



開發人員指南

# Amazon Translate



# Amazon Translate: 開發人員指南

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

Amazon 的商標和商業外觀不得用於任何非 Amazon 的產品或服務，也不能以任何可能造成客戶混淆、任何貶低或使 Amazon 名譽受損的方式使用 Amazon 的商標和商業外觀。所有其他非 Amazon 擁有的商標均為其各自擁有者的財產，這些擁有者可能附屬於 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
首次使用 .....	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
輸入內容的支援格式 .....	8
自訂您的翻譯 .....	8
自動語言檢測 .....	9
例外狀況處理 .....	9
設定 .....	10
註冊 AWS 帳戶 .....	10
建立管理使用者 .....	10
設定 AWS CLI .....	11
授予程式設計存取權 .....	12
使用 AWS SDK .....	13
開始使用 .....	15
開始使用 (主控台) .....	15
開始使用 (AWS CLI) .....	17
使用指令行 Translate 文字 .....	17
下一步驟 .....	17
開始使用 (SDK) .....	17
使用適用於 Java 的 SDK .....	18
使用適用於 Python 的 AWS 開發套件 .....	18
其他 SDK 範例 .....	18
翻譯處理模式 .....	19
即時翻譯 .....	19
即時翻譯 (主控台) .....	19
即時翻譯 .....	23
非同步批次處理 .....	26
區域可用性 .....	26

先決條件 .....	27
執行工作 .....	32
監控和分析 .....	38
獲得結果 .....	40
自訂您的翻譯 .....	42
使用 do-not-translate 標籤 .....	42
搭 do-not-translate 配主控台使用 .....	42
do-not-translate 與 API 搭配使用 .....	43
使用自訂術語自訂 .....	43
建立自訂術語 .....	44
使用自訂術語 .....	46
使用開發套件的範 Python .....	47
加密您的術語 .....	49
最佳實務 .....	50
使用簡潔 .....	51
使用簡潔度設定 .....	51
支援的語言 .....	51
遮蓋褻瀆 .....	52
使用褻瀆設定 .....	52
不支援語言 .....	53
設定形式 .....	53
使用形式設定 .....	54
支援的語言 .....	54
使用 parallel 資料自訂 .....	55
區域可用性 .....	56
Amazon Translate 平行數據輸入文件 .....	57
新增 parallel 資料 .....	62
檢視和管理 parallel 資料 .....	65
程式碼範例 .....	69
動作 .....	69
描述翻譯工作 .....	70
列出翻譯工作 .....	73
開始翻譯工作 .....	76
停止翻譯工作 .....	80
Translate 文字 .....	83
案例 .....	87

開始使用翻譯工作 .....	87
跨服務範例 .....	90
建置 Amazon Transcribe 串流應用程式 .....	90
建立 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
標記 .....	100
標記新資源 .....	100
檢視、編輯和刪除標籤 .....	102
安全 .....	104
資料保護 .....	104
靜態加密 .....	105
傳輸中加密 .....	106
身分和存取權管理 .....	106
物件 .....	107
使用身分驗證 .....	107
使用政策管理存取權 .....	110
Amazon Translate 如何與 IAM 搭配使用 .....	112
身分型政策範例 .....	118
AWS 受管政策 .....	124
故障診斷 .....	126
監控 .....	128
使用監控 CloudWatch .....	131
使用記錄 Amazon Translate API 呼叫 AWS CloudTrail .....	132
CloudWatch Amazon Translate 的指標和維度 .....	135
使用監控 EventBridge .....	136
法規遵循驗證 .....	139
恢復能力 .....	140
基礎架構安全 .....	140
VPC 端點 (AWS PrivateLink) .....	140
亞馬遜 VPC 端點翻譯的注意事項 .....	141
為 Amazon Translate 創建接口 VPC 端點 .....	141
為 Amazon Translate 建立 VPC 端點政策 .....	141
指引及配額 .....	143
支援的 AWS 區域 .....	143
合規 .....	143

---

限流 .....	143
指導方針 .....	143
Service Quotas .....	143
文件歷史記錄 .....	147
API 參考 .....	157
AWS 詞彙表 .....	158
.....	clix

# 什麼是 Amazon Translate ？

Amazon Translate 是一種文字翻譯服務，使用先進的機器學習技術，隨需提供高品質的翻譯。您可以使用 Amazon Translate 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 支援的語言的相關資訊，請參閱。

## 主題

- [使用案例](#)
- [您是 Amazon Translate 的首次使用者嗎？](#)
- [Amazon Translate 定價](#)
- [Amazon Translate API 參考](#)

## 使用案例

使用 Amazon Translate 來執行以下操作：

透過整合 Amazon Translate，在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。
- Translate 人際通訊，例如電子郵件、遊戲內聊天、客戶服務聊天等，以便客戶和員工能夠以自己偏好的語言進行聯繫。

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，以啟用與語言無關的處理：

- 從非結構化文字 (例如使用 [Amazon Comprehend](#) 的社交媒體串流) 擷取具名實體、情緒和關鍵片語。
- 使用 [Amazon 轉錄](#) 功能提供多種語言的字幕和即時字幕。
- 說翻譯的內容與 [Amazon Polly](#)。
- Translate 存放在 [Amazon S3](#) 中的文件儲存庫。
- Translate 存放在以下資料庫中的文字：[Amazon DynamoDB](#)、[Amazon Aurora](#) 和 [Amazon Redshift](#)。
- 無縫整合工作流程與 [AWS Lambda](#) 或 [AWS Glue](#)。

## 您是 Amazon Translate 的首次使用者嗎？

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

1. [Amazon Translate 如何工作](#)— 介紹 Amazon Translate。
2. [開始使用 Amazon Translate](#)— 說明如何設定 AWS 帳戶並開始使用 Amazon Translate。
3. [使用 AWS 開發套件的 Amazon Translate 程式碼範例](#)— 使用程式碼範例探索 Amazon Translate API。

您也可以使用下列資源來了解 Amazon Translate 服務：

- M [AWS Machine Learning 部落格](#) 包含有關 Amazon Translate 的實用文章。
- [Amazon Translate 深入了解視頻系列](#) 提供有關 Amazon Translate 的介紹視頻。

## Amazon Translate 定價

與其他 AWS 產品一樣，使用 Amazon Translate 沒有合約或最低承諾。如需使用 Amazon Translate 譯費用的詳細資訊，請參閱 [Amazon Translate 定價](#)。

## Amazon Translate API 參考

Amazon Translate API 參考現在是一個單獨的文檔。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) 和目標語言 (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。如果將來源語言設定為 auto，則可能需要支付使用 auto 偵測的費用。如需詳細資訊，請參閱 [自動語言檢測](#)。

### 主題

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

## 支援的語言

Amazon 翻譯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，表格遵循 [RFC 5646](#) 格式附加破折號，後面接著 [ISO 3166 2](#) 位數國家/地區代碼。例如，西班牙文墨西哥變體的語言代碼為 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
亞美尼亞人	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	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
波斯文 (波斯文)	fa
菲律賓語, 他加	tl
芬蘭文	fi
French	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	gu
海地克里奥尔语	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
愛爾蘭人	ga
義大利文	it
日文	ja
坎那達	kn
哈薩克人	kk
韓文	ko
拉脫維亞文	lv
立陶宛	lt
馬其頓語	mk
馬來文	ms
馬拉雅拉姆	ml
馬爾濟斯	mt

語言	語言代碼
馬拉地語	mr
Mongolian	mn
挪威文 (巴克摩尔)	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	pa
羅馬尼亞文	ro
俄文	ru
塞爾維亞文	sr
僧伽羅語	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	uz
越南文	vi
威爾斯文	cy

## Amazon Translate 功能支援的語言

以下各節說明 Amazon Translate 功能支援的語言。

- 實時文檔翻譯 — 支持從英語翻譯到任何支持的語言，從任何支持的語言翻譯為英語。如需即時翻譯的詳細資訊，請參閱[即時翻譯](#)。
- 簡潔 — 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 中掩蓋褻瀆的單詞和短語](#)。
- 形式性 — 如需此功能支援的語言，請參閱[在 Amazon Translate 中設置形式](#)。

# Amazon Translate 如何工作

使用 Amazon Translate 服務將內容從來源語言 (輸入內容的語言) 翻譯成目標語言 (您為翻譯輸出選取的語言)。在批次工作中，您可以將檔案從一或多個來源語言翻譯成一或多種目標語言。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

- [輸入內容的支援格式](#)
- [自訂您的翻譯](#)
- [自動語言檢測](#)
- [例外狀況處理](#)

## 輸入內容的支援格式

Amazon Translate，支持以下格式的輸入內容：

- 即時翻譯：
  - 輸入文字 — UTF-8 格式的純文字。Amazon Translate 提供輸出內容為 UTF-8 文本。
  - 一個輸入檔案 — 包含純文字 (.txt)、HTML (.html) 或字 (.docx) 內容的檔案。Amazon Translate 以檔案的形式提供輸出內容，其格式與輸入檔案相同。
- 對於批次轉譯工作：
  - 輸入檔案的集合 — 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)、字 (.docx)、Excel (.xlsx)、(.pptx) 和 XLIFF 1.2 PowerPoint (.xlf)。Amazon Translate 以文件形式提供輸出內容。每個輸出檔案的檔案格式與輸入檔案格式相符。

## 自訂您的翻譯

您可以使用下列功能自訂使用 Amazon 翻譯產生的翻譯：

- 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 (在 HTML 內容中)。
- 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。
- 簡潔 — 減少大多數翻譯的翻譯輸出長度 (與翻譯輸出相比而不簡潔)。實時文本翻譯支持簡潔。
- 褻瀆 — 遮罩翻譯輸出中的褻瀆字詞和片語。

- 形式性 — 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、色調和字詞選擇。

如需詳細資訊，請參閱[自訂翻譯](#)。

## 自動語言檢測

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，請指定auto為來源語言。Amazon Translate 會代表您呼叫 Amazon Comprehend，以確定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。如需有關亞馬遜定價的資訊，請參閱 [Amazon Comprehend 定價](#)。

## 例外狀況處理

如果您指定的來源或目標語言不受支援，Amazon Translate 會傳回下列例外狀況：

- `UnsupportedLanguagePairException`— Amazon Translate 支持所有支持的語言之間的翻譯。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的可信度較低，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，請先完成以下任務。

## 設定工作

- [註冊 AWS 帳戶](#)
- [建立管理使用者](#)
- [安裝和配置AWS Command Line Interface \( AWS CLI \)](#)
- [授予程式設計存取權](#)
- [透過 AWS SDK 使用此服務](#)

## 註冊 AWS 帳戶

如果您還沒有 AWS 帳戶，請完成以下步驟建立新帳戶。

### 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，並在電話鍵盤輸入驗證碼。

註冊 AWS 帳戶時，會建立 AWS 帳戶根使用者。根使用者有權存取該帳戶中的所有 AWS 服務和資源。作為最佳安全實務，[將管理存取權指派給管理使用者](#)，並且僅使用根使用者來執行[需要根使用者存取權的任務](#)。

註冊程序完成後，AWS 會傳送一封確認電子郵件給您。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。

## 建立管理使用者

註冊後，請保護AWS 帳戶AWS 帳戶根使用者、啟用和建立系統管理使用者AWS IAM Identity Center，這樣您就不會將 root 使用者用於日常工作。



## 保護您的 AWS 帳戶根使用者

1. 選擇根使用者 並輸入您的 AWS 帳戶電子郵件地址，以帳戶擁有者身分登入 [AWS Management Console](#)。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)。

如需指示，請參閱《IAM 使用者指南》中的[為 AWS 帳戶根使用者啟用虛擬 MFA 裝置 \(主控台\)](#)。

## 建立管理使用者

1. 啟用 IAM 身分識別中心。

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

2. 在 IAM 身分中心中，將管理存取權授與管理使用者。

[若要取得有關使用IAM Identity Center 目錄做為身分識別來源的自學課程，請參閱《使用指南》IAM Identity Center 目錄中的「以預設值設定使用AWS IAM Identity Center者存取」。](#)

## 以管理員的身分登入

- 若要使用您的 IAM 身分中心使用者登入，請使用建立 IAM 身分中心使用者時傳送至您電子郵件地址的登入 URL。

如需有關如何使用 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](#)。

## 安裝和配置AWS Command Line Interface ( AWS CLI )

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

### 安裝及設定 AWS CLI

1. 安裝AWS CLI。如需指示，請參閱AWS Command Line Interface使用指南中的下列主題：

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

2. 設定 AWS CLI。如需指示，請參閱AWS Command Line Interface使用指南中的下列主題：

## 設定 AWS Command Line Interface

### 授予程式設計存取權

若使用者想要與 AWS Management Console 之外的 AWS 互動，則需要程式設計存取權。授予程式設計存取權的方式取決於存取 AWS 的使用者類型。

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	By
人力身分 (IAM Identity Center 中管理的使用者)	使用臨時憑證簽署對 AWS CLI、AWS SDKs 或 AWS APIs 的程式設計請求。	請依照您要使用的介面所提供的指示操作。 <ul style="list-style-type: none"> <li>關於 AWS CLI，請參閱 AWS Command Line Interface 使用者指南 中的 <a href="#">設定 AWS CLI 來使用 AWS IAM Identity Center</a>。</li> <li>關於 AWS SDKs、工具和 AWS APIs，請參閱 AWSSDKs 和工具參考指南 中的 <a href="#">IAM Identity Center 驗證</a>。</li> </ul>
IAM	使用臨時憑證簽署對 AWS CLI、AWS SDKs 或 AWS APIs 的程式設計請求。	請遵循 IAM 使用者指南 中 <a href="#">使用臨時憑證搭配 AWS 資源</a> 中的指示。
IAM	(不建議使用) 使用長期憑證簽署 AWS CLI、AWS SDKs 或 AWS APIs 的程式設計要求。	請依照您要使用的介面所提供的指示操作。 <ul style="list-style-type: none"> <li>關於 AWS CLI，請參閱 AWS Command Line Interface 使用者指南 中的 <a href="#">使</a></li> </ul>

哪個使用者需要程式設計存取權？	到	By
		<p><a href="#">用 IAM 使用者憑證進行驗證</a></p> <ul style="list-style-type: none"> <li>• 關於 AWS SDKs 和工具，請參閱 <a href="#">AWSSDKs 和工具參考指南</a> 中的 <a href="#">使用長期憑證進行驗證</a>。</li> <li>• 關於 AWS API，請參閱 <a href="#">IAM 使用者指南</a> 中的 <a href="#">管理 IAM 使用者的存取金鑰</a>。</li> </ul>

## 透過 AWS SDK 使用此服務

AWS 軟體開發套件 (SDK) 適用於許多常用的程式設計語言。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	程式碼範例
<a href="#">AWS SDK for C++</a>	<a href="#">AWS SDK for C++ 程式碼範例</a>
<a href="#">AWS SDK for Go</a>	<a href="#">AWS SDK for Go 程式碼範例</a>
<a href="#">AWS SDK for Java</a>	<a href="#">AWS SDK for Java 程式碼範例</a>
<a href="#">AWS SDK for JavaScript</a>	<a href="#">AWS SDK for JavaScript 程式碼範例</a>
<a href="#">適用於 Kotlin 的 AWS SDK</a>	<a href="#">適用於 Kotlin 的 AWS SDK 程式碼範例</a>
<a href="#">AWS SDK for .NET</a>	<a href="#">AWS SDK for .NET 程式碼範例</a>
<a href="#">AWS SDK for PHP</a>	<a href="#">AWS SDK for PHP 程式碼範例</a>
<a href="#">AWS SDK for Python (Boto3)</a>	<a href="#">AWS SDK for Python (Boto3) 程式碼範例</a>
<a href="#">AWS SDK for Ruby</a>	<a href="#">AWS SDK for Ruby 程式碼範例</a>

SDK 文件	程式碼範例
<a href="#">適用於 Rust 的 AWS SDK</a>	<a href="#">適用於 Rust 的 AWS SDK 程式碼範例</a>
<a href="#">適用於 SAP ABAP 的 AWS SDK</a>	<a href="#">適用於 SAP ABAP 的 AWS SDK 程式碼範例</a>
<a href="#">適用於 Swift 的 AWS SDK</a>	<a href="#">適用於 Swift 的 AWS SDK 程式碼範例</a>

