



開發人員指南

Amazon Translate



Amazon Translate: 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，這些擁有者可能附屬於 Amazon，或與 Amazon 有合作關係，亦或受到 Amazon 贊助。

Table of Contents

什麼是 Amazon Translate ?	1
使用案例	1
第一次使用	2
Amazon Translate 定價	2
Amazon Translate API 參考	2
支援的語言	3
支援的語言	3
Amazon Translate 功能支援的語言	7
運作方式	8
輸入內容支援的格式	8
自訂您的翻譯	8
自動語言偵測	9
例外狀況處理	9
設定	10
註冊 AWS 帳戶	10
建立具有管理存取權的使用者	10
設定 AWS CLI	12
授與程式設計存取權	12
使用 AWS SDKs	14
開始使用	15
開始使用 (主控台)	15
開始使用 (AWS CLI)	17
使用命令列翻譯文字	17
下一步驟	17
入門 (SDK)	17
使用適用於 Java 的 軟體開發套件	18
使用適用於 Python 的 AWS 開發套件	18
其他 SDK 範例	18
轉譯處理模式	19
即時翻譯	19
即時轉譯 (主控台)	19
即時轉譯 (API)	23
非同步批次處理	26
區域可用性	26

先決條件	27
執行任務	32
監控和分析	38
取得結果	40
自訂您的翻譯	43
使用do-not-translate標籤	43
搭配主控台使用do-not-translate	43
搭配 API do-not-translate	44
使用自訂術語自訂	44
建立自訂術語	45
使用自訂術語	47
使用適用於 Python 的 SDK 的範例	48
加密您的術語	50
最佳實務	51
使用簡潔	52
使用簡潔設定	52
支援的語言	52
遮罩褻瀆	53
使用褻瀆設定	54
不支援的語言	54
設定正式性	55
使用正規設定	55
支援的語言	56
使用平行資料自訂	56
區域可用性	58
Amazon Translate 的平行資料輸入檔案	58
新增平行資料	63
檢視和管理平行資料	66
程式碼範例	71
基本概念	71
動作	72
案例	90
建置 Amazon Transcribe 串流應用程式	90
建置 Amazon Lex 聊天機器人	91
建置 Amazon SNS 應用程式	92
建立應用程式以分析客戶意見回饋	93

開始執行翻譯任務	99
標記	103
標記新資源	104
檢視、編輯和刪除標籤	105
安全	107
資料保護	107
靜態加密	108
傳輸中加密	109
身分和存取權管理	109
目標對象	110
使用身分驗證	110
使用政策管理存取權	111
Amazon Translate 如何與 IAM 搭配使用	112
身分型政策範例	117
AWS 受管政策	124
疑難排解	126
監控	128
使用 CloudWatch 進行監控	130
使用記錄 Amazon Translate API 呼叫 AWS CloudTrail	132
Amazon Translate 的 CloudWatch 指標和維度	134
使用 EventBridge 進行監控	136
法規遵循驗證	138
恢復能力	139
基礎設施安全性	139
VPC 端點 (AWS PrivateLink)	140
Amazon Translate VPC 端點的考量事項	140
為 Amazon Translate 建立介面 VPC 端點	140
為 Amazon Translate 建立 VPC 端點政策	140
指南和配額	142
支援的 AWS 區域	142
合規	142
限流	142
指導方針	142
Service Quotas	143
文件歷史紀錄	146
API 參考	156

AWS 詞彙表	157
.....	clviii

什麼是 Amazon Translate ？

Amazon Translate 是一種文字翻譯服務，使用進階機器學習技術來隨需提供高品質的翻譯。您可以使用 Amazon Translate 來翻譯非結構化文字文件，或建置以多種語言運作的應用程式。[支援的語言和語言代碼](#) 如需 Amazon Translate 支援的語言相關資訊，請參閱。

主題

- [使用案例](#)
- [您是第一次使用 Amazon Translate 嗎？](#)
- [Amazon Translate 定價](#)
- [Amazon Translate API 參考](#)

使用案例

使用 Amazon Translate 執行下列動作：

透過整合 Amazon Translate，在您的應用程式中啟用多語言使用者體驗：

- 翻譯公司撰寫的內容，例如會議記錄、技術人員報告、知識庫文章、貼文等等。
- 翻譯人際通訊，例如電子郵件、遊戲內聊天、客戶服務聊天，以便客戶和員工可以使用他們偏好的語言進行連線。

處理和管理您公司的傳入資料：

- 分析多種語言的文字，例如社交媒體和新聞饋送。
- 以多種語言搜尋資訊，例如搜尋 eDiscovery 案例。

將 Amazon Translate 與其他 AWS 服務整合，以啟用語言無關的處理：

- 從非結構化文字擷取具名實體、情緒和金鑰片語，例如使用 [Amazon Comprehend](#) 的社交媒體串流。
- 使用 [Amazon Transcribe](#) 以多種語言提供字幕和即時字幕。
- 與 [Amazon Polly](#) 交談翻譯的內容。
- 翻譯存放在 [Amazon S3](#) 中的文件儲存庫。
- 翻譯儲存在下列資料庫中的文字：[Amazon DynamoDB](#)、[Amazon Aurora](#) 和 [Amazon Redshift](#)。
- 無縫整合工作流程與 [AWS Lambda](#) 或 [AWS Glue](#)。

您是第一次使用 Amazon Translate 嗎？

如果您是第一次使用 Amazon Translate，建議您從下列主題開始：

1. [Amazon Translate 的運作方式](#) – 推出 Amazon Translate。
2. [Amazon Translate 入門](#) – 說明如何設定 AWS 帳戶並開始使用 Amazon Translate。
3. [Amazon Translate AWS SDKs 的程式碼範例](#) – 使用程式碼範例來探索 Amazon Translate APIs。

您也可以使用下列資源來了解 Amazon Translate 服務：

- [AWS Machine Learning 部落格](#) 包含有關 Amazon Translate 的實用文章。
- [Amazon Translate Deep Dive Video 系列](#) 提供有關 Amazon Translate 的介紹影片。

Amazon Translate 定價

如同其他 AWS 產品，使用 Amazon Translate 沒有合約或最低承諾。如需使用 Amazon Translate 成本的詳細資訊，請參閱 [Amazon Translate 定價](#)。

Amazon Translate API 參考

Amazon Translate API 參考現在是單獨的文件。如需詳細資訊，請參閱 [Amazon Translate API 參考](#)。

支援的語言和語言代碼

Amazon Translate 提供來源語言（輸入語言）與目標語言（輸出語言）之間的翻譯。來源語言與目標語言的組合稱為語言配對。

Note

如果您為來源語言和目標語言指定相同的語言，Amazon Translate 不會向您收取翻譯費用。如果您將來源語言設定為 auto，則可能需要支付使用自動偵測的費用。如需詳細資訊，請參閱[自動語言偵測](#)。

主題

- [支援的語言](#)
- [Amazon Translate 功能支援的語言](#)

支援的語言

Amazon Translate 支援下表所列語言之間的文字翻譯。語言代碼欄使用 [ISO 639-1](#) 兩位數語言代碼。對於語言的國家/地區變體，資料表遵循附加破折號的 [RFC 5646](#) 格式，後面接著 [ISO 3166](#) 2 位數國家/地區碼。例如，墨西哥西班牙文變體的語言代碼為 es-MX。

語言	語言代碼
南非荷蘭文	af
阿爾巴尼亞文	sq
阿姆哈拉文	am
Arabic	ar
亞美尼亞文	hy
亞塞拜然文	az
孟加拉文	bn

語言	語言代碼
波士尼亞文	bs
保加利亞文	bg
加泰隆尼亞文	ca
簡體中文	zh
繁體中文	zh-TW
克羅埃西亞文	hr
捷克文	cs
丹麥文	da
達里文	fa-AF
荷蘭文	nl
英文	en
Estonian	et
Farsi (波斯文)	fa
菲律賓文、塔加洛文	tl
芬蘭文	fi
法文	fr
法文 (加拿大)	fr-CA
喬治亞文	ka
德文	de
Greek	el

語言	語言代碼
古吉拉特文	gu
海地克里奧爾	ht
豪沙文	ha
Hebrew	he
北印度文	hi
匈牙利文	hu
冰島文	is
印尼文	id
愛爾蘭文	ga
義大利文	it
日文	ja
坎那達文	kn
哈薩克文	kk
韓文	ko
拉脫維亞文	lv
立陶宛文	lt
馬其頓文	mk
馬來文	ms
馬來亞拉姆文	ml
馬爾他文	mt

語言	語言代碼
馬拉地文	mr
Mongolian	mn
挪威文 (布克莫ål)	no
普什圖文	ps
Polish	pl
葡萄牙文 (巴西)	pt
葡萄牙語 (葡萄牙)	pt-PT
旁遮普文	pa
羅馬尼亞文	ro
俄文	ru
塞爾維亞文	sr
僧伽羅文	si
斯洛伐克文	sk
斯洛維尼亞文	sl
索馬利亞文	so
西班牙文	es
西班牙文 (墨西哥)	es-MX
史瓦西里文	sw
瑞典文	sv
坦米爾文	ta

語言	語言代碼
特拉古	te
Thai	th
Turkish	tr
烏克蘭文	uk
烏都文	ur
烏茲別克文	uz
越南文	vi
威爾斯文	cy

Amazon Translate 功能支援的語言

下列各節說明 Amazon Translate 功能支援的語言。

- 即時文件翻譯 – 支援從英文翻譯到任何支援的語言，以及從任何支援的語言翻譯到英文。如需即時翻譯的詳細資訊，請參閱 [即時翻譯](#)。
- Brevity – 如需此功能支援的語言，請參閱 [在 Amazon Translate 中使用簡潔](#)。
- 褻瀆遮罩 – 如需此功能支援的語言，請參閱 [在 Amazon Translate 中遮罩褻瀆單字和片語](#)。
- 形式 – 如需此功能支援的語言，請參閱 [在 Amazon Translate 中設定正式性](#)。

Amazon Translate 的運作方式

使用 Amazon Translate 服務將內容從來源語言（輸入內容的語言）轉譯為目標語言（您為轉譯輸出選擇的語言）。在批次任務中，您可以將檔案從一或多個來源語言轉譯為一或多個目標語言。如需支援語言的詳細資訊，請參閱 [支援的語言和語言代碼](#)。

主題

- [輸入內容支援的格式](#)
- [自訂您的翻譯](#)
- [自動語言偵測](#)
- [例外狀況處理](#)

輸入內容支援的格式

Amazon Translate 支援輸入內容的下列格式：

- 對於即時翻譯：
 - 輸入文字 – UTF-8 格式的純文字。Amazon Translate 以 UTF-8 文字提供輸出內容。
 - 一個輸入檔案 – 包含純文字 (.txt)、HTML (.html) 或 Word (.docx) 內容的檔案。Amazon Translate 會以與輸入檔案相同的格式提供輸出內容。
- 對於批次翻譯任務：
 - 輸入檔案集合 – 您上傳到 Amazon S3 位置的一或多個檔案。支援的檔案格式包括純文字 (.txt)、HTML (.html)、Word (.docx)、Excel (.xlsx)、PowerPoint (.pptx) 和 XLIFF 1.2 (.xlf)。Amazon Translate 以檔案的形式提供輸出內容。每個輸出檔案的檔案格式都符合輸入檔案格式。

自訂您的翻譯

您可以使用下列功能來自訂您使用 Amazon Translate 產生的翻譯：

- Do-not-translate 標籤 – 使用開始和結束標籤來指定您不想翻譯的內容（以 HTML 內容表示）。
- 自訂術語 – 定義您希望 Amazon Translate 如何翻譯特定術語，例如品牌名稱。
- Brevity – 減少大多數翻譯的翻譯輸出長度（相較於不含簡潔的翻譯輸出）。即時文字翻譯支援 Brevity。

- 褻瀆 – 遮罩翻譯輸出中的褻瀆字詞和片語。
- 形式 – 設定翻譯輸出中的語言形式層級。
- 平行資料 – 調整翻譯輸出，以反映您提供的範例翻譯範例中的樣式、音調和單字選擇。

如需詳細資訊，請參閱[自訂翻譯](#)。

自動語言偵測

Amazon Translate 可以自動偵測來源文字中使用的語言。若要使用自動語言偵測，請將指定 `auto` 為來源語言。Amazon Translate 會代表您呼叫 Amazon Comprehend，以判斷來源文字中使用的語言。選擇自動語言偵測，即表示您同意 Amazon Comprehend 的服務條款和協議。如需 Amazon Comprehend 定價的相關資訊，請參閱 [Amazon Comprehend 定價](#)。

例外狀況處理

如果您指定不支援的來源或目標語言，Amazon Translate 會傳回下列例外狀況：

- `UnsupportedLanguagePairException` – Amazon Translate 支援所有支援語言之間的翻譯。如果來源語言或目標語言不受支援，則會傳回此例外狀況。如需詳細資訊，請參閱[支援的語言](#)。
- `DetectedLanguageLowConfidenceException` – 如果您使用自動語言偵測，且 Amazon Translate 偵測到正確的來源語言，則會傳回此例外狀況。如果信心不足層級是可接受的，則您可以使用例外狀況中傳回的來源語言。

設定

第一次使用 Amazon Translate 之前，請先完成下列任務。

設定任務

- [註冊 AWS 帳戶](#)
- [建立具有管理存取權的使用者](#)
- [安裝和設定 AWS Command Line Interface \(AWS CLI\)](#)
- [授與程式設計存取權](#)
- [搭配 AWS SDK 使用此服務](#)

註冊 AWS 帳戶

如果您沒有 AWS 帳戶，請完成下列步驟來建立一個。

註冊 AWS 帳戶

1. 開啟 <https://portal.aws.amazon.com/billing/signup>。
2. 請遵循線上指示進行。

部分註冊程序需接收來電或簡訊，並在電話鍵盤輸入驗證碼。

當您註冊時 AWS 帳戶，AWS 帳戶根使用者會建立。根使用者有權存取該帳戶中的所有 AWS 服務和資源。作為安全最佳實務，請將管理存取權指派給使用者，並且僅使用根使用者來執行[需要根使用者存取權的任務](#)。

AWS 會在註冊程序完成後傳送確認電子郵件給您。您可以隨時登錄 <https://aws.amazon.com/> 並選擇我的帳戶，以檢視您目前的帳戶活動並管理帳戶。

建立具有管理存取權的使用者

註冊後 AWS 帳戶，請保護 AWS 帳戶根使用者、啟用 AWS IAM Identity Center 和建立管理使用者，以免將根使用者用於日常任務。

保護您的 AWS 帳戶根使用者

1. 選擇根使用者並輸入 AWS 帳戶 您的電子郵件地址，以帳戶擁有者[AWS 管理主控台](#)身分登入。在下一頁中，輸入您的密碼。

如需使用根使用者登入的說明，請參閱 AWS 登入 使用者指南中的[以根使用者身分登入](#)。

2. 若要在您的根使用者帳戶上啟用多重要素驗證 (MFA)。

如需說明，請參閱《IAM 使用者指南》中的[為您的 AWS 帳戶 根使用者 \(主控台 \) 啟用虛擬 MFA 裝置](#)。

建立具有管理存取權的使用者

1. 啟用 IAM Identity Center。

如需指示，請參閱《AWS IAM Identity Center 使用者指南》中的[啟用 AWS IAM Identity Center](#)。

2. 在 IAM Identity Center 中，將管理存取權授予使用者。

如需使用 IAM Identity Center 目錄 做為身分來源的教學課程，請參閱 AWS IAM Identity Center 《使用者指南》中的[使用預設值設定使用者存取 IAM Identity Center 目錄](#)。

以具有管理存取權的使用者身分登入

- 若要使用您的 IAM Identity Center 使用者簽署，請使用建立 IAM Identity Center 使用者時傳送至您電子郵件地址的簽署 URL。

如需使用 IAM Identity Center 使用者登入的說明，請參閱 AWS 登入 《使用者指南》中的[登入 AWS 存取入口網站](#)。

指派存取權給其他使用者

1. 在 IAM Identity Center 中，建立一個許可集來遵循套用最低權限的最佳實務。

如需指示，請參閱《AWS IAM Identity Center 使用者指南》中的[建立許可集](#)。

2. 將使用者指派至群組，然後對該群組指派單一登入存取權。

如需指示，請參閱《AWS IAM Identity Center 使用者指南》中的[新增群組](#)。

安裝和設定 AWS Command Line Interface (AWS CLI)

您可以使用 AWS CLI 對 Amazon Translate 進行互動式呼叫。

安裝和設定 AWS CLI

1. 安裝 AWS CLI。如需說明，請參閱AWS Command Line Interface 《使用者指南》中的下列主題：

[安裝或更新最新版本的 AWS Command Line Interface](#)

2. 設定 AWS CLI。如需說明，請參閱AWS Command Line Interface 《使用者指南》中的下列主題：

[設定 AWS Command Line Interface](#)

授與程式設計存取權

如果使用者想要與 AWS 外部互動，則需要程式設計存取 AWS 管理主控台。授予程式設計存取權的方式取決於正在存取的使用者類型 AWS。

若要授予使用者程式設計存取權，請選擇下列其中一個選項。

哪個使用者需要程式設計存取權？	到	根據
IAM	(建議) 使用主控台登入資料做為臨時登入資料，以簽署對 AWS CLI、AWS SDKs 程式設計請求。AWS APIs	請依照您要使用的介面所提供的指示操作。 <ul style="list-style-type: none"> • 如需 AWS CLI，請參閱AWS Command Line Interface 《使用者指南》中的登入以進行 AWS 本機開發。 • AWS SDKs，請參閱 AWS SDKs 和工具參考指南中的Login for AWS local development。

哪個使用者需要程式設計存取權？	到	根據
人力資源身分 (IAM Identity Center 中管理的使用者)	使用暫時登入資料簽署對 AWS CLI、AWS SDKs 程式設計請求。AWS APIs	請依照您要使用的介面所提供的指示操作。 <ul style="list-style-type: none"> • 如需 AWS CLI，請參閱 AWS Command Line Interface 《使用者指南》中的 設定 AWS CLI 要使用 AWS IAM Identity Center 的。 • AWS SDKs、工具和 AWS APIs，請參閱 AWS SDK 和工具參考指南中的 SDKs IAM Identity Center 身分驗證。
IAM	使用暫時登入資料簽署對 AWS CLI、AWS SDKs 程式設計請求。AWS APIs	遵循《IAM 使用者指南》中 將臨時登入資料與 AWS 資源搭配使用 的指示。
IAM	(不建議使用) 使用長期登入資料來簽署對 AWS CLI、AWS SDKs 程式設計請求。AWS APIs	請依照您要使用的介面所提供的指示操作。 <ul style="list-style-type: none"> • 如需 AWS CLI，請參閱 AWS Command Line Interface 《使用者指南》中的 使用 IAM 使用者憑證進行身分驗證。 • AWS SDKs 和工具，請參閱 AWS SDKs 和工具參考指南中的 使用長期憑證進行身分驗證。 • 對於 AWS APIs，請參閱《IAM 使用者指南》中的 管理 IAM 使用者的存取金鑰。

搭配 AWS SDK 使用此服務

AWS 軟體開發套件 (SDKs) 適用於許多熱門的程式設計語言。每個 SDK 都提供 API、程式碼範例和說明文件，讓開發人員能夠更輕鬆地以偏好的語言建置應用程式。

SDK 文件	代碼範例
適用於 C++ 的 AWS SDK	適用於 C++ 的 AWS SDK 程式碼範例
AWS CLI	AWS CLI 程式碼範例
適用於 Go 的 AWS SDK	適用於 Go 的 AWS SDK 程式碼範例
適用於 Java 的 AWS SDK	適用於 Java 的 AWS SDK 程式碼範例
適用於 JavaScript 的 AWS SDK	適用於 JavaScript 的 AWS SDK 程式碼範例
適用於 Kotlin 的 AWS SDK	適用於 Kotlin 的 AWS SDK 程式碼範例
適用於 .NET 的 AWS SDK	適用於 .NET 的 AWS SDK 程式碼範例
適用於 PHP 的 AWS SDK	適用於 PHP 的 AWS SDK 程式碼範例
AWS Tools for PowerShell	AWS Tools for PowerShell 程式碼範例
適用於 Python (Boto3) 的 AWS SDK	適用於 Python (Boto3) 的 AWS SDK 程式碼範例
適用於 Ruby 的 AWS SDK	適用於 Ruby 的 AWS SDK 程式碼範例
適用於 Rust 的 AWS SDK	適用於 Rust 的 AWS SDK 程式碼範例
適用於 SAP ABAP 的 AWS SDK	適用於 SAP ABAP 的 AWS SDK 程式碼範例
適用於 Swift 的 AWS SDK	適用於 Swift 的 AWS SDK 程式碼範例

可用性範例

找不到所需的內容嗎？請使用本頁面底部的提供意見回饋連結申請程式碼範例。

Amazon Translate 入門

開始使用 Amazon Translate 的最簡單方法是使用 主控台來翻譯一些文字。您也可以從命令列嘗試 API 操作。您也可以安裝其中一個 AWS SDKs，以使用 Amazon Translate API 操作。

主題

- [開始使用 \(主控台\)](#)
- [開始使用 \(AWS CLI\)](#)
- [入門 \(SDK\)](#)

開始使用 (主控台)

開始使用 Amazon Translate 的最簡單方法是使用 主控台來翻譯一些文字。您可以使用 主控台翻譯最多 10,000 個位元組的文字。如果您還沒有檢閱[Amazon Translate 的運作方式](#)中的概念和術語，我們建議您檢閱之後再繼續。

開啟 [Amazon Translate 主控台](#)。

如果這是您第一次使用 Amazon Translate，請選擇啟動即時翻譯。

在即時轉譯中，選擇目標語言。Amazon Translate 會自動偵測來源語言，或者您可以選擇來源語言。在左側文字方塊中輸入您要翻譯的文字。翻譯的文字會出現在右側文字方塊中。

Translation

Text | Documents

Source language: Auto (auto) | Target language: German (de)

Amazon Translate is a neural machine translation service for translating text to and from English across a breadth of supported languages. Powered by deep-learning technologies, Amazon Translate delivers fast, high-quality, and affordable language translation. |

Amazon Translate ist ein neuronaler maschineller Übersetzungsdienst für die Übersetzung von Text ins Englische und aus dem Englischen in einer Vielzahl unterstützter Sprachen. Amazon Translate basiert auf Deep-Learning-Technologien und bietet schnelle, qualitativ hochwertige und erschwingliche Sprachübersetzungen.

262 characters, 262 of 10000 bytes used. [Info](#)

Detected language: English (en)

► Additional settings

Is this translation what you expected? Please leave us [feedback](#)

在應用程式整合區段中，您可以看到 [TranslateText](#) 操作的 JSON 輸入和輸出。

▼ Application integration

Learn more about working with the Translate service using APIs for automation and larger volumes of text. [Info](#)

JSON request

```

1  {
2  "Text": "Amazon Translate is a neural machine
   translation service for translating text to
   and from English across a breadth of supported
   languages. Powered by deep-learning
   technologies, Amazon Translate delivers fast,
   high-quality, and affordable language
   translation. ",
3  "SourceLanguageCode": "auto",
4  "TargetLanguageCode": "de"
5  }

```

[Copy](#)

JSON response

```

1  {
2  "TranslatedText": "Amazon Translate ist ein
   neuronaler maschineller Übersetzungsdienst für
   die Übersetzung von Text ins Englische und aus
   dem Englischen in einer Vielzahl unterstützter
   Sprachen. Amazon Translate basiert auf Deep-
   Learning-Technologien und bietet schnelle,
   qualitativ hochwertige und erschwingliche
   Sprachübersetzungen. ",
3  "SourceLanguageCode": "en",
4  "TargetLanguageCode": "de"
5  }

```

[Copy](#)

開始使用 (AWS CLI)

在下列練習中，您可以使用 AWS 命令列界面 (AWS CLI) 來翻譯文字。若要完成練習，您需要熟悉 CLI 並擁有文字編輯器。如需詳細資訊，請參閱[安裝和設定 AWS Command Line Interface \(AWS CLI\)](#)。

若要從命令列使用 Amazon Translate，您需要從支援 Amazon Translate 服務的區域執行命令。如需可用端點和區域的清單，請參閱 AWS 一般參考中的 [Amazon Translate 區域和端點](#)。

使用命令列翻譯文字

下列範例示範如何使用命令列中的 [TranslateText](#) 操作來翻譯文字。此範例格式適用於 Unix、Linux 和 macOS。用於 Windows 時，請以插入號 (^) 取代每一行結尾處的 Unix 接續字元斜線 (\)。在命令列中輸入以下內容。

```
aws translate translate-text \  
    --region region \  
    --source-language-code "en" \  
    --target-language-code "es" \  
    --text "hello, world"
```

回應為以下 JSON：

```
{  
  "TargetLanguageCode": "es",  
  "Text": "Hola, mundo",  
  "SourceLanguageCode": "en"  
}
```

下一步驟

若要查看使用 Amazon Translate 的其他方法，請參閱 [Amazon Translate AWS SDKs 的程式碼範例](#)。

入門 (SDK)

AWS 提供適用於各種電腦語言 SDKs。SDK 會管理用戶端的許多 API 連線詳細資訊，例如簽章計算、請求重試處理和錯誤處理。如需詳細資訊，請參閱 [AWS 開發套件](#)。

下列範例示範如何使用 Java 和 Python 來使用 Amazon Translate [TranslateText](#) 操作。使用 SDKs 來了解 Amazon Translate API 和 作為您自己應用程式的建置區塊。

主題

- [使用 翻譯文字 適用於 Java 的 AWS SDK](#)
- [使用 翻譯文字 適用於 Python \(Boto\) 的 AWS SDK](#)
- [其他 SDK 範例](#)

使用 翻譯文字 適用於 Java 的 AWS SDK

AWS 提供 [GitHub 範例](#)，說明如何在 Java 中使用 [TranslateText](#) 操作。若要執行此範例，您需要適用於 Java 的 AWS SDK。如需安裝適用於 Java 的開發套件的指示，請參閱[設定適用於 Java 2.x 的 AWS 開發套件](#)。

使用 翻譯文字 適用於 Python (Boto) 的 AWS SDK

下列範例示範如何在 Python 中使用 [TranslateText](#) 操作。若要執行範例，請透過安裝 Python SDK AWS CLI。如需說明，請參閱 [the section called “設定 AWS CLI”](#)。

```
import boto3

translate = boto3.client(service_name='translate', region_name='region', use_ssl=True)

result = translate.translate_text(Text="Hello, World",
                                  SourceLanguageCode="en", TargetLanguageCode="de")
print('TranslatedText: ' + result.get('TranslatedText'))
print('SourceLanguageCode: ' + result.get('SourceLanguageCode'))
print('TargetLanguageCode: ' + result.get('TargetLanguageCode'))
```

如需支援的語言代碼清單，請參閱 [支援的語言和語言代碼](#)

其他 SDK 範例

如需使用 .NET 和 SAP ABAP 的範例，[Amazon Translate AWS SDKs的程式碼範例](#)請參閱。

轉譯處理模式

翻譯文件時，您可以使用兩種不同的翻譯處理模式：即時翻譯或非同步批次處理。您使用的模式會以目標文件的大小和類型為基礎，而且會影響提交翻譯工作和檢視其結果的方式。

- [即時翻譯](#) – 您提出同步請求來翻譯少量文字（或文字檔案），Amazon Translate 會立即以翻譯的文字回應。
- [非同步批次處理](#) – 您將文件集合放在 Amazon Simple Storage Service (Amazon S3) 位置，並啟動非同步處理任務進行翻譯。Amazon Translate 會將翻譯的輸出文件傳送至指定的 Amazon S3 位置。

即時翻譯

Amazon Translate 提供即時文件和文字翻譯操作，可立即傳回翻譯。您可以使用 主控台或 API 來執行即時轉譯。

主題

- [使用主控台即時翻譯](#)
- [使用 API 進行即時轉譯](#)

