斯德哥爾摩）區域推出。	2018 年 12 月 12 日
更新	已改善 CloudTrail 整合上的 Amazon Simple Workflow Service 主題。請參閱 使用記錄 API 呼叫 AWS CloudTrail 。	2018 年 8 月 7 日
更新	新增 CloudWatch PendingTasks 新指標的相關資訊。如需詳細資訊，請參閱 Amazon SWF 指標 。	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 Ruby Flow 的遷移選項 。	2018 年 3 月 9 日
更新	改善進階概念主題的導覽。請參閱 Amazon SWF 中的進階工作流程概念 。	2018 年 2 月 19 日
更新	新增有效的統計資料資訊，改善 CloudWatch 指標文件。請參閱 適用於 CloudWatch 的 Amazon SWF 指標 。	2017 年 12 月 4 日
更新	變更 TOC 以改善文件結構。新增「 API 和決策事件指標 」的相關資訊。	2017 年 11 月 9 日
更新	更新「 Amazon SWF 配額 」一節，以包括所有區域的調節限制。	2017 年 10 月 18 日
更新	將「 Amazon SWF 入門 」中的 task_list 變更為 workflowId，以免與 activity_list 混淆。	2017 年 7 月 25 日
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

變更	描述	變更日期
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之外，您還可以指定 Lambda 任務。如需詳細資訊，請參閱 AWS Lambda Amazon SWF 中的 任務 。	2015 年 7 月 21 日
設定任務優先順序的支援	Amazon SWF 現在包含在任務清單上設定任務優先順序的支援，並將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。「 在 Amazon SWF 中設定任務優先順序 」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	新增了新主題，說明如何使用 CloudTrail 記錄 Amazon SWF API 呼叫： 使用 記錄 API 呼叫 AWS CloudTrail 。	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： 適用於 CloudWatch 的 Amazon SWF 指標 提供支援的指標清單和說明，而 使用 檢視 CloudWatch 的 Amazon SWF 指標 AWS 管理主控台 則提供如何使用 檢視指標和設定警示的相關資訊 AWS 管理主控台。	2014 年 4 月 28 日
更新	新增章節： 「Amazon SWF 的其他資源和參考資訊 」。本節提供一些服務參考資訊，並提供 Amazon SWF 開發人員的其他文件、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 Amazon SWF 入門 。	2013 年 10 月 25 日
更新	新增 AWS CLI 資訊和範例 。	2013 年 8 月 26 日
更新	更新和修正。	2013 年 8 月 1 日

變更	描述	變更日期
更新	更新文件以描述如何使用 IAM 進行存取控制。	2013 年 2 月 22 日
初始版本	這是 Amazon Simple Workflow Service 開發人員指南的第一個版本。	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。