#### 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。

# 開始使用 Amazon Translate

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

## 主題

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

## 開始使用 (主控台)

開始使用 Amazon Translate 最簡單的方法是使用 Console 翻譯。您可以使用控制台翻譯多達 10,000 個字節的文本。如果您還沒有檢閱[Amazon Translate 如何工作](#)中的概念和術語，我們建議您檢閱之後再繼續。

打開 [Amazon Translate 控制台](#)。

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

在即時翻譯中，選擇目標語言。Amazon Translate 會自動偵測來源語言，或者您也可以選擇來源語言。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](#)作業的 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)

在以下練習中，您會使用 AWS 命令列界面 (AWS CLI) 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[安裝和配置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](#)。

### 使用指令行 Translate 文字

以下範例顯示如何從命令列使用 [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

要查看其他使用 Amazon Translate 的方法，請參閱[使用 AWS 開發套件的 Amazon Translate 程式碼範例](#)。

## 開始使用 (SDK)

AWS 為各種計算機語言提供 SDK。SDK 會管理用戶端的許多 API 連線詳細資料，例如簽名計算、要求重試處理和錯誤處理。如需詳細資訊，請參閱 [AWS 開發套件](#)。

以下實例演示瞭如何使用 Java 和 Python 使用 Amazon Translate [TranslateText](#) 操作。使用開發套件來了解 Amazon Translate API，以及做為您自己應用程式的建置區塊。

## 主題

- [使用翻譯文字 AWS SDK for Java](#)
- [使用翻譯文字 AWS SDK for Python \(Boto\)](#)
- [其他 SDK 範例](#)

## 使用翻譯文字 AWS SDK for Java

AWS 提供了 [GitHub](#) 如何在 Java 中使用 [TranslateText](#) 操作的示例。若要執行這個範例，您需要 AWS SDK for Java。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 AWS SDK for Python (Boto)

下面的例子演示了如何在 Python 中使用該 [TranslateText](#) 操作。若要執行此範例 Python 請透過 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 [使用 AWS 開發套件的 Amazon Translate 程式碼範例](#) 的範例，請參閱。



# 翻譯處理模式

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](#) — 您提出同步要求翻譯少量文字 (或文字檔)，Amazon Translate 會立即回應翻譯的文字。
- [非同步批次處理](#) — 您將一組文件放在 Amazon Simple Storage Service (Amazon S3) 位置，然後啟動非同步處理任務以進行翻譯。Amazon Translate 會將翻譯的輸出文件傳送到指定的 Amazon S3 位置。

## 即時翻譯

Amazon Translate 提供即時文件和文字翻譯作業，可立即傳回翻譯。您可以使用主控台或 API 執行即時翻譯。

### 主題

- [使用主控台進行即時翻譯](#)
- [使用 API 進行即時翻譯](#)

## 使用主控台進行即時翻譯

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] 文字方塊中，或將輸入文字做為檔案提供。或者，您可以設定所需的形式層級、粗俗遮罩、簡潔性和自訂術語等功能。

您可以將 auto 語言偵測與即時翻譯搭配使用，但可能會產生費用。如需詳細資訊，請參閱[自動語言檢測](#)。

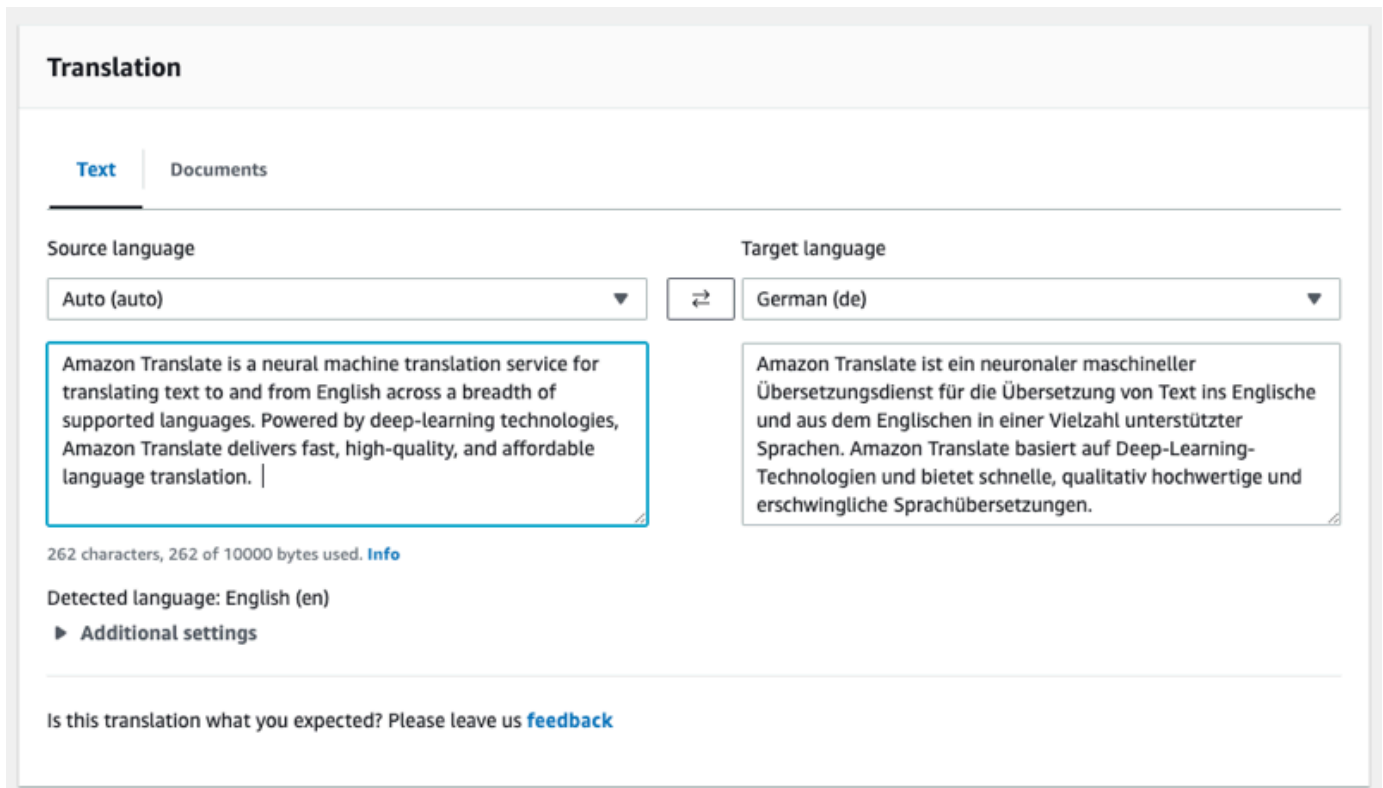
### 主題

- [Translate 文字](#)
- [Translate 文件](#)
- [檢視等效的 API 要求和回應資料](#)
- [使用翻譯功能](#)

## Translate 文字

使用 Amazon Translate 主控台可翻譯高達 10,000 位元組的文字。

1. 打開 [Amazon Translate 控制台](#)。
2. 在左側導覽功能表中，選擇 [即時翻譯]。
3. 對於來源語言，請選取來源文字的語言，或將值保持為 auto 以進行自動偵測。
4. 在「目標語言」中，選取一種語言。
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」文字方塊中顯示已翻譯的文字。



## Translate 文件

即時文件翻譯支援從英文翻譯成任何支援的語言，以及從任何支援的語言翻譯成英文。

若要使用 Amazon 翻譯主控台翻譯文件：

1. 打開 [Amazon Translate 控制台](#)。
2. 在左側導覽功能表中，選擇 [即時翻譯]。
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](#)

- 對於來源語言，請選取來源文字的語言，或選取 auto 以進行自動偵測。
- 在「目標語言」中，選取一種語言。如果來源語言不是英文，您必須為目標語言選取英文。
- 在「上傳檔案」下，選擇「選擇檔案」並輸入來源檔案的路徑。檔案大小上限為 100 KB。
- 在「文件類型」中，選取翻譯來源檔案的格式。文檔翻譯支持純文本，HTML 或 Word (.docx) 輸入文件。
- 選擇「Translate」。

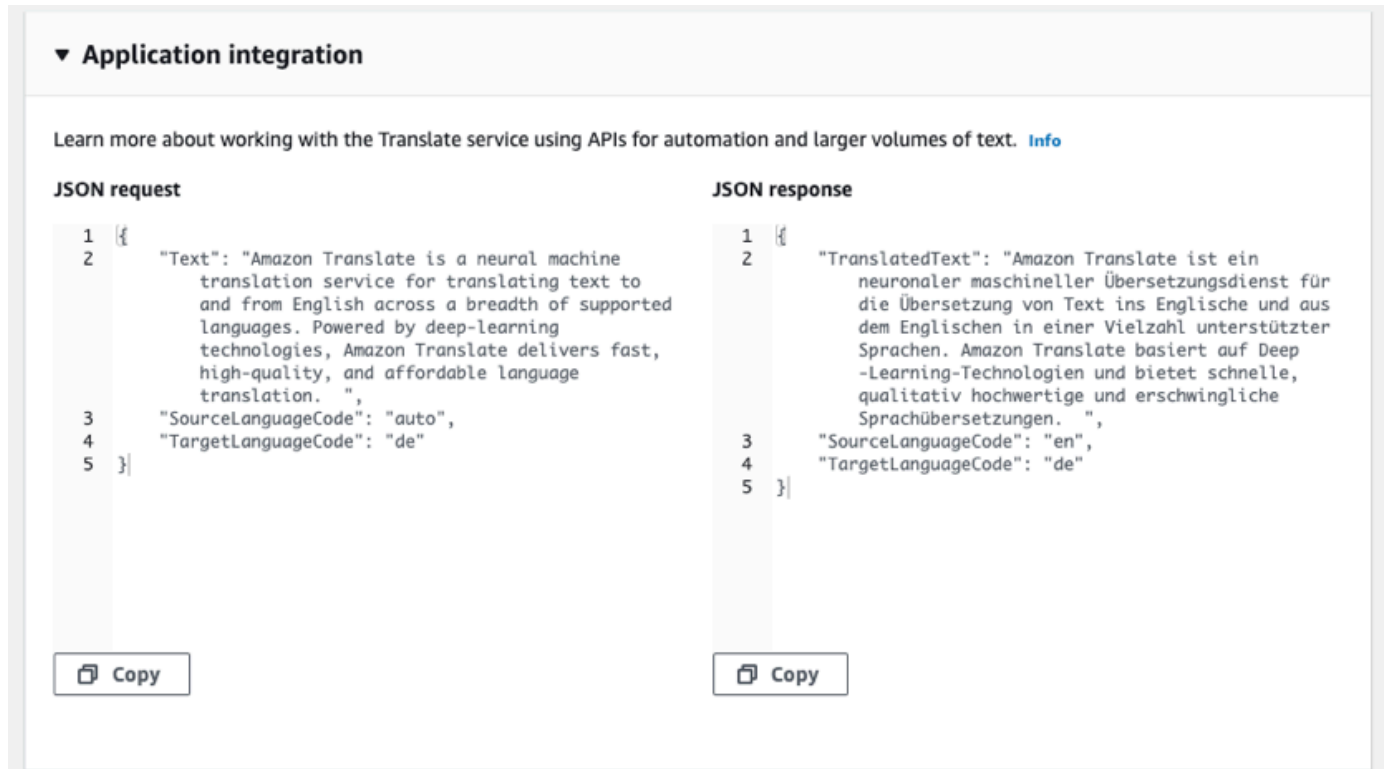
翻譯工作完成後，請選擇 [下載翻譯]，將翻譯的文件下載至您的本機硬碟。已翻譯文件的格式 (文字、HTML 或 Word) 與輸入文件相符。

## 檢視等效的 API 要求和回應資料

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

- 在「翻譯」面板下，展開「應用程式整合」面板。

主控台會以 JSON 格式顯示對等的翻譯要求資料。



The screenshot displays the 'Application integration' section of the Amazon Translate console. It features a heading 'Application integration' and a sub-heading 'Learn more about working with the Translate service using APIs for automation and larger volumes of text. Info'. Below this, there are two columns: 'JSON request' and 'JSON response'. The 'JSON request' column shows a JSON object with 'Text', 'SourceLanguageCode', and 'TargetLanguageCode' fields. The 'JSON response' column shows a JSON object with 'TranslatedText', 'SourceLanguageCode', and 'TargetLanguageCode' fields. Both columns have a 'Copy' button below them.

```
1 {
2   "Text": "Amazon Translate is a neural machine
3     translation service for translating text to
4     and from English across a breadth of supported
5     languages. Powered by deep-learning
6     technologies, Amazon Translate delivers fast,
7     high-quality, and affordable language
8     translation. ",
9   "SourceLanguageCode": "auto",
10  "TargetLanguageCode": "de"
11 }
```

```
1 {
2   "TranslatedText": "Amazon Translate ist ein
3     neuronaler maschineller Übersetzungsdienst für
4     die Übersetzung von Text ins Englische und aus
5     dem Englischen in einer Vielzahl unterstützter
6     Sprachen. Amazon Translate basiert auf Deep
7     -Learning-Technologien und bietet schnelle,
8     qualitativ hochwertige und erschwingliche
9     Sprachübersetzungen. ",
10  "SourceLanguageCode": "en",
11  "TargetLanguageCode": "de"
12 }
```

2. 您可以複製 JSON 要求以在 [TranslateText](#) 或 [TranslateDocument](#) API 作業中使用。
3. JSON 回應面板中的 JSON 輸出符合 API 所產生的輸出。

## 使用翻譯功能

若要搭配 Amazon 翻譯主控台使用翻譯功能：

1. 打開 [Amazon Translate 控制台](#)。
2. 在左側導覽功能表中，選擇 [即時翻譯]。
3. 如先前程序所述，提供來源語言、目標語言和輸入資料 (文字或文件)。
4. 在 [其他設定] 底下，您可以選擇使用下列設定來自訂翻譯工作的輸出：

### 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翻譯。

如需詳細資訊，請參閱 [使用自訂術語自訂翻譯](#)。

## 簡潔

減少大多數翻譯的翻譯輸出長度（與翻譯輸出相比，沒有簡潔性）。Amazon Translate 支持翻譯文本的簡潔性，但不支持翻譯文檔。為了簡潔起見，如果來源語言和目標語言形成不支援的語言配對，Amazon Translate 會忽略簡潔設定。

如需有關支援語言的資訊，請參閱[在 Amazon Translate 中使用簡潔](#)。

## 褻瀆

在翻譯輸出中掩蓋褻瀆的單詞和短語。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 中掩蓋褻瀆的單詞和短語](#)。

## 形式

對於某些目標語言，您可以將「形式化」設定為正式或非正式。如果形式不支援目標語言，Amazon Translate 會忽略形式設定。

如需詳細資訊，請參閱[在 Amazon Translate 中設置形式](#)。

5. 對於文件 Translate，請選擇「翻譯」，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文字。

## 使用 API 進行即時翻譯

Amazon Translate 提供下列即時翻譯操作來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）。

這些同步作業會將翻譯結果直接傳回至您的應用程式。如果您在這些操作中使用 auto 語言偵測，可能會產生費用。如需詳細資訊，請參閱[自動語言檢測](#)。

## Translate 文字

使用此[TranslateText](#)作業翻譯單一文字區塊。

## 使用指令行 Translate 文字

下面的例子演示了如何從命令行使用 [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"
```

此命令會以下列 JSON 回應：

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

## 使用 JSON 檔案 Translate 文字

此範例顯示如何使用 JSON 檔案翻譯較長的文字區塊。您可以在命令列上指定來源語言和目標語言，或在 JSON 檔案中指定它們。

### Note

範例 JSON 檔案已格式化以提高可讀性。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 AWS CLI 中執行以下命令：

```
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"
}
```

## Translate 文件

使用此 [TranslateDocument](#) 作業翻譯文字、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成任何支援的語言，以及從任何支援的語言翻譯成英文。您可以指定源語言代碼或使用 auto 檢測。

使用命令列 Translate 文件

下面的例子演示了如何從命令行使用 [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
```

此命令會以下列 JSON 回應：

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

## Amazon Translate 進行異步批次處理

若要翻譯大型文件集合 (最多 5 GB 的大小)，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，例如社交媒體張貼或使用者的評論，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，向 Amazon Translate 提供 IAM 角色，該角色對輸入 Amazon S3 資料夾及其所有子資料夾具有讀取權限。該角色還必須具有對輸出 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. 從指定的輸出值區擷取批次轉譯工作的結果。

## 區域可用性

下列 AWS 區域支援批次翻譯：

- 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)
- 亞太區域 (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