使用主控台即時翻譯

若要使用主控台進行即時翻譯，請將輸入文字貼到來源語言文字方塊，或提供輸入文字做為檔案。或者，您可以設定所需正式程度、褻瀆遮罩、簡潔和自訂術語等功能。

您可以搭配即時翻譯使用自動語言偵測，但可能會產生費用。如需詳細資訊，請參閱[自動語言偵測](#)。

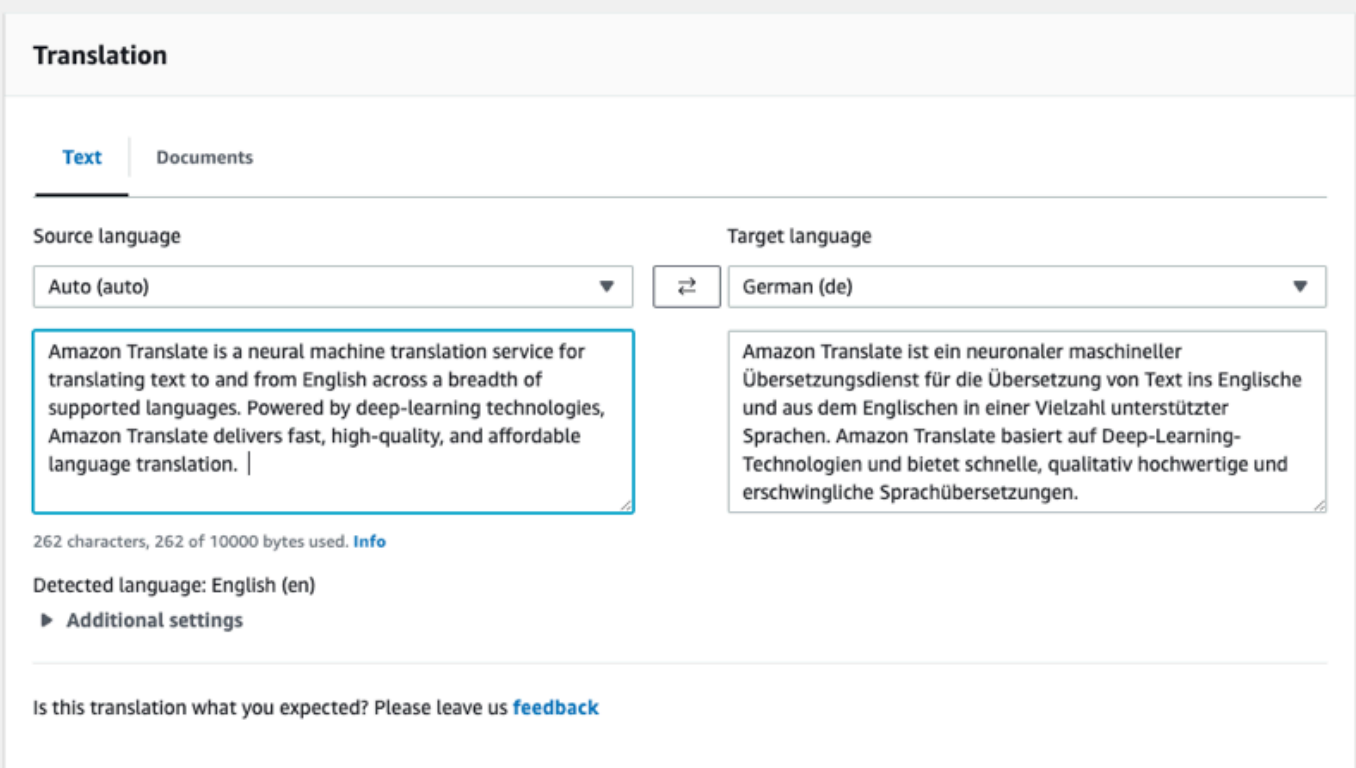
主題

- [翻譯文字](#)
- [翻譯文件](#)
- [檢視同等 API 請求和回應資料](#)
- [使用翻譯功能](#)

翻譯文字

使用 Amazon Translate 主控台來翻譯最多 10,000 個位元組的文字。

1. 開啟 [Amazon Translate 主控台](#)。
2. 在左側的導覽功能表中，選擇即時轉譯。
3. 針對來源語言，選取來源文字的語言，或將值保留為自動以進行自動偵測。
4. 針對目標語言，選取語言。
5. 在來源語言文字方塊中輸入或貼上文字。主控台會在目標語言文字方塊中顯示翻譯的文字。



The screenshot shows the Amazon Translate web interface for text translation. At the top, there's a 'Translation' header with two tabs: 'Text' (selected) and 'Documents'. Below the tabs, there are two dropdown menus for 'Source language' (set to 'Auto (auto)') and 'Target language' (set to 'German (de)'). A double-headed arrow icon is between the dropdowns. Two text input areas are shown: the left one contains the English source text 'Amazon Translate is a neural machine translation service for translating text to and from English across a breadth of supported languages. Powered by deep-learning technologies, Amazon Translate delivers fast, high-quality, and affordable language translation.' and the right one contains the German translation 'Amazon Translate ist ein neuronaler maschineller Übersetzungsdienst für die Übersetzung von Text ins Englische und aus dem Englischen in einer Vielzahl unterstützter Sprachen. Amazon Translate basiert auf Deep-Learning-Technologien und bietet schnelle, qualitativ hochwertige und erschwingliche Sprachübersetzungen.' Below the input areas, it shows '262 characters, 262 of 10000 bytes used.' and 'Detected language: English (en)'. There is a link for 'Additional settings' and a feedback prompt at the bottom: 'Is this translation what you expected? Please leave us [feedback](#)'.

翻譯文件

即時文件翻譯支援從英文翻譯到任何支援的語言，以及從任何支援的語言翻譯到英文。

若要使用 Amazon Translate 主控台翻譯文件：

1. 開啟 [Amazon Translate 主控台](#)。
2. 在左側的導覽功能表中，選擇即時轉譯。
3. 在翻譯面板中，選擇文件索引標籤。

Translation

Text | **Document**

Source language Target language

Choose English for either the source language or the target language.

Upload file

Supported file extensions include .html and .txt. The maximum file size is 100 KB.

Document type

▶ **Additional settings**

Is this translation what you expected? Please leave us [feedback](#)

4. 針對來源語言，選取來源文字的語言，或選取自動進行自動偵測。
5. 針對目標語言，選取語言。如果來源語言不是英文，您必須為目標語言選取英文。
6. 在上傳檔案下，選擇選擇檔案，然後輸入來源檔案的路徑。檔案大小上限為 100 KB。
7. 針對文件類型，選取轉譯來源檔案的格式。文件翻譯支援純文字、HTML 或 Word (.docx) 輸入檔案。
8. 選擇翻譯。

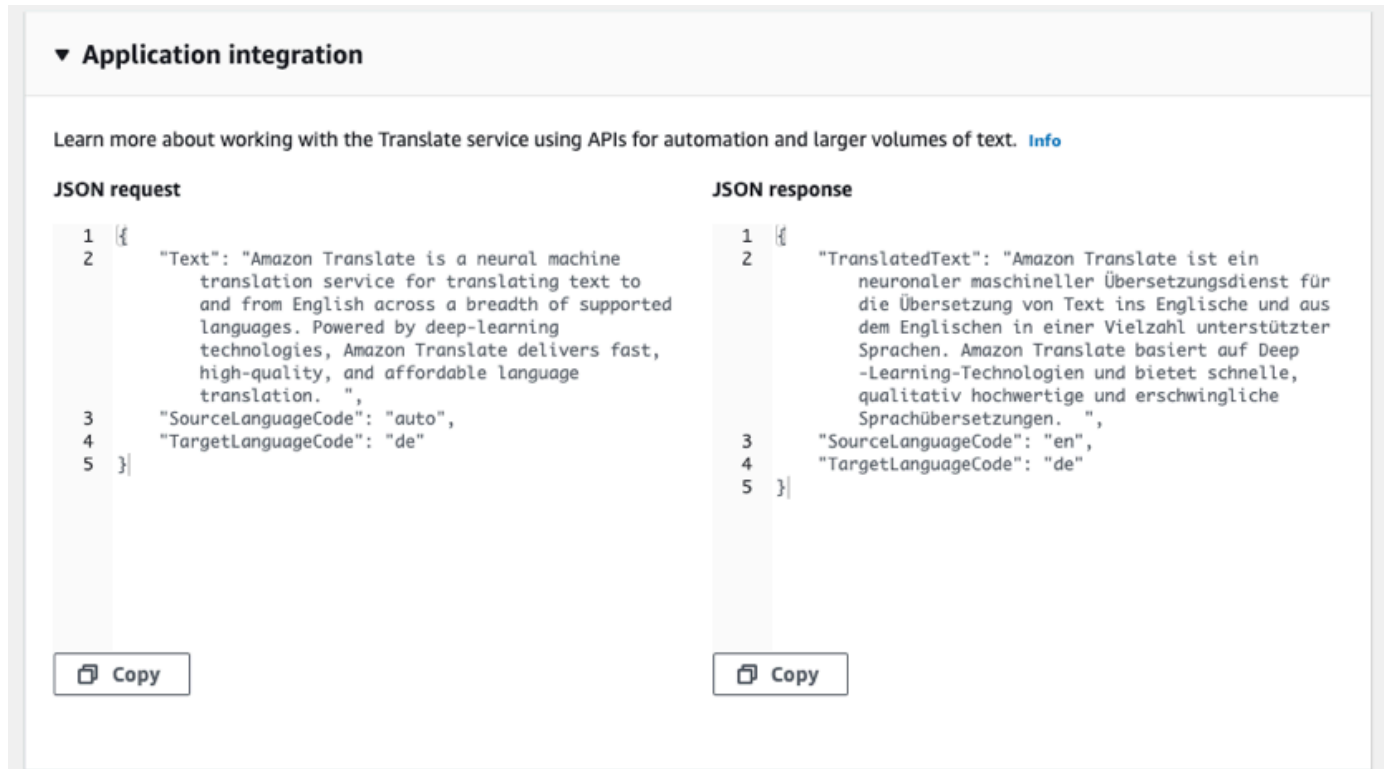
翻譯任務完成後，請選擇下載翻譯，將翻譯文件下載到本機硬碟。翻譯文件的格式（文字、HTML 或 Word）符合輸入文件。

檢視同等 API 請求和回應資料

使用主控台翻譯輸入文字或文件後，您可以檢視同等 API 請求資料和 JSON 格式的回應資料。

1. 在轉譯面板下方，展開應用程式整合面板。

主控台會以 JSON 格式顯示同等的轉譯請求資料。



The screenshot displays the 'Application integration' section of the Amazon Translate console. It features a header with a dropdown arrow and the text 'Application integration'. Below this, there is a link to learn more about working with the Translate service using APIs for automation and larger volumes of text. The main content is divided into two columns: 'JSON request' and 'JSON response'. The 'JSON request' column shows a JSON object with a 'Text' field containing an English paragraph about Amazon Translate, and 'SourceLanguageCode' and 'TargetLanguageCode' fields set to 'auto' and 'de' respectively. The 'JSON response' column shows the corresponding JSON object with a 'TranslatedText' field containing the German translation of the text, and 'SourceLanguageCode' and 'TargetLanguageCode' fields set to 'en' and 'de' respectively. Both columns have a 'Copy' button at the bottom.

2. 您可以複製要在 [TranslateText](#) 或 [TranslateDocument](#) API 操作中使用的 JSON 請求。
3. JSON 回應面板中的 JSON 輸出符合 API 產生的輸出。

使用翻譯功能

若要搭配 Amazon Translate 主控台使用翻譯功能：

1. 開啟 [Amazon Translate 主控台](#)。
2. 在左側的導覽功能表中，選擇即時轉譯。
3. 提供來源語言、目標語言和輸入資料（文字或文件），如先前程序所述。
4. 在其他設定下，您可以選擇使用以下設定自訂翻譯任務的輸出：

自訂術語

選取自訂術語檔案。如果檔案在輸入文字中有來源詞彙的項目，Amazon Translate 會使用術語檔案中的轉譯。

如需詳細資訊，請參閱 [使用自訂術語自訂翻譯](#)。

Brevity

減少大多數翻譯的翻譯輸出長度（相較於不含簡潔的翻譯輸出）。Amazon Translate 支援翻譯文字的簡潔性，但不支援翻譯文件。如果來源和目標語言形成不支援的語言對以求簡潔，Amazon Translate 會忽略簡潔設定。

如需支援的語言資訊，請參閱 [在 Amazon Translate 中使用簡潔](#)。

褻瀆

遮罩翻譯輸出中的褻瀆單字和片語。Amazon Translate 不支援所有支援語言的褻瀆遮罩。

如需詳細資訊，請參閱 [在 Amazon Translate 中遮罩褻瀆單字和片語](#)。

形式

對於某些目標語言，您可以將形式設定為正式或非正式。如果正規不支援目標語言，Amazon Translate 會忽略正規設定。

如需詳細資訊，請參閱 [在 Amazon Translate 中設定正式性](#)。

5. 針對文件翻譯，選擇翻譯以使用所選功能翻譯文件。

對於文字翻譯，當您選擇每個功能時，主控台會將翻譯功能套用至翻譯的文字。

使用 API 進行即時轉譯

Amazon Translate 提供下列即時翻譯操作，以支援互動式應用程式：

- [TranslateText](#) – 翻譯文字區塊。
- [TranslateDocument](#) – 翻譯檔案的內容（純文字、HTML 或 .docx）。

這些同步操作會將轉譯結果直接傳回至您的應用程式。如果您搭配這些操作使用自動語言偵測，可能會產生費用。如需詳細資訊，請參閱 [自動語言偵測](#)。

翻譯文字

使用 [TranslateText](#) 操作來翻譯單一文字區塊。

使用命令列翻譯文字

下列範例示範如何從命令列使用 [TranslateText](#) 操作。此範例格式適用於 Unix、Linux 和 macOS。用於 Windows 時，請以插入號 (^) 取代每一行結尾處的 Unix 接續字元斜線 (\)。

在命令列輸入下列命令。

```
aws translate translate-text \  
    --region region \  
    --source-language-code "en" \  
    --target-language-code "es" \  
    --text "hello, world"
```

命令會以下列 JSON 回應：

```
{  
  "TargetLanguageCode": "es",  
  "TranslatedText": "Hola, mundo",  
  "SourceLanguageCode": "en"  
}
```

使用 JSON 檔案翻譯文字

此範例示範如何使用 JSON 檔案來翻譯較長的文字區塊。您可以在命令列指定來源和目標語言，或在 JSON 檔案中指定它們。

Note

範例 JSON 檔案的格式具有可讀性。重新格式化 "Text" 欄位，以移除換行符號。此範例格式適用於 Unix、Linux 和 macOS。用於 Windows 時，請以插入號 (^) 取代每一行結尾處的 Unix 接續字元斜線 (\)。

使用 JSON 檔案翻譯文字

1. 將下列文字複製到名為 `translate.json` 的 JSON 檔案。

```
{  
  "Text": "Amazon Translate translates documents between languages in  
  real time. It uses advanced machine learning technologies  
  to provide high-quality real-time translation. Use it to
```

```
translate documents or to build applications that work in
multiple languages.",
"SourceLanguageCode": "en",
"TargetLanguageCode": "fr"
}
```

2. 在中 AWS CLI，執行下列命令：

```
aws translate translate-text \
    --region region \
    --cli-input-json file://translate.json > translated.json
```

此命令輸出 JSON 檔案，其中包含以下 JSON 文字：

```
{
  "TargetLanguageCode": "fr",
  "TranslatedText": "Amazon Translate traduit les documents entre
les langue en temps réel. Il utilise des technologies
avancées d'apprentissage de la machine pour fournir
une traduction en temps réel de haute qualité. Utilisez-le
pour traduire des documents ou pour créer des applications
qui fonctionnent en plusieurs langues.",
  "SourceLanguageCode": "en"
}
```

翻譯文件

使用 [TranslateDocument](#) 操作來翻譯文字、HTML 或 Word (.docx) 文件，並將翻譯結果直接傳回至您的應用程式。

即時文件翻譯支援從英文翻譯到任何支援的語言，以及從任何支援的語言翻譯到英文。您可以指定來源語言程式碼或使用自動偵測。

使用命令列翻譯文件

下列範例示範如何從命令列使用 [TranslateDocument](#) 操作。此範例格式適用於 Unix、Linux 和 macOS。用於 Windows 時，請以插入號 (^) 取代每一行結尾處的 Unix 接續字元斜線 (\)。

在命令列輸入下列命令。

```
aws translate translate-document \
```

```
--region region \  
--source-language-code "en" \  
--target-language-code "es" \  
--document-content fileb://source-lang.txt  
--document ContentType=text/plain  
--query "TranslatedDocument.Content"  
--output text | base64  
--decode > target-lang.txt
```

命令會以下列 JSON 回應：

```
{  
  "SourceLanguageCode": "en",  
  "TargetLanguageCode": "es",  
  "TranslatedDocument": {  
    "Content": blob  
  }  
}
```

使用 Amazon Translate 進行非同步批次處理

若要翻譯大型文件集合（大小上限為 5 GB），請使用 Amazon Translate 非同步批次處理操作 [StartTextTranslationJob](#)。這最適合收集簡短文件，例如社交媒體貼文或使用者的評論，或任何不需要即時翻譯的情況。

若要執行非同步批次翻譯，您通常可以執行下列步驟：

1. 將一組文件存放在 Amazon S3 儲存貯體內的輸入資料夾中。
2. 開始批次翻譯工作。
3. 在您的請求中，為 Amazon Translate 提供對輸入 Amazon S3 資料夾及其所有子資料夾具有讀取存取權的 IAM 角色。此角色也必須具有輸出 Amazon S3 儲存貯體的讀取和寫入存取權。
4. 監控批次翻譯工作的進度。
5. 從指定的輸出儲存貯體擷取批次轉譯任務的結果。

區域可用性

下列 AWS 區域支援批次轉譯：

- 美國東部 (俄亥俄)

- 美國東部 (維吉尼亞北部)
- 美國西部 (加利佛尼亞北部)
- 美國西部 (奧勒岡)
- 亞太區域 (孟買)
- 亞太區域 (首爾)
- 亞太區域 (新加坡)
- 亞太地區 (雪梨)
- 亞太區域 (東京)
- 加拿大 (中部)
- 歐洲 (法蘭克福)
- 歐洲 (愛爾蘭)
- 歐洲 (倫敦)
- 歐洲 (巴黎)
- 歐洲 (斯德哥爾摩)

主題

- [批次翻譯任務的先決條件](#)
- [執行批次翻譯任務](#)
- [監控和分析批次翻譯任務](#)
- [取得批次翻譯結果](#)

批次翻譯任務的先決條件

必須符合下列先決條件，Amazon Translate 才能執行成功的批次翻譯任務：

- 包含輸入和輸出文件的 Amazon S3 儲存貯體必須與您呼叫的 API 端點位於相同的 AWS 區域。
- 批次輸入文件的集合大小必須為 5 GB 或更小。
- 批次翻譯工作中最多可提交 100 萬個文件。
- 每個輸入文件都必須為 20 MB 或更少，且必須包含少於 100 萬個字元。
- 您的輸入檔案必須位於 Amazon S3 儲存貯體的資料夾中。如果您將輸入檔案新增至儲存貯體的頂層，Amazon Translate 會在您嘗試執行批次翻譯任務時擲回錯誤。此要求適用於輸入檔案。輸出檔案不需要資料夾，Amazon Translate 可以將它們放在 Amazon S3 儲存貯體的最上層。

- 您的輸入檔案資料夾可以包含巢狀資料夾。請確定沒有任何巢狀資料夾名為 details，否則當您嘗試執行批次轉譯任務時，Amazon Translate 會擲回錯誤。

支援檔案格式

Amazon Translate 支援批次翻譯任務的下列檔案類型：

- 純文字。
- HTML。
- Word 文件 (.docx)。
- PowerPoint 簡報檔案 (.pptx)。
- Excel 手冊檔案 (.xlsx)。
- XML 當地語系化交換檔案格式 (XLIFF) 檔案 (.xlf)。Amazon Translate 僅支援 XLIFF 1.2 版。

Amazon Translate 要求檔案進行 UTF-8 編碼。

先決條件許可

在您可以執行批次翻譯任務之前，AWS 您的帳戶必須在 IAM 中具有服務角色。此角色必須具有授予 Amazon Translate 的許可政策：

- 在 Amazon S3 中讀取對輸入資料夾及其所有子資料夾的存取權。
- 您的輸出儲存貯體的讀取和寫入權限。

它還必須包含允許 Amazon Translate 擔任角色並取得其許可的信任政策。此信任政策必須允許 `translate.amazonaws.com` 服務主體執行 `sts:AssumeRole` 動作。

當您使用 Amazon Translate 主控台建立批次翻譯任務時，您可以選擇允許 Amazon Translate 自動為您建立此角色。當您使用 AWS CLI 或 Amazon Translate API 執行批次翻譯任務時，您可以在請求中提供角色的 Amazon Resource Name (ARN)。

如需詳細資訊，請參閱 IAM 使用者指南 https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles_create_for_service.html 中的建立角色以委派許可給 AWS 服務。

Example 許可政策

下列範例許可政策會授予對 Amazon S3 儲存貯體中輸入資料夾的讀取存取權。它授予對輸出儲存貯體的讀取和寫入存取權。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "s3:GetObject",
      "Resource": [
        "arn:aws:s3:::input-bucket-name/*",
        "arn:aws:s3:::output-bucket-name/*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": "s3:ListBucket",
      "Resource": [
        "arn:aws:s3:::input-bucket-name",
        "arn:aws:s3:::output-bucket-name"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:PutObject"
      ],
      "Resource": "arn:aws:s3:::output-bucket-name/*"
    }
  ]
}
```

Example信任政策

下列信任政策允許 Amazon Translate 擔任政策所屬的 IAM 角色。

我們建議您驗證使用信任政策 AWS 的帳戶，以緩解[混淆代理人](#)問題。此範例使用 `aws:SourceArn` 和 `aws:SourceAccount` 條件金鑰來驗證來源帳戶。輸入提交批次翻譯任務 AWS 的帳戶。

JSON

```
{
```

```
"Version": "2012-10-17",
"Statement": [
  {
    "Effect": "Allow",
    "Principal": {
      "Service": "translate.amazonaws.com"
    },
    "Action": "sts:AssumeRole",
    "Condition": {
      "ArnLike": {
        "aws:SourceArn": "arn:aws:translate:*:111122223333:"
      },
      "StringEquals": {
        "aws:SourceAccount": "111122223333"
      }
    }
  }
]
```

自訂加密的先決條件許可

您可以在 Amazon Translate 中自訂加密設定，但您必須先將許可新增至 IAM 中的服務角色。

Amazon Translate 會加密您在執行批次翻譯任務時產生的翻譯輸出。根據預設，它會使用執行此加密 AWS 受管金鑰。這種類型的金鑰是由建立，AWS 並儲存在帳戶中的 AWS Key Management Service (AWS KMS) 中。不過，您無法自行管理此 KMS 金鑰。它由代您管理和使用 AWS。

或者，您可以選擇使用客戶受管金鑰來加密輸出，這是您在 AWS 帳戶中建立、擁有和管理的 KMS 金鑰。

您的金鑰必須具有可讓 Amazon Translate 使用金鑰政策。金鑰政策透過將其許可授予允許 Amazon Translate 存取 Amazon S3 儲存貯體的服務角色來執行此操作。

金鑰政策允許服務角色執行加密輸出所需的 AWS KMS 操作，如下列範例政策陳述式所示。

Example KMS 金鑰政策陳述式

```
{
  "Effect": "Allow",
  "Principal":
  {
```

```
"AWS":
[
  "arn:aws:iam::111122223333:role/AmazonTranslateServiceRoleS3FullAccess"
],
"Action":
[
  "kms:Decrypt",
  "kms:GenerateDataKey",
  "kms:CreateGrant",
  "kms:RetireGrant",
  "kms:DescribeKey"
],
"Resource": "*"
}
```

如需詳細資訊，請參閱《AWS Key Management Service 開發人員指南》中的 [金鑰政策 AWS KMS](#)

從另一個 AWS 帳戶使用 AWS KMS 金鑰的許可

如果您想要使用與使用 Amazon Translate 不同 AWS 帳戶中的 KMS 金鑰，則必須：

1. 在 IAM 中更新 Amazon Translate 的服務角色。
2. 在 中更新金鑰政策 AWS KMS。

若要更新您的服務角色，請連接允許其使用其他 AWS 帳戶中的 KMS 金鑰執行必要 AWS KMS 操作的政策，如下列範例所示。

Example 授予不同帳戶中 KMS 金鑰存取權的 IAM 政策

```
{
  "Effect": "Allow",
  "Action":
  [
    "kms:Decrypt",
    "kms:GenerateDataKey",
    "kms:CreateGrant",
    "kms:RetireGrant",
    "kms:DescribeKey"
  ],
  "Resource": "arn:aws:kms:us-west-2:111122223333:key/key-id"
}
```

```
}
```

若要更新您的 KMS 金鑰政策，請將服務角色和管理使用者新增為允許使用金鑰的主體，如下列範例政策陳述式所示。

Example 允許 IAM 角色使用金鑰的 KMS 金鑰政策陳述式

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": [
      "arn:aws:iam::444455556666:role/AmazonTranslateServiceRoleS3FullAccess",
      "arn:aws:iam::444455556666:admin"
    ]
  },
  "Action": [
    "kms:Decrypt",
    "kms:CreateGrant",
    "kms:GenerateDataKey",
    "kms:RetireGrant",
    "kms:DescribeKey"
  ],
  "Resource": "*"
}
```

如需詳細資訊，請參閱《AWS Key Management Service 開發人員指南》中的[允許其他帳戶中的使用者使用 AWS KMS 金鑰](#)

執行批次翻譯任務

您可以使用 Amazon Translate 主控台、AWS CLI 或 Amazon Translate API 執行批次翻譯任務。

Note

批次翻譯工作是長時間執行的作業，可能需要大量時間才能完成。例如，小型資料集上的批次轉譯可能需要幾分鐘的時間，而非常大型的資料集可能需要最多 2 天或更久的時間。完成時間也取決於資源的可用性。

Amazon Translate 主控台

若要使用 Amazon Translate 主控台執行翻譯任務，請使用批次翻譯頁面來建立任務：

1. 開啟 [Amazon Translate 主控台](#)。
2. 在左側的導覽功能表中，選擇批次翻譯。
3. 在翻譯任務頁面上，選擇建立任務。主控台會顯示建立翻譯任務頁面。
4. 在任務設定下，執行下列動作：
 - a. 針對名稱，輸入批次翻譯任務的自訂名稱。
 - b. 針對來源語言，選取來源檔案的語言。如果您不知道來源檔案的語言，或輸入文件包含不同的來源語言，請選取 auto。Amazon Translate 會自動偵測每個檔案的來源語言。
 - c. 針對目標語言，最多選取 10 種語言。Amazon Translate 會將每個來源檔案轉譯為每種目標語言。
5. 在輸入資料下，執行下列動作：
 - a. 針對輸入 S3 位置，指定在 Amazon S3 中包含轉譯來源檔案的輸入資料夾。若要在 Amazon S3 中導覽至資料夾以提供資料夾，請選擇選取資料夾。
 - b. 針對檔案格式，選取轉譯來源檔案的格式。
6. 在輸出資料下，執行下列動作：
 - a. 對於輸出 S3 位置，在 Amazon S3 中指定輸出資料夾，Amazon Translate 會在其中放置轉譯輸出。若要在 Amazon S3 中導覽至資料夾以提供資料夾，請選擇選取資料夾。
 - b. 或者，如果您想要使用您在 () 中管理的客戶受管金鑰來加密輸出，請選擇自訂加密設定 (進階)AWS KMS。AWS Key Management Service

根據預設，Amazon Translate 會使用代表您建立、管理和使用的 KMS 金鑰來加密翻譯輸出 AWS。如果您想要改用自己的 KMS 金鑰來加密輸出，請選擇此選項。

如果您想要使用目前 AWS 帳戶中的 KMS 金鑰，請在選擇 AWS Key Management Service 金鑰下選取它。或者，如果您想要使用來自不同 AWS 帳戶的 KMS 金鑰，請輸入該金鑰的 Amazon Resource Name (ARN)。