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

## 批次翻譯工作的先決條件

必須符合下列先決條件，Amazon Translate 才能成功執行批次翻譯任務：

- 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。
- 每個輸入文件必須少於 20 MB，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。如果您將輸入檔案新增至儲存貯體的最上層，Amazon Translate 會在您嘗試執行批次翻譯任務時擲回錯誤。此要求適用於輸入文件。輸出文件不需要任何文件夾，Amazon Translate 可以將它們放在 Amazon S3 存儲桶的頂層。

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

## 支援的檔案格式

Amazon Translate 支援批次翻譯任務的檔案類型如下：

- 純文字。
- HTML。
- 字文檔 ( .docx )。
- PowerPoint 簡報檔案 (.pptx)。
- Excel 工作簿文件 ( .xlsx )。
- XML 本地化交換檔案格式 (XLIFF) 檔案 (.xlf)。Amazon Translate 只支援 XLIFF 1.2 版。

Amazon Translate per 要求檔案經過 UTF-8 編碼。

## 必要權限

您的AWS帳戶必須在 IAM 中具有服務角色，才能執行批次翻譯工作。此角色必須具有授予 Amazon Translate 的許可政策：

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

它還必須包含信任政策，該政策可允許 Amazon Translate per Guide。此信任原則必須允許translate.amazonaws.com服務主體執行sts:AssumeRole動作。

使用 Amazon Translate 翻譯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 side API 執行批次翻譯任務時，請務必提供要求的 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](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 Example Manarper 政策授權存取 Amazon S3 儲存貯體的 Example Manarper。它授權讀取和寫入存取權。

```
{
  "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 擔任該政策所屬的 IAM 角色。

我們建議您驗證使用信任原則的AWS帳戶，以減輕[混淆的副](#)問題。此範例使用aws:SourceArn和aws:SourceAccount條件鍵來驗證來源帳戶。輸入提交批次換算工作的AWS科目。

```
{
  "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權限

您可以在 Amazon Translate 中自訂加密設定，但首先必須將許可新增至 IAM 中的服務角色。

Amazon Translate 翻譯會加密您在執行批次翻譯任務時產生的翻譯輸出。默認情況下，它使用 AWS 受管金鑰。這種類型的密鑰由創建AWS並存儲在您的帳戶AWS Key Management Service ( AWS KMS ) 中。不過，您無法自行管理此 KMS 金鑰。它僅由代表您管理和使用AWS。

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

您的金鑰必須有一個金鑰政策，讓 Amazon Translate per Guide。金鑰政策透過授予允許 Amazon Translate 存取 Amazon S3 儲存貯體的服務角色的許可來達成此目的。

金鑰原則可讓服務角色執行加密輸出所需的AWS KMS作業，如下列範例原則陳述式所示。

### Example KMS 金鑰政策聲明

```

{
  "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AWS KMS[中的主要政策](#)

使用其他AWS帳戶AWS KMS金鑰的權限

如果您想要使用與使用 Amazon TranslateAWS 帳戶不同的 KMS 金鑰，您必須：

1. 更新 IAM 中 Amazon Translate 的服務角色。
2. 更新中的金鑰原則AWS KMS。

若要更新您的服務角色，請附加政策，以允許其使用其他AWS帳戶中的 KMS 金鑰執行必要AWS KMS 作業，如下列範例所示。

Example IAM 政策：授權存取不同帳戶的 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 金鑰原則，請將服務角色和 admin 使用者新增為允許使用金鑰的主體，如下列原則陳述式範例所示。

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 Developer Guide](#) [Developer Guide](#) [Developer Guide](#) [Developer Guide](#) [Developer Guide](#) [Developer Guide](#) [AWS KMS](#)

## 執行批次翻譯工作

您可以使用 Amazon 翻譯主控台、或 Amazon Translate 譯 API 來執行 AWS CLI 批次 Amazon Translate 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，小型資料集的批次翻譯可能需要幾分鐘的時間，而非常大的資料集可能需要 2 天或更長時間。完成時間也取決於資源的可用性。

## Amazon Translate 控制

若要使用 Amazon Translate 主控台執行翻譯任務，請使用 Batch 翻譯頁面建立任務：

1. 打開 [Amazon Translate 控制台](#)。
2. 在左側的導覽選單中，選擇「Batch 轉譯」。
3. 在 [翻譯工作] 頁面上，選擇 [建立工作]。主控台會顯示 [建立翻譯工作] 頁面。
4. 在 Job 設定之下，執行下列操作：

- a. 在名稱中，輸入批次轉譯工作的自訂名稱。
  - b. 對於「來源語言」，請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 auto 偵測每個檔案的來源語言。
  - c. 針對目標語言，最多可選取 10 種語言。Amazon Translate 會將每個來源檔案翻譯成每種目標語言。
5. 在「輸入資料」下，執行下列操作：
- a. 對於輸入 S3 位置，請指定包含 Amazon S3 中翻譯來源檔案的輸入資料夾。若要透過在 Amazon S3 中導覽至該資料夾來提供資料夾，請選擇「選取資料夾」。
  - b. 在檔案格式中，選取翻譯來源檔案的格式。
6. 在「輸出資料」下，執行下列操作：
- a. 對於輸出 S3 位置，請在 Amazon S3 中指定輸出資料夾，Amazon Translate 放置翻譯輸出的位置。若要透過在 Amazon S3 中導覽至該資料夾來提供資料夾，請選擇「選取資料夾」。
  - b. 如果您要使用您在 () 中管理的客戶管理金鑰來加密輸出，請選擇 [自訂加密設定 AWS Key Management Service (進階AWS KMS)]。

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

如果您要使用目前AWS帳戶中的 KMS 金鑰，請在 [選擇金鑰] 下方選取該金鑰AWS Key Management Service鑰。或者，如果您想要使用其他 AWS 帳戶的 KMS 金鑰，請輸入該金鑰的 Amazon 資源名稱 (ARN)。

#### Note

在您可以使用自己的 KMS 金鑰之前，必須先將許可新增至 IAM 中 Amazon Translate 的服務角色。如果您想要使用其他帳戶的 KMS 金鑰，您還必須更新中的金鑰政策 AWS KMS。如需詳細資訊，請參閱[自訂加密的先決條件權限](#)。

7. 在 [自訂-選用] 底下，您可以選擇使用下列設定來自訂翻譯工作的輸出：

#### 褻瀆

在翻譯輸出中掩蓋褻瀆的單詞和短語。如果您為工作指定多種目標語言，則所有目標語言都必須支援粗俗遮罩。如果任何目標語言不支援粗言遮罩，翻譯工作就不會遮罩任何目標語言的不雅用語。

如需詳細資訊，請參閱[在 Amazon Translate 中掩蓋褻瀆的單詞和短語](#)。

## 簡潔

Amazon Translate 不支援批次翻譯任務的簡潔性。

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「形式化」設定為正式或非正式。如果您為工作指定多種目標語言，翻譯會忽略任何不支援的目標語言的格式設定。

如需詳細資訊，請參閱[在 Amazon Translate 中設置形式](#)。

## 自訂術語

包含範例來源詞彙和每個字詞所需的翻譯。如果您為工作指定多種目標語言，翻譯會針對每個要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

由示範如何翻譯文字區段的範例組成。如果您為工作指定多種目標語言，parallel 資料檔案必須包含所有目標語言的翻譯。

當您將 parallel 資料新增至批次轉譯工作時，您會建立「使用中自訂翻譯」工作。

### Note

使用中自訂翻譯工作的定價比其他不使用 parallel 資料的工作更高。如需詳細資訊，請參閱[Amazon Translate 定價](#)。

如需詳細資訊，請參閱[使用 parallel 資料自訂翻譯 \(使用中自訂翻譯\)](#)。

8. 在存取許可下，向 Amazon Translate 提供 IAM 角色，以授與 Amazon S3 中的輸入和輸出檔案所需許可：
  - 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」。對於 IAM 角色，請選擇輸入和輸出 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 中的角色名稱前綴為 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，並且 KMS 金鑰必須具有允許角色使用金鑰的金鑰政策。

如需更多資訊，請參閱 [自訂加密的先決條件權限](#)

#### 9. 選擇 建立任務。

主控台會返回「翻譯工作」頁面，其中工作建立狀態會顯示在頁面頂端的橫幅中。幾分鐘後，您的工作會顯示在表格中。

#### 10. 在「名稱」資料欄中選擇工作名稱，以開啟工作詳細資訊頁面。

當您執行翻譯工作時，[狀態] 欄位會顯示 [進行中]。

#### 11. 當狀態變為「已完成」時，請選擇「輸出檔案位置」下的連結來移至您的翻譯輸出。主控台會移至 Amazon S3 中的輸出儲存貯體。

#### 12. 若要下載輸出檔案，請選取每個輸出檔案的核取方塊，然後選擇「下載」。

## AWS CLI

若要使用執行轉譯工作AWS CLI，請使用[start-text-translation-job](#)命令，並為parallel-data-names參數指定 parallel 資料資源的名稱。

### Example S tart-text-translation-job 命令

下列範例透過提交存放在 Amazon S3 輸入儲存貯體中的 Excel 檔案來執行翻譯任務。此工作是由包含在要求中的 parallel 資料自訂。

```
$ aws translate start-text-translation-job \  
> --input-data-config ContentType=application/vnd.openxmlformats-officedocument.spreadsheetml.sheet,S3Uri=s3://my-s3-bucket/input/ \  
> --output-data-config S3Uri=s3://my-s3-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 會以任務 ID 和狀態回應：

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

如果要自訂翻譯工作的輸出，可以使用下列參數：

### --settings

設定翻譯輸出的設定，包括下列選項：

打開翻譯輸出的簡潔性。Amazon Translate 不支援批次翻譯任務的簡潔性。如需詳細資訊，請參閱在 [Amazon Translate 中使用簡潔](#)。

啟用褻瀆以遮罩褻瀆的單詞和短語。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 中掩蓋褻瀆的單詞和短語](#)。如果任何目標語言不支援粗言遮罩，翻譯工作就不會遮罩任何目標語言的不雅用語。

設定翻譯輸出中的形式程度。將Formality參數設定為FORMAL或INFORMAL。如果您為工作指定多種目標語言，翻譯會忽略任何不支援的目標語言的格式設定。如需詳細資訊，請參閱在 [Amazon Translate 中設置形式](#)。

### --terminology-names

要新增至翻譯工作的自訂術語資源名稱。此資源列出來源詞彙範例，以及每個字詞所需的翻譯。如果您為工作指定多種目標語言，翻譯會針對每個要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](#) 指令。

如需詳細資訊，請參閱 [使用自訂術語自訂翻譯](#)。

### --parallel-data-names

要新增至翻譯工作的 parallel 資料資源名稱。此資源包含示範如何翻譯文字區段的範例。如果您為工作指定多種目標語言，parallel 資料檔案必須包含所有目標語言的翻譯。

將 parallel 資料新增至翻譯工作時，即建立「使用中自訂翻譯」工作。

此參數只接受一個 parallel 資料資源。

**Note**

使用中自訂翻譯工作的定價比其他不使用 parallel 資料的工作更高。如需詳細資訊，請參閱 [Amazon Translate 定價](#)。

如需可用 parallel 資料資源的清單，請使用 [list-parallel-data](#) 指令。

如需詳細資訊，請參閱 [使用 parallel 資料自訂翻譯 \(使用中自訂翻譯\)](#)。

若要檢查翻譯工作的狀態，請使用 [describe-text-translation-job](#) 指令。

Example D describe-text-translation-job 命令

下列範例會透過提供工作 ID 來检查工作狀態。此 ID 是由 Amazon Translate 在 start-text-translation-job 命令啟動任務時提供的。

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

Amazon Translate 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://my-s3-bucket/input/",
      "ContentType": "application/vnd.openxmlformats-officedocument.spreadsheetml.sheet"
    },
    "OutputDataConfig": {
      "S3Uri": "s3://my-s3-bucket/output/111122223333-TranslateText-4446f95f20c88a4b347449d3671fbe3d/"
    },
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/my-iam-role"
  }
}

```

如果該JobStatus值是IN\_PROGRESS，請等待幾分鐘的時間，然後[describe-text-translation-job](#)後再次運行，直到狀態為止COMPLETED。工作完成時，您可以在下S3Uri欄位所提供的位置下載翻譯結果OutputDataConfig。

## Amazon Translate API

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

## 監控和分析批次翻譯工作

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。若要監視特定工作，請使用 [DescribeTextTranslationJob](#) 作業。您也可以使用此 [ListTextTranslationJobs](#) 作業來擷取帳戶中所有翻譯工作的資訊。若要將結果限制為符合特定條件的工作，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

下列範例使用檢查工作的狀態AWS CLI執行[DescribeTextTranslationJob](#)命令：

```
$ 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/"
  }
}
}

```

當批次翻譯工作的狀態為 `IN_PROGRESS` 且由 [StopTextTranslationJob](#) 作業使用時，您可以停止該工作。

Example `stop-text-translation-job` 命令

下列範例停止批次轉譯：AWS CLI執行[StopTextTranslationJob](#)命令：

```
$ 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

一旦工作的狀態為 `COMPLETED` 或者 `COMPLETED_WITH_ERROR`，您的輸出文件在您指定的 Amazon S3 資料夾中可用。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`。

如果批次翻譯工作的狀態為 `FAILED`，[DescribeTextTranslationJob](#) 作業回應會包含一個 `Message` 欄位，其中會說明工作未順利完成的原因。

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 自定義翻譯

您可以使用下列設定來自訂使用 Amazon 翻譯產生的翻譯：

- **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（在 HTML 內容中）。
- **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。
- **簡潔** — 減少大多數翻譯的翻譯輸出長度（與翻譯輸出相比而不簡潔）。使用簡潔的實時文本翻譯。
- **褻瀆** — 遮罩翻譯輸出中的褻瀆字詞和片語。
- **形式性** — 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、色調和文字選擇。

## 主題

- [do-not-translate 在 Amazon Translate 中使用](#)
- [使用自訂術語自訂翻譯](#)
- [在 Amazon Translate 中使用簡潔](#)
- [在 Amazon Translate 中掩蓋褻瀆的單詞和短語](#)
- [在 Amazon Translate 中設置形式](#)
- [使用 parallel 資料自訂翻譯 \(使用中自訂翻譯\)](#)

## do-not-translate 在 Amazon Translate 中使用

對於 HTML 內容，您可以在不想翻譯的文本中添加 do-not-translate 標記。此功能可用於控制台和 API 操作。

## 主題

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

## 搭 do-not-translate 配主控台使用

在來源 HTML 內容中，`translate="no"` 在包圍您不想翻譯的內容的 HTML 標籤中指定。例如，若要將下列文字從英文翻譯成德文：



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

文本「盧浮宮博物館」需要保持法語，因此我們使用 `span` 標籤來跳過此內容的翻譯：

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

這句話將由此產生的翻譯成德語：

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

## do-not-translate 與 API 搭配使用

您可以搭 `do-not-translate` 配即時 API 作業 (`TranslateText` 和 `TranslateDocument`) 和非同步 `StartTextTranslationJob` API 作業搭配使用。在您為 API 要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翻譯的內容。