Note

您必須先將許可新增至 IAM 中 Amazon Translate 的服務角色，才能使用自己的 KMS 金鑰。如果您想要使用不同帳戶的 KMS 金鑰，您還必須在 [中更新金鑰政策 AWS KMS](#)。如需詳細資訊，請參閱[自訂加密的先決條件許可](#)。

7. 在自訂 - 選用下，您可以選擇使用以下設定自訂翻譯任務的輸出：

褻瀆

遮罩翻譯輸出中的褻瀆單字和片語。如果您為任務指定多種目標語言，所有目標語言都必須支援褻瀆遮罩。如果任何目標語言不支援褻瀆遮罩，翻譯任務不會遮罩任何目標語言的褻瀆遮罩。

如需詳細資訊，請參閱[在 Amazon Translate 中遮罩褻瀆單字和片語](#)。

Brevity

Amazon Translate 不支援批次翻譯工作的簡潔性。

如需詳細資訊，請參閱[在 Amazon Translate 中使用簡潔](#)。

形式

對於某些目標語言，您可以將形式設定為正式或非正式。如果您為任務指定多種目標語言，轉譯會忽略任何不支援的目標語言的正規設定。

如需詳細資訊，請參閱[在 Amazon Translate 中設定正式性](#)。

自訂術語

包含範例來源詞彙和每個詞彙所需的翻譯。如果您為任務指定多種目標語言，轉譯會針對每個請求的目標語言使用指定的術語，該語言在術語檔案中具有來源術語的項目。

如需詳細資訊，請參閱[使用自訂術語自訂翻譯](#)。

平行資料

由範例組成，顯示您希望如何翻譯文字區段。如果您為任務指定多種目標語言，平行資料檔案必須包含所有目標語言的翻譯。

當您將平行資料新增至批次翻譯任務時，您會建立作用中的自訂翻譯任務。

Note

作用中的自訂轉譯任務定價比不使用平行資料的其他任務高。如需詳細資訊，請參閱 [Amazon Translate 定價](#)。

如需詳細資訊，請參閱 [使用平行資料自訂翻譯（主動自訂翻譯）](#)。

8. 在存取許可下，向 Amazon Translate 提供 IAM 角色，將必要的許可授予 Amazon S3 中的輸入和輸出檔案：
 - 如果您的帳戶中已有此 IAM 角色，請選擇使用現有的 IAM 角色，然後在 IAM 角色下選取該角色。
 - 如果您的帳戶中還沒有此 IAM 角色，請選擇建立 IAM 角色。針對 IAM 角色，選擇輸入和輸出 S3 儲存貯體。對於角色名稱，請提供自訂名稱。當您建立翻譯任務時，Amazon Translate 會自動建立角色。IAM 中的角色名稱字首為 AmazonTranslateServiceRole-。

Note

如果您選擇使用自己的 KMS 金鑰加密翻譯輸出，則無法選擇建立 IAM 角色。在這種情況下，您必須使用預先存在的 IAM 角色，而您的 KMS 金鑰必須具有允許角色使用金鑰的金鑰政策。

如需詳細資訊，請參閱 [自訂加密的先決條件許可](#)

9. 選擇建立作業。

主控台會返回翻譯任務頁面，其中任務建立狀態會顯示在頁面頂端的橫幅中。幾分鐘後，您的任務會顯示在表格中。

10. 在名稱欄中選擇任務名稱，以開啟任務詳細資訊頁面。

當您的翻譯任務執行時，狀態欄位會顯示進行中。

11. 當狀態變成已完成時，選擇輸出檔案位置下方的連結，前往您的翻譯輸出。主控台會前往 Amazon S3 中的輸出儲存貯體。
12. 若要下載輸出檔案，請選取每個檔案的核取方塊，然後選擇下載。

AWS CLI

若要使用 執行翻譯任務 AWS CLI，請使用 [start-text-translation-job](#) 命令，並指定 `parallel-data-names` 參數的平行資料資源名稱。

Example Start-text-translation-job 命令

下列範例會透過提交存放在 Amazon S3 輸入儲存貯體中的 Excel 檔案來執行翻譯任務。此任務由請求中包含的平行資料自訂。

```
$ aws translate start-text-translation-job \  
> --input-data-config ContentType=application/vnd.openxmlformats-officedocument.spreadsheetml.sheet,S3Uri=s3://amzn-s3-demo-bucket/input/ \  
> --output-data-config S3Uri=s3://amzn-s3-demo-bucket/output/ \  
> --data-access-role-arn arn:aws:iam::111122223333:role/my-iam-role \  
> --source-language-code en \  
> --target-language-codes es it \  
> --job-name my-translation-job
```

如果命令成功，Amazon Translate 會回應任務 ID 和狀態：

```
{  
  "JobId": "4446f95f20c88a4b347449d3671fbe3d",  
  "JobStatus": "SUBMITTED"  
}
```

如果您想要自訂翻譯任務的輸出，您可以使用下列參數：

--settings

設定轉換輸出的設定，包括下列選項：

在轉譯輸出中開啟簡潔性。Amazon Translate 不支援批次翻譯工作的簡潔性。如需詳細資訊，請參閱在 [Amazon Translate 中使用簡潔](#)。

啟用褻瀆性以遮罩褻瀆單字和片語。若要啟用，請將褻瀆參數設定為 `Profanity=MASK`。如需詳細資訊，請參閱在 [Amazon Translate 中遮罩褻瀆單字和片語](#)。如果任何目標語言不支援褻瀆遮罩，翻譯任務不會遮罩任何目標語言的褻瀆遮罩。

設定轉譯輸出中的正式程度。將 `Formality` 參數設定為 `FORMAL` 或 `INFORMAL`。如果您為任務指定多種目標語言，轉譯會忽略任何不支援的目標語言的正規設定。如需詳細資訊，請參閱在 [Amazon Translate 中設定正式性](#)。

--terminology-names

要新增至翻譯任務的自訂術語資源名稱。此資源會列出範例來源詞彙，以及每個詞彙所需的翻譯。如果您為任務指定多種目標語言，轉譯會針對每個請求的目標語言使用指定的術語，該語言在術語檔案中具有來源術語的項目。

此參數僅接受一個自訂術語資源。

如需可用的自訂術語資源清單，請使用 [list-terminologies](#) 命令。

如需詳細資訊，請參閱 [使用自訂術語自訂翻譯](#)。

--parallel-data-names

要新增至翻譯任務的平行資料資源名稱。此資源包含範例，顯示您希望如何翻譯文字區段。如果您為任務指定多種目標語言，平行資料檔案必須包含所有目標語言的翻譯。

當您將平行資料新增至翻譯任務時，您可以建立作用中的自訂翻譯任務。

此參數僅接受一個平行資料資源。

Note

作用中的自訂轉譯任務定價比不使用平行資料的其他任務高。如需詳細資訊，請參閱 [Amazon Translate 定價](#)。

如需可用的平行資料資源清單，請使用 [list-parallel-data](#) 命令。

如需詳細資訊，請參閱 [使用平行資料自訂翻譯（主動自訂翻譯）](#)。

若要檢查翻譯任務的狀態，請使用 [describe-text-translation-job](#) 命令。

Example Describe-text-translation-job 命令

下列範例會提供任務 ID 來檢查任務狀態。此 ID 是由 Amazon Translate 在 start-text-translation-job 命令啟動任務時提供。

```
$ aws translate describe-text-translation-job \  
> --job-id 4446f95f20c88a4b347449d3671fbe3d
```

Amazon Translate 會回應任務屬性，其中包括其狀態：

```
{
  "TextTranslationJobProperties": {
    "JobId": "4446f95f20c88a4b347449d3671fbe3d",
    "JobName": "my-translation-job",
    "JobStatus": "COMPLETED",
    "JobDetails": {
      "TranslatedDocumentsCount": 0,
      "DocumentsWithErrorsCount": 0,
      "InputDocumentsCount": 1
    },
    "SourceLanguageCode": "en",
    "TargetLanguageCodes": [
      "es",
      "it"
    ],
    "SubmittedTime": 1598661012.468,
    "InputDataConfig": {
      "S3Uri": "s3://amzn-s3-demo-bucket/input/",
      "ContentType": "application/vnd.openxmlformats-officedocument.spreadsheetml.sheet"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://amzn-s3-demo-bucket/output/111122223333-TranslateText-4446f95f20c88a4b347449d3671fbe3d/"
    },
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/my-iam-role"
  }
}
```

如果該JobStatus值為 IN_PROGRESS，請等待幾分鐘的時間通過，然後[describe-text-translation-job](#)再次執行，直到狀態為 為止COMPLETED。當任務完成時，您可以在 下 S3Uri 欄位提供的位置下載翻譯結果OutputDataConfig。

Amazon Translate API

若要使用 Amazon Translate API 提交批次翻譯任務，請使用 [StartTextTranslationJob](#) 操作。

監控和分析批次翻譯任務

您可以使用任務的 ID 來監控其進度，並取得其輸出文件的 Amazon S3 位置。若要監控特定任務，請使用 [DescribeTextTranslationJob](#) 操作。您也可以使用 [ListTextTranslationJobs](#) 操作來擷取您帳戶中

所有翻譯任務的資訊。若要將結果限制為符合特定條件的任務，請使用 [ListTextTranslationJobs](#) 操作的 `filter` 參數。您可以依工作名稱、工作狀態或提交工作的日期和時間來篩選結果。

Example `describe-text-translation-job` 命令

下列範例使用 執行 [DescribeTextTranslationJob](#) 命令 AWS CLI ，檢查任務的狀態：

```
$ aws translate describe-text-translation-job --job-id 1c1838f470806ab9c3e0057f14717bed
```

此命令會傳回下列輸出：

```
{
  "TextTranslationJobProperties": {
    "InputDataConfig": {
      "ContentType": "text/plain",
      "S3Uri": "s3://input-bucket-name/folder"
    },
    "EndTime": 1576551359.483,
    "SourceLanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::012345678901:role/service-role/AmazonTranslateInputOutputAccess",
    "JobId": "1c1838f470806ab9c3e0057f14717bed",
    "TargetLanguageCodes": [
      "fr"
    ],
    "JobName": "batch-test",
    "SubmittedTime": 1576544017.357,
    "JobStatus": "COMPLETED",
    "Message": "Your job has completed successfully.",
    "JobDetails": {
      "InputDocumentsCount": 77,
      "DocumentsWithErrorsCount": 0,
      "TranslatedDocumentsCount": 77
    },
    "OutputDataConfig": {
      "S3Uri": "s3://bucket-name/output/012345678901-TranslateText-1c1838f470806ab9c3e0057f14717bed/"
    }
  }
}
```

您可以使用 [StopTextTranslationJob](#) IN_PROGRESS 操作，在其狀態為 時停止批次翻譯任務。

Example stop-text-translation-job 命令

下列範例使用 執行 [StopTextTranslationJob](#) 命令 AWS CLI ，以 停止批次翻譯：

```
$ aws translate stop-text-translation-job --job-id 5236d36ce5192abdb3e2519f3ab8b065
```

此命令會傳回下列輸出：

```
{
  "TextTranslationJobProperties": {
    "InputDataConfig": {
      "ContentType": "text/plain",
      "S3Uri": "s3://input-bucket-name/folder"
    },
    "SourceLanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::012345678901:role/service-role/AmazonTranslateInputOutputAccess",
    "TargetLanguageCodes": [
      "fr"
    ],
    "JobName": "canceled-test",
    "SubmittedTime": 1576558958.167,
    "JobStatus": "STOP_REQUESTED",
    "JobId": "5236d36ce5192abdb3e2519f3ab8b065",
    "OutputDataConfig": {
      "S3Uri": "s3://output-bucket-name/012345678901-TranslateText-5236d36ce5192abdb3e2519f3ab8b065/"
    }
  }
}
```

取得批次翻譯結果

一旦任務的狀態為 `COMPLETED` 或 `COMPLETED_WITH_ERROR`，您的輸出文件即可在您指定的 Amazon S3 資料夾中使用。輸出文件名稱與輸入文件名稱相符，而且會新增目標語言代碼作為字首。例如，如果您將名為 `mySourceText.txt` 的文件翻譯為法文，則會呼叫輸出文件 `fr.mySourceText.txt`。

如果批次翻譯任務的狀態為 `FAILED`，[DescribeTextTranslationJob](#) 操作回應會包含一個 `Message` 欄位，說明任務無法順利完成的原因。

每個批次翻譯工作也會產生一個輔助檔案，其中包含執行翻譯的資訊，例如翻譯的字元總數和發生的錯誤數量。會在輸出資料夾的 `details` 子資料夾中產生這個名為 `target-language-code.auxiliary-translation-details.json` 的檔案。

以下是批次翻譯輔助檔案的範例。

```
{
  "sourceLanguageCode": "en",
  "targetLanguageCode": "fr",
  "charactersTranslated": "105",
  "documentCountWithCustomerError": "0",
  "documentCountWithServerError": "0",
  "inputDataPrefix": "s3://input-bucket-name/folder",
  "outputDataPrefix": "s3://output-bucket-name/012345678901-
TranslateText-1c1838f470806ab9c3e0057f14717bed/",
  "details": [
    {
      "sourceFile": "mySourceText.txt",
      "targetFile": "fr.mySourceText.txt",
      "auxiliaryData": {
        "appliedTerminologies": [
          {
            "name": "TestTerminology",
            "terms": [
              {
                "sourceText": "Amazon",
                "targetText": "Amazon"
              }
            ]
          }
        ]
      }
    },
    {
      "sourceFile": "batchText.txt",
      "targetFile": "fr.batchText.txt",
      "auxiliaryData": {
        "appliedTerminologies": [
          {
            "name": "TestTerminology",
            "terms": [
              {
                "sourceText": "Amazon",
```

```
    "targetText": "Amazon"  
  }  
]  
}  
]  
}  
]  
}  
]  
}
```

使用 Amazon Translate 自訂翻譯

您可以使用下列設定來自訂您使用 Amazon Translate 產生的翻譯：

- Do-not-translate 標籤 – 使用開始和結束標籤來指定您不想翻譯的內容（以 HTML 內容表示）。
- 自訂術語 – 定義您希望 Amazon Translate 如何翻譯特定術語，例如品牌名稱。
- Brevity – 減少大多數翻譯的翻譯輸出長度（相較於不含簡潔的翻譯輸出）。透過即時文字翻譯使用簡潔性。
- 褻瀆性 – 在翻譯輸出中遮罩褻瀆文字和片語。
- 形式 – 設定翻譯輸出中的語言形式程度。
- 平行資料 – 調整翻譯輸出，以反映您提供的範例翻譯範例中的樣式、音調和單字選項。

主題

- [在 do-not-translate Amazon Translate](#)
- [使用自訂術語自訂翻譯](#)
- [在 Amazon Translate 中使用簡潔](#)
- [在 Amazon Translate 中遮罩褻瀆單字和片語](#)
- [在 Amazon Translate 中設定正式性](#)
- [使用平行資料自訂翻譯（主動自訂翻譯）](#)

在 do-not-translate Amazon Translate

對於 HTML 內容，您可以將 do-not-translate 標籤新增至您不想翻譯的文字。此功能適用於 主控台和 API 操作。

主題

- [搭配主控台使用 do-not-translate](#)
- [搭配 API do-not-translate](#)

搭配主控台使用 do-not-translate

在來源 HTML 內容中，在 HTML 標籤 `translate="no"` 中指定您不想翻譯的內容。例如，若要將下列文字從英文翻譯為德文：

```
In French, the Louvre Museum is Musée du Louvre.
```

「Musée du Louvre」文字需要保留法文，因此我們使用跨度標籤略過此內容的翻譯：

```
<p>In French, the Louvre Museum is <span translate="no">Musée du Louvre</span>.</p>
```

此句子具有產生的德文翻譯：

```
<p>Auf Französisch ist <span translate="no">Musée du Louvre</span> das Louvre-Museum.</p>
```

搭配 API do-not-translate

您可以使用do-not-translate搭配即時 API 操作 (TranslateText 和 TranslateDocument) 和非同步 StartTextTranslationJob API 操作。在您為 API 請求提供的來源文字中，您可以使用任何類型的 HTML 元素來指定需要略過轉譯的內容。

在下列範例中，我們希望將一些文字從英文翻譯為西班牙文，但保留一些英文文字：

```
aws translate translate-text \  
  --source-language-code "en" \  
  --target-language-code "es" \  
  --region us-west-2 \  
  --text "<p>You can translate this paragraph to any language.</p> <p translate=no>But do not translate this.</p>"
```

此 API 請求會傳回下列西班牙文翻譯：

```
{  
  "TranslatedText": "<p>Puede traducir este párrafo a cualquier idioma.</p>  
    <p translate=no>But do not translate this.</p>",  
  "SourceLanguageCode": "en",  
  "TargetLanguageCode": "es"  
}
```

使用自訂術語自訂翻譯

使用自訂術語與您的翻譯請求，以確保您的品牌名稱、角色名稱、模型名稱和其他唯一內容已翻譯為所需的結果。

您可以建立術語檔案，並將其上傳至您的 Amazon Translate 帳戶。如需有關檔案大小和術語檔案數量的資訊，請參閱 [Service Quotas](#)。翻譯文字時，您可以選擇是否要使用的自訂術語檔案。當 Amazon Translate 找到來源文字與術語檔案之間的相符項目時，它會使用術語檔案中的轉譯。

請考慮下列範例：Amazon Photos 為 Amazon Prime 成員提供免費的相片和影片儲存。在法文中，名稱不會翻譯：它仍為 Amazon Photos。

當您使用 Amazon Translate 將 Amazon Photos 翻譯為法文，而沒有任何其他內容時，結果是 Photos d'Amazon，這不是所需的翻譯。

如果您為 Amazon Photos 一詞新增自訂術語項目，指定法文翻譯為 Amazon Photos，Amazon Translate 會使用自訂術語將片語翻譯為所需的結果。

Amazon Translate 不保證會針對每個轉譯使用目標詞彙。自訂術語使用轉譯內容中來源和目標術語的意義來決定是否使用目標術語。如需詳細資訊，請參閱 [最佳實務](#)。

主題

- [建立自訂術語](#)
- [使用自訂術語](#)
- [使用的自訂術語範例 適用於 Python \(Boto\) 的 AWS SDK](#)
- [加密您的術語](#)
- [最佳實務](#)

建立自訂術語

您可以透過建立術語檔案來定義自訂術語。Amazon Translate 支援 CSV、TSV 或 TMX 檔案格式。檔案中的每個項目都包含每個目標語言的來源術語和同等（已翻譯）術語。

建立術語檔案之後，您會將檔案上傳至 Amazon Translate 帳戶。

Important

自訂術語中的來源文字區分大小寫。在翻譯期間，Amazon Translate 在輸入文件中找到完全相符項目時，會使用自訂術語。

術語檔案格式

下列範例顯示 CSV 格式的術語檔案。

CSV (逗號分隔值)

```
en,fr,es  
Amazon Photos,Amazon Photos,Amazon Photos
```

下列範例顯示 TMX 格式的術語檔案。TMX 檔案使用翻譯軟體常用的 XML 格式。

TMX (翻譯記憶 eXchange)

```
<?xml version="1.0" encoding="UTF-8"?>  
<tmx version="1.4">  
  <header  
    creationtool="XYZTool" creationtoolversion="0"  
    datatype="PlainText" segtype="sentence"  
    adminlang="en-us" srclang="en"  
    o-tmf="test"/>  
  <body>  
    <tu>  
      <tuv xml:lang="en">  
        <seg>Amazon Photos</seg>  
      </tuv>  
      <tuv xml:lang="fr">  
        <seg>Amazon Photos</seg>  
      </tuv>  
      <tuv xml:lang="es">  
        <seg>Amazon Photos</seg>  
      </tuv>  
    </tu>  
  </body>  
</tmx>
```

方向性

當您上傳自訂術語檔案時，您可以設定自訂術語的方向性值。方向性表示您的術語檔案指定一種來源語言或多種來源語言。

針對方向性，請設定下列其中一個值：

單向

術語檔案包含一種來源語言（清單中的第一個語言）。所有其他語言都是目標語言。

例如，在 CSV 檔案中，第一欄包含來源語言的文字，而所有其他欄則包含目標語言的文字。

多向

檔案中的任何語言可以是來源語言或目標語言。例如，如果您的術語檔案包含英文、法文和西班牙文的文字，您可以將該檔案用於翻譯下列語言對的任務：

- 英文到法文
- 英文到西班牙文
- 法文到英文
- 從法文到西班牙文
- 西班牙文到英文
- 西班牙文到法文

相反地，您需要為這六個翻譯任務建立三個單向術語檔案（每個來源語言一個）。

使用自訂術語

若要在使用 [TranslateText](#) 操作翻譯文字時使用自訂術語，請包含選用 `TerminologyNames` 參數。

例如，如果您將名為 `Amazon_Family.csv` 的下列術語檔案上傳至您的帳戶：

```
en,fr
Amazon Family,Amazon Famille
```

您可以使用下列 CLI 命令，使用自訂術語翻譯文字。

Note

此範例格式適用於 Unix、Linux 和 macOS。用於 Windows 時，請以插入號 (^) 取代每一行結尾處的 Unix 接續字元斜線 (\)。

```
aws translate translate-text \  
  --region region \  
  --source-language-code "en" \  
  --target-language-code "fr" \  
  --terminology-names "Amazon_Family" \  
  --text "Have you ever stored videos in Amazon Family?"
```

這使用選取的自訂術語，將此文字翻譯為「Avez-vous déjà fait des achats avec Amazon Famille？」而不是直接（但不想要）翻譯 "Avez-vous déjà fait des achats avec Famille Amazon ?"

下列範例示範如何在 Python 中使用相同的術語檔案。

```
import boto3

translate = boto3.client(service_name='translate')

print("Translating 'Have you ever shopped with Amazon Family?' from English to French with the 'Amazon_Family' custom terminology...")
response = translate.translate_text(Text="Have you ever shopped with Amazon Family?", TerminologyNames=["Amazon_Family"], SourceLanguageCode="en", TargetLanguageCode="fr")
print("Translated text: " + response.get('TranslatedText'))
print("\n")
```

如需搭配自訂術語使用 Amazon Translate 操作的詳細資訊，請參閱 [API 操作](#)。

使用的自訂術語範例適用於 Python (Boto) 的 AWS SDK

下列範例示範如何在 Python 中使用自訂術語操作。若要執行範例，請透過安裝 Python SDK AWS CLI。如需說明，請參閱「[the section called “設定 AWS CLI”](#)」。

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-

import boto3

translate = boto3.client(service_name='translate')

# The terminology file 'my-first-terminology.csv' has the following contents:
...
en,fr
Amazon Family,Amazon Famille
...

# Read the terminology from a local file
with open('/tmp/my-first-terminology.csv', 'rb') as f:
    data = f.read()

file_data = bytearray(data)

print("Importing the terminology into Amazon Translate...")
```

```
response = translate.import_terminology(Name='my-first-terminology',
MergeStrategy='OVERWRITE', TerminologyData={"File": file_data, "Format": 'CSV'})
print("Terminology imported: "),
print(response.get('TerminologyProperties'))
print("\n")

print("Getting the imported terminology...")
response = translate.get_terminology(Name='my-first-terminology',
TerminologyDataFormat='CSV')
print("Received terminology: "),
print(response.get('TerminologyProperties'))
print("The terminology data file can be downloaded here: " +
response.get('TerminologyDataLocation').get('Location'))
print("\n")

print("Listing the first 10 terminologies for the account...")
response = translate.list_terminologies(MaxResults=10)
print("Received terminologies: "),
print(response.get('TerminologyPropertiesList'))
print("\n")

print("Translating 'Amazon Family' from English to French with no terminology...")
response = translate.translate_text(Text="Amazon Family", SourceLanguageCode="en",
TargetLanguageCode="fr")
print("Translated text: " + response.get('TranslatedText'))
print("\n")

print("Translating 'Amazon Family' from English to French with the 'my-first-
terminology' terminology...")
response = translate.translate_text(Text="Amazon Family", TerminologyNames=["my-
first-terminology"], SourceLanguageCode="en", TargetLanguageCode="fr")
print("Translated text: " + response.get('TranslatedText'))
print("\n")

# The terminology file 'my-updated-terminology.csv' has the following contents:
'''
en,fr
Amazon Family,Amazon Famille
Prime Video, Prime Video
'''

# Read the terminology from a local file
with open('/tmp/my-updated-terminology.csv', 'rb') as f:
    data = f.read()
```

```
file_data = bytearray(data)

print("Updating the imported terminology in Amazon Translate...")
response = translate.import_terminology(Name='my-first-terminology',
MergeStrategy='OVERWRITE', TerminologyData={"File": file_data, "Format": 'CSV'})
print("Terminology updated: "),
print(response.get('TerminologyProperties'))
print("\n")

print("Translating 'Prime Video' from English to French with no terminology...")
response = translate.translate_text(Text="Prime Video", SourceLanguageCode="en",
TargetLanguageCode="fr")
print("Translated text: " + response.get('TranslatedText'))
print("\n")

print("Translating 'Prime Video' from English to French with the 'my-first-
terminology' terminology...")
response = translate.translate_text(Text="Prime Video", TerminologyNames=["my-
first-terminology"], SourceLanguageCode="en", TargetLanguageCode="fr")
print("Translated text: " + response.get('TranslatedText'))
print("\n")

print("Cleaning up by deleting 'my-first-terminology'...")
translate.delete_terminology(Name="my-first-terminology")
print("Terminology deleted.")
```

加密您的術語

Amazon Translate 竭盡全力保護您的所有資料，而您的自訂術語也不例外。在建立時，每個自訂術語都會經過加密，只有您能存取。

共有三種加密選項可用：

- 使用 AWS encryption. AWS encryption 是保護您的資訊的預設選項。
- 使用與您的帳戶相關聯的加密金鑰。主控台的功能表可讓您選擇要使用的相關聯加密金鑰。
- 使用與您的帳戶無關的加密金鑰。主控台會顯示輸入欄位，供您輸入加密金鑰的 Amazon Resource Name (ARN)。

最佳實務

使用自訂術語時，請使用下列一般最佳實務：

- 讓您的自訂術語保持簡潔。僅包含您需要控制翻譯值的詞彙。
- 自訂術語會區分大小寫。如果您想要文字大寫和非大寫版本的目標轉譯，請包含每個版本的項目。
- 自訂術語不是用來控制間距、標點符號或大小寫的工具。例如，避免下列類型的項目：
 - 新增空格 – EN : USA FR : U S A
 - 新增標點符號 – 英文 : 美國 法文 : 美國
 - 變更大寫 – EN : USA FR : Usa
- 請勿為相同的來源片語包含不同的轉譯。例如：
 - 項目編號 1 – 英文 : Amazon FR : Amazon
 - 項目編號 2 – 英文 : Amazon FR : Amazone
- 您可以為 Amazon Translate 支援的任何語言建立自訂術語檔案。

Amazon Translate 不保證自訂術語會針對每個轉譯使用目標術語。若要使用自訂術語達到高準確度，請在建立術語檔案的內容時遵循以下最佳實務：

- 自訂術語可與任何單字搭配使用，包括動詞和同義詞。正確名稱，例如品牌名稱和產品名稱，是理想的項目。
- 目標術語在目標語言中應流暢。對於包含許多特殊字元或格式的目標術語，不建議使用自訂術語。
- 您可以在術語檔案中包含多字片語或子句。不過，包含多個單字的術語不太可能以目標語言流暢讀取。
- 自訂術語使用轉譯內容中來源和目標術語的意義，來決定是否使用目標術語。如果目標術語不適合特定翻譯內容，Amazon Translate 可能不會使用目標術語。

例如，如果您的術語檔案包含以下英文到法文的項目：

EN: order, FR: commande (根據英文 "to order" 翻譯為法文 "mander")。

Amazon Translate 在翻譯下列句子時不會使用此項目，因為翻譯內容不相符：

「為了讓我們協助您，請分享您的名稱。」

避免這種情況的建議：

- 請確定每種語言的目標術語在語意上等於來源術語。

- 避免具有多個意義的來源或目標術語。

在 Amazon Translate 中使用簡潔

在語言之間翻譯時，有時翻譯輸出會比所需時間更長（以字元計數為單位）。如果沒有額外的字元空間，較長的輸出在某些情況下可能會導致問題（例如字幕、字幕、標題或表單欄位）。

當您使用 Amazon Translate 執行即時文字翻譯時，可以開啟簡潔設定。Brevity 可減少大多數翻譯的翻譯輸出長度（相較於不含簡潔的翻譯輸出）。

主題

- [使用簡潔設定](#)
- [支援的語言](#)

使用簡潔設定

您可以使用簡潔設定搭配即時文字翻譯。Amazon Translate 不支援即時文件翻譯或非同步翻譯工作的簡潔性。

若要在即時文字翻譯請求中使用簡潔，請執行下列其中一項：

- 主控台 – 在即時轉譯頁面的文字索引標籤中，於其他設定下，選擇 Brevity 設定。
- AWS CLI – 在 `translate-text` 命令中，將 `--settings` 參數中的簡潔度設定為 `Brevity=ON`。如需詳細資訊，請參閱 AWS CLI 命令參考中的[翻譯文字](#)。
- AWS API – 在 [TranslateText](#) API 操作中，在設定參數中設定簡潔性。

支援的語言

Amazon Translate 支援以下語言對的簡潔性：

- 從任何[來源語言](#)到下表中的其中一種語言。
- 從下表中的任何語言到英文。

語言	語言代碼
法文	fr
德文	de
義大利文	it
葡萄牙文 (巴西)	pt
西班牙文	es

如果您使用不支援的語言對請求翻譯簡潔，翻譯會繼續關閉簡潔設定。

在 Amazon Translate 中遮罩褻瀆單字和片語

當您使用 Amazon Translate 執行翻譯時，您可以啟用褻瀆設定來遮罩翻譯輸出中的褻瀆單字和片語。

若要遮罩褻瀆單字和片語，Amazon Translate 會將它們取代為 gawlix 字串“? \$#@\$”。此 5 個字元序列用於每個褻瀆單字或片語，無論單字的長度或數量為何。

Amazon Translate 不會在來源語言和目標語言相同的翻譯請求中遮罩褻瀆性。

在某些情況下，來源輸入中的褻瀆文字在翻譯的輸出中自然會變得不雅。在這種情況下，不會套用遮罩。

Amazon Translate 會以文字方式偵測每個褻瀆單字或片語，而非情境式。這表示它可能會掩蓋褻瀆單字，即使其在內容中不具冒犯性。例如，如果 Amazon Translate 偵測到“jerk”為褻瀆單字，則它會將片語“jerk chicken”撰寫為“? \$#@\$ chicken”，即使“jerk chicken”是無禮的。（此處，「急衝」僅用作範例。Amazon Translate 不會將該字詞偵測為褻瀆。）

主題

- [使用褻瀆設定](#)
- [不支援的語言](#)

使用褻瀆設定

您可以在 Amazon Translate 中使用褻瀆設定搭配這兩種類型的轉譯操作：即時轉譯和非同步批次處理。

若要遮罩即時翻譯請求中的褻瀆，請執行下列任一動作：

- 在 Amazon Translate 主控台的即時轉譯頁面上，在其他設定下，啟用褻瀆設定。Amazon Translate
- 在的 `translate-text` 命令中 AWS CLI，將 `--settings` 參數設定為 `Profanity=MASK`。如需詳細資訊，請參閱《AWS CLI 命令參考》中的[翻譯文字](#)。
- 在 Amazon Translate API 中的 `TranslateText` 動作中，使用 `Settings` 參數來設定褻瀆遮罩。如需詳細資訊，請參閱 API 參考中的 [TranslateText](#)。

若要在非同步批次操作中遮罩褻瀆，請參閱 [執行批次翻譯任務](#)。

不支援的語言

您可以在翻譯為 Amazon Translate 支援的任何目標語言時遮罩褻瀆，但以下情況除外：

Language	語言代碼
孟加拉文	bn
北印度文	hi
馬來亞拉姆文	ml
旁遮普文	pa
僧伽羅文	si
越南文	vi

如需 Amazon Translate 支援的所有語言，請參閱 [支援的語言和語言代碼](#)。

在 Amazon Translate 中設定正式性

您可以選擇性地指定所需的正式程度，以將轉譯為支援的目標語言。正式性設定控制轉譯輸出中正式語言使用量（也稱為[評價或註冊](#)）的層級。正式性設定可用於即時轉譯和非同步批次處理。

Formality 支援下列值：

- 非正式 – 翻譯文字中的所有句子都使用與非正式溝通相關聯的語言建構。例如，翻譯文字使用熟悉的第二個人代名詞形式及其動詞協議（或適用於日文的 Kudaketa 形式）。
- 正式 – 翻譯文字中的所有句子都使用與正式、禮貌的溝通相關聯的語言建構。例如，翻譯文字使用正式形式的第二個人代名詞及其動詞協議（或日文的 Teineigo 格式）。

例如，句子「確定嗎？」可以有兩個正確的德文翻譯：「Sind Sie sicher？」用於正式註冊和「Bist du sicher？」適用於非正式的。

如果 Amazon Translate 不支援目標語言的正式程度，或者您未指定正式性參數，轉譯任務會忽略正式性設定。

主題

- [使用正規設定](#)
- [支援的語言](#)

使用正規設定

若要在即時轉譯請求中設定正式性，請執行下列其中一項操作：

- 在 Amazon Translate 主控台的即時轉譯頁面上的其他設定下，啟用形式設定，然後選取其中一個值。Amazon Translate
- 在 Amazon Translate API 的 [TranslateText](#) 操作中使用設定參數。
- 對於 `translate-text` 命令 AWS CLI，將 `--settings` 參數設定為 `Formality=FORMAL` 或 `Formality=INFORMAL`。如需詳細資訊，請參閱《AWS CLI 命令參考》中的[翻譯文字](#)。

若要在批次轉譯請求中設定正規，請在開始轉譯任務時設定正規參數。如需詳細資訊和範例，請參閱[執行批次翻譯任務](#)。

對於 CLI 或 API 請求，回應中的 `AppliedSettings` 欄位包含請求的正規設定（如果有的話）。如果目標語言不支援正式性，則回應中的 `AppliedSettings` 值為 `NULL`。

支援的語言

Amazon Translate 支援從任何來源語言轉譯為下列目標語言的正規設定。

Language	語言代碼
荷蘭文	nl
法文	fr
法文 (加拿大)	fr-CA
德文	de
北印度文	hi
義大利文	it
日文	ja
韓文	ko
葡萄牙語 (葡萄牙)	pt-PT
西班牙文	es
西班牙文 (墨西哥)	es-MX

如需 Amazon Translate 支援的所有語言，請參閱 [支援的語言和語言代碼](#)。

使用平行資料自訂翻譯（主動自訂翻譯）

將平行資料新增至 Amazon Translate，以自訂批次翻譯任務的輸出。平行資料包含顯示您希望如何翻譯文字區段的範例。它包含來源語言的文字範例集合，而且對於每個範例，它包含一或多個目標語言所需的翻譯輸出。

當您將平行資料新增至批次轉譯任務時，您會建立作用中的自訂轉譯任務。當您執行這些任務時，Amazon Translate 會在執行時間使用您的平行資料來產生自訂的機器翻譯輸出。它會調整翻譯，以反映它在平行資料中找到的樣式、色調和單字選擇。透過平行資料，您可以針對特定領域特有的詞彙或片語量身打造翻譯，例如生命科學、法律或金融。

Note

作用中的自訂轉譯任務定價比不使用平行資料的其他任務高。如需詳細資訊，請參閱 [Amazon Translate 定價](#)。

例如，下列平行資料是在 CSV 檔案中定義：

```
"en","fr"  
"How are you?","Comment ça va ?"
```

在此範例中，英文 (en) 是來源語言，而法文 (fr) 是目標語言。此範例顯示來源片語「您的狀況如何？」應該翻譯為法文。此範例輸入檔案匯入 Amazon Translate 之後，即可套用至翻譯任務，以影響其輸出。在這類任務期間，Amazon Translate 會翻譯「您的表現如何？」非正式的「評論 ça va ?」而不是正式的「Comment allez-vous ?」例如，任務可能會收到下列來源文字：

```
Hello, how are you?  
How are you?  
Hi, how are you?  
How are you doing?
```

在此文字中，任務會產生下列轉譯：

```
Bonjour, comment ça va ?  
Comment ça va ?  
Salut, comment ça va ?  
Comment ça va ?
```

相反地，如果任務在沒有平行資料的情況下執行，輸出可能包含更正式的「comment allez-vous」：

```
Bonjour, comment allez-vous ?  
Comment allez-vous ?  
Salut, comment allez-vous ?  
Comment allez-vous ?
```

透過使用平行資料自訂批次翻譯任務，您可以像使用您透過翻譯範例訓練的自訂翻譯模型一樣的方式影響輸出。使用主動自訂轉譯時，訓練自訂模型是不必要的，而且您可以避免此類訓練所需的時間和費用。隨著翻譯需求隨時間變更，您可以透過更新平行資料來精簡輸出，這比重新訓練自訂模型更容易。

區域可用性

作用中自訂轉譯可在下列區域使用：

- 美國東部 (維吉尼亞北部)
- 美國西部 (奧勒岡)
- 歐洲 (愛爾蘭)

主題

- [Amazon Translate 的平行資料輸入檔案](#)
- [將平行資料新增至 Amazon Translate](#)
- [在 Amazon Translate 中檢視和管理平行資料](#)

Amazon Translate 的平行資料輸入檔案

您必須先建立包含翻譯範例的輸入檔案，才能在 Amazon Translate 中建立平行資料資源。您的平行資料輸入檔案必須使用 Amazon Translate 支援的語言。如需這些語言的清單，請參閱 [支援的語言和語言代碼](#)。

平行資料範例

下表中的文字提供可格式化為平行資料輸入檔案的翻譯區段範例：

zh	es
Amazon Translate 是一種神經機器翻譯服務。	Amazon Translate es un servicio de traducción automática basado en neuronas.
神經機器翻譯是一種使用深度學習模型的語言翻譯自動化形式。	La traducción automática neuronal es una forma de automatizar la traducción de lenguajes utilizando modelos de aprendizaje profundo.
Amazon Translate 可讓您將國際使用者的內容當地語系化。	Amazon Translate le permite localizar el contenido para usuarios internacionales.

資料表的第一列提供語言代碼。第一語言是英文 (en)，是來源語言。西班牙文 (es) 是目標語言。第一欄提供來源文字的範例。另一個資料欄包含翻譯的範例。當此平行資料自訂批次任務時，Amazon Translate 會調整轉譯以反映範例。

輸入檔案格式

Amazon Translate 支援下列平行資料輸入檔案格式：

- Translation Memory eXchange (TMX)
- 逗號分隔值 (CSV)
- Tab 分隔值 (TSV)

TMX

Example TMX 輸入檔案

下列範例 TMX 檔案以 Amazon Translate 接受的格式定義平行資料。在此檔案中，英文 (en) 是來源語言。西班牙文 (es) 是目標語言。做為平行資料的輸入檔案，它提供數個範例，Amazon Translate 可用來自訂批次任務的輸出。

```
<?xml version="1.0" encoding="UTF-8"?>
<tmx version="1.4">
  <header srclang="en"/>
  <body>
    <tu>
      <tuv xml:lang="en">
        <seg>Amazon Translate is a neural machine translation service.</seg>
      </tuv>
      <tuv xml:lang="es">
        <seg>Amazon Translate es un servicio de traducción automática basado
en redes neuronales.</seg>
      </tuv>
    </tu>
    <tu>
      <tuv xml:lang="en">
        <seg>Neural machine translation is a form of language translation
automation that uses deep learning models.</seg>
      </tuv>
      <tuv xml:lang="es">
        <seg>La traducción automática neuronal es una forma de automatizar
la traducción de lenguajes utilizando modelos de aprendizaje profundo.</seg>
      </tuv>
    </tu>
  </body>
</tmx>
```

```
        </tuv>
    </tu>
    <tu>
        <tuv xml:lang="en">
            <seg>Amazon Translate allows you to localize content for
international users.</seg>
        </tuv>
        <tuv xml:lang="es">
            <seg>Amazon Translate le permite localizar contenido para usuarios
internacionales.</seg>
        </tuv>
    </tu>
</body>
</tmx>
```

TMX 要求

當您在 TMX 檔案中定義平行資料時，請記住 Amazon Translate 的下列要求：

- Amazon Translate 支援 TMX 1.4b。如需詳細資訊，請參閱 Globalization and Localization Association 網站上的 [TMX 1.4b 規格](#)。
- header 元素必須包含 srclang 屬性。此屬性的值會決定平行資料的來源語言。
- body 元素必須至少包含一個轉譯單位 (tu) 元素。
- 每個 tu 元素必須至少包含兩個轉譯單位變體 (tuv) 元素。其中一個 tuv 元素必須具有與元素 srclang 中指派給屬性的值相同的 xml:lang 屬性 header。
- 所有 tuv 元素都必須有 xml:lang 屬性。
- 所有 tuv 元素都必須有區段 (seg) 元素。
- 處理輸入檔案時，如果 Amazon Translate 遇到空白 tu 或僅包含空格的 tuv 元素，則會略過特定或 seg 元素：
 - 如果 seg 元素對應至來源語言，Amazon Translate 會略過 tu 元素佔用的 seg 元素。
 - 如果 seg 元素對應至目標語言，Amazon Translate 只會略過 tuv 元素佔用的 seg 元素。
- 處理輸入檔案時，如果 Amazon Translate 遇到超過 1000 個位元組的 tuv 元素，則會略過特定 tu 或 seg 元素：
 - 如果 seg 元素對應至來源語言，Amazon Translate 會略過 tu 元素佔用的 seg 元素。
 - 如果 seg 元素對應至目標語言，Amazon Translate 只會略過 tuv 元素佔用的 seg 元素。
- 如果輸入檔案包含具有相同來源文字的多個 tu 元素，Amazon Translate 會執行下列其中一項操作：

- 如果 `tu` 元素具有 `changedate` 屬性，則會使用具有最近日期的 元素。
- 否則，它會使用最接近檔案結尾的 元素。

CSV

下列範例 CSV 檔案以 Amazon Translate 接受的格式定義平行資料。在此檔案中，英文 (en) 是來源語言。西班牙文 (es) 是目標語言。做為平行資料的輸入檔案，它提供數個範例，Amazon Translate 可用來自訂批次任務的輸出。

Example CSV 輸入檔案

```
en,es
Amazon Translate is a neural machine translation service.,Amazon Translate es un
servicio de traducción automática basado en redes neuronales.
Neural machine translation is a form of language translation automation that uses
deep learning models.,La traducción automática neuronal es una forma de automatizar
la traducción de lenguajes utilizando modelos de aprendizaje profundo.
Amazon Translate allows you to localize content for international users.,Amazon
Translate le permite localizar contenido para usuarios internacionales.
```

CSV 要求

當您在 CSV 檔案中定義平行資料時，請記住 Amazon Translate 的下列要求：

- 第一列包含語言代碼。第一個程式碼是來源語言，每個後續程式碼都是目標語言。
- 第一欄中的每個欄位都包含來源文字。後續資料欄中的每個欄位都包含目標轉譯。
- 如果任何欄位中的文字包含逗號，則文字必須以雙引號 (") 字元括住。
- 文字欄位不能跨越多行。
- 欄位不能以下列字元開頭：+、-、=、@。無論 欄位是否以雙引號 (") 括住，此要求都適用。
- 如果欄位中的文字包含雙引號 (")，則必須使用雙引號逸出。例如，文字例如：

```
34" monitor
```

必須寫入為：

```
34"" monitor
```

- 處理您的輸入檔案時，如果 Amazon Translate 遇到空白或僅包含空格的欄位，則會略過特定行或欄位：
 - 如果來源文字欄位為空，Amazon Translate 會略過其佔用的行。
 - 如果目標轉譯欄位為空，Amazon Translate 只會略過該欄位。
- 處理輸入檔案時，如果遇到超過 1000 個位元組的欄位，Amazon Translate 會略過特定行或欄位：
 - 如果來源文字欄位超過位元組限制，Amazon Translate 會略過其佔用的行。
 - 如果目標轉譯欄位超過位元組限制，Amazon Translate 只會略過該欄位。
- 如果輸入檔案包含具有相同來源文字的多個記錄，Amazon Translate 會使用最接近檔案結尾的記錄。

TSV

下列範例 TSV 檔案會以 Amazon Translate 接受的格式定義平行資料。在此檔案中，英文 (en) 是來源語言。西班牙文 (es) 是目標語言。做為平行資料的輸入檔案，它提供數個範例，Amazon Translate 可用來自訂批次任務的輸出。

Example TSV 輸入檔案

```
en es
Amazon Translate is a neural machine translation service. Amazon Translate es un
servicio de traducción automática basado en redes neuronales.
Neural machine translation is a form of language translation automation that uses
deep learning models. La traducción automática neuronal es una forma de automatizar
la traducción de lenguajes utilizando modelos de aprendizaje profundo.
Amazon Translate allows you to localize content for international users. Amazon
Translate le permite localizar contenido para usuarios internacionales.
```

TSV 要求

當您在 TSV 檔案中定義平行資料時，請記住 Amazon Translate 的下列要求：

- 第一列包含語言代碼。第一個程式碼是來源語言，每個後續程式碼都是目標語言。
- 第一欄中的每個欄位都包含來源文字。後續資料欄中的每個欄位都包含目標轉譯。
- 如果任何欄位中的文字包含標籤字元，則文字必須以雙引號 (") 字元括住。
- 文字欄位不能跨越多行。
- 欄位不能以下列字元開頭：+、-、=、@。無論欄位是否以雙引號 (") 括住，此要求都適用。

- 如果欄位中的文字包含雙引號 (")，則必須使用雙引號逸出。例如，文字例如：

```
34" monitor
```

必須寫入為：

```
34"" monitor
```

- 處理輸入檔案時，如果遇到空白或僅包含空格的欄位，Amazon Translate 會略過特定行或欄位：
 - 如果來源文字欄位為空白，Amazon Translate 會略過其佔用的行。
 - 如果目標轉譯欄位為空，Amazon Translate 只會略過該欄位。
- 處理輸入檔案時，如果遇到超過 1000 個位元組的欄位，Amazon Translate 會略過特定行或欄位：
 - 如果來源文字欄位超過位元組限制，Amazon Translate 會略過其佔用的行。
 - 如果目標轉譯欄位超過位元組限制，Amazon Translate 只會略過該欄位。
- 如果輸入檔案包含具有相同來源文字的多個記錄，Amazon Translate 會使用最接近檔案結尾的記錄。

將平行資料新增至 Amazon Translate

若要將平行資料新增至 Amazon Translate，請從 Amazon S3 匯入平行資料輸入檔案。之後，您可以使用平行資料來自訂批次翻譯任務產生的輸出。

先決條件

在將平行資料新增至 Amazon Translate 之前，您必須：

- 具有平行資料輸入檔案。若要建立服務角色，請參閱[Amazon Translate 的平行資料輸入檔案](#)。
- AWS 您的帳戶中有 Amazon S3 儲存貯體。若要建立儲存貯體，請參閱 [《Amazon Simple Storage Service 使用者指南》中的如何建立 S3 儲存貯體？](#)。
- 將輸入檔案上傳至 Amazon S3 儲存貯體。如需詳細資訊，請參閱 [《Amazon Simple Storage Service 使用者指南》中的如何將檔案和資料夾上傳至 S3 儲存貯體？](#)。

新增平行資料 (Amazon Translate 主控台)

若要使用 Amazon Translate 主控台新增平行資料，請使用平行資料頁面：

1. 開啟 [Amazon Translate 主控台](#)。
2. 在左側的導覽功能表中，選擇自訂，然後選擇平行資料。
3. 在平行資料頁面上，選擇建立平行資料。主控台會顯示建立平行資料頁面。
4. 提供下列資訊：

名稱

平行資料資源的自訂名稱。您必須指派帳戶和區域中唯一的名稱。

描述 - 選用

自訂描述。

S3 上的平行資料位置

Amazon S3 中平行資料輸入檔案的位置。若要透過導覽至 Amazon S3 中的檔案來提供位置，請選擇選取檔案。

檔案格式

平行資料輸入檔案的格式。支援的格式包括 Translation Memory eXchange (TMX)、逗號分隔值 (CSV) 和標籤分隔值 (TSV)。

5. 在加密金鑰下，選擇 AWS KMS key 來保護平行資料。這些 KMS 金鑰由 AWS Key Management Service () 管理 AWS KMS。如需的詳細資訊 AWS KMS，請參閱 [AWS Key Management Service 開發人員指南](#)。

使用 AWS 擁有的金鑰

使用 Amazon Translate 擁有和管理的 KMS 金鑰。這是預設選項，如果您未選擇其他方法，則會用來加密您的資訊。如需詳細資訊，請參閱《AWS Key Management Service 開發人員指南》中的 [AWS 擁有的金鑰](#)。

使用目前帳戶的金鑰

使用您在 AWS KMS AWS 帳戶中管理的其中一個 KMS 金鑰。如果您選擇此選項，則功能表會提供 KMS 金鑰清單以供選擇。如需更多資訊，請參閱 AWS Key Management Service 開發人員指南中的 [客戶受管金鑰](#)。

使用不同帳戶的金鑰

使用在不同 AWS 帳戶中 AWS KMS 在中管理的 KMS 金鑰。如果您選擇此選項，主控台會提供欄位，供您輸入 KMS 金鑰的 Amazon Resource Name (ARN)。

如需加密金鑰的詳細資訊，請參閱 [AWS Key Management Service 開發人員指南](#)。

6. 選擇建立平行資料。

主控台會返回平行資料頁面，其中匯入狀態會顯示在頁面頂端的橫幅中。幾分鐘後，您的平行資料資源會顯示在資料表中。當狀態欄中的值為作用中時，平行資料已準備好供您在批次翻譯任務中使用。

故障診斷的錯誤檔案

如果 Amazon Translate 在處理輸入檔案時產生任何錯誤或警告，主控台會提供錯誤檔案供您下載，以檢閱錯誤訊息。此檔案的內容類似下列範例：

```
{
  "summary": {
    "record_error_count": 1,
    "record_skipped_count": 0
  },
  "messages": [
    {
      "content": "Number 1 TU element",
      "message": "Invalid TMX format. One tu element should contain exactly one tuv element with the source language code: en"
    }
  ]
}
```

新增平行資料 (AWS CLI)

若要使用 新增平行資料 AWS CLI，請使用 `create-parallel-data` 命令。

Example `create-parallel-data` 命令

下列範例會透過從 Amazon S3 匯入 TSV 檔案來建立平行資料物件：

```
$ aws translate create-parallel-data \
```

```
> --name my-parallel-data \  
> --parallel-data-config S3Uri=s3://input-bucket/parallel-data-file.tsv,Format=TSV
```

如果命令成功，Amazon Translate 會以新平行資料物件的狀態回應：

```
{  
  "Name": "my-parallel-data",  
  "Status": "CREATING"  
}
```

您可以使用 `get-parallel-data` 命令來監控平行資料的持續狀態。當狀態為 `ACTIVE`，平行資料已準備好供您在批次翻譯任務中使用。如需 `get-parallel-data` 命令的範例，請參閱 [檢視平行資料物件的詳細資訊](#)。

使用您的平行資料

現在您已建立平行資料資源，您可以將其套用至批次翻譯任務以自訂輸出。若要執行批次任務，請參閱 [執行批次翻譯任務](#)。

在 Amazon Translate 中檢視和管理平行資料

您可以檢視已新增至 Amazon Translate 的所有平行資料資源，也可以存取每個資源的詳細摘要。隨著翻譯需求變更，您可以透過更新平行資料來精簡翻譯輸出。

檢視和管理平行資料 (Amazon Translate 主控台)

若要在 Amazon Translate 主控台中檢視和管理平行資料，請使用平行資料頁面：

檢視平行資料資源的清單

1. 開啟 [Amazon Translate 主控台](#)。
2. 在左側導覽功能表中，選擇自訂，然後選擇平行資料。本頁面上的表格列出您已新增至 Amazon Translate 的平行資料資源。

檢視平行資料資源的詳細資訊

- 在平行資料頁面上，選擇名稱欄中平行資料資源的名稱。主控台會開啟詳細資訊頁面，其中包含狀態、上次更新日期、來源語言和目標語言等資訊。

更新平行資料資源

1. 將平行資料的更新版本上傳為 Amazon S3 儲存貯體中的新輸入檔案。
2. 在 Amazon Translate 主控台中，前往平行資料頁面。
3. 選取您要更新的平行資料，然後選擇更新。主控台會顯示更新平行資料頁面。
4. 提供下列資訊：

描述 - 選用

已更新的描述。

S3 上的平行資料位置

Amazon S3 中更新平行資料輸入檔案的位置。若要透過導覽至 Amazon S3 中的檔案來提供位置，請選擇選取檔案。

選取平行資料檔案格式

平行資料輸入檔案的格式。支援的格式包括 Translation Memory eXchange (TMX)、逗號分隔值 (CSV) 和標籤分隔值 (TSV)。

5. 選擇儲存。Amazon Translate 會將舊的平行資料取代為新的輸入檔案。

檢視和管理平行資料 (AWS CLI)

您可以使用 AWS CLI 檢視和更新平行資料資源。

檢視平行資料資源的清單

若要檢視您已新增至 Amazon Translate 的平行資料資源清單，請使用 `list-parallel-data` 命令。

Example `list-parallel-data` 命令

下列範例會傳回平行資料資源及其屬性的清單。

```
$ aws translate list-parallel-data
```

如果命令成功，Amazon Translate 會傳回如下所示的陣列：

```
{
  "ParallelDataPropertiesList": [
    {
```

```

    "Name": "my-parallel-data",
    "Arn": "arn:aws:translate:us-west-2:111122223333:parallel-data/my-parallel-
data",
    "Status": "ACTIVE",
    "SourceLanguageCode": "en",
    "TargetLanguageCodes": [
        "es",
        "ja",
        "zh"
    ],
    "ParallelDataConfig": {
        "S3Uri": "s3://input-bucket/parallel-data-file.tsv",
        "Format": "TSV"
    },
    "ImportedDataSize": 2283,
    "ImportedRecordCount": 3,
    "FailedRecordCount": 0,
    "CreatedAt": 1598597751.406,
    "LastUpdatedAt": 1598597911.675
  }
]
}

```

檢視平行資料物件的詳細資訊

若要查詢單一平行資料資源的詳細資訊，請使用 `get-parallel-data` 命令。此命令會傳回平行資料的屬性以及預先簽章的 S3 URL，您可以在其中下載用來建立它的輸入檔案。

Example `get-parallel-data` 命令

下列範例會取得 `my-parallel-data` 物件的屬性和下載位置：

```

$ aws translate get-parallel-data \
> --name my-parallel-data

```

如果命令成功，Amazon Translate 會傳回屬性和下載位置：

```

{
  "ParallelDataProperties": {
    "Name": "my-parallel-data",
    "Arn": "arn:aws:translate:us-west-2:111122223333:parallel-data/my-parallel-
data",
    "Status": "ACTIVE",

```

```
    "SourceLanguageCode": "en",
    "TargetLanguageCodes": [
      "es",
      "ja",
      "zh"
    ],
    "ParallelDataConfig": {
      "S3Uri": "s3://input-bucket/parallel-data-file.tsv",
      "Format": "TSV"
    },
    "ImportedDataSize": 2283,
    "ImportedRecordCount": 3,
    "FailedRecordCount": 0,
    "CreatedAt": 1598597751.406,
    "LastUpdatedAt": 1598597911.675
  },
  "DataLocation": {
    "RepositoryType": "S3",
    "Location": "pre-signed S3 URL"
  }
}
```

更新平行資料資源

若要更新平行資料資源，請先將新的輸入檔案上傳至 Amazon S3 輸入儲存貯體。然後，使用 `update-parallel-data` 命令並指定您要更新的平行資料資源。Amazon Translate 會將舊的平行資料取代為新輸入檔案中的資訊。

Example `update-parallel-data` 命令

下列命令 `my-parallel-data` 會使用來自 Amazon S3 的新輸入檔案進行更新：

```
$ aws translate update-parallel-data \
> --name my-parallel-data \
> --parallel-data-config S3Uri=s3://input-bucket/parallel-data-file.tsv,Format=TSV
```

如果命令成功，Amazon Translate 會提供如下所示的回應：