在下面的例子中，我們想要將一些文本從英語翻譯成西班牙語，但保留一些英文文本：

```
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 要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

搭配翻譯要求使用自訂術語，以確保您的品牌名稱、字元名稱、型號名稱和其他獨特內容都能翻譯成想要的結果。

您可以建立術語檔案，並將其上傳到您的 Amazon Translate 帳戶。如需檔案大小和術語檔案數目的資訊，請參閱 [Service Quotas](#)。翻譯文字時，您可以選擇性地選擇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，就會使用術語檔案中的翻譯。

考慮以下示例：Amazon 照片提供免費的照片和視頻存儲 Amazon Prime 會員。在法語中，名稱沒有翻譯：它仍然是 Amazon 照片。

當您使用 Amazon Translate 在沒有任何其他上下文的情況下將 Amazon 照片翻譯成法語時，結果是「亞馬遜照片」，這不是理想的翻譯。

如果您新增「Amazon 相片」一詞的自訂術語條目，並指定法文翻譯為 Amazon 相片，Amazon Translate 會使用自訂術語將片語翻譯成想要的結果。

Amazon Translate 不保證每個翻譯都會使用目標字詞。自訂術語會使用翻譯內容中來源與目標詞彙的意義來決定是否要使用目標詞彙。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](#)
- [使用自訂術語範例 AWS SDK for Python \(Boto\)](#)
- [加密您的術語](#)
- [最佳實務](#)

## 建立自訂術語

您可以藉由建立術語檔案來定義自訂術語。亞馬遜翻譯支援 CSV、TSV 或 TMX 檔案格式。檔案中的每個項目都包含來源字詞和每個目標語言的對等 (翻譯) 字詞。

建立術語檔案後，您可以將檔案上傳到您的 Amazon 翻譯帳戶。

### 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的項目時，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 檔案使用翻譯軟體經常使用的 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

上傳自訂術語檔案時，您可以設定自訂術語的方向性值。方向性指出您的術語檔案是指定一種來源語言還是多種來源語言。

對於方向性，請設定下列其中一個值：

### 單向

術語檔案包含一種來源語言 (清單中的第一種語言)。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，而所有其他欄都包含目標語言的文字。

## 多方向

檔案中的任何語言都可以是來源語言或目標語言。例如，如果術語檔案包含英文、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配對的工作：

- 英文譯成法文
- 英語到西班牙語
- 法文譯成英文
- 法文至西班牙文
- 西班牙語到英語
- 西班牙文至法文

相反地，您需要為這六個翻譯工作建立三個單向術語檔案 (每個來源語言各一個)。

## 使用自訂術語

若要在使用作 [TranslateText](#) 業轉換文字時使用「自訂術語」，請包含選用 `TerminologyNames` 參數。

例如，如果您將下列名為的術語檔案上傳 `Amazon_Family.csv` 至您的帳戶：

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

您可以使用下列 CLI 指令，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?"
```

這使用選定的自定義術語將此文本翻譯為「亞馬遜家庭是否合適？」而不是直接（但不受歡迎的）翻譯「與家庭亞馬遜家庭的幫助嗎？」

下面的例子演示了如何在 Python 中使用相同的術語文件。

```
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")
```

如需將 Amazon 翻譯操作搭配自訂術語搭配使用的詳細資訊，請參閱 [API](#) 操作。

## 使用自訂術語範例 AWS SDK for Python (Boto)

下面的例子演示了如何在 Python 中使用自定義術語操作。若要執行此範例 Python 請透過 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 致力於保護您的所有資料，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加密。AWS加密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。主控台會顯示一個輸入欄位，供您輸入加密金鑰的 Amazon 資源名稱 (ARN)。

## 最佳實務

使用自訂術語時，請使用下列一般最佳作法：

- 保持您的自訂術語整齊。僅包含您需要控制轉譯值的詞彙。
- 自訂術語會區分大小寫。如果您想要單字的大寫和非大寫版本的目標轉譯，請為每個版本包含一個項目。
- 自訂術語並不是用來控制間距、標點符號或大小寫的工具。例如，請避免下列類型的項目：
  - 新增空格 — 英文：美國 FR：U S A
  - 新增標點符號 — EN：美國 FR：美國
  - 改變大小寫 — 英：美國 FR：美國
- 請勿為同一個來源片語包含不同的翻譯。例如：
  - 條目 #1-EN: Amazon FR: Amazon
  - 條目 #2-EN: Amazon FR: 亞馬遜
- 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翻譯使用目標術語。若要使用自訂術語達到高精確度，請在建立術語檔案的內容時遵循下列最佳作法：

- 自定義術語適用於任何單詞，包括動詞和同詞。適當的名稱，例如品牌名稱和產品名稱，是理想的條目。
- 目標術語應該精通目標語言。對於包含許多特殊字元或格式的目標字詞，不建議使用自訂術語。
- 您可以在術語檔案中包含多個字詞語或子句。但是，包含多個單詞的術語在目標語言中流利閱讀的可能性較低。
- 自訂術語會使用翻譯內容中來源與目標詞彙的意義來決定是否要使用目標詞彙。如果目標字詞不適合指定的翻譯環境，Amazon Translate 可能不會使用目標字詞。

例如，如果您的術語檔案包含下列英文到法文的項目：

EN: order, FR: commande ( 基於英語「訂購」翻譯成法語「指揮官」 )。

翻譯下列句子時，Amazon Translate 不會使用此項目，因為翻譯上下文不相符：

「為了讓我們能夠幫助您，請分享您的名字。」

避免這種情況的建議：



- 請確定每種語言的目標詞語在語意上等同於來源字詞。
- 避免使用具有多重意義的來源或目標術語。

## 在 Amazon Translate 中使用簡潔

在語言之間進行翻譯時，有時翻譯輸出比預期的長度 (以字元計數計算)。如果沒有額外字元的空間，較長的輸出可能會在某些情況下造成問題 (例如註解、字幕、標題或表格欄位)。

使用 Amazon Translate 翻譯執行即時文字翻譯時，您可以開啟簡潔設定。簡潔度減少了大多數翻譯的翻譯輸出長度 (與翻譯輸出相比，沒有簡潔性)。

### 主題

- [使用簡潔度設定](#)
- [支援的語言](#)

## 使用簡潔度設定

您可以將簡潔設定與即時文字翻譯搭配使用。Amazon Translate 不支援即時文件翻譯或非同步翻譯任務的簡潔性。

若要在即時文字翻譯請求中使用簡潔性，請執行下列其中一項操作：

- 主控台 — 在即時翻譯頁面的 [文字] 索引標籤的 [其他設定] 底下，選擇 [簡潔] 設定。
- AWS CLI — 在 `translate-text` 指令中，將 `--settings` 參數中的簡潔性設定為 `Brevity=0N`。若要取得更多資訊，請參閱 [〈AWS CLI 指令參考〉中的轉換文字](#)。
- AWSAPI — 在 [TranslateText](#) API 操作中，在「設置」參數中配置簡潔性。

## 支援的語言

Amazon Translate 支持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語。

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

如果您要求使用不支援的語言配對進行翻譯的簡潔性，翻譯會在關閉簡潔設定的情況下繼續進行。

## 在 Amazon Translate 中掩蓋褻瀆的單詞和短語

使用 Amazon Translate 執行翻譯時，您可以啟用褻瀆設定，以遮罩翻譯輸出中不雅的字詞和片語。

為了掩蓋褻瀆的單詞和短語，Amazon Translate 將其替換為 grawlix 字符串「? \$ # @ \$」。這個 5 個字符的序列用於每個褻瀆的單詞或短語，而不管單詞的長度或數量如何。

在來源語言和目標語言相同的翻譯請求中，Amazon Translate 不會遮蓋不雅用語。

在某些情況下，來源輸入中的褻瀆字詞在翻譯的輸出中可能自然變得不冒犯。在這種情況下，不會套用遮罩。

Amazon Translate 可以從字面上檢測每個褻瀆的單詞或短語，而不是上下文。這意味著即使在上下文中是無害的，它也可能掩蓋了褻瀆的單詞。例如，如果 Amazon Translate 檢測到「混蛋」作為一個褻瀆的詞，那麼它會寫短語「混蛋雞」為「? \$ # @ \$ 雞肉」，即使「混蛋雞」是無攻擊性的。（在這裡，「混蛋」僅用作示例。Amazon Translate 不會將該單詞檢測為褻瀆。）

### 主題

- [使用褻瀆設定](#)
- [不支援語言](#)

## 使用褻瀆設定

您可以在 Amazon Translate 中使用這兩種類型的翻譯操作：即時翻譯和非同步批次處理。

若要在即時翻譯要求中遮罩不雅用語，請執行下列任一項作業：

- 在 Amazon Translate 翻譯主控台的 [即時翻譯] 頁面的 [其他設定] 下，啟用 [褻瀆] 設定。
- 在的translate-text指令中AWS CLI，將--settings參數設定為Profanity=MASK。若要取得更多資訊，請參閱〈AWS CLI指令參考〉中的[轉換文字](#)。
- 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[TranslateText](#)中的。

若要在非同步批次作業中遮罩不雅的字詞，請參閱[執行批次翻譯工作](#)。

## 不支援語言

當您翻譯成 Amazon Translate 支援的任何目標語言時，您可以遮罩不雅用語，但下列情況除外：

語言	語言代碼
孟加拉文	bn
北印度文	hi
馬拉雅拉姆	ml
旁遮普	pa
僧伽羅語	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 中設置形式

您可以選擇性地指定翻譯為支援的目標語言的形式程度。形式設定會控制翻譯輸出中正式語言使用的層級 (也稱為[敬語](#)或[暫存器](#))。形式設定可用於即時翻譯和非同步批次處理。

形式支持以下值：

- 非正式 — 翻譯文本中的所有句子都使用與非正式溝通相關的語言結構。例如，翻譯文本使用熟悉的第二人稱代詞形式和它們的動詞協議 (或日語 Kudaketa 形式)。

- 正式 — 翻譯文本中的所有句子都使用與正式有禮貌的溝通相關的語言結構。例如，翻譯文本使用第二人稱代詞的正式形式和它們的動詞協議（或日語的 Teineigo 形式）。

例如，句子「你確定嗎？」在德語中可以有兩個正確的翻譯：「你想知道嗎？」正式登記冊和 '比斯特杜西赫?' 對於非正式的一個。

如果 Amazon Translate 不支援目標語言的形式層級，或者您未指定形式參數，則翻譯工作會忽略形式設定。

## 主題

- [使用形式設定](#)
- [支援的語言](#)

## 使用形式設定

若要在即時翻譯要求中設定形式，請執行下列其中一個動作：

- 在 Amazon Translate 主控台的「即時翻譯」頁面的「其他設定」下，啟用「形式」設定，然後選取其中一個值。
- 在 Amazon Translate API 的 [TranslateText](#) 操作中使用設置參數。
- 對於中的 translate-text 指令 AWS CLI，將 --settings 參數設定為 Formality=FORMAL 或 Formality=INFORMAL。若要取得更多資訊，請參閱 [AWS CLI 指令參考](#) 中的 [轉換文字](#)。

若要在批次轉譯請求中設定形式，請在開始翻譯工作時設定 Formality 參數。如需詳細資訊和範例，請參閱 [執行批次翻譯工作](#)。

針對 CLI 或 API 要求，回應中的 AppliedSettings 欄位包含來自要求的形式設定 (如果有的話)。如果目標語言不支持形式，則響應中的 AppliedSettings 值為 NULL。

## 支援的語言

Amazon Translate 支援從任何來源語言翻譯成下列目標語言的形式設定。

語言	語言代碼
荷蘭文	nl

語言	語言代碼
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 parallel 資料自訂翻譯 (使用中自訂翻譯)

加parallel 資料到 Amazon Translate，以自訂批次翻譯任務的輸出。平行資料由範例組成，顯示您希望如何轉譯文字區段。它包含來源語言中的文字範例集合，每個範例都包含一或多種目標語言中所需的翻譯輸出。

當您將 parallel 資料新增至批次轉譯工作時，您會建立活動自定義翻譯任務。當您執行這些任務時，Amazon Translate 會在執行階段使用您的 parallel 資料來產生自訂的機器翻譯輸出。它會調整翻譯，以反映在 parallel 資料中找到的樣式、色調和字詞選擇。透過 parallel 資料，您可以針對特定領域 (例如生命科學、法律或財務) 特有的字詞或片語量身打造翻譯。

### Note

使用中自訂翻譯工作的定價比其他不使用 parallel 資料的工作更高。如需詳細資訊，請參閱《《[Amazon Translate 定價](#)》。

例如，在 CSV 檔案中定義下列 parallel 資料：

```
"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。這個例子顯示了源短語「你好嗎？」應翻譯成法文。將此範例輸入檔案匯入 Amazon Translate 後，即可將其套用至翻譯工作，以影響其輸出。在這樣的工作期間，Amazon Translate 翻譯為「你好嗎？」進入非正式的「評論 ça va？」而不是正式的「有評論？」例如，工作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?
```

從此文字中，工作產生下列翻譯：

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

相反，如果作業在沒有並行數據的情況下運行，則輸出可能包括更正式的「評論 ALLEZ-VOUS」：

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

透過使用 parallel 資料自訂批次翻譯工作，您會以類似於使用自訂翻譯模型 (使用翻譯範例訓練的自訂翻譯模型) 的方式影響輸出。使用「使用中自訂翻譯」時，不需要訓練自訂模型，而且您可以避免此類訓練所需的時間和費用。隨著翻譯需求隨著時間的推移而改變，您可以更新 parallel 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

下列區域提供使用中自訂翻譯：

- 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 平行數據輸入文件](#)
- [將您的 parallel 資料新增至 Amazon Translate](#)
- [在 Amazon Translate 中查看和管理 parallel 數據](#)

## Amazon Translate 平行數據輸入文件

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 parallel 資料資源。您的 parallel 資料輸入檔案必須使用 Amazon Translate 支援的語言。如需這些語言的列表，請參閱[支援的語言和語言代碼](#)。

### 範例 parallel 資料

下表中的文字提供可格式化為 parallel 資料輸入檔案的轉譯區段範例：

zh	es
Amazon Translate 是一種神經機器翻譯服務。	Amazon Translate 是神經元自動翻譯服務。
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。	神經元自動化的自動化形式自動化工師使用自動化工具的形式。
Amazon Translate 可讓您為國際使用者本地化內容。	Amazon Translate 為國際使用者的最佳本地化。

表格的第一列提供語言代碼。第一種語言英文 (en) 是來源語言。西班牙文是目標語言。第一欄提供來源文字的範例。另一欄包含翻譯範例。當此 parallel 資料自訂批次任務時，Amazon Translate 會調整翻譯以反映範例。

### 輸入檔案格式

Amazon Translate 支援 parallel 資料輸入檔案的下列格式：

- 翻譯記憶庫交換 (TMX)
- 逗點分隔值 (CSV)

- 定位點分隔值 (TSV)

## TMX

### Example TMX 輸入檔案

下列範例 TMX 檔案會以 Amazon Translate 接受的格式定義 parallel 資料。在這個檔案中，英文 (en) 是來源語言。西班牙文 (es) 是目標語言。作為 parallel 資料的輸入檔案，它提供了幾個範例，Amazon Translate 可用來量身打造批次任務的輸出。

```
<?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>
    <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>
```



```
    </tu>
  </body>
</tmx>
```

## TMX 要求

在 TMX 檔案中定義 parallel 資料時，請記住 Amazon Translate 的以下要求：

- Amazon Translate 支持 TMX 1.4b。如需詳細資訊，請參閱[尺寸規格](#)在全球化與本地化協會網站上。
- 所以此header元素必須包含srclang屬性。此屬性的值決定 parallel 資料的來源語言。
- 所以此body元素至少必須包含一個 Translate 單元 (tu) 元素。
- 每個tu元素必須包含至少兩個翻譯單位變體 (tuv) 元素。其中一項tuv元素必須具有xml:lang與指定給的屬性具有相同值的屬性srclang中的屬性header元素。
- 全部tuv元素必須具有xml:lang屬性。
- 全部tuv元素必須具有區段 (seg) 元素。
- 在處理您的輸入文件時，Amazon Translate 會跳過某些tu或者tuv元素，如果，它，遇到seg是空的或只包含空格的元素：
  - 如果seg元素對應於源語言，Amazon Translate 跳過tu該元素seg元素佔據。
  - 如果seg元素對應於一個目標語言，Amazon Translate 只跳過tuv該元素seg元素佔據。
- 在處理您的輸入文件時，Amazon Translate 會跳過某些tu或者tuv元素，如果，它，遇到seg超過1000 個字節的元素：
  - 如果seg元素對應於源語言，Amazon Translate 跳過tu該元素seg元素佔據。
  - 如果seg元素對應於一個目標語言，Amazon Translate 只跳過tuv該元素seg元素佔據。
- 如果輸入文件包含多個tuAmazon Translate 具有相同來源文字的元素會執行下列其中一項操作：
  - 如果tu元素具有changedate屬性，它使用的元素與最新的日期。
  - 否則，它會使用最接近檔案結尾的元素。

## CSV

下列範例 CSV 檔案會以 Amazon Translate 接受的格式定義 parallel 資料。在這個檔案中，英文 (en) 是來源語言。西班牙文 (es) 是目標語言。作為 parallel 資料的輸入檔案，它提供了幾個範例，Amazon Translate 可用來量身打造批次任務的輸出。

## 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

在 CSV 檔案中定義 parallel 資料時，請記住 Amazon Translate 的以下要求：

- 第一行由語言代碼組成。第一個代碼是源語言，後續每個代碼都是一種目標語言。
- 第一欄中的每個欄位都包含來源文字。後續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為開頭：+ , - , = , @。無論欄位是否以雙引號 ( " ) 括起來，都適用此要求。
- 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，文字如：

```
34" monitor
```