```
{
  "Name": "my-parallel-data",
  "Status": "ACTIVE",
  "LatestUpdateAttemptStatus": "UPDATING",
```

```
"LatestUpdateAttemptAt": 1598601455.844  
}
```

在此回應中，`Status` 欄位會提供預先存在的平行資料物件狀態，而 `LatestUpdateAttemptStatus` 欄位會提供目前更新嘗試的狀態。

Amazon Translate AWS SDKs的程式碼範例

下列程式碼範例示範如何使用 Amazon Translate 搭配 AWS 軟體開發套件 (SDK)。

Actions 是大型程式的程式碼摘錄，必須在內容中執行。雖然動作會告訴您如何呼叫個別服務函數，但您可以在其相關情境中查看內容中的動作。

案例是向您展示如何呼叫服務中的多個函數或與其他 AWS 服務組合來完成特定任務的程式碼範例。

如需 AWS SDK 開發人員指南和程式碼範例的完整清單，請參閱 [搭配 AWS SDK 使用此服務](#)。此主題也包含有關入門的資訊和舊版 SDK 的詳細資訊。

程式碼範例

- [Amazon Translate using AWS SDKs 的基本範例](#)
 - [Amazon Translate AWS SDKs的動作](#)
 - [DescribeTextTranslationJob 搭配 AWS SDK 使用](#)
 - [ListTextTranslationJobs 搭配 AWS SDK 使用](#)
 - [StartTextTranslationJob 搭配 AWS SDK 使用](#)
 - [StopTextTranslationJob 搭配 AWS SDK 使用](#)
 - [TranslateText 搭配 AWS SDK 或 CLI 使用](#)
- [使用 AWS SDKs 的 Amazon Translate 案例](#)
 - [建置 Amazon Transcribe 串流應用程式](#)
 - [建立 Amazon Lex 聊天機器人與網站訪客互動](#)
 - [建置可轉譯訊息的發佈和訂閱應用程式](#)
 - [建立可分析客戶意見回饋並合成音訊的應用程式](#)
 - [使用 AWS SDK 開始使用 Amazon Translate 任務](#)

Amazon Translate using AWS SDKs 的基本範例

下列程式碼範例示範如何搭配 AWS SDK 使用 Amazon Translate 的基本功能。

範例

- [Amazon Translate AWS SDKs的動作](#)
 - [DescribeTextTranslationJob 搭配 AWS SDK 使用](#)

- [ListTextTranslationJobs 搭配 AWS SDK 使用](#)
- [StartTextTranslationJob 搭配 AWS SDK 使用](#)
- [StopTextTranslationJob 搭配 AWS SDK 使用](#)
- [TranslateText 搭配 AWS SDK 或 CLI 使用](#)

Amazon Translate AWS SDKs的動作

下列程式碼範例示範如何使用 AWS SDKs 執行個別 Amazon Translate 動作。每個範例均包含 GitHub 的連結，您可以在連結中找到設定和執程式碼的相關說明。

這些摘錄會呼叫 Amazon Translate API，是必須在內容中執行之大型程式的程式碼摘錄。您可以在 [使用 AWS SDKs 的 Amazon Translate 案例](#) 中查看內容中的動作。

下列範例僅包含最常使用的動作。如需完整清單，請參閱 [《Amazon Translate API 參考》](#)。

範例

- [DescribeTextTranslationJob 搭配 AWS SDK 使用](#)
- [ListTextTranslationJobs 搭配 AWS SDK 使用](#)
- [StartTextTranslationJob 搭配 AWS SDK 使用](#)
- [StopTextTranslationJob 搭配 AWS SDK 使用](#)
- [TranslateText 搭配 AWS SDK 或 CLI 使用](#)

DescribeTextTranslationJob 搭配 AWS SDK 使用

下列程式碼範例示範如何使用 DescribeTextTranslationJob。

動作範例是大型程式的程式碼摘錄，必須在內容中執行。您可以在下列程式碼範例的內容中看到此動作：

- [開始執行翻譯任務](#)

.NET

適用於 .NET 的 SDK

Note

GitHub 上提供更多範例。尋找完整範例，並了解如何在 [AWS 程式碼範例儲存庫](#) 中設定和執行。

```
using System;
using System.Threading.Tasks;
using Amazon.Translate;
using Amazon.Translate.Model;

/// <summary>
/// The following example shows how to retrieve the details of
/// a text translation job using Amazon Translate.
/// </summary>
public class DescribeTextTranslation
{
    public static async Task Main()
    {
        var client = new AmazonTranslateClient();

        // The Job Id is generated when the text translation job is started
        // with a call to the StartTextTranslationJob method.
        var jobId = "1234567890abcdef01234567890abcde";

        var request = new DescribeTextTranslationJobRequest
        {
            JobId = jobId,
        };

        var jobProperties = await DescribeTranslationJobAsync(client,
request);

        DisplayTranslationJobDetails(jobProperties);
    }

    /// <summary>
    /// Retrieve information about an Amazon Translate text translation job.
```

```
    /// </summary>
    /// <param name="client">The initialized Amazon Translate client
    object.</param>
    /// <param name="request">The DescribeTextTranslationJobRequest object.</
    param>
    /// <returns>The TextTranslationJobProperties object containing
    /// information about the text translation job.</returns>
    public static async Task<TextTranslationJobProperties>
    DescribeTranslationJobAsync(
        AmazonTranslateClient client,
        DescribeTextTranslationJobRequest request)
    {
        var response = await client.DescribeTextTranslationJobAsync(request);
        if (response.HttpStatusCode == System.Net.HttpStatusCode.OK)
        {
            return response.TextTranslationJobProperties;
        }
        else
        {
            return null;
        }
    }

    /// <summary>
    /// Displays the properties of the text translation job.
    /// </summary>
    /// <param name="jobProperties">The properties of the text translation
    /// job returned by the call to DescribeTextTranslationJobAsync.</param>
    public static void
    DisplayTranslationJobDetails(TextTranslationJobProperties jobProperties)
    {
        if (jobProperties is null)
        {
            Console.WriteLine("No text translation job properties found.");
            return;
        }

        // Display the details of the text translation job.
        Console.WriteLine($"{jobProperties.JobId}: {jobProperties.JobName}");
    }
}
```

- 如需 API 詳細資訊，請參閱《適用於 .NET 的 AWS SDK API 參考》中的 [DescribeTextTranslationJob](#)。

SAP ABAP

適用於 SAP ABAP 的開發套件

Note

GitHub 上提供更多範例。尋找完整範例，並了解如何在 [AWS 程式碼範例儲存庫](#) 中設定和執行。

```
"Gets the properties associated with an asynchronous batch translation job."  
"Includes properties such as name, ID, status, source and target languages,  
and input/output Amazon Simple Storage Service (Amazon S3) buckets."  
TRY.  
    oo_result = lo_xl8->describetexttranslationjob(      "oo_result is  
returned for testing purposes."  
    iv_jobid      = iv_jobid ).  
    MESSAGE 'Job description retrieved.' TYPE 'I'.  
CATCH /aws1/cx_xl8internalserverex.  
    MESSAGE 'An internal server error occurred. Retry your request.' TYPE  
'E'.  
CATCH /aws1/cx_xl8resourcenotfoundex.  
    MESSAGE 'The resource you are looking for has not been found.' TYPE 'E'.  
CATCH /aws1/cx_xl8toomanyrequestsex.  
    MESSAGE 'You have made too many requests within a short period of time.'  
TYPE 'E'.  
ENDTRY.
```

- 如需 API 詳細資訊，請參閱《適用於 SAP ABAP 的 AWS SDK API 參考》中的 [DescribeTextTranslationJob](#)。

如需 AWS SDK 開發人員指南和程式碼範例的完整清單，請參閱 [搭配 AWS SDK 使用此服務](#)。此主題也包含有關入門的資訊和舊版 SDK 的詳細資訊。

ListTextTranslationJobs 搭配 AWS SDK 使用

下列程式碼範例示範如何使用 ListTextTranslationJobs。

.NET

適用於 .NET 的 SDK

Note

GitHub 上提供更多範例。尋找完整範例，並了解如何在 [AWS 程式碼範例儲存庫](#) 中設定和執行。

```
using System;
using System.Collections.Generic;
using System.Threading.Tasks;
using Amazon.Translate;
using Amazon.Translate.Model;

/// <summary>
/// List Amazon Translate translation jobs, along with details about each
job.
/// </summary>
public class ListTranslationJobs
{
    public static async Task Main()
    {
        var client = new AmazonTranslateClient();
        var filter = new TextTranslationJobFilter
        {
            JobStatus = "COMPLETED",
        };

        var request = new ListTextTranslationJobsRequest
        {
            MaxResults = 10,
            Filter = filter,
        };

        await ListJobsAsync(client, request);
    }
}
```

```
    /// <summary>
    /// List Amazon Translate text translation jobs.
    /// </summary>
    /// <param name="client">The initialized Amazon Translate client
object.</param>
    /// <param name="request">An Amazon Translate
    /// ListTextTranslationJobsRequest object detailing which text
    /// translation jobs are of interest.</param>
    public static async Task ListJobsAsync(
        AmazonTranslateClient client,
        ListTextTranslationJobsRequest request)
    {
        ListTextTranslationJobsResponse response;

        do
        {
            response = await client.ListTextTranslationJobsAsync(request);

            ShowTranslationJobDetails(response.TextTranslationJobPropertiesList);

            request.NextToken = response.NextToken;
        }
        while (response.NextToken is not null);
    }

    /// <summary>
    /// List existing translation job details.
    /// </summary>
    /// <param name="properties">A list of Amazon Translate text
    /// translation jobs.</param>
    public static void
    ShowTranslationJobDetails(List<TextTranslationJobProperties> properties)
    {
        properties.ForEach(prop =>
        {
            Console.WriteLine($"{prop.JobId}: {prop.JobName}");
            Console.WriteLine($"Status: {prop.JobStatus}");
            Console.WriteLine($"Submitted time: {prop.SubmittedTime}");
        });
    }
}
```

- 如需 API 詳細資訊，請參閱《適用於 .NET 的 AWS SDK API 參考》中的 [ListTextTranslationJobs](#)。

SAP ABAP

適用於 SAP ABAP 的開發套件

Note

GitHub 上提供更多範例。尋找完整範例，並了解如何在 [AWS 程式碼範例儲存庫](#) 中設定和執行。

```
"Gets a list of the batch translation jobs that you have submitted."

DATA lo_filter TYPE REF TO /aws1/cl_xl8textxlationjobfilt.

"Create an ABAP object for filtering using jobname."
lo_filter = NEW #( iv_jobname = iv_jobname ).

TRY.
    oo_result = lo_xl8->listtexttranslationjobs(      "oo_result is returned
for testing purposes."
        io_filter      = lo_filter ).
    MESSAGE 'Jobs retrieved.' TYPE 'I'.
CATCH /aws1/cx_xl8internalserverex.
    MESSAGE 'An internal server error occurred. Retry your request.' TYPE
'E'.
CATCH /aws1/cx_xl8invalidfilterex.
    MESSAGE 'The filter specified for the operation is not valid. Specify a
different filter.' TYPE 'E'.
CATCH /aws1/cx_xl8invalidrequestex.
    MESSAGE 'The request that you made is not valid.' TYPE 'E'.
CATCH /aws1/cx_xl8toomanyrequestsex.
    MESSAGE 'You have made too many requests within a short period of time.'
TYPE 'E'.
ENDTRY.
```

- 如需 API 詳細資訊，請參閱《適用於 SAP ABAP 的 AWS SDK API 參考》中的 [ListTextTranslationJobs](#)。

如需 AWS SDK 開發人員指南和程式碼範例的完整清單，請參閱 [搭配 AWS SDK 使用此服務](#)。此主題也包含有關入門的資訊和舊版 SDK 的詳細資訊。

StartTextTranslationJob 搭配 AWS SDK 使用

下列程式碼範例示範如何使用 StartTextTranslationJob。

動作範例是大型程式的程式碼摘錄，必須在內容中執行。您可以在下列程式碼範例的內容中看到此動作：

- [開始執行翻譯任務](#)

.NET

適用於 .NET 的 SDK

Note

GitHub 上提供更多範例。尋找完整範例，並了解如何在 [AWS 程式碼範例儲存庫](#) 中設定和執行。

```
using System;
using System.Collections.Generic;
using System.Threading.Tasks;
using Amazon.Translate;
using Amazon.Translate.Model;

/// <summary>
/// This example shows how to use Amazon Translate to process the files in
/// an Amazon Simple Storage Service (Amazon S3) bucket. The translated
results
/// will also be stored in an Amazon S3 bucket.
/// </summary>
public class BatchTranslate
{
    public static async Task Main()
```

```
{
    var contentType = "text/plain";

    // Set this variable to an S3 bucket location with a folder."
    // Input files must be in a folder and not at the bucket root."
    var s3InputUri = "s3://amzn-s3-demo-bucket1/FOLDER/";
    var s3OutputUri = "s3://amzn-s3-demo-bucket2/";

    // This role must have permissions to read the source bucket and to
read and
    // write to the destination bucket where the translated text will be
stored.
    var dataAccessRoleArn = "arn:aws:iam::0123456789ab:role/
S3TranslateRole";

    var client = new AmazonTranslateClient();

    var inputConfig = new InputDataConfig
    {
        ContentType = contentType,
        S3Uri = s3InputUri,
    };

    var outputConfig = new OutputDataConfig
    {
        S3Uri = s3OutputUri,
    };

    var request = new StartTextTranslationJobRequest
    {
        JobName = "ExampleTranslationJob",
        DataAccessRoleArn = dataAccessRoleArn,
        InputDataConfig = inputConfig,
        OutputDataConfig = outputConfig,
        SourceLanguageCode = "en",
        TargetLanguageCodes = new List<string> { "fr" },
    };

    var response = await StartTextTranslationAsync(client, request);

    if (response.HttpStatusCode == System.Net.HttpStatusCode.OK)
    {
        Console.WriteLine($"{response.JobId}: {response.JobStatus}");
    }
}
```

```

    }

    /// <summary>
    /// Start the Amazon Translate text translation job.
    /// </summary>
    /// <param name="client">The initialized AmazonTranslateClient object.</
param>
    /// <param name="request">The request object that includes details such
    /// as source and destination bucket names and the IAM Role that will
    /// be used to access the buckets.</param>
    /// <returns>The StartTextTranslationResponse object that includes the
    /// details of the request response.</returns>
    public static async Task<StartTextTranslationJobResponse>
StartTextTranslationAsync(AmazonTranslateClient client,
StartTextTranslationJobRequest request)
    {
        var response = await client.StartTextTranslationJobAsync(request);
        return response;
    }
}

```

- 如需 API 詳細資訊，請參閱《適用於 .NET 的 AWS SDK API 參考》中的 [StartTextTranslationJob](#)。

SAP ABAP

適用於 SAP ABAP 的開發套件

Note

GitHub 上提供更多範例。尋找完整範例，並了解如何在 [AWS 程式碼範例儲存庫](#) 中設定和執行。

```

"Starts an asynchronous batch translation job."
"Use batch translation jobs to translate large volumes of text across
multiple documents at once."

```

```
DATA lo_inputdataconfig TYPE REF TO /aws1/cl_xl8inputdataconfig.
```

```

DATA lo_outputdataconfig TYPE REF TO /aws1/cl_xl8outputdataconfig.
DATA lt_targetlanguagecodes TYPE /aws1/
cl_xl8tgtlanguagecodes00=>tt_targetlanguagecodestrlist.
DATA lo_targetlanguagecodes TYPE REF TO /aws1/cl_xl8tgtlanguagecodes00.

"Create an ABAP object for the input data config."
lo_inputdataconfig = NEW #( iv_s3uri = iv_input_data_s3uri
                           iv_contenttype = iv_input_data_contenttype ).

"Create an ABAP object for the output data config."
lo_outputdataconfig = NEW #( iv_s3uri = iv_output_data_s3uri ).

"Create an internal table for target languages."
lo_targetlanguagecodes = NEW #( iv_value = iv_targetlanguagecode ).
INSERT lo_targetlanguagecodes INTO TABLE lt_targetlanguagecodes.

TRY.
    oo_result = lo_xl8->starttexttranslationjob(      "oo_result is returned
for testing purposes."
    io_inputdataconfig = lo_inputdataconfig
    io_outputdataconfig = lo_outputdataconfig
    it_targetlanguagecodes = lt_targetlanguagecodes
    iv_dataaccessrolelearn = iv_dataaccessrolelearn
    iv_jobname = iv_jobname
    iv_sourcelanguagecode = iv_sourcelanguagecode ).
    MESSAGE 'Translation job started.' TYPE 'I'.
CATCH /aws1/cx_xl8internalserverex.
    MESSAGE 'An internal server error occurred. Retry your request.' TYPE
'E'.
CATCH /aws1/cx_xl8invparamvalueex.
    MESSAGE 'The value of the parameter is not valid.' TYPE 'E'.
CATCH /aws1/cx_xl8invalidrequestex.
    MESSAGE 'The request that you made is not valid.' TYPE 'E'.
CATCH /aws1/cx_xl8resourcenotfoundex.
    MESSAGE 'The resource you are looking for has not been found.' TYPE 'E'.
CATCH /aws1/cx_xl8toomanyrequestsex.
    MESSAGE 'You have made too many requests within a short period of time.'
TYPE 'E'.
CATCH /aws1/cx_xl8unsuppdedlanguage00.
    MESSAGE 'Amazon Translate does not support translation from the language
of the source text into the requested target language.' TYPE 'E'.
ENDTRY.

```

- 如需 API 詳細資訊，請參閱《適用於 SAP ABAP 的 AWS SDK API 參考》中的 [StartTextTranslationJob](#)。

如需 AWS SDK 開發人員指南和程式碼範例的完整清單，請參閱 [搭配 AWS SDK 使用此服務](#)。此主題也包含有關入門的資訊和舊版 SDK 的詳細資訊。

StopTextTranslationJob 搭配 AWS SDK 使用

下列程式碼範例示範如何使用 StopTextTranslationJob。

.NET

適用於 .NET 的 SDK

Note

GitHub 上提供更多範例。尋找完整範例，並了解如何在 [AWS 程式碼範例儲存庫](#) 中設定和執行。

```
using System;
using System.Threading.Tasks;
using Amazon.Translate;
using Amazon.Translate.Model;

/// <summary>
/// Shows how to stop a running Amazon Translation Service text translation
/// job.
/// </summary>
public class StopTextTranslationJob
{
    public static async Task Main()
    {
        var client = new AmazonTranslateClient();
        var jobId = "1234567890abcdef01234567890abcde";

        var request = new StopTextTranslationJobRequest
        {
            JobId = jobId,
        };
    }
}
```

```
        await StopTranslationJobAsync(client, request);
    }

    /// <summary>
    /// Sends a request to stop a text translation job.
    /// </summary>
    /// <param name="client">Initialized AmazonTrnslateClient object.</param>
    /// <param name="request">The request object to be passed to the
    /// StopTextJobAsync method.</param>
    public static async Task StopTranslationJobAsync(
        AmazonTranslateClient client,
        StopTextTranslationJobRequest request)
    {
        var response = await client.StopTextTranslationJobAsync(request);
        if (response.HttpStatusCode == System.Net.HttpStatusCode.OK)
        {
            Console.WriteLine($"{response.JobId} as status:
{response.JobStatus}");
        }
    }
}
```

- 如需 API 詳細資訊，請參閱《適用於 .NET 的 AWS SDK API 參考》中的 [StopTextTranslationJob](#)。

SAP ABAP

適用於 SAP ABAP 的開發套件

Note

GitHub 上提供更多範例。尋找完整範例，並了解如何在 [AWS 程式碼範例儲存庫](#) 中設定和執行。

"Stops an asynchronous batch translation job that is in progress."

TRY.

```
oo_result = lo_xl8->stoptexttranslationjob( "oo_result is returned
for testing purposes."
    iv_jobid          = iv_jobid ).
MESSAGE 'Translation job stopped.' TYPE 'I'.
CATCH /aws1/cx_xl8internalserverex.
MESSAGE 'An internal server error occurred.' TYPE 'E'.
CATCH /aws1/cx_xl8resourcenotfoundex.
MESSAGE 'The resource you are looking for has not been found.' TYPE 'E'.
CATCH /aws1/cx_xl8toomanyrequestsex.
MESSAGE 'You have made too many requests within a short period of time.'
TYPE 'E'.
ENDTRY.
```

- 如需 API 詳細資訊，請參閱《適用於 SAP ABAP 的 AWS SDK API 參考》中的 [StopTextTranslationJob](#)。

如需 AWS SDK 開發人員指南和程式碼範例的完整清單，請參閱 [搭配 AWS SDK 使用此服務](#)。此主題也包含有關入門的資訊和舊版 SDK 的詳細資訊。

TranslateText 搭配 AWS SDK 或 CLI 使用

下列程式碼範例示範如何使用 TranslateText。

.NET

適用於 .NET 的 SDK

Note

GitHub 上提供更多範例。尋找完整範例，並了解如何在 [AWS 程式碼範例儲存庫](#) 中設定和執行。

```
using System;
using System.IO;
using System.Threading.Tasks;
using Amazon.S3;
using Amazon.S3.Transfer;
using Amazon.Translate;
using Amazon.Translate.Model;
```

```
/// <summary>
/// Take text from a file stored a Amazon Simple Storage Service (Amazon S3)
/// object and translate it using the Amazon Transfer Service.
/// </summary>
public class TranslateText
{
    public static async Task Main()
    {
        // If the region you want to use is different from the region
        // defined for the default user, supply it as a parameter to the
        // Amazon Translate client object constructor.
        var client = new AmazonTranslateClient();

        // Set the source language to "auto" to request Amazon Translate to
        // automatically detect te language of the source text.

        // You can get a list of the languages supposed by Amazon Translate
        // in the Amazon Translate Developer's Guide here:
        //     https://docs.aws.amazon.com/translate/latest/dg/what-is.html
        string srcLang = "en"; // English.
        string destLang = "fr"; // French.

        // The Amazon Simple Storage Service (Amazon S3) bucket where the
        // source text file is stored.
        string srcBucket = "amzn-s3-demo-bucket";
        string srcTextFile = "source.txt";

        var srcText = await GetSourceTextAsync(srcBucket, srcTextFile);
        var destText = await TranslatingTextAsync(client, srcLang, destLang,
srcText);

        ShowText(srcText, destText);
    }

    /// <summary>
    /// Use the Amazon S3 TransferUtility to retrieve the text to translate
    /// from an object in an S3 bucket.
    /// </summary>
    /// <param name="srcBucket">The name of the S3 bucket where the
    /// text is stored.
    /// </param>
    /// <param name="srcTextFile">The key of the S3 object that
    /// contains the text to translate.</param>
```

```
    /// <returns>A string representing the source text.</returns>
    public static async Task<string> GetSourceTextAsync(string srcBucket,
string srcTextFile)
    {
        string srcText = string.Empty;

        var s3Client = new AmazonS3Client();
        TransferUtility utility = new TransferUtility(s3Client);

        using var stream = await utility.OpenStreamAsync(srcBucket,
srcTextFile);

        StreamReader file = new System.IO.StreamReader(stream);

        srcText = file.ReadToEnd();
        return srcText;
    }

    /// <summary>
    /// Use the Amazon Translate Service to translate the document from the
    /// source language to the specified destination language.
    /// </summary>
    /// <param name="client">The Amazon Translate Service client used to
    /// perform the translation.</param>
    /// <param name="srcLang">The language of the source text.</param>
    /// <param name="destLang">The destination language for the translated
    /// text.</param>
    /// <param name="text">A string representing the text to ranslate.</
param>
    /// <returns>The text that has been translated to the destination
    /// language.</returns>
    public static async Task<string>
TranslatingTextAsync(AmazonTranslateClient client, string srcLang, string
destLang, string text)
    {
        var request = new TranslateTextRequest
        {
            SourceLanguageCode = srcLang,
            TargetLanguageCode = destLang,
            Text = text,
        };

        var response = await client.TranslateTextAsync(request);
```

```
        return response.TranslatedText;
    }

    /// <summary>
    /// Show the original text followed by the translated text.
    /// </summary>
    /// <param name="srcText">The original text to be translated.</param>
    /// <param name="destText">The translated text.</param>
    public static void ShowText(string srcText, string destText)
    {
        Console.WriteLine("Source text:");
        Console.WriteLine(srcText);
        Console.WriteLine();
        Console.WriteLine("Translated text:");
        Console.WriteLine(destText);
    }
}
```

- 如需 API 詳細資訊，請參閱《適用於 .NET 的 AWS SDK API 參考》中的 [TranslateText](#)。

PowerShell

Tools for PowerShell V4

範例 1：將指定的英文文字轉換為法文。要轉換的文字也可以當作 -Text 參數來傳遞。

```
"Hello World" | ConvertTo-TRNTargetLanguage -SourceLanguageCode en -
TargetLanguageCode fr
```

- 如需 API 詳細資訊，請參閱《AWS Tools for PowerShell Cmdlet 參考 (V4)》中的 [TranslateText](#)。

Tools for PowerShell V5


範例 1：將指定的英文文字轉換為法文。要轉換的文字也可以當作 -Text 參數來傳遞。

```
"Hello World" | ConvertTo-TRNTargetLanguage -SourceLanguageCode en -
TargetLanguageCode fr
```

- 如需 API 詳細資訊，請參閱《AWS Tools for PowerShell Cmdlet 參考 (V5)》中的 [TranslateText](#)。

SAP ABAP

適用於 SAP ABAP 的開發套件

 Note

GitHub 上提供更多範例。尋找完整範例，並了解如何在 [AWS 程式碼範例儲存庫](#) 中設定和執行。