必須寫成：

```
34"" monitor
```

- 在處理輸入檔案時，如果 Amazon Translate 遇到空白或僅包含空白字元的欄位，則會略過某些行或欄位：
  - 如果來源文字欄位為空白，Amazon Translate 會略過它所佔用的行。
  - 如果目標翻譯欄位為空白，Amazon Translate 譯只會略過該欄位。
- 在處理輸入檔案時，如果遇到超過 1000 位元組的欄位，Amazon Translate 會略過某些行或欄位：
  - 如果來源文字欄位超過位元組限制，Amazon Translate 會略過它所佔用的行。
  - 如果目標翻譯欄位超過位元組限制，Amazon Translate 譯只會略過該欄位。

- 如果輸入檔案包含多個具有相同來源文字的記錄，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 parallel 資料。在這個檔案中，英文 (en) 是來源語言。西班牙文 (es) 是目標語言。作為 parallel 資料的輸入檔案，它提供了幾個範例，Amazon Translate 可用來量身打造批次任務的輸出。

### 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

在 TSV 檔案中定義 parallel 資料時，請記住 Amazon Translate 的以下要求：

- 第一行由語言代碼組成。第一個代碼是源語言，後續每個代碼都是一種目標語言。
- 第一欄中的每個欄位都包含來源文字。後續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定位字元，則文字必須以雙引號 ( " ) 字元括起來。
- 文字欄位不能跨越多行。
- 欄位不能以下列字元為開頭：+ , - , = , @。無論欄位是否以雙引號 ( " ) 括起來，都適用此要求。
- 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，文字如：

```
34" monitor
```

必須寫成：

```
34"" monitor
```

- 在處理輸入檔案時，如果遇到空白或僅包含空白字元的欄位，Amazon Translate 會略過某些行或欄位：
  - 如果來源文字欄位為空白，Amazon Translate 會略過它所佔用的行。
  - 如果目標翻譯欄位為空白，Amazon Translate 譯只會略過該欄位。
- 在處理輸入檔案時，如果遇到超過 1000 位元組的欄位，Amazon Translate 會略過某些行或欄位：
  - 如果來源文字欄位超過位元組限制，Amazon Translate 會略過它所佔用的行。
  - 如果目標翻譯欄位超過位元組限制，Amazon Translate 譯只會略過該欄位。
- 如果輸入檔案包含多個具有相同來源文字的記錄，Amazon Translate 會使用最接近檔案結尾的記錄。

## 將您的 parallel 資料新增至 Amazon Translate

若要將 parallel 資料新增至 Amazon Translate，請從 Amazon S3 匯入 parallel 資料輸入檔案。之後，您可以使用 parallel 資料來自訂批次轉譯工作所產生的輸出。

### 先決條件

您必須先執行下列動作，才能將 parallel 資料新增至 Amazon Translate：

- 有一個 parallel 的數據輸入文件。若要建立服務角色，請參閱[Amazon Translate 平行數據輸入文件](#)。
- 在你的 Amazon S3 儲存貯體AWS帳戶。若要建立，請參閱[如何建立 S3 儲存貯體？](#)在Amazon Simple Storage Service 範例。
- 將輸入檔案上傳至 Amazon S3 儲存貯體。如需詳細資訊，請參閱[如何將檔案與資料夾上傳至 S3 儲存貯體？](#)在Amazon Simple Storage Service 範例。

## 新增 parallel 資料 (Amazon Translate 主控台)

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

1. Open[Amazon Translate 主控台](#)。
2. 從左側的導覽選單中，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 parallel 資料。此主控台會顯示建立 parallel 資料頁面。

#### 4. 提供下列資訊：

##### 名稱

parallel 資料 Resource 的自訂名稱。您必須指定帳戶和區域中唯一的名稱。

##### 描述-可選的

自訂描述。

##### S3 上的並行資料位置

parallel 資料輸入檔案在 Amazon S3 中的位置。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

parallel 資料輸入檔案的格式。支援的格式包括翻譯記憶庫交換 (TMX)、逗號分隔值 (CSV) 和定位點分隔值 (TSV)。

5. 下加密金鑰，選擇一個AWS KMS key以保護您的 parallel 數據。這些 KMS 金鑰由AWS Key Management Service(AWS KMS)。如需 AWS KMS 的詳細資訊，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。這是默認選項，如果您不選擇其他方法，則用於加密您的信息。如需詳細資訊，請參閱[AWS 擁有的金鑰](#)在AWS Key Management Service開發人員指南。

##### 使用當前帳戶的密鑰

使用您管理的其中一個 KMS 金鑰AWS KMS在您的AWS帳戶。如果您選擇此選項，功能表會提供 KMS 金鑰清單供您選擇。如需詳細資訊，請參閱[客戶受管金鑰](#)在AWS Key Management Service開發人員指南。

##### 使用來自不同帳戶的金鑰

使用在中管理的 KMS 金鑰AWS KMS在一個不同的AWS帳戶。如果您選擇此選項，則主控台會提供一個欄位供您輸入 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 parallel 資料。

控制台返回平行資料頁面，此頁面頂端的橫幅會顯示匯入狀態。幾分鐘後，您的 parallel 資料資源就會顯示在表格中。當中的值狀態直欄是作用中parallel 資料已準備好供您在批次轉譯工作中使用。

### 用於疑難排解的錯誤檔

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，您可以下載該檔案以檢閱錯誤訊息。此檔案的內容類似於下列範例：

```
{
  "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 parallel 資料 (AWS CLI)

若要使用新增 parallel 資料AWS CLI，使用create-parallel-data指令。

Example create-parallel-data 命令

下列範例透過從 Amazon S3 匯入 TSV 檔案來建立 parallel 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 parallel 資料物件的狀態回應：

```
{
  "Name": "my-parallel-data",
```

```
"Status": "CREATING"  
}
```

您可 parallel 使用 `get-parallel-data` 指令。當狀態為 `ACTIVE` parallel 資料已準備好供您在批次轉譯工作中使用。對於一個例子 `get-parallel-data` 指令，請參閱[檢視 parallel 資料物件的詳細資訊的步驟](#)。

## 使用 parallel 資料

現在您已經建立了 parallel 資料資源，您可以將其套用至批次轉譯工作以自訂輸出。若要執行批次工作，請參閱[執行批次翻譯工作](#)。

## 在 Amazon Translate 中查看和管理 parallel 數據

您可以檢視已新增至 Amazon Translate 的所有 parallel 資料資源，也可以存取每個資料的詳細摘要。隨著翻譯需求的變更，您可以更新 parallel 資料，以精簡您的翻譯輸出。

### 檢視和管理 parallel 資料 (Amazon Translate 主控台)

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

#### 檢視 parallel 資料資源清單

1. 開啟[Amazon Translate 主控台](#)。
2. 在左側的導覽選單中，選擇自訂，然後選擇平行資料。此頁面上的表格列出您已新增至 Amazon Translate 的 parallel 資料資源。

#### 檢視 parallel 資料資源的詳細資訊

- 在「」平行資料頁面上，選擇 parallel 資料資源的名稱名稱直欄。主控台會開啟詳細資料頁面，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 parallel 資料資源

1. 將 parallel 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 parallel 資料，然後選擇更新。此主控台會顯示 parallel 資料頁面。
4. 提供下列資訊：

## 描述-可選

已更新的說明。

## S3 上的並行資料位置

已更新 parallel 資料輸入檔案在 Amazon S3 中的位置。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。

## 選擇 parallel 資料檔案格式

parallel 資料輸入檔案的格式。支援的格式包括翻譯記憶庫交換 (TMX)、逗號分隔值 (CSV) 和定位點分隔值 (TSV)。

5. 選擇 Save (儲存)。Amazon Translate 會以新的輸入檔案取代舊的 parallel 資料。

## 檢視和管理 parallel 資料 (AWS CLI)

您可以使用 AWS CLI 以檢視和更新您的 parallel 資料資源。

### 檢視 parallel 資料資源清單

若要檢視已新增至 Amazon Translate 的 parallel 資料資源清單，請使用 `list-parallel-data` 指令。

### Example `list-parallel-data` 命令

下列範例會傳回 parallel 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 parallel 資料物件的詳細資訊的步驟

若要查詢單一 parallel 資料資源的詳細資料，請使用 `get-parallel-data` 指令。此命令會傳回 parallel 資料的屬性以及預先簽署的 S3 URL，您可以在其中下載用來建立資料的輸入檔案。

#### Example `get-parallel-data` 命令

下列範例會取得 `my-parallel-data` 物件：

```
$ 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 parallel 資料資源

若要更新 parallel 資料資源，請先將新的輸入檔案上傳到 Amazon S3 輸入儲存貯體。然後，使用 `update-parallel-data` 命令，然後指定您要更新的 parallel 資料資源。Amazon Translate 會以新輸入檔案中的資訊取代舊的 parallel 資料。

### Example `update-parallel-data` 命令

以下命令更新 `my-parallel-data` 使用 Amazon S3 的新輸入檔案：

```
$ 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 會提供如下的回應：

```
{
  "Name": "my-parallel-data",
  "Status": "ACTIVE",
  "LatestUpdateAttemptStatus": "UPDATING",
  "LatestUpdateAttemptAt": 1598601455.844
}
```

在此回應中，`Status` 欄位提供預先存在的 parallel 資料物件的狀態，以及 `LatestUpdateAttemptStatus` 欄位會提供目前更新嘗試的狀態。

# 使用 AWS 開發套件的 Amazon Translate 程式碼範例

下列程式碼範例說明如何搭配 AWS 軟體開發套件 (SDK) 使用 Amazon Translate。

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和跨服務範例中查看內容中的動作。

Scenarios (案例) 是向您展示如何呼叫相同服務中的多個函數來完成特定任務的程式碼範例。

Cross-service examples (跨服務範例) 是跨多個 AWS 服務執行的應用程式範例。

如需 AWS SDK 開發人員指南和程式碼範例的完整清單，請參閱[透過 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

- [使用 AWS SDK 進行 Amazon Translate 的操作](#)
  - [使用 AWS 開發套件描述 Amazon Translate 工作](#)
  - [使用 AWS 開發套件列出 Amazon Translate 任務](#)
  - [使用開 AWS 發套件開始 Amazon Translate 任務](#)
  - [使用 AWS 開發套件停止 Amazon Translate 工作](#)
  - [使用 AWS 開發套件使用亞馬遜 Translate 文字](#)
- [使用 AWS 開發套件的 Amazon Translate 案例](#)
  - [使用開 AWS 發套件開始使用 Amazon Translate 任務](#)
- [使 AWS 用軟體開發套件的 Amazon Translate 跨服務範例](#)
  - [建置 Amazon Transcribe 串流應用程式](#)
  - [建立 Amazon Lex 聊天機器人來吸引您的網站訪客](#)
  - [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 進行 Amazon Translate 的操作

下列程式碼範例示範如何使用 AWS SDK 執行個別 Amazon Translate 動作。這些摘錄呼叫 Amazon Translate API，是大型程式的程式碼摘錄，這些程式必須在內容中執行。每個範例都包含一個連結 GitHub，您可以在其中找到設定和執程式碼的指示。

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 參考資料](#)。

## 範例

- [使用 AWS 開發套件描述 Amazon Translate 工作](#)
- [使用 AWS 開發套件列出 Amazon Translate 任務](#)
- [使用開 AWS 發套件開始 Amazon Translate 任務](#)
- [使用 AWS 開發套件停止 Amazon Translate 工作](#)
- [使用 AWS 開發套件使用亞馬遜 Translate 文字](#)

## 使用 AWS 開發套件描述 Amazon Translate 工作

下列程式碼範例說明如何描述 Amazon Translate 譯工作。

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：

- [開始使用翻譯工作](#)

### .NET

#### AWS SDK for .NET

#### Note

還有更多關於 GitHub。尋找完整範例，並了解如何在[AWS 設定和執行程式碼範例儲存庫](#)。

```
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 AWS SDK for .NET API 參考[DescribeTextTranslationJob](#)中的。

## SAP ABAP

### 適用於 SAP ABAP 的開發套件

#### Note

還有更多關於 GitHub。尋找完整範例，並了解如何在[AWS 設定和執行程式碼範例儲存庫](#)。

```

    "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."
        EXPORTING
            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'.
  ENDRY.
```

- 如需 API 詳細資訊，請參閱 AWS SDK [DescribeTextTranslationJob](#) 中的 SAP ABAP API 參考資料。

如需 AWS SDK 開發人員指南和程式碼範例的完整清單，請參閱[透過 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 開發套件列出 Amazon Translate 任務

下列程式碼範例顯示如何列出 Amazon Translate 譯工作。

.NET

AWS SDK for .NET

### Note

還有更多關於 GitHub。尋找完整範例，並了解如何在[AWS 設定和執行程式碼範例儲存庫](#)。

```
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 AWS SDK for .NET API 參考[ListTextTranslationJobs](#)中的。

## SAP ABAP

### 適用於 SAP ABAP 的開發套件

#### Note

還有更多關於 GitHub。尋找完整範例，並了解如何在[AWS 設定和執行程式碼範例儲存庫](#)。

"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."

```
CREATE OBJECT lo_filter
```

```
EXPORTING
```

```
    iv_jobname = iv_jobname.
```

```
TRY.
```

```
oo_result = lo_xl8->listtexttranslationjobs( "oo_result is returned
for testing purposes."
    EXPORTING
        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 AWS SDK [ListTextTranslationJobs](#) 中的 SAP ABAP API 參考資料。

如需 AWS SDK 開發人員指南和程式碼範例的完整清單，請參閱 [透過 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。

## 使用開 AWS 發套件開始 Amazon Translate 任務

下列程式碼範例顯示如何開始 Amazon Translate 譯工作。

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：

- [開始使用翻譯工作](#)

## .NET

### AWS SDK for .NET

#### Note

還有更多關於 GitHub。尋找完整範例，並了解如何在[AWS 設定和執行程式碼範例儲存庫](#)。

```
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://DOC-EXAMPLE-BUCKET1/FOLDER/";
        var s3OutputUri = "s3://DOC-EXAMPLE-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 AWS SDK for .NET API 參考[StartTextTranslationJob](#)中的。

## SAP ABAP

### 適用於 SAP ABAP 的開發套件

#### Note

還有更多關於 GitHub。尋找完整範例，並了解如何在[AWS 設定和執行程式碼範例儲存庫](#)。

```
"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."
CREATE OBJECT lo_inputdataconfig
  EXPORTING
    iv_s3uri          = iv_input_data_s3uri
    iv_contenttype    = iv_input_data_contenttype.

"Create an ABAP object for the output data config."
CREATE OBJECT lo_outputdataconfig
  EXPORTING
    iv_s3uri = iv_output_data_s3uri.

"Create an internal table for target languages."
CREATE OBJECT lo_targetlanguagecodes
  EXPORTING
```

```
        iv_value = iv_targetlanguagecode.
INSERT lo_targetlanguagecodes INTO TABLE lt_targetlanguagecodes.

TRY.
    oo_result = lo_xl8->starttexttranslationjob(      "oo_result is returned
for testing purposes."
        EXPORTING
            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 AWS SDK [StartTextTranslationJob](#) 中的 SAP ABAP API 參考資料。

如需 AWS SDK 開發人員指南和程式碼範例的完整清單，請參閱[透過 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 開發套件停止 Amazon Translate 工作

下列程式碼範例顯示如何停止 Amazon Translate 譯工作。

## .NET

### AWS SDK for .NET

#### Note

還有更多關於 GitHub。尋找完整範例，並了解如何在[AWS 設定和執行程式碼範例儲存庫](#)。

```
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 AWS SDK for .NET API 參考[StopTextTranslationJob](#)中的。

## SAP ABAP

### 適用於 SAP ABAP 的開發套件

#### Note

還有更多關於 GitHub。尋找完整範例，並了解如何在[AWS 設定和執行程式碼範例儲存庫](#)。

```

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

TRY.
    oo_result = lo_xl8->stoptexttranslationjob(      "oo_result is returned
for testing purposes."
    EXPORTING
        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 AWS SDK [StopTextTranslationJob](#) 中的 SAP ABAP API 參考資料。

如需 AWS SDK 開發人員指南和程式碼範例的完整清單，請參閱[透過 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 開發套件使用亞馬遜 Translate 文字

下列程式碼範例會示範如何使用 Amazon 翻譯文字來翻譯文字。

.NET

AWS SDK for .NET

### Note

還有更多關於 GitHub。尋找完整範例，並了解如何在[AWS 設定和執行程式碼範例儲存庫](#)。

```
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 the 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 = "DOC-EXAMPLE-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 translate.</
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 AWS SDK for .NET API 參考 [TranslateText](#) 中的。

## SAP ABAP

### 適用於 SAP ABAP 的開發套件

#### Note

還有更多關於 GitHub。尋找完整範例，並了解如何在 [AWS 設定和執行程式碼範例儲存庫](#)。

```

"Translates input text from the source language to the target language."
TRY.
    oo_result = lo_xl8->translatetext(      "oo_result is returned for
testing purposes."
    EXPORTING
        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_xl8serviceunavaillex .
    MESSAGE 'The Amazon Translate service is temporarily unavailable.' TYPE
'E'.

```

```
CATCH /aws1/cx_xl8textsize1mtexcde .
    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 AWS SDK [TranslateText](#) 中的 SAP ABAP API 參考資料。

如需 AWS SDK 開發人員指南和程式碼範例的完整清單，請參閱[透過 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 開發套件的 Amazon Translate 案例

下列程式碼範例說明如何使用 AWS SDK 在 Amazon Translate 中實作常見案例。這些案例說明如何透過在 Amazon Translate 中呼叫多個函數來完成特定任務。每個案例都包含一個連結 GitHub，您可以在其中找到如何設定和執程式碼的指示。

### 範例

- [使用開 AWS 發套件開始使用 Amazon Translate 任務](#)

## 使用開 AWS 發套件開始使用 Amazon Translate 任務

以下程式碼範例顯示做法：

- 啟動非同步批次翻譯工作。
- 等待非同步工作完成。
- 描述非同步工作。

## SAP ABAP

## 適用於 SAP ABAP 的開發套件

 Note

還有更多關於 GitHub。尋找完整範例，並了解如何在[AWS 設定和執行程式碼範例儲存庫](#)。

```
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."
CREATE OBJECT lo_inputdataconfig
  EXPORTING
    iv_s3uri          = iv_input_data_s3uri
    iv_contenttype = iv_input_data_contenttype.

"Create an ABAP object for the output data config."
CREATE OBJECT lo_outputdataconfig
  EXPORTING
    iv_s3uri = iv_output_data_s3uri.

"Create an internal table for target languages."
CREATE OBJECT lo_targetlanguagecodes
  EXPORTING
    iv_value = iv_targetlanguagecode.
INSERT lo_targetlanguagecodes INTO TABLE lt_targetlanguagecodes.

TRY.
  DATA(lo_translationjob_result) = lo_xl8->starttexttranslationjob(
    EXPORTING
      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."
        EXPORTING
            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'.
    ENDRTRY.
```

- 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[透過 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 AWS 用軟體開發套件的 Amazon Translate 跨服務範例

下列範例應用程式使用 AWS 開發套件將 Amazon Translate 與其他 AWS 服務應用程式結合。每個範例都包含一個連結 GitHub，您可以在其中找到如何設定和執行應用程式的指示。

### 範例

- [建置 Amazon Transcribe 串流應用程式](#)
- [建立 Amazon Lex 聊天機器人來吸引您的網站訪客](#)
- [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](#)

## 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) 的開發套件

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。

有關如何設置和運行的完整源代碼和說明，請參閱中的完整示例[GitHub](#)。

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[透過 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 聊天機器人來吸引您的網站訪客

以下代碼示例演示瞭如何創建聊天機器人以吸引您的網站訪問者。

### 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，以吸引您的網站訪客。

有關如何設置和運行的完整源代碼和說明，請參閱中的完整示例[GitHub](#)。

此範例中使用的服務

- Amazon Comprehend
- Amazon Lex
- Amazon Translate

## JavaScript

### 適用於 JavaScript (v3) 的開發套件

示範如何使用 Amazon Lex API 在 Web 應用程式中建立 Chatbot，以吸引您的網站訪客。

如需有關如何設定和執行的完整原始程式碼和指示，請參閱開發人員 AWS SDK for JavaScript 員指南中的[建立 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[透過 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

#### AWS SDK for .NET

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。

有關如何設置和運行的完整源代碼和說明，請參閱中的完整示例[GitHub](#)。

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。

有關如何設置和運行的完整源代碼和說明，請參閱中的完整示例[GitHub](#)。

有關如何設置和運行使用 Java Async API 的示例的完整源代碼和說明，請參閱上[GitHub](#)的完整示例。

此範例中使用的服務

- Amazon SNS
- Amazon Translate

## Kotlin

### 適用於 Kotlin 的 SDK

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。

有關如何創建 Web 應用程序的完整源代碼和說明，請參閱中的完整示例[GitHub](#)。

有關如何創建本機 Android 應用程序的完整源代碼和說明，請參閱上的完整示例[GitHub](#)。

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[透過 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

#### AWS SDK for .NET

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指示，請參閱中的專案 [GitHub](#)。

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指示，請參閱中的專案 [GitHub](#)。

此範例中使用的服務

- Amazon Comprehend
- Lambda
- Amazon Polly
- Amazon Textract

- Amazon Translate

## JavaScript

### 適用於 JavaScript (v3) 的開發套件

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指示，請參閱中的專案 [GitHub](#)。下列摘錄顯示如何在 AWS SDK for JavaScript Lambda 函數內部使用。

```
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

### 適用於 Ruby 的開發套件

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指示，請參閱中的專案 [GitHub](#)。

### 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 [透過 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：組織資源並提供基於標籤的訪問控制。您可以將標籤新增至資源，然後建立 IAM 政策，以根據資源的標籤允許或限制對資源的存取。

政策可以根據請求中提供的標籤（請求標籤）或與您正在調用的資源（資源標籤）關聯的標籤來允許或禁止操作。如需具 IAM 使用的詳細資訊，請參閱[使用標籤控制存取權限](#)在 IAM User Guide。

搭配 Amazon Translate 使用標籤的注意事項：

- 每個資源最多可新增 50 個使用者標籤。
- 您可以在建立資源時或建立資源後隨時新增標籤。
- 一個標籤鍵是必要欄位，但是標籤值是選用的。
- 標籤在資源之間不一定是唯一的，但是給定資源的標籤必須具有唯一的鍵。
- 標籤鍵與值皆區分大小寫。
- 標籤金鑰最多可包含 128 個字元；一個標籤值最多可包含 256 個字元。
- AWS系統標籤以字首開始aws：在標籤鍵或值中。您不可新增、編輯或刪除具此字首的標籤名稱或值。系統標籤不包含在每個資源的標籤配額中。

## Note

如果您打算跨多個使用標記架構AWS服務和資源，請記得其他服務可能有字元使用限制。

## 主題

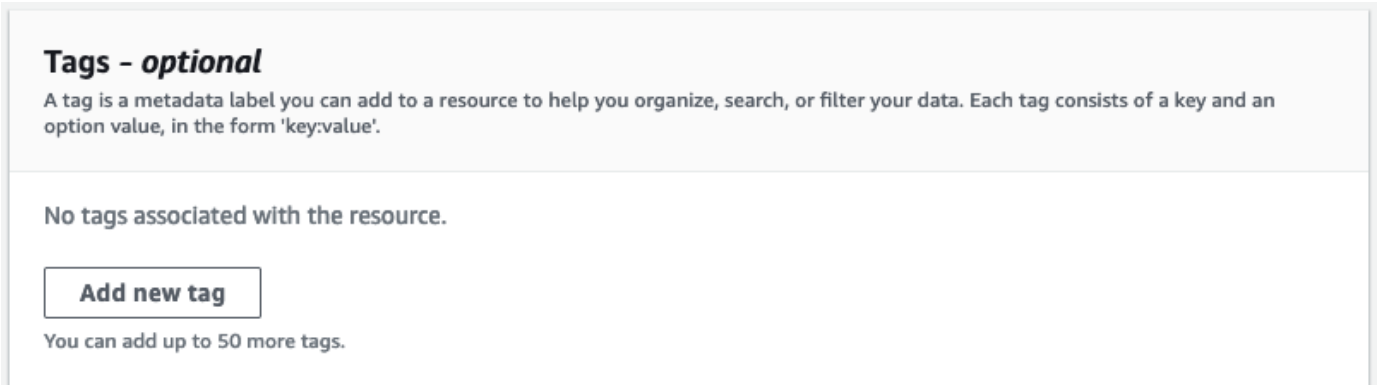
- [標記新資源](#)
- [檢視、更新和刪除與資源相關聯的標籤](#)

# 標記新資源

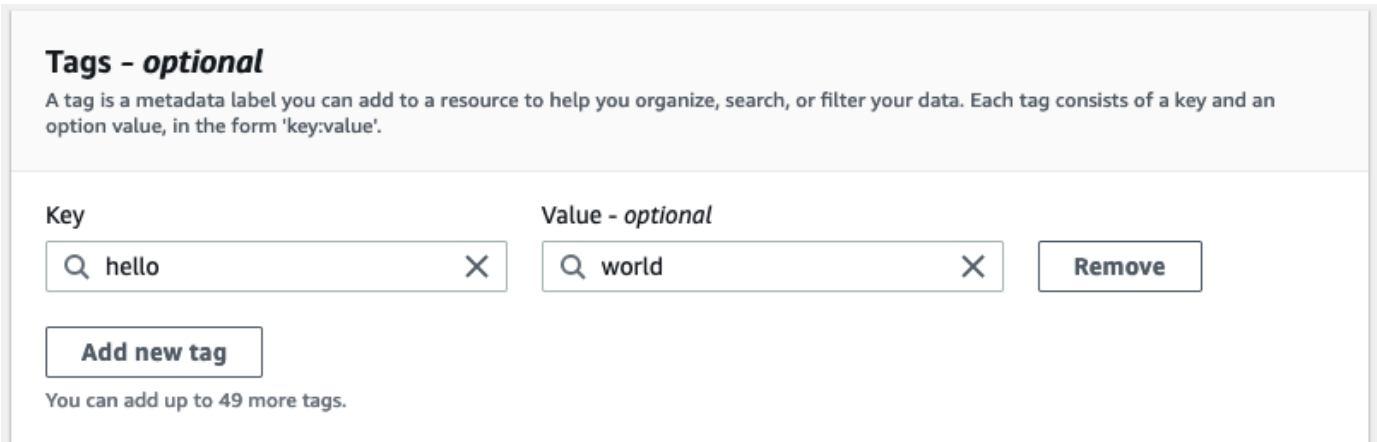
您可以在建立ParallelData或自訂術語資源時新增標籤。

## 若要將標籤新增至新資源 (主控台)

1. 登入 [Amazon Translate 主控台](#)。
2. 在左側導覽窗格中，選取要建立的資源 (Parallel data或Custom terminology)。
3. 選擇建立 parallel 資料或建立術語。主控台會顯示資源的主要「建立」頁面。在此頁面的最後，您會看到「標籤- 可選」面板。



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。



5. 重複步驟 4，直到新增所有標籤為止。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'.

<p><b>Key</b></p> <div style="border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;"> <input type="text" value="hello"/> <span style="float: right; font-size: 0.8em;">×</span> </div> <div style="border: 1px solid #ccc; padding: 2px; margin-bottom: 5px; border-color: red;"> <input type="text" value="Enter key"/> </div> <p style="font-size: 0.8em; color: red;">⚠ You must specify a tag key</p> <div style="border: 1px solid #ccc; padding: 2px; margin-top: 5px; width: fit-content;"> <b>Add new tag</b> </div> <p style="font-size: 0.8em; margin-top: 5px;">You can add up to 48 more tags.</p>	<p><b>Value - optional</b></p> <div style="border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;"> <input type="text" value="world"/> <span style="float: right; font-size: 0.8em;">×</span> </div> <div style="border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;"> <input type="text" value="Enter value"/> </div> <div style="border: 1px solid #ccc; padding: 2px; margin-top: 5px; width: fit-content; margin-left: auto;"> <b>Remove</b> </div> <div style="border: 1px solid #ccc; padding: 2px; margin-top: 5px; width: fit-content; margin-left: auto;"> <b>Remove</b> </div>
--	---

6. 選擇「建立 parallel 資料」或「建立術語」以建立資源。

您也可以使用 Amazon Translate [CreateParallelDataAPI](#) 操作添加標籤。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。

若要更新現有資源 (主控台) 的標籤

1. 登入 [Amazon Translate 主控台](#)。
2. 在左側導覽窗格中，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。控制台會開啟資源的詳細資訊頁面。
4. 向下捲動，直到看到「標記」面板。在這裡，您可以看到與所選資源關聯的所有標籤。

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

選取 [管理標籤] 以編輯或移除資源中的標籤。

5. 選擇要修改的文字，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機碼和值。
7. 修改完標籤後，請選取 [儲存]。

# Amazon Translate 中的安全

雲安全 AWS 是最高的優先級。身為 AWS 客戶，您可以從資料中心和網路架構中獲益，該架構專為滿足對安全性最敏感的組織的需求而打造。

安全是 AWS 與您之間共同承擔的責任。[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 AWS 雲端中執行 AWS 服務的基礎架構。AWS 還為您提供可以安全使用的服務。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 [合規計畫範圍內的 AWS 服務](#)。
- 雲端中的安全性 — 您的責任取決於您使用的 AWS 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 以符合安全性與合規性目標。您也會學到如何使用其他 AWS 服務，協助您監控和保護 AWS 資源。

## 主題

- [Amazon Translate 中的數據保護](#)
- [Amazon Translate 的 Identity and Access Management](#)
- [監控 Amazon Translate](#)
- [Amazon Translate 的合規驗證](#)
- [Amazon Translate 的彈性](#)
- [Amazon Translate 基礎設施安全](#)
- [Amazon Translate 和接口 VPC 端點 \( \)AWS PrivateLink](#)

## Amazon Translate 中的數據保護

Amazon Translate 符合 AWS [共同的責任模型](#)，其中包括有關資料保護的法規和準則。AWS 負責保護運行所有 AWS 服務的全球基礎設施。AWS 保持對此基礎架構上託管的數據的控制，包括用於處理客戶內容和個人數據的安全配置控制。AWS 客戶和 APN 合作夥伴 (擔任資料控制者或資料處理者) 必須對放置在 AWS 雲端中的任何個人資料負責。

基於資料保護目的，我們建議您使用 AWS Identity and Access Management (IAM) 保護 AWS 帳戶登入資料並設定角色，以便每位使用者僅獲得履行其工作職責所需的權限。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 資源進行通訊。
- 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性控制。
- 使用 Amazon Macie 等進階受管安全服務，協助探索和保護存放在 Amazon 簡單儲存服務 (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。這包括當您使用主控台、API 或 AWS 開發套件使用 Amazon Translate 或其他 AWS 服務時。AWS CLI 您輸入 Amazon Translate 或其他服務的任何資料都可能會被拾取，以便包含在診斷日誌中。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

對於使用 Amazon Translate 執行的批次翻譯任務，您的翻譯輸入和輸出都會在靜態時加密。但是，每種加密方法都不同。

Amazon Translate 也使用預設金鑰加密的亞馬遜彈性區塊存放區 (Amazon EBS) 磁碟區。

## 翻譯輸入

當您使用 Amazon Translate 批次翻譯文件時，您會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。使用此選項，每個物件都會使用 Amazon S3 管理的唯一金鑰加密。

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](#)。

## 翻譯輸出

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

如需 SSE-KMS 的詳細資訊，請參閱 Amazon 簡單儲存服務 [使用者指南中的使用 AWS Key Management Service \(SSE-KMS\) 使用伺服器端加密來保護資料](#)。

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 可以使用下列其中一種金鑰類型：

### AWS 受管金鑰

默認情況下，Amazon Translate 使用 AWS 受管金鑰。此類型的 KMS 金鑰是由您的帳戶建立 AWS 並儲存在您的帳戶中。但是，您無法自行管理此 KMS 金鑰。它僅由代表您管理和使用 AWS。

### 客戶受管金鑰

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

在您可以使用自己的 KMS 金鑰之前，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。如果您想要使用不同 AWS 帳戶中的 KMS 金鑰，您也必須更新中的金鑰原則 AWS KMS。如需詳細資訊，請參閱 [自訂加密的先決條件權限](#)。

執行批次翻譯工作時，您可以選擇使用客戶管理金鑰。如需詳細資訊，請參閱 [執行批次翻譯工作](#)。

## 傳輸中加密

為了加密傳輸中的資料，Amazon Translate 會將 TLS 1.2 與 AWS 憑證搭配使用。

## Amazon Translate 的 Identity and Access Management

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

### 主題



- [物件](#)
- [使用身分驗證](#)
- [使用政策管理存取權](#)
- [Amazon Translate 如何與 IAM 搭配使用](#)
- [Amazon Translate 基於身份的政策示例](#)
- [AWS Amazon Translate 的受管政策](#)
- [疑難排解 Amazon Translate 身分和存取](#)

## 物件

您使用 AWS Identity and Access Management (IAM) 的方式會有所不同，具體取決於您在 Amazon Translate 中所做的工作。

**服務使用者** — 如果您使用 Amazon Translate 服務完成工作，則管理員會為您提供所需的登入資料和許可。當您使用更多 Amazon Translate 功能完成工作時，您可能需要額外的許可。了解存取許可的管理方式可協助您向管理員請求正確的許可。如果您無法存取 Amazon Translate 中的功能，請參閱[疑難排解 Amazon Translate 身分和存取](#)。

**服務管理員** — 如果您負責公司的 Amazon 翻譯資源，您可能擁有完整的 Amazon Translate 存取權。您的任務是確定服務使用者應存取哪些 Amazon Translate 功能和資源。接著，您必須將請求提交給您的 IAM 管理員，來變更您服務使用者的許可。檢閱此頁面上的資訊，了解 IAM 的基本概念。若要進一步了解貴公司如何將 IAM 與 Amazon Translate 搭配使用，請參閱[Amazon Translate 如何與 IAM 搭配使用](#)。

**IAM 管理員** — 如果您是 IAM 管理員，您可能想要了解如何撰寫政策以管理 Amazon Translate 存取權限的詳細資訊。若要檢視您可以在 IAM 中使用的 Amazon Translate 身分型政策範例，請參閱[Amazon Translate 基於身份的政策示例](#)。

## 使用身分驗證

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

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

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

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

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

## AWS 帳戶 根使用者

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

## 聯合身分

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

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

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

## IAM 使用者和群組

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

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

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

## IAM 角色

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

使用暫時憑證的 IAM 角色在下列情況中非常有用：

- 聯合身分使用者存取 – 若要向聯合身分指派許可，請建立角色，並為角色定義許可。當聯合身分進行身分驗證時，該身分會與角色建立關聯，並獲授予由角色定義的許可。如需有關聯合角色的相關資訊，請參閱 [IAM 使用者指南](#)中的為第三方身分提供者建立角色。如果您使用 IAM Identity Center，則需要設定許可集。為控制身分驗證後可以存取的內容，IAM Identity Center 將許可集與 IAM 中的角色相關聯。如需有關許可集的資訊，請參閱 AWS IAM Identity Center 使用者指南中的[許可集](#)。
- 暫時 IAM 使用者許可 – IAM 使用者或角色可以擔任 IAM 角色來暫時針對特定任務採用不同的許可。
- 跨帳戶存取權 – 您可以使用 IAM 角色，允許不同帳戶中的某人 (信任的委託人) 存取您帳戶中的資源。角色是授予跨帳戶存取權的主要方式。但是，對於某些策略 AWS 服務，您可以將策略直接附加到資源 (而不是使用角色作為代理)。若要了解跨帳戶存取權角色和資源型政策間的差異，請參閱 IAM 使用者指南中的[IAM 角色與資源類型政策的差異](#)。
- 跨服務訪問 — 有些 AWS 服務 使用其他 AWS 服務功能。例如，當您在服務中進行呼叫時，該服務通常會在 Amazon EC2 中執行應用程式或將物件儲存在 Amazon Simple Storage Service (Amazon S3) 中。服務可能會使用呼叫主體的許可、使用服務角色或使用服務連結角色來執行此作業。
- 轉寄存取工作階段 (FAS) — 當您使用 IAM 使用者或角色在中執行動作時 AWS，您會被視為主體。使用某些服務時，您可能會執行某個動作，進而在不同服務中啟動另一個動作。FAS 會使用主體呼叫的權限 AWS 服務，並結合要求 AWS 服務 向下游服務發出要求。只有當服務收到需要與其 AWS 服務 他資源互動才能完成的請求時，才會發出 FAS 請求。在此情況下，您必須具有執行這兩個動作的許可。如需提出 FAS 請求時的政策詳細資訊，請參閱 [《轉發存取工作階段》](#)。
- 服務角色 – 服務角色是服務擔任的 [IAM 角色](#)，可代表您執行動作。IAM 管理員可以從 IAM 內建立、修改和刪除服務角色。如需更多資訊，請參閱 IAM 使用者指南中的[建立角色以委派許可給 AWS 服務](#)。

- 服務連結角色 — 服務連結角色是連結至 AWS 服務服務可以擔任代表您執行動作的角色。服務連結角色會顯示在您的中，AWS 帳戶且屬於服務所有。IAM 管理員可以檢視，但不能編輯服務連結角色的許可。
- 在 Amazon EC2 上執行的應用程式 — 您可以使用 IAM 角色來管理在 EC2 執行個體上執行的應用程式以及發出 AWS CLI 或 AWS API 請求的臨時登入資料。這是在 EC2 執行個體內儲存存取金鑰的較好方式。若要將 AWS 角色指派給 EC2 執行個體並提供給其所有應用程式，請建立連接至執行個體的執行個體設定檔。執行個體設定檔包含該角色，並且可讓 EC2 執行個體上執行的程式取得暫時憑證。如需更多資訊，請參閱 IAM 使用者指南中的[利用 IAM 角色來授予許可給 Amazon EC2 執行個體上執行的應用程式](#)。

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

## 使用政策管理存取權

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

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

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

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

## 身分型政策

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

身分型政策可進一步分類成內嵌政策或受管政策。內嵌政策會直接內嵌到單一使用者、群組或角色。受管理的策略是獨立策略，您可以將其附加到您的 AWS 帳戶。受管政策包括 AWS 受管政策和客戶管理

的策略。若要了解如何在受管政策及內嵌政策間選擇，請參閱 IAM 使用者指南中的[在受管政策和內嵌政策間選擇](#)。

## 資源型政策

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

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

## 存取控制清單 (ACL)

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

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

## 其他政策類型

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

- 許可界限 – 許可範圍是一種進階功能，可供您設定身分型政策能授予 IAM 實體 (IAM 使用者或角色) 的最大許可。您可以為實體設定許可界限。所產生的許可會是實體的身分型政策和其許可界限的交集。會在 Principal 欄位中指定使用者或角色的資源型政策則不會受到許可界限限制。所有這類政策中的明確拒絕都會覆寫該允許。如需許可範圍的更多相關資訊，請參閱 IAM 使用者指南中的[IAM 實體許可範圍](#)。
- 服務控制策略 (SCP) — SCP 是 JSON 策略，用於指定中組織或組織單位 (OU) 的最大權限。AWS Organizations 是一種用於分組和集中管理您企業擁有的多個 AWS 帳戶。若您啟用組織中的所有功能，您可以將服務控制策略 (SCP) 套用到任何或所有帳戶。SCP 限制成員帳戶中實體的權限，包括每個 AWS 帳戶根使用者帳戶。如需組織和 SCP 的更多相關資訊，請參閱 AWS Organizations 使用者指南中的[SCP 運作方式](#)。
- 工作階段政策 – 工作階段政策是一種進階政策，您可以在透過編寫程式的方式建立角色或聯合使用者的暫時工作階段時，作為參數傳遞。所產生工作階段的許可會是使用者或角色的身分型政策和工作

階段政策的交集。許可也可以來自資源型政策。所有這類政策中的明確拒絕都會覆寫該允許。如需更多資訊，請參閱 IAM 使用者指南中的 [工作階段政策](#)。

## 多種政策類型

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

## Amazon Translate 如何與 IAM 搭配使用

在您使用 IAM 管理 Amazon Translate 的存取權限之前，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 支持
<a href="#">身分型政策</a>	是
<a href="#">資源型政策</a>	否
<a href="#">政策動作</a>	是
<a href="#">政策資源</a>	是
<a href="#">政策條件索引鍵 (服務特定)</a>	是
<a href="#">ACL</a>	否
<a href="#">ABAC(政策中的標籤)</a>	部分
<a href="#">臨時憑證</a>	是
<a href="#">轉送存取工作階段 (FAS)</a>	是
<a href="#">服務角色</a>	是
<a href="#">服務連結角色</a>	否

若要深入瞭解 Amazon Translate 和其他 AWS 服務如何與大多數 IAM 功能搭配使用，請參閱 IAM 使用者指南中的可與 IAM 搭配使用的[AWS 服務](#)。

## Amazon Translate 基於身份的政策

支援身分型政策 是

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

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

### Amazon Translate 基於身份的政策示例

若要檢視 Amazon Translate 以身分識別為基礎的政策範例，請參閱。[Amazon Translate 基於身份的政策示例](#)

## Amazon Translate 中基於資源的政策

支援以資源基礎的政策 否

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

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

## 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哪些內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

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

政策會使用動作來授予執行相關聯動作的許可。

若要查看 Amazon Translate 動作清單，請參閱服務授權參考中 [由 Amazon Translate 定義的動作](#)。

Amazon Translate 中的政策動作會在動作前使用下列前置詞：