```
"Translates input text from the source language to the target language."
TRY.
    oo_result = lo_xl8->translatetext(      "oo_result is returned for
testing purposes."
    iv_text      = iv_text
    iv_sourcelanguagecode = iv_sourcelanguagecode
    iv_targetlanguagecode = iv_targetlanguagecode ).
    MESSAGE 'Translation completed.' TYPE 'I'.
CATCH /aws1/cx_xl8detectedlanguage00.
    MESSAGE 'The confidence that Amazon Comprehend accurately detected the
source language is low.' TYPE 'E'.
CATCH /aws1/cx_xl8internalserverex.
    MESSAGE 'An internal server error occurred.' TYPE 'E'.
CATCH /aws1/cx_xl8invalidrequestex.
    MESSAGE 'The request that you made is not valid.' TYPE 'E'.
CATCH /aws1/cx_xl8resourcenotfoundex.
    MESSAGE 'The resource you are looking for has not been found.' TYPE 'E'.
CATCH /aws1/cx_xl8serviceunavailex.
    MESSAGE 'The Amazon Translate service is temporarily unavailable.' TYPE
'E'.
CATCH /aws1/cx_xl8textsizegmtexcdex.
    MESSAGE 'The size of the text you submitted exceeds the size limit. '
TYPE 'E'.
CATCH /aws1/cx_xl8toomanyrequestsex.
    MESSAGE 'You have made too many requests within a short period of time.'
TYPE 'E'.
CATCH /aws1/cx_xl8unsuppdedlanguage00.
    MESSAGE 'Amazon Translate does not support translation from the language
of the source text into the requested target language. ' TYPE 'E'.
ENDTRY.
```

- 如需 API 詳細資訊，請參閱《適用於 SAP ABAP 的 AWS SDK API 參考》中的 [TranslateText](#)。

如需 AWS SDK 開發人員指南和程式碼範例的完整清單，請參閱 [搭配 AWS SDK 使用此服務](#)。此主題也包含有關入門的資訊和舊版 SDK 的詳細資訊。

使用 AWS SDKs 的 Amazon Translate 案例

下列程式碼範例示範如何在 Amazon Translate AWS SDKs 中實作常見案例。這些案例示範如何呼叫 Amazon Translate 中的多個函數，或與其他 AWS 服務結合，藉以完成特定任務。每個案例均包含完整原始碼的連結，您可在連結中找到如何設定和執行程式碼的相關指示。

案例的目標是獲得中等水平的經驗，協助您了解內容中的服務動作。

範例

- [建置 Amazon Transcribe 串流應用程式](#)
- [建立 Amazon Lex 聊天機器人與網站訪客互動](#)
- [建置可轉譯訊息的發佈和訂閱應用程式](#)
- [建立可分析客戶意見回饋並合成音訊的應用程式](#)
- [使用 AWS SDK 開始使用 Amazon Translate 任務](#)

建置 Amazon Transcribe 串流應用程式

下面的程式碼範例說明如何建置可即時記錄、轉錄和翻譯直播音訊並透過電子郵件傳送結果的應用程式。

JavaScript

適用於 JavaScript (v3) 的 SDK

說明如何使用 Amazon Transcribe 建置應用程式，該應用程式可即時記錄、轉錄和翻譯直播音訊，並可使用 Amazon Simple Email Service (Amazon SES) 透過電子郵件傳送結果。

如需完整的原始碼和如何設定及執行的指示，請參閱 [GitHub](#) 上的完整範例。

此範例中使用的服務

- Amazon Comprehend

- Amazon SES
- Amazon Transcribe
- Amazon Translate

如需 AWS SDK 開發人員指南和程式碼範例的完整清單，請參閱 [搭配 AWS SDK 使用此服務](#)。此主題也包含有關入門的資訊和舊版 SDK 的詳細資訊。

建立 Amazon Lex 聊天機器人與網站訪客互動

下列程式碼範例示範如何建立聊天機器人，與網站訪客互動。

Java

適用於 Java 2.x 的 SDK

示範如何使用 Amazon Lex API 在 Web 應用程式中建立 Chatbot，與網站訪客的互動。

如需完整的原始碼和如何設定及執行的指示，請參閱 [GitHub](#) 上的完整範例。

此範例中使用的服務

- Amazon Comprehend
- Amazon Lex
- Amazon Translate

JavaScript

適用於 JavaScript (v3) 的 SDK

示範如何使用 Amazon Lex API 在 Web 應用程式中建立 Chatbot，與網站訪客的互動。

如需完整的原始程式碼和如何設定和執行的指示，請參閱適用於 JavaScript 的 AWS SDK 開發人員指南中的 [建置 Amazon Lex 聊天機器人](#) 的完整範例。

此範例中使用的服務

- Amazon Comprehend
- Amazon Lex
- Amazon Translate

如需 AWS SDK 開發人員指南和程式碼範例的完整清單，請參閱 [搭配 AWS SDK 使用此服務](#)。此主題也包含有關入門的資訊和舊版 SDK 的詳細資訊。

建置可轉譯訊息的發佈和訂閱應用程式

以下程式碼範例示範如何建立具有訂閱和發布功能並且可轉譯訊息的應用程式。

.NET

適用於 .NET 的 SDK

示範如何使用 Amazon Simple Notification Service .NET API 來建立具有訂閱和發布功能的 Web 應用程式。此外，此範例應用程式也會轉譯訊息。

如需完整的原始碼和如何設定及執行的指示，請參閱 [GitHub](#) 上的完整範例。

此範例中使用的服務

- Amazon SNS
- Amazon Translate

Java

適用於 Java 2.x 的 SDK

示範如何使用 Amazon Simple Notification Service Java API 來建立具有訂閱和發布功能的 Web 應用程式。此外，此範例應用程式也會轉譯訊息。

如需完整的原始碼和如何設定及執行的指示，請參閱 [GitHub](#) 上的完整範例。

如需完整的原始碼和如何設定及執行使用 Java Async API 之範例的指示，請參閱 [GitHub](#) 上的完整範例。

此範例中使用的服務

- Amazon SNS
- Amazon Translate

Kotlin

SDK for Kotlin

示範如何使用 Amazon SNS Kotlin API 來建立具有訂閱和發布功能的應用程式。此外，此範例應用程式也會轉譯訊息。

如需完整的原始碼和如何建立 Web 應用程式的指示，請參閱 [GitHub](#) 上的完整範例。

如需完整的原始碼和如何建立原生 Android 應用程式的指示，請參閱 [GitHub](#) 上的完整範例。

此範例中使用的服務

- Amazon SNS
- Amazon Translate

如需 AWS SDK 開發人員指南和程式碼範例的完整清單，請參閱 [搭配 AWS SDK 使用此服務](#)。此主題也包含有關入門的資訊和舊版 SDK 的詳細資訊。

建立可分析客戶意見回饋並合成音訊的應用程式

下列程式碼範例示範如何建立應用程式，以分析客戶評論卡、從其原始語言進行翻譯、判斷對方情緒，以及透過翻譯後的文字產生音訊檔案。

.NET

適用於 .NET 的 SDK

此範例應用程式會分析和存儲客戶的意見回饋卡。具體來說，它滿足了紐約市一家虛構飯店的需求。飯店以實體評論卡的形式收到賓客以各種語言撰寫的意見回饋。這些意見回饋透過 Web 用戶端上傳至應用程式。評論卡的影像上傳後，系統會執行下列步驟：

- 文字內容是使用 Amazon Textract 從影像中擷取。
- Amazon Comprehend 會決定擷取文字及其用語的情感。
- 擷取的文字內容會使用 Amazon Translate 翻譯成英文。
- Amazon Polly 會使用擷取的文字內容合成音訊檔案。

完整的應用程式可透過 AWS CDK 部署。如需原始程式碼和部署的說明，請參閱 [GitHub](#) 中的專案。

此範例中使用的服務

- Amazon Comprehend
- Lambda
- Amazon Polly
- Amazon Textract
- Amazon Translate

Java

適用於 Java 2.x 的 SDK

此範例應用程式會分析和存儲客戶的意見回饋卡。具體來說，它滿足了紐約市一家虛構飯店的需求。飯店以實體評論卡的形式收到賓客以各種語言撰寫的意見回饋。這些意見回饋透過 Web 用戶端上傳至應用程式。評論卡的影像上傳後，系統會執行下列步驟：

- 文字內容是使用 Amazon Textract 從影像中擷取。
- Amazon Comprehend 會決定擷取文字及其用語的情感。
- 擷取的文字內容會使用 Amazon Translate 翻譯成英文。
- Amazon Polly 會使用擷取的文字內容合成音訊檔案。

完整的應用程式可透過 AWS CDK 部署。如需原始程式碼和部署的說明，請參閱 [GitHub](#) 中的專案。

此範例中使用的服務

- Amazon Comprehend
- Lambda
- Amazon Polly
- Amazon Textract
- Amazon Translate

JavaScript

適用於 JavaScript (v3) 的 SDK

此範例應用程式會分析和存儲客戶的意見回饋卡。具體來說，它滿足了紐約市一家虛構飯店的需求。飯店以實體評論卡的形式收到賓客以各種語言撰寫的意見回饋。這些意見回饋透過 Web 用戶端上傳至應用程式。評論卡的影像上傳後，系統會執行下列步驟：

- 文字內容是使用 Amazon Textract 從影像中擷取。
- Amazon Comprehend 會決定擷取文字及其用語的情感。
- 擷取的文字內容會使用 Amazon Translate 翻譯成英文。
- Amazon Polly 會使用擷取的文字內容合成音訊檔案。

完整的應用程式可透過 AWS CDK 部署。如需原始程式碼和部署的說明，請參閱 [GitHub](#) 中的專案。以下摘錄顯示如何在 Lambda 函數內適用於 JavaScript 的 AWS SDK 使用。

```
import {
  ComprehendClient,
  DetectDominantLanguageCommand,
  DetectSentimentCommand,
} from "@aws-sdk/client-comprehend";

/**
 * Determine the language and sentiment of the extracted text.
 *
 * @param {{ source_text: string }} extractTextOutput
 */
export const handler = async (extractTextOutput) => {
  const comprehendClient = new ComprehendClient({});

  const detectDominantLanguageCommand = new DetectDominantLanguageCommand({
    Text: extractTextOutput.source_text,
  });

  // The source language is required for sentiment analysis and
  // translation in the next step.
  const { Languages } = await comprehendClient.send(
    detectDominantLanguageCommand,
  );

  const languageCode = Languages[0].LanguageCode;
```

```
const detectSentimentCommand = new DetectSentimentCommand({
  Text: extractTextOutput.source_text,
  LanguageCode: languageCode,
});

const { Sentiment } = await comprehendClient.send(detectSentimentCommand);

return {
  sentiment: Sentiment,
  language_code: languageCode,
};
};
```

```
import {
  DetectDocumentTextCommand,
  TextractClient,
} from "@aws-sdk/client-textract";

/**
 * Fetch the S3 object from the event and analyze it using Amazon Textract.
 *
 * @param {import("@types/aws-lambda").EventBridgeEvent<"Object Created">}
  eventBridgeS3Event
 */
export const handler = async (eventBridgeS3Event) => {
  const textractClient = new TextractClient();

  const detectDocumentTextCommand = new DetectDocumentTextCommand({
    Document: {
      S3Object: {
        Bucket: eventBridgeS3Event.bucket,
        Name: eventBridgeS3Event.object,
      },
    },
  });

  // Textract returns a list of blocks. A block can be a line, a page, word, etc.
  // Each block also contains geometry of the detected text.
  // For more information on the Block type, see https://docs.aws.amazon.com/textract/latest/dg/API\_Block.html.
  const { Blocks } = await textractClient.send(detectDocumentTextCommand);

  // For the purpose of this example, we are only interested in words.
```

```
const extractedWords = Blocks.filter((b) => b.BlockType === "WORD").map(
  (b) => b.Text,
);

return extractedWords.join(" ");
};
```

```
import { PollyClient, SynthesizeSpeechCommand } from "@aws-sdk/client-polly";
import { S3Client } from "@aws-sdk/client-s3";
import { Upload } from "@aws-sdk/lib-storage";

/**
 * Synthesize an audio file from text.
 *
 * @param {{ bucket: string, translated_text: string, object: string}}
 * sourceDestinationConfig
 */
export const handler = async (sourceDestinationConfig) => {
  const pollyClient = new PollyClient({});

  const synthesizeSpeechCommand = new SynthesizeSpeechCommand({
    Engine: "neural",
    Text: sourceDestinationConfig.translated_text,
    VoiceId: "Ruth",
    OutputFormat: "mp3",
  });

  const { AudioStream } = await pollyClient.send(synthesizeSpeechCommand);

  const audioKey = `${sourceDestinationConfig.object}.mp3`;

  // Store the audio file in S3.
  const s3Client = new S3Client();
  const upload = new Upload({
    client: s3Client,
    params: {
      Bucket: sourceDestinationConfig.bucket,
      Key: audioKey,
      Body: AudioStream,
      ContentType: "audio/mp3",
    },
  });
};
```

```
    await upload.done();
    return audioKey;
};
```

```
import {
  TranslateClient,
  TranslateTextCommand,
} from "@aws-sdk/client-translate";

/**
 * Translate the extracted text to English.
 *
 * @param {{ extracted_text: string, source_language_code: string }}
  textAndSourceLanguage
 */
export const handler = async (textAndSourceLanguage) => {
  const translateClient = new TranslateClient({});

  const translateCommand = new TranslateTextCommand({
    SourceLanguageCode: textAndSourceLanguage.source_language_code,
    TargetLanguageCode: "en",
    Text: textAndSourceLanguage.extracted_text,
  });

  const { TranslatedText } = await translateClient.send(translateCommand);

  return { translated_text: TranslatedText };
};
```

此範例中使用的服務

- Amazon Comprehend
- Lambda
- Amazon Polly
- Amazon Textract
- Amazon Translate

Ruby

SDK for Ruby

此範例應用程式會分析和存儲客戶的意見回饋卡。具體來說，它滿足了紐約市一家虛構飯店的需求。飯店以實體評論卡的形式收到賓客以各種語言撰寫的意見回饋。這些意見回饋透過 Web 用戶端上傳至應用程式。評論卡的影像上傳後，系統會執行下列步驟：

- 文字內容是使用 Amazon Textract 從影像中擷取。
- Amazon Comprehend 會決定擷取文字及其用語的情感。
- 擷取的文字內容會使用 Amazon Translate 翻譯成英文。
- Amazon Polly 會使用擷取的文字內容合成音訊檔案。

完整的應用程式可透過 AWS CDK 部署。如需原始程式碼和部署的說明，請參閱 [GitHub](#) 中的專案。

此範例中使用的服務

- Amazon Comprehend
- Lambda
- Amazon Polly
- Amazon Textract
- Amazon Translate

如需 AWS SDK 開發人員指南和程式碼範例的完整清單，請參閱 [搭配 AWS SDK 使用此服務](#)。此主題也包含有關入門的資訊和舊版 SDK 的詳細資訊。


使用 AWS SDK 開始使用 Amazon Translate 任務

以下程式碼範例顯示做法：

- 啟動非同步批次翻譯任務。
- 等候非同步任務完成。
- 描述非同步任務。

SAP ABAP

適用於 SAP ABAP 的開發套件

 Note

GitHub 上提供更多範例。尋找完整範例，並了解如何在 [AWS 程式碼範例儲存庫](#) 中設定和執行。

```
DATA lo_inputdataconfig TYPE REF TO /aws1/cl_xl8inputdataconfig.
DATA lo_outputdataconfig TYPE REF TO /aws1/cl_xl8outputdataconfig.
DATA lt_targetlanguagecodes TYPE /aws1/
cl_xl8tgtlanguagecodes00=>tt_targetlanguagecodestrlist.
DATA lo_targetlanguagecodes TYPE REF TO /aws1/cl_xl8tgtlanguagecodes00.

"Create an ABAP object for the input data config."
lo_inputdataconfig = NEW #( iv_s3uri = iv_input_data_s3uri
                           iv_contenttype = iv_input_data_contenttype ).

"Create an ABAP object for the output data config."
lo_outputdataconfig = NEW #( iv_s3uri = iv_output_data_s3uri ).

"Create an internal table for target languages."
lo_targetlanguagecodes = NEW #( iv_value = iv_targetlanguagecode ).
INSERT lo_targetlanguagecodes INTO TABLE lt_targetlanguagecodes.

TRY.
  DATA(lo_translationjob_result) = lo_xl8->starttexttranslationjob(
    io_inputdataconfig = lo_inputdataconfig
    io_outputdataconfig = lo_outputdataconfig
    it_targetlanguagecodes = lt_targetlanguagecodes
    iv_dataaccessrolelearn = iv_dataaccessrolelearn
    iv_jobname = iv_jobname
    iv_sourcelanguagecode = iv_sourcelanguagecode ).
  MESSAGE 'Translation job started.' TYPE 'I'.
CATCH /aws1/cx_xl8internalserverex.
  MESSAGE 'An internal server error occurred. Retry your request.' TYPE
'E'.
CATCH /aws1/cx_xl8invparamvalueex.
  MESSAGE 'The value of the parameter is not valid.' TYPE 'E'.
```

```

CATCH /aws1/cx_xl8invalidrequestex.
    MESSAGE 'The request that you made is not valid.' TYPE 'E'.
CATCH /aws1/cx_xl8resourcenotfoundex.
    MESSAGE 'The resource you are looking for has not been found.' TYPE 'E'.
CATCH /aws1/cx_xl8toomanyrequestsex.
    MESSAGE 'You have made too many requests within a short period of time.'
TYPE 'E'.
CATCH /aws1/cx_xl8unsuppdedlanguage00.
    MESSAGE 'Amazon Translate does not support translation from the language
of the source text into the requested target language.' TYPE 'E'.
ENDTRY.

"Get the job ID."
DATA(lv_jobid) = lo_translationjob_result->get_jobid( ).

"Wait for translate job to complete."
DATA(lo_des_translation_result) = lo_xl8-
>describetexttranslationjob( iv_jobid = lv_jobid ).
    WHILE lo_des_translation_result->get_textxlationjobproperties( )-
>get_jobstatus( ) <> 'COMPLETED'.
        IF sy-index = 30.
            EXIT.                "Maximum 900 seconds."
        ENDIF.
    WAIT UP TO 30 SECONDS.
    lo_des_translation_result = lo_xl8->describetexttranslationjob( iv_jobid =
lv_jobid ).
ENDWHILE.

TRY.
    oo_result = lo_xl8->describetexttranslationjob(      "oo_result is
returned for testing purposes."
    iv_jobid      = lv_jobid ).
    MESSAGE 'Job description retrieved.' TYPE 'I'.
    CATCH /aws1/cx_xl8internalserverex.
        MESSAGE 'An internal server error occurred. Retry your request.' TYPE
'E'.
    CATCH /aws1/cx_xl8resourcenotfoundex.
        MESSAGE 'The resource you are looking for has not been found.' TYPE 'E'.
    CATCH /aws1/cx_xl8toomanyrequestsex.
        MESSAGE 'You have made too many requests within a short period of time.'
TYPE 'E'.
ENDTRY.

```

- 如需 API 詳細資訊，請參閱《適用於 SAP ABAP 的 AWS SDK API 參考》中的下列主題。
 - [DescribeTextTranslationJob](#)
 - [StartTextTranslationJob](#)

如需 AWS SDK 開發人員指南和程式碼範例的完整清單，請參閱 [搭配 AWS SDK 使用此服務](#)。此主題也包含有關入門的資訊和舊版 SDK 的詳細資訊。

標記您的 資源

標籤是您可以與 Amazon Translate 資源建立關聯的中繼資料。一個標籤包含一對索引鍵/值對。您可以將標籤新增至平行資料和自訂術語資源。

標籤有兩個主要功能：組織您的資源，並提供標籤型存取控制。您可以將標籤新增至資源，然後建立 IAM 政策，以允許或限制根據其標籤存取資源。

政策可以根據您請求中提供的標籤 (request-tags) 或與您呼叫之資源相關聯的標籤 (resource-tags)，來允許或不允許 操作。如需搭配 IAM 使用標籤的詳細資訊，請參閱《IAM 使用者指南》中的[使用標籤控制存取](#)。

搭配 Amazon Translate 使用標籤的考量：

- 每個資源最多可以新增 50 個使用者標籤。
- 您可以在建立資源時或建立資源後的任何時間新增標籤。
- 標籤索引鍵是必要欄位，但標籤值是選用的。
- 標籤在資源之間不一定是唯一的，但指定資源的標籤必須具有唯一的金鑰。
- 標籤鍵與值皆區分大小寫。
- 標籤索引鍵最多可有 128 個字元；標籤值最多可有 256 個字元。
- AWS 系統標籤以標籤索引鍵或值aws:中的字首開頭。您無法新增、編輯或刪除具有此字首的標籤名稱或值。每個資源的標籤配額中不包含系統標籤。

Note

如果您計劃跨多個 AWS 服務和資源使用標記結構描述，請記住，其他服務對允許的字元可能有不同的要求。

主題

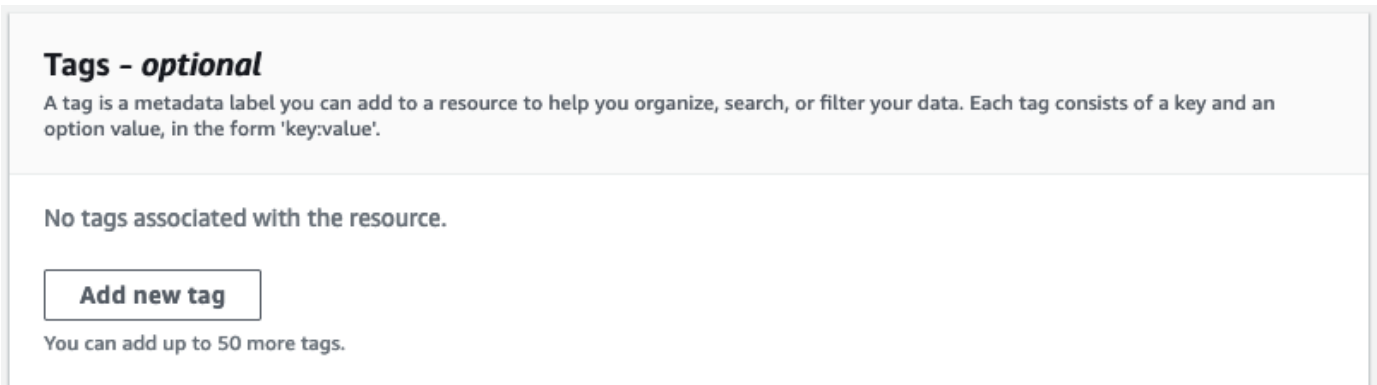
- [標記新資源](#)
- [檢視、更新和刪除與資源相關聯的標籤](#)

標記新資源

您可以在建立標籤時，將標籤新增至 ParallelData 或自訂術語資源。

將標籤新增至新資源（主控台）

1. 登入 [Amazon Translate 主控台](#)。
2. 從左側導覽窗格中，選取您要建立的資源 (Parallel data 或 Custom terminology)。
3. 選擇建立平行資料或建立術語。主控台會顯示資源的主要「建立」頁面。在此頁面的結尾，您會看到「標籤 - 選用」面板。



Tags - optional

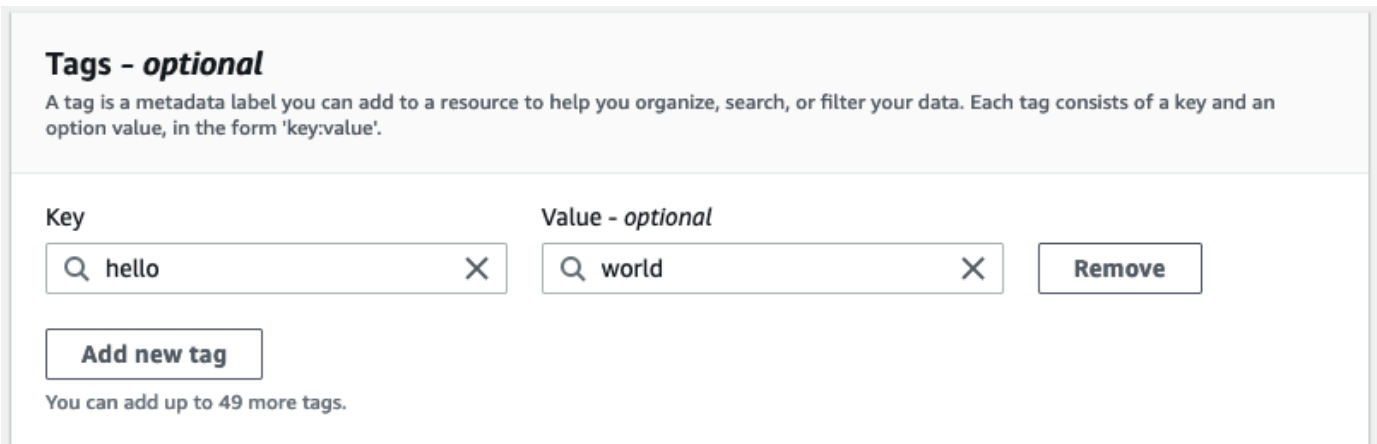
A tag is a metadata label you can add to a resource to help you organize, search, or filter your data. Each tag consists of a key and an option value, in the form 'key:value'.

No tags associated with the resource.

Add new tag

You can add up to 50 more tags.

4. 選擇新增標籤以新增資源的標籤。輸入標籤索引鍵和選用的標籤值。



Tags - optional

A tag is a metadata label you can add to a resource to help you organize, search, or filter your data. Each tag consists of a key and an option value, in the form 'key:value'.

Key: ×

Value - optional: × Remove

Add new tag

You can add up to 49 more tags.

5. 重複步驟 4，直到您新增所有標籤。此資源的每個金鑰都必須是唯一的。

Tags - optional

A tag is a metadata label you can add to a resource to help you organize, search, or filter your data. Each tag consists of a key and an option value, in the form 'key:value'.

Key	Value - optional	
<input style="border: 1px solid #ccc;" type="text" value="hello"/>	<input style="border: 1px solid #ccc;" type="text" value="world"/>	<input type="button" value="Remove"/>
<input style="border: 1px solid #ccc;" type="text" value="Enter key"/>	<input style="border: 1px solid #ccc;" type="text" value="Enter value"/>	<input type="button" value="Remove"/>

⚠ You must specify a tag key

You can add up to 48 more tags.

6. 選擇建立平行資料或建立術語來建立資源。

您也可以使用 Amazon Translate [CreateParallelData](#) API 操作新增標籤。下列範例顯示如何使用 create-parallel-data CLI 命令新增標籤。

```
aws translate create-parallel-data \
--name "myTest" \
--parallel-data-config "{\"format\": \"CSV\", \
    \"S3Uri\": \"s3://test-input/TEST.csv\"}" \
--tags "[{\"Key\": \"color\", \"Value\": \"orange\"}]"
```

檢視、更新和刪除與資源相關聯的標籤

您可以檢視、更新或刪除與 Parallel data 或 Custom terminology 資源相關聯的標籤。

更新現有資源的標籤（主控台）

1. 登入 [Amazon Translate 主控台](#)。
2. 從左側導覽窗格中，選取 Parallel data 或 Custom terminology。
3. 選取包含您要檢視、更新或刪除之標籤的資源。主控台會開啟資源的詳細資訊頁面。
4. 向下捲動，直到您看到標籤面板。在這裡，您可以看到與所選資源相關聯的所有標籤。

Tags (2)		Manage tags
Key	▲	Value
color		orange
type		PDF

選取管理標籤，從您的資源編輯或移除標籤。

5. 選擇您要修改的文字，然後編輯標籤。您也可以選取移除來移除標籤。
6. 若要新增標籤，請選取新增標籤，然後在空白欄位中輸入索引鍵和值。
7. 修改標籤完成後，請選取儲存。

Amazon Translate 的安全性

的雲端安全性 AWS 是最高優先順序。身為 AWS 客戶，您可以受益於資料中心和網路架構，這些架構是為了滿足最安全敏感組織的需求而建置。

安全性是 AWS 與您之間共同責任。[共同責任模型](#) 將此描述為雲端的安全和雲端內的安全：

- 雲端的安全性 – AWS 負責保護在 Cloud AWS 中執行 AWS 服務的基礎設施。AWS 也為您提供可安全使用的服務。在 [AWS 合規計畫](#) 中，第三方稽核員會定期測試並驗證我們的安全功效。若要了解適用於 Amazon Translate 的合規計畫，請參閱 [AWS 合規計畫範圍內的服務](#)。
- 雲端的安全性 – 您的責任取決於您使用 AWS 的服務。您也必須對其他因素負責，包括資料的機密性、您的公司的要求和適用法律和法規。

本主題有助於您了解，共同責任模型在使用 AWS 時的適用情形。下列主題說明如何設定 AWS 以符合您的安全與合規目標。您也會了解如何使用其他 AWS 服務來協助您監控和保護 AWS 資源。

主題

- [Amazon Translate 中的資料保護](#)
- [Amazon Translate 的 Identity and Access Management](#)
- [監控 Amazon Translate](#)
- [Amazon Translate 的合規驗證](#)
- [Amazon Translate 中的彈性](#)
- [Amazon Translate 中的基礎設施安全性](#)
- [Amazon Translate 和介面 VPC 端點 \(AWS PrivateLink\)](#)

Amazon Translate 中的資料保護

Amazon Translate 符合 AWS [共同的責任模型](#)，其中包括資料保護的法規和指導方針。AWS 負責保護執行所有 AWS 服務的 全球基礎設施。AWS 維護對此基礎設施上託管資料的控制，包括處理客戶內容和個人資料的安全組態控制。AWS 客戶和 APN 合作夥伴、做為資料控制者或資料處理者，負責他們放入 AWS 雲端的任何個人資料。

基於資料保護目的，建議您保護 AWS 帳戶登入資料並使用 AWS Identity and Access Management (IAM) 設定角色，以便每個使用者只獲得完成其任務所需的許可。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。
- 使用 設定 API 和使用者活動記錄 AWS CloudTrail。
- 使用 AWS 加密解決方案，以及 AWS 服務中的所有預設安全控制。
- 使用進階受管安全服務，例如 Amazon Macie，可協助探索和保護存放在 Amazon Simple Storage Service (Amazon S3) 中的個人資料。

我們強烈建議您絕對不要將客戶帳戶號碼等敏感的識別資訊，放在自由格式的欄位中，例如 Name (名稱) 欄位。這包括當您使用 Amazon Translate 或使用主控台 AWS CLI、API 或 AWS SDKs 的其他 AWS 服務時。您在 Amazon Translate 或其他服務中輸入的任何資料都可能被選入診斷日誌中。當您提供外部伺服器的 URL 時，請勿在驗證您對該伺服器請求的 URL 中包含憑證資訊。

如需關於資料保護的詳細資訊，請參閱 AWS 安全部落格上的 [AWS 共同責任模型和歐盟《一般資料保護規範》\(GDPR\) 部落格文章](#)。

主題

- [靜態加密](#)
- [傳輸中加密](#)

靜態加密

對於您使用 Amazon Translate 執行的批次翻譯任務，您的翻譯輸入和輸出都會靜態加密。不過，每個的加密方法都不同。

Amazon Translate 也會使用使用預設金鑰加密的 Amazon Elastic Block Store (Amazon EBS) 磁碟區。

轉譯輸入

當您使用 Amazon Translate 批次翻譯文件時，您可以將一組輸入文件存放在 Amazon S3 儲存貯體中。若要靜態加密這些文件，您可以使用 Amazon S3 提供的 SSE-S3 伺服器端加密選項。Amazon S3 使用此選項時，每個物件都會使用由 Amazon S3 管理的唯一金鑰進行加密。

如需詳細資訊，請參閱《Amazon Simple Storage Service 使用者指南》中的 [透過 Amazon S3 受管加密金鑰 \(SSE-S3\) 使用伺服器端加密來保護資料](#)。

轉譯輸出

當 Amazon Translate 完成批次翻譯任務時，會將輸出放入您 AWS 帳戶中的 Amazon S3 儲存貯體。為了加密靜態輸出，Amazon Translate 會使用 Amazon S3 提供的 SSE-KMS 加密選項。使用此選項時，您的輸出會使用存放在 AWS Key Management Service () 中的金鑰加密 AWS KMS。

如需 SSE-KMS 的詳細資訊，請參閱《Amazon Simple Storage Service 使用者指南》中的[使用伺服器端加密搭配 AWS Key Management Service \(SSE-KMS\) 保護資料](#)。

如需 KMS 金鑰的詳細資訊，請參閱《AWS Key Management Service 開發人員指南》中的[AWS KMS keys](#)。

對於此加密，Amazon Translate 可以使用下列任一類型的金鑰：

AWS 受管金鑰

根據預設，Amazon Translate 會使用 AWS 受管金鑰。這種類型的 KMS 金鑰是由建立 AWS 並存放在您的帳戶中。不過，您無法自行管理此 KMS 金鑰。它僅由代您管理和使用 AWS。

客戶自管金鑰

或者，您可以選擇使用客戶受管金鑰來加密輸出，這是您在 AWS 帳戶中建立、擁有和管理的 KMS 金鑰。

您必須先將許可新增至 Amazon Translate 用來存取 Amazon S3 中輸出儲存貯體的 IAM 服務角色，才能使用自己的 KMS 金鑰。如果您想要使用不同帳戶中的 KMS 金鑰 AWS，您還必須在其中更新金鑰政策 AWS KMS。如需詳細資訊，請參閱[自訂加密的先決條件許可](#)。

執行批次翻譯任務時，您可以選擇使用客戶受管金鑰。如需詳細資訊，請參閱[執行批次翻譯任務](#)。

傳輸中加密

若要加密傳輸中的資料，Amazon Translate 會使用 TLS 1.2 搭配 AWS 憑證。

Amazon Translate 的 Identity and Access Management

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 Translate 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](#)
- [Amazon Translate 如何與 IAM 搭配使用](#)
- [Amazon Translate 的身分型政策範例](#)
- [AWS Amazon Translate 的 受管政策](#)
- [對 Amazon Translate 身分和存取進行故障診斷](#)

目標對象

如何使用 AWS Identity and Access Management (IAM) 會因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 Translate 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 Translate 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 Translate 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，該身分可完整存取所有 AWS 服務和資源。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來 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。您無法在資源型政策中使用來自 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，請參閱《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 AWS 在涉及多個政策類型時決定是否允許請求，請參閱《IAM 使用者指南》中的[政策評估邏輯](#)。

Amazon Translate 如何與 IAM 搭配使用

在您使用 IAM 管理 Amazon Translate 的存取權之前，請先了解哪些 IAM 功能可與 Amazon Translate 搭配使用。

您可以搭配 Amazon Translate 使用的 IAM 功能

IAM 功能	Amazon Translate 支援
身分型政策	是
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
轉送存取工作階段 (FAS)	是
服務角色	是
服務連結角色	否

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

Amazon Translate 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 Translate 的身分型政策範例

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

Amazon Translate 中的資源型政策

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](#)。委託人可以包含帳戶、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 Translate 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 Translate 動作的清單，請參閱《服務授權參考》中的 [Amazon Translate 定義的動作](#)。

Amazon Translate 中的政策動作在動作之前使用以下字首：

```
translate
```

如需在單一陳述式中指定多個動作，請用逗號分隔。

```
"Action": [  
    "translate:ListLanguages",  
    "translate:TranslateText"
```

```
] ]
```

您也可以使用萬用字元 (*) 來指定多個動作。例如，若要指定開頭是 List 文字的所有動作，請包含以下動作：

```
"Action": "translate:List*"
```

請勿使用萬用字元來指定服務的所有動作。當您在政策中指定許可時，請使用授予最低權限的最佳實務。

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

Amazon Translate 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 Translate 資源類型及其 ARNs，請參閱《服務授權參考》中的 [Amazon Translate 定義的資源](#)。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 Translate 定義的動作](#)。

如需如何在 Amazon Translate 政策中使用資源的範例，請參閱 [在政策中指定資源](#)。

Amazon Translate 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 Translate 條件索引鍵的清單，請參閱《服務授權參考》中的 [Amazon Translate 的條件索引鍵](#)。若要了解您可以使用條件金鑰的動作和資源，請參閱 [Amazon Translate 定義的動作](#)。

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

Amazon Translate 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 Translate

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 \(ABAC\)](#)。

如需標記 Amazon Translate 資源的詳細資訊，請參閱 [標記您的資源](#)。

搭配 Amazon Translate 使用臨時登入資料

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](#)。

轉送 Amazon Translate 的存取工作階段

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，並結合請求 AWS 服務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 Translate 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 Translate 功能。只有在 Amazon Translate 提供指引時，才能編輯服務角色。

若要使用 Amazon Translate 非同步操作，您必須授予 Amazon Translate 存取包含輸入文件的 Amazon S3 儲存貯體。您可以透過在帳戶中建立具有信任政策的服務角色來信任 Amazon Translate 服務主體來執行此操作。

如需政策範例，請參閱[批次翻譯任務的先決條件](#)。

Amazon Translate 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 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。

如需建立或管理服務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 Translate 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 Translate 資源的許可。他們也無法使用 AWS 管理主控台 AWS CLI 或 AWS API 執行任務。IAM 管理員必須建立 IAM 政策，授予對其所需的特定資源執行特定 API 操作的許可。管理員接著必須將這些政策連接至需要這些許可的使用者或角色。

若要了解如何使用下列範例 JSON 政策文件建立 IAM 身分型政策，請參閱《IAM 使用者指南》中的[在 JSON 標籤上建立政策](#)。

主題

- [身分型政策最佳實務](#)
- [允許存取 Amazon Translate 主控台](#)
- [允許使用者檢視他們自己的許可](#)
- [在政策中指定資源](#)
- [搭配自訂術語使用客戶受管金鑰的許可](#)

身分型政策最佳實務

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 Translate 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 AWS 帳戶中使用。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 例如 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](#)。

允許存取 Amazon Translate 主控台

若要存取 Amazon Translate 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 AWS 帳戶中 Amazon Translate 資源的詳細資訊。如果您建立的身分型政策比最低必要許可更嚴格，則主控台對該政策的實體（使用者、群組或角色）將無法如預期運作。

對於 Amazon Translate 主控台許可，您可以將 `TranslateFullAccess` AWS 受管政策連接至實體。如需詳細資訊，請參閱 [AWS Amazon Translate 的受管政策](#)。

您也需要下列政策中所示動作的許可。這些許可包含在 `TranslateFullAccess` 政策中。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iam:ListRoles",
        "iam:GetRole",
        "s3:ListAllMyBuckets",
        "s3:ListBucket",
        "s3:GetBucketLocation"
      ],
      "Resource": "*"
    }
  ]
}
```

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。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": "*"
  }
]
}

```

在政策中指定資源

對於許多 Amazon Translate API 動作，您可以透過指定動作允許（或不允許）的資源來限制政策的範圍。如需可指定資源的動作清單，請參閱 [Amazon Translate 定義的動作](#)。您可以在政策中指定下列資源：

- Custom terminology – 使用下列 ARN 格式：

```
arn:partition:translate:region:account:terminology/terminology-name/LATEST
```

- Parallel data – 使用下列 ARN 格式：

```
arn:partition:translate:region:account:parallel-data/parallel-data-name
```

您可以使用萬用字元在政策中指定多個資源。下列範例政策允許所有 Amazon Translate 動作的所有自訂術語資源。

Example

```
{
  "Sid": "Example1",
  "Effect": "Allow",
  "Action": "translate:*",
  "Resource": [
    "arn:aws:translate:us-west-2:123456789012:terminology/*"
  ]
}
```

下列範例政策拒絕存取 GetParallelData 動作的特定平行資料資源。

Example

```
{
  "Sid": "Example2",
  "Effect": "Deny",
  "Action": "translate:GetParallelData",
  "Resource": [
    "arn:aws:translate:us-west-2:123456789012:parallel-data/test-parallel-
data"
  ]
}
```

搭配自訂術語使用客戶受管金鑰的許可

如果您搭配 Amazon Translate 自訂術語使用 AWS Key Management Service (AWS KMS) 客戶受管金鑰，您可能需要 KMS 金鑰政策中的其他許可。

若要使用客戶受管金鑰呼叫 ImportTerminology 操作，請將下列許可新增至現有的 KMS 金鑰政策。

JSON

```
{
  "Id": "key-consolepolicy-3",
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Allow access for use with Amazon Translate",
      "Effect": "Allow",
      "Principal": {
        "AWS": "IAM USER OR ROLE ARN"
      },
      "Action": [
        "kms:CreateAlias",
        "kms:CreateGrant",
        "kms:DescribeKey",
        "kms:GenerateDataKey",
        "kms:GetKeyPolicy",
        "kms:PutKeyPolicy",
        "kms:RetireGrant"
      ],
      "Resource": "*"
    }
  ]
}
```

若要針對使用 KMS 客戶受管金鑰匯入的自訂術語呼叫 `GetTerminology` 操作，請在 KMS 金鑰政策中新增下列許可。

JSON

```
{
  "Id": "key-consolepolicy-3",
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Allow access for use with Amazon Translate",
      "Effect": "Allow",
      "Principal": {
        "AWS": "IAM USER OR ROLE ARN"
      }
    }
  ]
}
```

```

    },
    "Action": [
        "kms:Decrypt",
        "kms:GetKeyPolicy",
        "kms:PutKeyPolicy"
    ],
    "Resource": "*"
  }
]
}

```

若要針對使用客戶受管金鑰匯入的自訂術語呼叫 `ListTerminologies` 或 `DeleteTerminology` 操作，您不需要有任何特殊 AWS KMS 許可。

若要搭配所有自訂術語操作使用客戶受管金鑰，請在 KMS 金鑰政策中新增下列許可。

JSON

```

{
  "Id": "key-consolepolicy-3",
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Allow access for use with Amazon Translate",
      "Effect": "Allow",
      "Principal": {
        "AWS": "IAM USER OR ROLE ARN"
      },
      "Action": [
        "kms:CreateGrant",
        "kms:Decrypt",
        "kms:DescribeKey",
        "kms:GenerateDataKey",
        "kms:GetKeyPolicy",
        "kms:PutKeyPolicy",
        "kms:RetireGrant"
      ],
      "Resource": "*"
    }
  ]
}

```

如需 Amazon Translate 操作和資源的詳細資訊，請參閱《服務授權參考》中的 [Amazon Translate 的動作、資源和條件索引鍵](#)。

AWS Amazon Translate 的 受管政策

AWS 受管政策是由 AWS 受管政策建立和管理的獨立政策旨在為許多常用案例提供許可，以便您可以開始將許可指派給使用者、群組和角色。

請記住，AWS 受管政策可能不會授予特定使用案例的最低權限許可，因為這些許可可供所有 AWS 客戶使用。我們建議您定義特定於使用案例的 [客戶管理政策](#)，以便進一步減少許可。

您無法變更 AWS 受管政策中定義的許可。如果 AWS 更新受管政策中定義的許可，則更新會影響政策連接的所有主體身分（使用者、群組和角色）。當新的 AWS 服務 啟動或新的 API 操作可用於現有服務時，AWS 最有可能更新 AWS 受管政策。

如需詳細資訊，請參閱 IAM 使用者指南中的 [AWS 受管政策](#)。

主題

- [AWS 受管政策：TranslateFullAccess](#)
- [AWS 受管政策：TranslateReadOnly](#)
- [Amazon Translate AWS 受管政策的更新](#)

AWS 受管政策：TranslateFullAccess

此政策授予 Amazon Translate 資源、Amazon Comprehend DetectDominantLanguage API 操作和所需 CloudWatch API 操作的完整存取權。此政策也會授予清單並取得 Amazon S3 儲存貯體和 IAM 角色的許可。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "translate:*",
        "comprehend:DetectDominantLanguage",
        "cloudwatch:GetMetricStatistics",
        "cloudwatch:ListMetrics",
```

```

        "s3:ListAllMyBuckets",
        "s3:ListBucket",
        "s3:GetBucketLocation",
        "iam:ListRoles",
        "iam:GetRole"
    ],
    "Effect": "Allow",
    "Resource": "*"
}
]
}

```

AWS 受管政策：TranslateReadOnly

此政策授予許可，以存取不會修改與您帳戶相關聯資源的 Amazon Translate API 操作。此政策也授予存取 Amazon Comprehend DetectDominantLanguage API 操作和所需 CloudWatch API 操作的許可。

JSON

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "translate:TranslateText",
        "translate:TranslateDocument",
        "translate:GetTerminology",
        "translate:ListTerminologies",
        "translate:ListTextTranslationJobs",
        "translate:DescribeTextTranslationJob",
        "translate:GetParallelData",
        "translate:ListParallelData",
        "comprehend:DetectDominantLanguage",
        "cloudwatch:GetMetricStatistics",
        "cloudwatch:ListMetrics"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}

```

```
}
```

Amazon Translate AWS 受管政策的更新

檢視自此服務開始追蹤 Amazon Translate AWS 受管政策更新以來的詳細資訊。如需此頁面變更的自動提醒，請訂閱 Amazon Translate [文件歷史記錄](#) 頁面上的 RSS 摘要。

變更	描述	Date
TranslateReadOnly – 更新至現有政策	Amazon Translate 現在允許 TranslateReadOnly 政策中的 TranslateDocument 動作	2023 年 5 月 23 日
Amazon Translate 已開始追蹤變更	Amazon Translate 開始追蹤其 AWS 受管政策的變更。	2023 年 5 月 23 日

對 Amazon Translate 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 Translate 和 IAM 時可能遇到的常見問題。

主題

- [我無權在 Amazon Translate 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 Translate 資源](#)

我無權在 Amazon Translate 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 IAM 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 `translate:GetWidget` 許可時發生。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
translate:GetWidget on resource: my-example-widget
```

在此情況下，必須更新 Mateo 政策，允許他使用 `translate: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至 Amazon Translate。

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 `marymajor` 的 IAM 使用者嘗試使用主控台在 Amazon Translate 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 Translate 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 Translate 是否支援這些功能，請參閱 [Amazon Translate 如何與 IAM 搭配使用](#)。
- 若要了解如何提供您擁有 AWS 帳戶的資源存取權，請參閱《[IAM 使用者指南](#)》中的在您擁有 AWS 帳戶的另一個中為 IAM 使用者提供存取權。
- 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](#)。

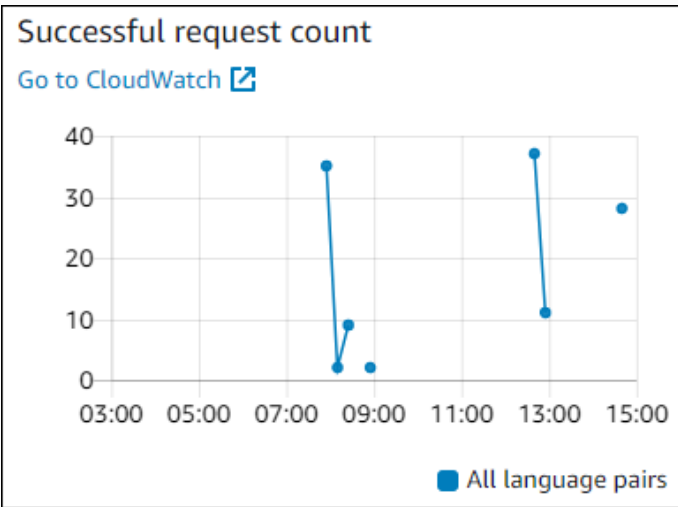
監控 Amazon Translate

監控是維護 Amazon Translate 和 解決方案可靠性、可用性和效能的重要部分。AWS 提供各種工具，可讓您用來監控 Amazon Translate。您可以設定其中一些工具，為您監控解決方案。建議您盡可能自動化監控任務。

Amazon Translate 提供預先設定的圖表，顯示解決方案最重要的指標。每個圖形都會提供一個視窗，顯示您解決方案的效能。若要取得您的解決方案如何隨時間執行的不同檢視，您可以變更圖形顯示的時間範圍。

您也可以使用 Amazon CloudWatch 來監控 Amazon Translate。使用 CloudWatch，您可以自動化監控解決方案的特定指標。每當您設定的指標超出閾值，就會收到通知。您也可以使用 CloudWatch API 來建立適合您需求的自訂監控應用程式。如需詳細資訊，請參閱《Amazon CloudWatch 使用者指南》中的 [什麼是 Amazon CloudWatch](#)。

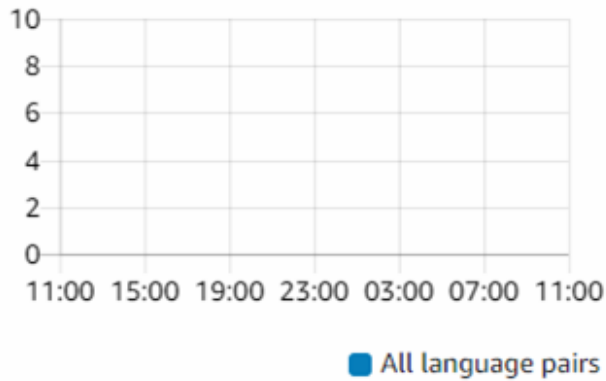
下表說明 Amazon Translate 提供的每個預先設定圖形。

圖表	Description
 <p>Successful request count</p> <p>Go to CloudWatch</p> <p>03:00 05:00 07:00 09:00 11:00 13:00 15:00</p> <p>All language pairs</p>	<p>成功請求計數</p> <p>在指定期間內對 Amazon Translate 提出的成功請求數量。</p>

圖表

Description

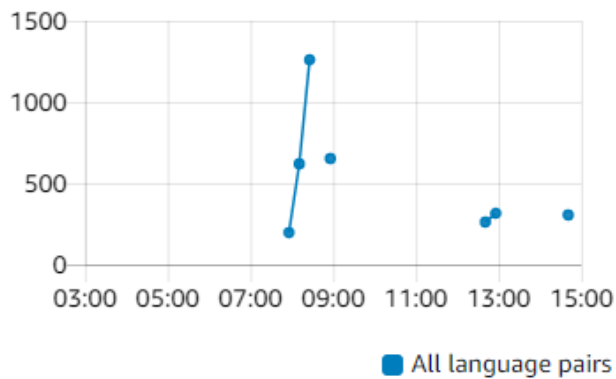
Throttled request count

[Go to CloudWatch](#)

已調節請求計數

在指定期間內對 Amazon Translate 限流的請求數量。使用此資訊來判斷您的應用程式是否太快將請求傳送至 Amazon Translate。

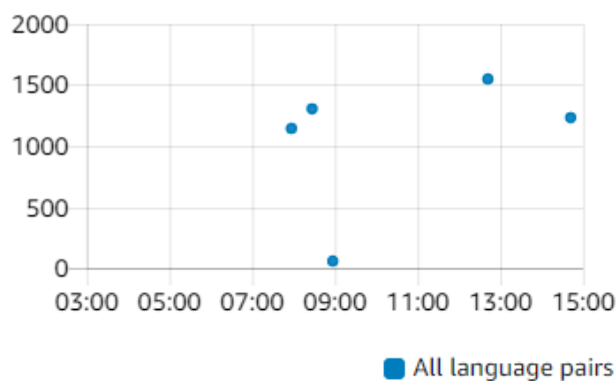
Average response time (milliseconds)

[Go to CloudWatch](#)

平均回應時間

Amazon Translate 在指定期間內處理您的請求所需的平均時間長度。

Character count

[Go to CloudWatch](#)

字元計數

您在指定期間內傳送至 Amazon Translate 的字元總數。這是您將付費的字元數。

圖表	Description
 <p>User error count Go to CloudWatch ↗</p> <p>4 2 0</p> <p>03:00 05:00 07:00 09:00 11:00 13:00 15:00</p> <p>■ All language pairs</p>	<p>使用者錯誤計數</p> <p>在指定時段發生的使用者錯誤數目。使用者錯誤位於 HTTP 錯誤碼範圍 400-499 內。</p>
 <p>System error count Go to CloudWatch ↗</p> <p>4 2 0</p> <p>11:00 15:00 19:00 23:00 03:00 07:00 11:00</p> <p>■ All language pairs</p>	<p>系統錯誤計數</p> <p>在指定時段發生的系統錯誤數目。系統錯誤位於 HTTP 錯誤碼範圍 500-599 內。</p>

監控 Amazon Translate

使用 Amazon CloudWatch，您可以取得個別 Amazon Translate 操作的指標或帳戶的全域 Amazon Translate 指標。使用指標來追蹤 Amazon Translate 解決方案的運作狀態，並設定警示，在一或多個指標超出定義的閾值時通知您。例如，您可以監控在特定期間內對 Amazon Translate 提出的請求數量、查看請求的延遲，或在錯誤超過閾值時發出警示。

了解 Amazon Translate 的 CloudWatch 指標

若要取得 Amazon Translate 操作的指標，請指定下列資訊：

- 指標維度。維度是一組用來識別指標的名稱與數值對。Amazon Translate 有兩個維度：
 - Operation
 - Language pair
- 指標名稱，例如 SuccessfulRequestCount 或 RequestCharacters。如需指標的完整清單，請參閱[Amazon Translate 的 CloudWatch 指標](#)。

您可以使用 AWS 管理主控台、AWS CLI 或 CloudWatch API 取得 Amazon Translate 的指標。您可以透過其中一個 Amazon AWS 軟體開發套件 (SDKs) 或 CloudWatch API 工具來使用 CloudWatch API。

下表列出 CloudWatch 指標的一些常見用途。這些是協助您開始的建議，而不是完整清單。

運作方式？	監控此指標
追蹤成功請求的數目	SuccessfulRequestCount 指標的 sum 統計資料
了解我的應用程式是否已經達到最大傳輸量	ThrottledCount 指標的 sum 統計資料
尋找應用程式的回應時間	ResponseTime 指標的 average 統計資料
尋找應用程式的錯誤數目	ServerErrorCount 和 UserErrorCount 指標的 sum 統計資料
尋找應計費字元數目	CharacterCount 指標的 sum 統計資料

您必須擁有適當的 CloudWatch 許可，以使用 CloudWatch 監控 Amazon Translate 如需詳細資訊，請參閱《[Amazon CloudWatch 使用者指南](#)》中的 [Amazon CloudWatch 的身分驗證和存取控制](#)。

Amazon CloudWatch

檢視 Amazon Translate 指標

在 CloudWatch 主控台中檢視 Amazon Translate 指標。

若要檢視指標 (CloudWatch 主控台)

1. 登入 AWS 管理主控台，並在 <https://console.aws.amazon.com/cloudwatch/> 開啟 CloudWatch 主控台。

2. 選擇 Metrics (指標)、選擇 All Metrics (所有指標)，然後選擇 AWS/Translate (AWS/翻譯)。
3. 選擇維度、選擇指標名稱，然後選擇 Add to graph (新增至圖形)。
4. 選擇日期範圍的值。所指定日期範圍的指標計數會顯示在圖形中。

使用 記錄 Amazon Translate API 呼叫 AWS CloudTrail

Amazon Translate 已與 整合 AWS CloudTrail，此服務提供 Amazon Translate 中 IAM 實體 AWS 或服務所採取動作的記錄。CloudTrail 會將 Amazon Translate 的所有 API 呼叫擷取為事件。這包括從 Amazon Translate 主控台進行的呼叫，以及對 Amazon Translate API 操作的程式碼呼叫。如果您建立 CloudTrail 追蹤，您可以啟用 CloudTrail 事件持續交付至 Amazon Simple Storage Service (Amazon S3) 儲存貯體，包括 Amazon Translate 的事件。即使您未設定追蹤，依然可以透過 CloudTrail 主控台的事件歷史記錄檢視最新事件。您可以使用 CloudTrail 收集的資訊來判斷對 Amazon Translate 提出的請求、提出請求的 IP 地址、提出請求的人員、提出請求的時間，以及其他詳細資訊。

若要進一步了解 CloudTrail，請參閱「[AWS CloudTrail 使用者指南](#)」。

主題

- [CloudTrail 中的 Amazon Translate 資訊](#)
- [了解 Amazon Translate 日誌檔案項目](#)

CloudTrail 中的 Amazon Translate 資訊

當您建立 AWS 帳戶時，會在您的帳戶上啟用 CloudTrail。當 Amazon Translate 中發生活動時，該活動會與事件歷史記錄中的其他 AWS 服務事件一起記錄在 CloudTrail 事件中。您可以在 AWS 帳戶中檢視、搜尋和下載最近的事件。如需詳細資訊，請參閱《使用 CloudTrail 事件歷史記錄檢視事件》<https://docs.aws.amazon.com/awscloudtrail/latest/userguide/view-cloudtrail-events.html>。

若要持續記錄您 AWS 帳戶中的事件，包括 Amazon Translate 的事件，請建立追蹤。線索能讓 CloudTrail 將日誌檔案交付至 Amazon S3 儲存貯體。依預設，當您使用主控台建立線索時，該線索會套用到所有 AWS 區域。該追蹤會記錄來自 AWS 分割區中所有區域的事件，並將日誌檔案交付到您指定的 S3 儲存貯體。您可以設定其他 AWS 服務，以進一步分析和處理 CloudTrail 日誌中收集的事件資料。如需詳細資訊，請參閱下列內容：

- [建立追蹤的概觀](#)
- [CloudTrail 支援的服務和整合](#)
- [設定 CloudTrail 的 Amazon SNS 通知](#)

- [從多個策略接收 CloudTrail 日誌檔案](#)，以及[從多個帳戶接收 CloudTrail 日誌檔案](#)

CloudTrail 會記錄所有 Amazon Translate 動作，並記錄在 [API 參考區段](#) 中。例如，對 DeleteTerminology、ImportTerminology 和 TranslateText 動作發出的呼叫會在 CloudTrail 記錄檔案中產生專案。

每一筆事件或日誌專案都會包含產生請求者的資訊。此資訊可協助您判斷下列事項：

- 是否使用根使用者登入資料提出請求
- 提出該請求時，是否使用了特定角色或聯合身分使用者的臨時安全憑證
- 請求是否由其他 AWS 服務提出

如需詳細資訊，請參閱 [CloudTrail userIdentity 元素](#)。

了解 Amazon Translate 日誌檔案項目

追蹤是一種組態，能讓事件以日誌檔案的形式交付到您指定的 Amazon S3 儲存貯體。CloudTrail 日誌檔案包含一或多個日誌專案。一個事件為任何來源提出的單一請求，並包含請求動作、請求的日期和時間、請求參數等資訊。CloudTrail 日誌檔並非依公有 API 呼叫的堆疊追蹤排序，因此不會以任何特定順序出現。