```
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。在原則中指定權限時，請使用授與最少權限的最佳作法。

若要檢視 Amazon Translate 以身分識別為基礎的政策範例，請參閱 [Amazon Translate 基於身份的政策示例](#)



## 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哪些內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

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

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

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

若要查看 Amazon Translate 資源類型及其 ARN 的清單，請參閱服務授權參考中 [由 Amazon Translate 定義的資源](#)。若要了解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 [指定策略中的資源](#)。

## Amazon Translate 的政策條件密鑰

支援服務特定政策條件金鑰 是

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

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

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

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

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

若要查看 Amazon Translate 條件金鑰清單，請參閱服務授權參考中的 [Amazon Translate 條件金鑰](#)。若要了解您可以使用條件金鑰的動作和資源，請參閱 [Amazon Translate 定義的動作](#)。

若要檢視 Amazon Translate 以身分識別為基礎的政策範例，請參閱 [Amazon Translate 基於身份的政策示例](#)

## Amazon Translate 中的 ACL

支援 ACL	否
--------	---

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

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

ABAC 在成長快速的環境中相當有幫助，並能在政策管理變得繁瑣時提供協助。

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

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

如需 ABAC 的詳細資訊，請參閱《IAM 使用者指南》中的 [什麼是 ABAC?](#)。若要查看含有設定 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

支援臨時憑證 是

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

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

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

## Amazon Translate 的轉寄存取工作階段

支援轉寄存取工作階段 (FAS) 是

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

## Amazon 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。如需更多資訊，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可能中斷 Amazon Translate 功能。只有在 Amazon Translate 提供指導時，才能編輯服務角色。

若要使用 Amazon 翻譯非同步操作，您必須授予 Amazon Translate 存取權限，存取包含輸入文件的 Amazon S3 儲存貯體。您可以透過信任政策在帳戶中建立服務角色，以信任 Amazon Translate 服務主體來執行此操作。

如需政策範例，請參閱「[批次翻譯工作的先決條件](#)」。

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

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

## Amazon Translate 基於身份的政策示例

依預設，使用者和角色沒有建立或修改 Amazon Translate 資源的權限。他們也無法使用 AWS Management Console、AWS CLI、或 AWS API 執行工作。IAM 管理員必須建立 IAM 政策，以授與權限，才能在所需的特定資源上執行特定 API 操作。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。

若要了解如何使用以下 JSON 政策文件範例建立 IAM 身分型政策，請參閱 IAM 使用者指南中的[在 JSON 索引標籤上建立政策](#)。

### 主題

- [以身分為基礎的原則最佳做法](#)
- [允許存取 Amazon Translate 主控台](#)
- [允許使用者檢視他們自己的許可](#)
- [指定策略中的資源](#)

- [使用具有自訂術語的客戶管理金鑰的權限](#)

## 以身分為基礎的原則最佳做法

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

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

## 允許存取 Amazon Translate 主控台

若要存取 Amazon Translate 主控台，您必須擁有最少一組許可。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。如果您建立的以身分識別為基礎的原則，其限制性高於最低所需權限，則控制台將無法如預期用於具有該原則的實體 (使用者、群組或角色) 運作。

對於 Amazon Translate 主控台許可，您可以將受 TranslateFullAccess AWS 管政策附加到實體。如需詳細資訊，請參閱 [AWS Amazon Translate 的受管政策](#)。

您還需要下列策略中顯示的動作的權限。這些權限包含在TranslateFullAccess策略中。

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

您不需要為僅對 AWS CLI 或 AWS API 進行呼叫的使用者允許最低主控台權限。反之，只需允許存取符合他們嘗試執行之 API 操作的動作就可以了。如需詳細資訊，請參閱《IAM 使用者指南》中的[新增許可到使用者](#)。

## 允許使用者檢視他們自己的許可

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

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupsForUser",
        "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": "*"
  }
]
}

```

## 指定策略中的資源

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](#)。您可以在策略中指定下列資源：

- 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策略中指定多個資源。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/*"
    ]
}
```

下列範例原則會拒絕存取GetParallelData動作的特定 parallel 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"
    ]
}
```

## 使用具有自訂術語的客戶管理金鑰的權限

如果您將 AWS Key Management Service (AWS KMS) 客戶受管金鑰與 Amazon Translate 自訂術語搭配使用，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作業，請將下列權限新增至您現有的 KMS 金鑰原則。

```
{
    "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` 作業，請在 KMS 金鑰原則中新增下列權限。

```

{
  "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": "*"
    }
  ]
}

```

若要針對使用客戶管理金鑰匯入的自訂術語呼叫 `ListTerminologies` 或 `DeleteTerminology` 作業，您不需要擁有任何特殊 AWS KMS 權限。

若要將客戶受管金鑰與所有自訂術語作業搭配使用，請在 KMS 金鑰原則中新增下列權限。

```

{
  "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、資源和條件金鑰](#)。

## AWS Amazon Translate 的受管政策

受 AWS 管理的策略是由建立和管理的獨立策略 AWS。AWS 受管理的策略旨在為許多常見使用案例提供權限，以便您可以開始將權限指派給使用者、群組和角色。

請記住，AWS 受管理的政策可能不會為您的特定使用案例授與最低權限權限，因為這些權限可供所有 AWS 客戶使用。我們建議您定義使用案例專屬的 [客戶管理政策](#)，以便進一步減少許可。

您無法變更受 AWS 管理策略中定義的權限。如果 AWS 更新 AWS 受管理原則中定義的權限，則此更新會影響附加原則的所有主體識別 (使用者、群組和角色)。AWS 當新的啟動或新 AWS 服務的 API 操作可用於現有服務時，最有可能更新 AWS 受管理策略。

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 受管理的策略：TranslateFullAccess](#)
- [AWS 受管理的策略：TranslateReadOnly](#)
- [AWS 受管政策的 Amazon Translate 更新](#)

## AWS 受管理的策略：TranslateFullAccess

此政策授予對 Amazon Translate 資源、Amazon Comprehend DetectDominantLanguage API 操作和所需 CloudWatch API 操作的完整存取權。該政策還授予列表並取得 Amazon S3 儲存貯體和 IAM 角色的許可。

```
{
  "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 受管理的策略：TranslateReadOnly

此政策授予存取 Amazon Translate API 操作的權限，這些操作不會修改與您帳戶相關聯的資源。此政策也授予存取 Amazon Comprehend DetectDominantLanguage API 操作和所需 CloudWatch API 操作的權限。

```
{
  "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": "*"
}
]
}

```

## AWS 受管政策的 Amazon Translate 更新

檢視有關 Amazon Translate AWS 受管政策更新的詳細資訊，因為此服務開始追蹤這些變更。如需有關此頁面變更的自動警示，請訂閱 Amazon Translate [文件歷史記錄](#) 頁面上的 RSS 摘要。

變更	描述	日期
<a href="#">TranslateReadOnly</a> – 更新現有政策	Amazon Translate 現在允許在 TranslateReadOnly 政策中 TranslateDocument 執行操作	2023 年 5 月 23 日
Amazon Translate 開始跟踪更改	Amazon Translate 開始追蹤其 AWS 受管政策的變更。	2023 年 5 月 23 日

## 疑難排解 Amazon Translate 身分和存取

使用下列資訊協助您診斷和修正使用 Amazon Translate 和 IAM 時可能會遇到的常見問題。

### 主題

- [我沒有授權在 Amazon Translate 中執行操作](#)
- [我沒有授權執行 iam : PassRole](#)