以下範例顯示的是展示 TranslateText 動作的 CloudTrail 日誌項目。

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
    "arn": "arn:aws:iam::111122223333:user/Administrator",
    "accountId": "111122223333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "userName": "Administrator"
  },
  "eventTime": "2019-09-03T20:32:50Z",
  "eventSource": "translate.amazonaws.com",
  "eventName": "TranslateText",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "192.0.2.0",
  "userAgent": "aws-cli/1.16.207 Python/3.4.7
Linux/4.9.184-0.1.ac.235.83.329.metal1.x86_64 boto3/1.12.197",
  "requestParameters": {
```

```

    "text": "HIDDEN_DUE_TO_SECURITY_REASONS",
    "sourceLanguageCode": "en",
    "targetLanguageCode": "fr"
  },
  "responseElements": {
    "translatedText": "HIDDEN_DUE_TO_SECURITY_REASONS",
    "sourceLanguageCode": "en",
    "targetLanguageCode": "fr"
  },
  "requestID": "f56da956-284e-4983-b6fc-59befa20e2bf",
  "eventID": "1dc75278-84d7-4bb2-861a-493d08d67391",
  "eventType": "AwsApiCall",
  "recipientAccountId": "111122223333"
}

```

Amazon Translate 的 CloudWatch 指標和維度

若要監控解決方案的效能，請使用 Amazon CloudWatch 指標和 Amazon Translate 的維度。

Amazon Translate 的 CloudWatch 指標

指標	Description
CharacterCount	<p>請求中的計費字元數。</p> <p>有效維度：語言對、操作</p> <p>有效統計資料：平均、最大值、最小值、總和</p> <p>單位：計數</p>
ResponseTime	<p>回應請求所花費的時間。</p> <p>有效維度：語言對、操作</p> <p>有效統計資料：資料範例、平均</p> <p>單位：對於資料範例，計數。對於平均統計資料，毫秒。</p>
ServerErrorCount	<p>伺服器錯誤的次數。伺服器錯誤的 HTTP 回應碼範圍為 500 到 599。</p> <p>有效維度：操作</p>

指標	Description
	<p>有效的統計資訊：平均、總和</p> <p>單位：計數</p>
SuccessfulRequestCount	<p>成功轉譯請求的數量。成功請求的回應碼為 200 到 299。</p> <p>有效維度：操作</p> <p>有效的統計資訊：平均、總和</p> <p>單位：計數</p>
ThrottledCount	<p>需要限流的請求數量。使用 ThrottledCount 來判斷您的應用程式向 Amazon Translate 傳送請求的速度是否比您的帳戶設定接受請求的速度更快。如需詳細資訊，請參閱 《》中的 Amazon Translate 限制 Amazon Web Services 一般參考。</p> <p>有效維度：操作</p> <p>有效的統計資訊：平均、總和</p> <p>單位：計數</p>
UserErrorCount	<p>發生的使用者錯誤數目。使用者錯誤的 HTTP 回應碼範圍是 400 到 499。</p> <p>有效維度：操作</p> <p>有效的統計資訊：平均、總和</p> <p>單位：計數</p>

Amazon Translate 的 CloudWatch 維度

使用以下維度篩選 Amazon Translate 指標。指標會依來源語言和目標語言分組。

維度	Description
LanguagePair	將指標限制為僅包含指定語言的指標。
Operation	將指標限制為只有具有指定操作的指標。

使用 Amazon EventBridge 監控 Amazon Translate 事件 EventBridge

Amazon Translate 與 Amazon EventBridge 整合，以通知您影響翻譯任務和平行資料資源的變更。來自 AWS 服務的事件會以近乎即時的方式交付至 EventBridge。您可編寫簡單的規則，來指示您在意的事件，以及當事件符合規則時所要自動執行的動作。例如，可自動啟動的動作包括：

- 叫用 AWS Lambda 函數
- 叫用 AWS Systems Manager Run 命令
- 將事件轉傳至 Amazon Kinesis Data Streams
- 啟用 AWS Step Functions 狀態機器
- 通知 Amazon SNS 主題或 Amazon SQS 佇列

如需詳細資訊，請參閱「Amazon EventBridge 使用者指南」中的[建立對事件做出反應的 Amazon EventBridge 規則](#)。

Amazon Translate 事件

以下是來自 Amazon Translate 的範例事件。

批次翻譯任務的事件

您可以使用 Amazon Translate 主控台或 [StartTextTranslationJob](#) 操作來執行批次翻譯任務。Amazon Translate 會在這些任務完成時傳送事件，無論是成功還是失敗。這些事件類似下列範例。

```
{
  "version": "0",
  "id": "CWE-event-id",
  "detail-type": "Translate TextTranslationJob State Change",
  "source": "aws.translate",
  "account": "111122223333",
  "time": "2017-04-22T03:31:47Z",
  "region": "us-east-1",
```

```
"resources": [],
"detail": {
  "jobId": "01234567-0123-0123-0123-012345678901",
  "jobStatus": "STATUS"
}
}
```

jobStatus 屬性的值取決於 Amazon Translate 傳送事件的目標任務狀態。這些 jobStatus 值為：

- COMPLETED – 任務已成功完成，且輸出可用。
- COMPLETED_WITH_ERROR – 任務已完成，但發生錯誤。您可以在任務的輸出中分析錯誤。
- STOPPED – 任務已停止。
- FAILED – 任務未完成。若要取得詳細資訊，請使用 [DescribeTextTranslationJob](#) 操作。

平行資料資源的事件

當您使用 Amazon Translate 建立或更新平行資料資源時，它會傳送事件來指出操作是否成功。

您可以使用 Amazon Translate 主控台或 [CreateParallelData](#) 操作來建立平行資料資源。當您這樣做時，Amazon Translate 會傳送類似下列的事件。

```
{
  "version": "0",
  "id": "CWE-event-id",
  "detail-type": "Translate Parallel Data State Change",
  "source": "aws.translate",
  "account": "111122223333",
  "time": "2017-04-22T03:31:47Z",
  "region": "us-east-1",
  "resources": [arn:aws:translate:us-east-1:111122223333:parallel-data/
ExampleParallelData],
  "detail": {
    "operation": "CreateParallelData",
    "name": "ExampleParallelData",
    "status": "STATUS"
  }
}
```

status 屬性的值為：

- ACTIVE – CreateParallelData 操作成功，且資源已準備好供您使用。

- FAILED – CreateParallelData操作失敗。

您可以使用 Amazon Translate 主控台或 [UpdateParallelData](#) 操作來更新平行資料資源。當您這樣做時，Amazon Translate 會傳送類似下列的事件。

```
{
  "version": "0",
  "id": "CWE-event-id",
  "detail-type": "Translate Parallel Data State Change",
  "source": "aws.translate",
  "account": "111122223333",
  "time": "2017-04-22T03:31:47Z",
  "region": "us-east-1",
  "resources": [arn:aws:translate:us-east-1:111122223333:parallel-data/
ExampleParallelData],
  "detail": {
    "operation": "UpdateParallelData",
    "name": "ExampleParallelData",
    "status": "STATUS",
    "latestUpdateAttemptStatus": "STATUS",
    "latestUpdateAttemptAt": "2017-04-22T03:31:47Z"
  }
}
```

status 屬性會提供先前版本的平行資料資源狀態，該資源正由更新取代。數值為：

- ACTIVE – 先前版本已成功建立或更新。
- FAILED – 無法建立或更新先前的版本。

latestUpdateAttemptStatus 屬性會提供由更新所建立之平行資料資源新版本的狀態。數值為：

- ACTIVE – UpdateParallelData操作成功，且已更新的資源已準備好供您使用。
- FAILED – UpdateParallelData操作失敗。

Amazon Translate 的合規驗證

在多個合規計畫中，第三方稽核人員會評估 Amazon Translate 的安全性和 AWS 合規性。這些計畫包括 PCI、FedRAMP、HIPAA 等等。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](#)。

您使用 Amazon Translate 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](#) – 這些部署指南討論在 AWS 上部署以安全及合規為重心基準環境的架構考量和步驟。
- [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](#) – AWS 此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 內安全狀態的全方位檢視 AWS，可協助您檢查是否符合安全產業標準和最佳實務。

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

Amazon Translate 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，這些可用區域以低延遲、高輸送量和高備援聯網連接。透過可用區域，您所設計與操作的應用程式和資料庫，就能夠在可用區域之間自動容錯移轉，而不會發生中斷。可用區域的可用性、容錯能力和擴充能力，均較單一或多個資料中心的傳統基礎設施還高。

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

Amazon Translate 中的基礎設施安全性

作為受管服務，Amazon Translate 受到 [Amazon Web Services：安全程序概觀](#) 白皮書中所述的 AWS 全球網路安全程序的保護。

若要透過網路存取 Amazon Translate，您可以使用 AWS 已發佈的 API 呼叫。用戶端必須支援 TLS 1.2 或更新版本。用戶端也必須支援具備完美轉送私密 (PFS) 的密碼套件，例如臨時 Diffie-Hellman (DHE) 或橢圓曲線臨時 Diffie-Hellman (ECDHE)。現代系統 (如 Java 7 和更新版本) 大多會支援這些模式。

此外，必須使用與 AWS Identity and Access Management (IAM) 主體相關聯的存取金鑰 ID 和私密存取金鑰來簽署請求。或者，您可以使用 [AWS Security Token Service](#) (AWS STS) 來產生暫時安全憑證來簽署請求。

Amazon Translate 和介面 VPC 端點 (AWS PrivateLink)

您可以建立介面 VPC 端點，在 VPC 和 Amazon Translate 之間建立私有連線。介面端點採用 [AWS PrivateLink](#) 技術，可讓您在沒有網際網路閘道、NAT 裝置、VPN 連接或 AWS Direct Connect 連接的情況下私下存取 Amazon Translate APIs。VPC 中的執行個體不需要公有 IP 地址，即可與 Amazon Translate APIs 通訊。VPC 和 Amazon Translate 之間的流量不會離開 Amazon 網路。

每個介面端點都是由您子網路中的一或多個[彈性網路介面](#)表示。

如需詳細資訊，請參閱《Amazon [VPC 使用者指南](#)》中的[介面 VPC 端點 \(AWS PrivateLink\)](#)。

Amazon Translate VPC 端點的考量事項

為 Amazon Translate 設定介面 VPC 端點之前，請務必檢閱《Amazon VPC 使用者指南》中的[介面端點屬性和限制](#)。

Amazon Translate 支援從您的 VPC 呼叫其所有 API 動作。

為 Amazon Translate 建立介面 VPC 端點

您可以使用 Amazon VPC 主控台或 () 為 Amazon Translate 服務建立 VPC 端點AWS CLI。AWS Command Line Interface 如需詳細資訊，請參閱《Amazon VPC 使用者指南》中的[建立介面端點](#)。

使用下列服務名稱為 Amazon Translate 建立 VPC 端點：

- `com.amazonaws.region.translate`

如果您為端點啟用私有 DNS，您可以使用區域的預設 DNS 名稱向 Amazon Translate 提出 API 請求，例如 `translate.us-east-1.amazonaws.com`。

如需詳細資訊，請參閱《Amazon VPC 使用者指南》中的[透過介面端點存取服務](#)。

為 Amazon Translate 建立 VPC 端點政策

您可以將端點政策連接至控制 Amazon Translate 存取的 VPC 端點。此政策會指定下列資訊：

- 可執行動作的主體。
- 可執行的動作。
- 可供執行動作的資源。

如需詳細資訊，請參閱 Amazon VPC 使用者指南中的[使用 VPC 端點控制對服務的存取](#)。

範例：Amazon Translate 即時轉譯動作的 VPC 端點政策

以下是 Amazon Translate 中即時轉譯的端點政策範例。連接到端點時，此政策會授予所有資源上所有主體的所列 Amazon Translate 動作的存取權。

```
{
  "Statement": [
    {
      "Principal": "*",
      "Effect": "Allow",
      "Action": [
        "translate:TranslateText",
      ],
      "Resource": "*"
    }
  ]
}
```

範例：Amazon Translate 批次轉譯動作的 VPC 端點政策

以下是 Amazon Translate 中批次翻譯的端點政策範例。連接到端點時，此政策會授予所有資源上所有主體的所列 Amazon Translate 動作的存取權。

```
{
  "Statement": [
    {
      "Principal": "*",
      "Effect": "Allow",
      "Action": [
        "translate:StartTextTranslationJob",
        "iam:PassRole"
      ],
      "Resource": "*"
    }
  ]
}
```

指南和配額

下列各節包含 Amazon Translate 指導方針和配額的相關資訊。

主題

- [支援的 AWS 區域](#)
- [合規](#)
- [限流](#)
- [指導方針](#)
- [Service Quotas](#)

支援的 AWS 區域

如需支援 Amazon Translate AWS 的區域清單，請參閱 AWS 一般參考中的 [Amazon Translate 端點和配額](#)。

合規

如需 Amazon Translate 合規計劃的詳細資訊，請參閱 [AWS 合規AWS 計劃範圍內的合規AWS、合規計劃和服務](#)。

限流

Amazon Translate 會擴展規模以服務客戶營運流量。如果您持續遇到調節的情況，請聯絡 [AWS Support](#)。

指導方針

為了持續改善分析模型的品質，Amazon Translate 可能會儲存您的資料。若要進一步了解，請參閱 [Amazon Translate 常見問答集](#)。

您可以聯絡 [AWS Support](#)，請我們刪除您的資料，而且不要保存未來與您的帳戶相關聯的資料。不過，由於刪除您的資料也可能刪除有助於改善翻譯的獨特訓練資料，這樣可能會降低翻譯品質。

Service Quotas

Amazon Translate 有下列服務準則和配額。

同步即時轉譯配額

描述	限制
字元編碼	UTF-8
最大輸入文字	10,000 位元組
每個文件的字元數上限	100,000
文件大小上限	100,000 位元組

非同步批次轉譯配額

描述	限制
字元編碼	UTF-8
每個文件的字元數上限	1,000,000
每個文件的大小上限	20 MB
單一文件中可翻譯文字的大小上限	1 MB
批次任務請求中的目標語言數量上限	10
批次中的文件數目上限	1,000,000
批次中文件總數的大小上限	5 GB
並行批次翻譯任務的數量上限	10
已排入佇列的批次翻譯任務數目上限	1000
StartTextTranslationJob API 動作的每秒交易數	5
DescribeTextTranslationJob API 動作的每秒交易數	10

描述	限制
ListTextTranslationJobs API 動作的每秒交易數	10
StopTextTranslationJob API 動作的每秒交易數	5

自訂術語配額

描述	限制
自訂術語檔案大小上限	10 MB
每個 AWS 區域每個 AWS 帳戶的自訂術語檔案數目上限	100
每個自訂術語檔案的目標語言數上限	10
每個自訂術語詞組的來源和目標文字長度上限	200 位元組
每個 TranslateText 或 StartTextTranslationJob 要求的術語檔案數目上限。	1
ImportTerminology API 動作的每秒交易數	5
GetTerminology API 動作的每秒交易數	10
ListTerminologies API 動作的每秒交易數	10
DeleteTerminology API 動作的每秒交易數	5

平行資料配額

描述	限制
每個 AWS 區域每個 AWS 帳戶的平行資料資源數目上限	1000
平行資料輸入檔案大小上限	5 GB
平行資料資源中的來源語言數量上限	1
平行資料輸入檔案中單一區段或記錄的大小上限	1000 個位元組

描述	限制
平行資料資源的並行建立或更新操作數量上限	1
CreateParallelData API 動作的每秒交易數	5
GetParallelData API 動作的每秒交易數	10
ListParallelData API 動作的每秒交易數	10
UpdateParallelData API 動作的每秒交易數	5
DeleteParallelData API 動作的每秒交易數	5

Amazon Translate 的文件歷史記錄

下表說明此 Amazon Translate 版本的文件。

變更	描述	日期
新功能：Brevity	翻譯現在支援即時文字翻譯的簡潔性。Brevity 可減少大多數翻譯的翻譯輸出長度（相較於不含簡潔的翻譯輸出）。如需詳細資訊，請參閱在 Amazon Translate 中使用簡潔 。	2023 年 10 月 31 日
即時翻譯的文件輸入語言自動偵測	您現在可以在將文件輸入即時翻譯（主控台或 API）時使用語言自動偵測。如需詳細資訊，請參閱 即時翻譯 。	2023 年 8 月 3 日
Word (.docx) 檔案作為即時轉譯的輸入	您現在可以使用 .docx 檔案（文字檔案和 HTML 檔案除外）做為即時轉譯（主控台或 API）的輸入。如需詳細資訊，請參閱 即時轉譯 。	2023 年 7 月 17 日
自訂術語的增強功能	翻譯現在支援增強自訂術語功能，以提高翻譯流暢性和準確性。如需詳細資訊，請參閱 使用自訂術語自訂翻譯 。	2023 年 6 月 30 日
文字或 HTML 檔案做為即時翻譯的輸入	您現在可以使用文字檔案或 HTML 檔案做為即時轉譯（主控台或 API）的輸入。如需詳細資訊，請參閱 即時轉譯 。	2023 年 5 月 23 日
TranslateReadOnly 政策中允許的新動作	Amazon Translate 現在允許 TranslateReadOnly 受管政策中的 Translate Document 動作。如需詳細資	2023 年 5 月 23 日

	訊，請參閱 AWS 受管政策：TranslateReadOnly 。	
Translate 現在支援非同步批次處理的其他區域。	翻譯現在支援非同步批次處理的其他區域。如需詳細資訊，請參閱 使用 Amazon Translate 進行非同步批次處理 。	2023 年 3 月 28 日
增加即時轉譯的輸入大小	您現在可以輸入最多 10,000 個字元進行即時翻譯。如需詳細資訊，請參閱 Amazon Translate 入門（主控台） 。	2022 年 12 月 16 日
支援批次模式的巢狀輸入資料夾	您現在可以提供巢狀輸入資料夾給批次翻譯任務。如需詳細資訊，請參閱 在 Amazon Translate 中執行批次翻譯任務 。Amazon Translate	2022 年 11 月 18 日
支援批次模式的自動語言偵測	您現在可以在批次翻譯任務中自動偵測來源語言。因此，您現在可以在批次翻譯任務中輸入具有不同來源語言的文件。如需詳細資訊，請參閱 在 Amazon Translate 中執行批次翻譯任務 。Amazon Translate	2022 年 11 月 18 日
支援多種目標語言	您現在可以在批次翻譯任務中指定多種目標語言。如需詳細資訊，請參閱 在 Amazon Translate 中執行批次翻譯任務 。Amazon Translate	2022 年 10 月 10 日
支援標籤	您現在可以在 Amazon Translate 中標記 ParallelData 和自訂術語資源。如需詳細資訊，請參閱 在 Amazon Translate 中標記您的資源 。	2022 年 10 月 6 日

形式支援其他語言	您現在可以在 Amazon Translate 中設定荷蘭文、韓文和墨西哥西班牙文的翻譯正式程度。如需詳細資訊，請參閱在 Amazon Translate 中設定正式性 。	2022 年 10 月 5 日
個別 API 參考	Amazon Translate API 參考現在是與 開發人員指南 不同的文件。如需詳細資訊，請參閱 Amazon Translate API 參考 。	2022 年 8 月 25 日
新功能	您現在可以設定翻譯輸出的正式程度。如需詳細資訊，請參閱在 Amazon Translate 中設定正式性 。	2022 年 2 月 22 日
新功能	您現在可以在翻譯輸出中遮罩褻瀆單字和片語。如需詳細資訊，請參閱 Amazon Translate 中的遮罩褻瀆單字和片語 。	2021 年 11 月 24 日
AWS PrivateLink 支援	您現在可以使用在 VPC 和 Amazon Translate 之間建立私有連線 AWS PrivateLink。如需詳細資訊，請參閱 Amazon Translate 和界面 VPC 端點 (AWS PrivateLink) 。	2021 年 11 月 24 日
平行資料更新	您現在可以使用 Amazon Translate 支援的任何語言來建立平行資料資源。您不再需要使用英文做為其中一種語言。如需平行資料的詳細資訊，請參閱 使用平行資料自訂翻譯 (主動自訂翻譯) 。	2021 年 11 月 15 日

自訂術語方向性	您現在可以建立多方向術語，其中任何語言可以是來源語言或目標語言。如需詳細資訊，請參閱 建立自訂術語 。	2021 年 11 月 11 日
新的語言	Amazon Translate 現在支援下列語言：愛爾蘭文、馬拉地文、葡萄牙文（葡萄牙文）和旁遮普文。如需 Amazon Translate 支援的所有語言，請參閱 支援的語言和語言代碼 。	2021 年 11 月 10 日
新的自訂加密設定	您現在可以使用自己管理的客戶受管金鑰來加密翻譯輸出 AWS Key Management Service。如需詳細資訊，請參閱 執行批次翻譯任務	2021 年 11 月 5 日
支援新的檔案格式	Amazon Translate 現在支援 XML Localization Interchange File Format (XLIFF) 檔案，以進行非同步批次處理。如需所有支援的格式，請參閱 支援的檔案格式 。	2021 年 6 月 9 日
EventBridge 整合	Amazon Translate 現在會將事件傳送至 Amazon EventBridge，以通知您影響翻譯任務和平行資料資源的變更。如需詳細資訊，請參閱 使用 Amazon EventBridge 監控 Amazon Translate 事件 EventBridge 。	2021 年 6 月 4 日

新配額

Amazon Translate 現在支援最多 1000 個排入佇列的批次翻譯任務。如需所有 Amazon Translate 配額，請參閱[指導方針和限制](#)。

2021 年 4 月 23 日

配額增加

平行資料輸入檔案的大小上限已從 1 MB 增加到 5 MB。如需所有 Amazon Translate 配額，請參閱[準則和限制](#)。

2021 年 3 月 31 日

新的語言

Amazon Translate 現在支援下列語言：亞美尼亞文、加泰隆尼亞文、波斯文（波斯文）、菲律賓他加祿文、古吉拉特文、海地克里奧爾文、冰島文、卡納達文、哈薩克文、立陶宛文、馬其頓文、馬來拉姆文、馬爾他文、蒙古文、辛哈拉文、特拉古文、烏茲別克文和威爾斯文。如需 Amazon Translate 支援的所有語言，請參閱[支援的語言和語言代碼](#)。

2020 年 11 月 23 日

新功能

您現在可以使用並行資料自訂批次翻譯任務，其中包含來源文字及其翻譯的範例。使用平行資料的任務稱為 Active Custom Translation 任務。在這些任務期間，Amazon Translate 會調整轉譯輸出，以反映平行資料中的範例。如需詳細資訊，請參閱[使用平行資料自訂翻譯（主動自訂翻譯）](#)。

2020 年 11 月 23 日

支援新的檔案格式	Amazon Translate 現在支援下列 Office Open XML 檔案格式做為非同步批次處理的輸入：Word 文件 (.docx)、PowerPoint 簡報 (.pptx)、Excel 工作手冊 (.xlsx)。如需詳細資訊，請參閱 啟動批次翻譯任務 。	2020 年 7 月 29 日
新的語言	Amazon Translate 現在支援西班牙文（墨西哥）語言進行翻譯。如需所有支援的語言，請參閱 支援的語言和語言代碼 。	2020 年 4 月 30 日
新區域	Amazon Translate 支援歐洲（倫敦）區域中的非同步批次處理。如需可使用非同步批次處理的所有 AWS 區域，請參閱 區域可用性 。	2020 年 4 月 20 日
新功能	Amazon Translate 新增非同步批次翻譯功能。如需詳細資訊，請參閱 非同步批次處理 。	2019 年 12 月 23 日
新區域	Amazon Translate 新增對亞太區域（香港）、亞太區域（雪梨）、歐洲（倫敦）、歐洲（巴黎）、歐洲（斯德哥爾摩）和美國西部（加利佛尼亞北部）區域的支援。如需 Amazon Translate 支援之 AWS 區域的完整清單，請參閱《Amazon Web Services 一般參考》中的 AWS 區域表 或 AWS 區域和端點 。	2019 年 11 月 25 日

新的語言

Amazon Translate 新增了翻譯的新語言：Afrikaan、Albanian、Amharic、Azerbaijani、Bengali、Bosnian、Bulgarian、Canadian-French、Croatian、Dari、Estonian、Georgian、Hausa、Latvian、Pashto、Serbian、Slovak、Slovenian、Somali、Swahili、Tagalog 和 Tamil。如需 Amazon Translate 可以直接翻譯的語言組合清單，請參閱[支援的語言](#)。

2019 年 11 月 25 日

新的語言

Amazon Translate 新增了翻譯的新語言：希臘文、匈牙利文、羅馬尼亞文、泰文、烏克蘭文、烏爾都文和越南文。如需 Amazon Translate 可以直接翻譯的語言組合清單，請參閱[支援的語言](#)。

2019 年 10 月 3 日

新功能

Amazon Translate 新增 [FedRAMP 合規](#)。如需詳細資訊，請參閱[合規](#)。

2019 年 7 月 31 日

新功能

Amazon Translate 新增 [SOC 合規](#)。如需詳細資訊，請參閱[合規](#)。

2019 年 5 月 30 日

新區域

Amazon Translate 新增對亞太區域（孟買）、亞太區域（新加坡）、亞太區域（東京）和加拿大（中部）區域的支援。如需 Amazon Translate 支援之 AWS 區域的完整清單，請參閱《Amazon Web Services 一般參考》中的 [AWS 區域表](#) 或 [AWS 區域和端點](#)。

2019 年 5 月 8 日

新的語言

Amazon Translate 新增了翻譯的新語言：印地文、馬來文、挪威文和波斯文。如需 Amazon Translate 可以直接翻譯的語言組合清單，請參閱 [支援的語言](#)。

2019 年 5 月 6 日

新區域

Amazon Translate 新增對歐洲（法蘭克福）和亞太區域（首爾）區域的支援。如需 Amazon Translate 支援之 AWS 區域的完整清單，請參閱《Amazon Web Services 一般參考》中的 [AWS 區域表](#) 或 [AWS 區域和端點](#)。

2019 年 2 月 28 日

新功能

Amazon Translate 新增 [PCI 合規](#)。如需詳細資訊，請參閱 [合規](#)。

2018 年 12 月 12 日

[新功能](#)

Amazon Translate 新增了四個新的 APIs 和自訂術語功能，可讓您更充分地控制翻譯。在翻譯請求中使用自訂術語，可讓您確保您的品牌名稱、角色名稱、模型名稱和其他唯一內容每一次都會完全依照您的想法進行翻譯，不受標準翻譯或其上下文影響。如需詳細資訊，請參閱[自訂術語](#)。

2018 年 11 月 27 日

[新的語言](#)

Amazon Translate 現在以下列語言翻譯文件：丹麥文、荷蘭文、芬蘭文、希伯來文、印尼文、韓文、波蘭文和瑞典文。Amazon Translate 會大幅減少不支援的語言對數量，進而持續改善直接翻譯。如需 Amazon Translate 可以直接翻譯的語言組合，請參閱[支援的語言](#)。

2018 年 11 月 20 日

[新功能](#)

Amazon Translate 會在英文以外的支援語言之間新增直接轉譯。如需 Amazon Translate 可以直接翻譯的語言組合，請參閱[支援的語言](#)。

2018 年 10 月 29 日

[新功能](#)

Amazon Translate 新增 [HIPAA](#) 合規。如需詳細資訊，請參閱[合規](#)。

2018 年 10 月 25 日

[新功能](#)

Amazon Translate 新增多種新語言進行翻譯：中文（傳統）、捷克文、義大利文、日文、俄文和土耳其文。如需 Amazon Translate 支援的語言清單，請參閱[支援的語言](#)。

2018 年 7 月 17 日

新功能

Amazon Translate 新增對自動來源語言偵測的支援。如需詳細資訊，請參閱 [Amazon Translate 的運作方式](#)。

2018 年 4 月 4 日

新的指南

這是 Amazon Translate 開發人員指南的第一個版本。

2017 年 11 月 29 日

API 參考

Amazon Translate API 參考現在是單獨的文件。如需詳細資訊，請參閱 [Amazon Translate API 參考](#)。

AWS 詞彙表

如需最新的 AWS 術語，請參閱 AWS 詞彙表 參考中的[AWS 詞彙表](#)。

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。