- [我想允許我以外的人訪 AWS 帳戶 問我的 Amazon Translate 資源](#)

## 我沒有授權在 Amazon Translate 中執行操作

如果您收到錯誤，告知您未獲授權執行動作，您的政策必須更新，允許您執行動作。

下列範例錯誤會在mateojackson 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 系統管理員。您的管理員提供您的簽署憑證。

## 我沒有授權執行 iam : PassRole

如果您收到未獲授權執行 `iam:PassRole` 動作的錯誤訊息，則必須更新您的政策以允許您將角色傳遞給 Amazon Translate。

有些 AWS 服務 允許您將現有角色傳遞給該服務，而不是建立新的服務角色或服務連結角色。如需執行此作業，您必須擁有將角色傳遞至該服務的許可。

當名為的 IAM 使用者marymajor嘗試使用主控台在 Amazon 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 沒有將角色傳遞至該服務的許可。

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

在這種情況下，Mary 的政策必須更新，允許她執行 `iam:PassRole` 動作。

如果您需要協助，請聯絡您的 AWS 系統管理員。您的管理員提供您的簽署憑證。

## 我想允許我以外的人訪 AWS 帳戶 問我的 Amazon 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](#)。
- 若要了解如何提供對您所擁有資源 AWS 帳戶的存取權，請參閱 [IAM 使用者指南中您擁有的另一 AWS 帳戶 個 IAM 使用者提供存取權限](#)。
- 若要了解如何將資源存取權提供給第三方 AWS 帳戶，請參閱 IAM 使用者指南中的[提供第三方 AWS 帳戶擁有的存取權](#)。
- 若要了解如何透過聯合身分提供存取權，請參閱 IAM 使用者指南中的[將存取權提供給在外部進行身分驗證的使用者 \(聯合身分\)](#)。
- 若要了解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 角色與資源型政策的差異](#)。

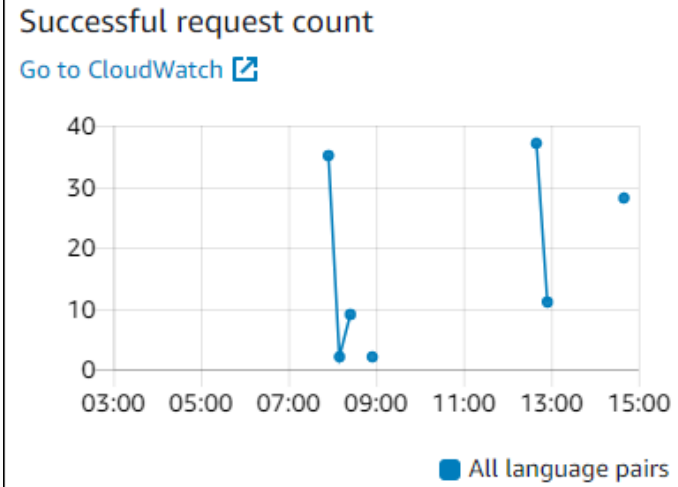
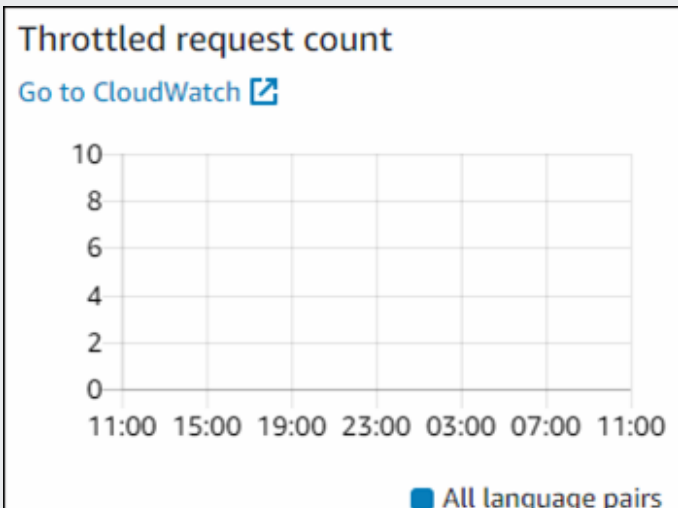
## 監控 Amazon Translate

監控是維持 Amazon Translate 和您的解決方案的可靠性、可用性和效能的重要組成部分。AWS 提供各種工具，您可以用來監控 Amazon Translate。您可以設定其中一些工具，為您監控解決方案。建議您盡量自動化監控任務。

Amazon Translate 提供預先設定的圖形，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。

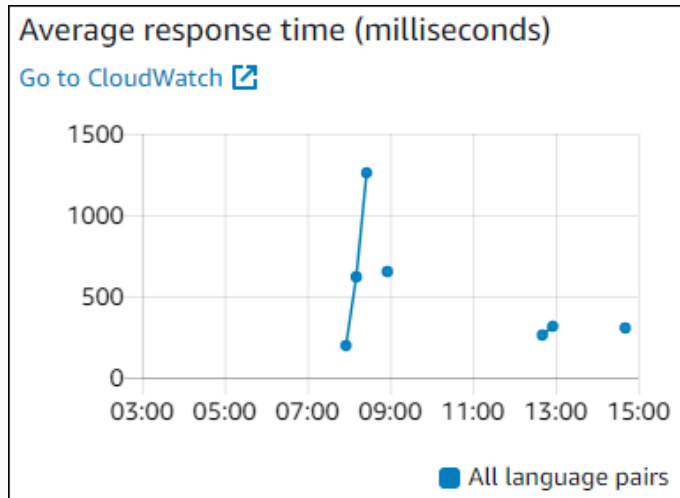
您還可以使用 Amazon CloudWatch 監控 Amazon Translate。使用 CloudWatch，您可以自動監控解決方案的特定指標。每當您設定的指標超出閾值，就會收到通知。您也可以使用 CloudWatch API 建立適合您需求的自訂監視應用程式。有關更多信息，請參閱 [Amazon 用 CloudWatch 戶指南 CloudWatch 中的 Amazon 是什麼](#)。

下表說明 Amazon Translate 提供的每個預先設定的圖形。

圖表	描述
<p><b>Successful request count</b></p> <p><a href="#">Go to CloudWatch</a></p>  <p>■ All language pairs</p>	<p><b>成功請求計數</b></p> <p>指定期間內成功向 Amazon Translate 發出的要求數目。</p>
<p><b>Throttled request count</b></p> <p><a href="#">Go to CloudWatch</a></p>  <p>■ All language pairs</p>	<p><b>已調節請求計數</b></p> <p>在指定時間範圍內限制的 Amazon Translate 請求數目。使用此資訊來判斷您的應用程式是否將請求傳送到 Amazon Translate 的速度過快。</p>

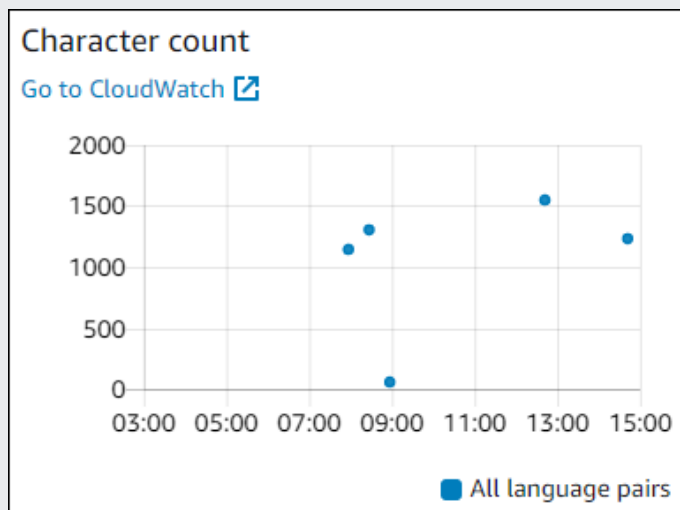
## 圖表

## 描述



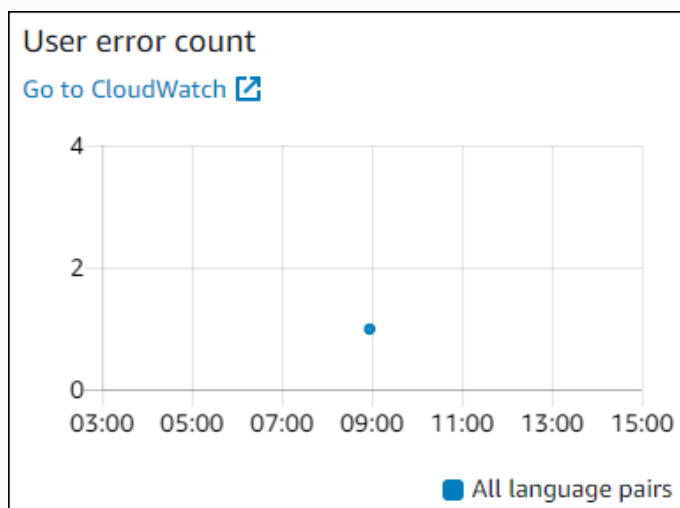
## 平均回應時間

Amazon Translate 在指定時間段內處理您的請求所花費的平均時間長度。



## 字元計數

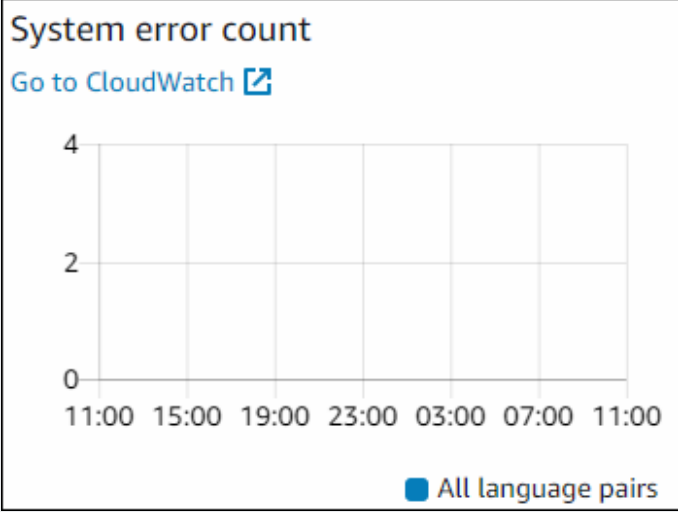
您在指定時間範圍內傳送至 Amazon Translate 的字元總數。這是您將付費的字元數。



## 使用者錯誤計數

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。



圖表	描述
 <p>The chart, titled 'System error count', shows a line graph for 'All language pairs'. The y-axis represents the number of errors, ranging from 0 to 4. The x-axis shows time intervals: 11:00, 15:00, 19:00, 23:00, 03:00, 07:00, and 11:00. The data points for all intervals are at 0. A link 'Go to CloudWatch' is visible at the top left of the chart area.</p>	<p><b>系統錯誤計數</b></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，您可以為您的帳戶取得個別 Amazon Translate 操作或全球 Amazon Translate 指標的指標。使用指標追蹤 Amazon Translate 解決方案的運作狀態，並設定警示，以在一個或多個指標超出定義的閾值時通知您。例如，您可以監控在特定時段內向 Amazon Translate 發出的請求數量、查看請求延遲，或在錯誤超過閾值時發出警示。

### 了解 CloudWatch Amazon Translate 的指標

若要取得 Amazon Translate 作業的指標，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 [CloudWatch Amazon Translate 指標](#)。

您可以使用 AWS Management Console、或 CloudWatch API 取得 Amazon Translate 的指標。AWS CLI 您可以透過其中 CloudWatch 一個 Amazon AWS 軟體開發套件 (開發套件) 或 CloudWatch API 工具使用 API。

下表列出 CloudWatch 測量結果的一些常見用法。這些是協助您開始的建議，而不是完整清單。

運作方式？	監視此測量結果
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 許可才能監控 Amazon Translate 如 CloudWatch 需詳細資訊，請參閱 Amazon CloudWatch 使用者指南 CloudWatch 中的 [Amazon 身份驗證和存取控制](#)。

## 檢視 Amazon Translate 指標

在 CloudWatch 主控台中檢視 Amazon Translate 指標。

若要檢視量度 (CloudWatch 主控台)

1. 請登入 AWS Management Console 並開啟 CloudWatch 主控台，網址為 <https://console.aws.amazon.com/cloudwatch/>。
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 與這項服務整合在一起 AWS CloudTrail，該服務可提供 IAM 實體或 AWS 服務在 Amazon Translate 中所採取的操作記錄。CloudTrail 以事件形式擷取 Amazon Translate 的所有 API 呼叫。這包括來自 Amazon Translate 主控台的呼叫，以及對 Amazon Translate API 操作的程式碼呼叫。如果您建立 CloudTrail 追蹤，您可以啟用將 CloudTrail 事件 (包括 Amazon Translate 的事件) 持續交付到 Amazon Simple Storage Service (Amazon S3) 儲存貯體。如果您未設定追蹤，您仍然可以在 [事件歷程記錄] 中檢視 CloudTrail 主控台中最近的事件。您可以使用收集 CloudTrail 到的資訊來判

斷向 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 用者指南](#)。

## 主題

- [Amazon Translate 信息 CloudTrail](#)
- [了解 Amazon Translate 日誌文件條目](#)

## Amazon Translate 信息 CloudTrail

CloudTrail 在您創建 AWS 帳戶時，您的帳戶已啟用。在 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 CloudTrail 事件一起記錄在事件中。您可以在帳戶中查看，搜索和下載最近的事 AWS 件。如需詳細資訊，請參閱[檢視具有事 CloudTrail 件記錄的事件](#)。

如需 AWS 帳戶中持續的事件記錄 (包括 Amazon Translate 的事件)，請建立追蹤。追蹤可 CloudTrail 將日誌檔交付到 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 CloudTrail 記錄中收集的事件資料並採取行動。如需詳細資訊，請參閱下列內容：

- [建立追蹤的概觀](#)
- [CloudTrail 支援的服務與整合](#)
- [設定 Amazon SNS 通知 CloudTrail](#)
- [從多個 tegions 接收 CloudTrail 日誌文件，並從多個 CloudTrail 帳戶接收日誌文件](#)

所有 Amazon Translate 動作都會記錄在 API 參考部分中，CloudTrail 並將其記錄在 [API 參考部分](#)中。例如，呼叫 `ImportTerminology` 和 `TranslateText` 動作會 `DeleteTerminology` 在 CloudTrail 記錄檔中產生項目。

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 root 使用者認證提出要求
- 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 使 userIdentity 元素](#)。

## 了解 Amazon Translate 日誌文件條目

追蹤是一種組態，可讓事件以日誌檔的形式傳遞到您指定的 Amazon S3 儲存貯體。CloudTrail 記錄檔包含一或多個記錄項目。事件代表來自任何來源的單一請求，包括有關請求的操作，動作的日期和時間，請求參數等信息。CloudTrail 日誌文件不是公共 API 調用的有序堆棧跟踪，因此它們不會以任何特定順序顯示。

下列範例顯示示範 TranslateText 動作的 CloudTrail 記錄項目。

```
{
  "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"
}
```

## CloudWatch Amazon Translate 的指標和維度

若要監控解決方案的效能，請使用 Amazon 翻譯的 Amazon CloudWatch 指標和維度。

### CloudWatch Amazon Translate 指標

指標	描述
CharacterCount	<p>請求中的可計費字元數。</p> <p>有效維度：語言配對、作業</p> <p>有效統計資料：平均值、最大值、最小值、總和</p> <p>單位：計數</p>
ResponseTime	<p>回應要求所花費的時間。</p> <p>有效維度：語言配對、作業</p> <p>有效統計資料：資料樣本、平均值</p> <p>單位：對於數據樣本，計數。對於平均統計值，毫秒。</p>
ServerErrorCount	<p>伺服器錯誤的次數。伺服器錯誤的 HTTP 回應碼範圍是 500 到 599。</p> <p>有效尺寸:操作</p> <p>有效的統計資訊：平均、總和</p> <p>單位：計數</p>
SuccessfulRequestCount	<p>成功翻譯要求的數目。一個成功的請求的響應代碼是 200 到 299。</p> <p>有效尺寸:操作</p> <p>有效的統計資訊：平均、總和</p> <p>單位：計數</p>
ThrottledCount	<p>受限制的要求數目。用ThrottledCount 於判斷您的應用程式傳送請求到 Amazon Translate 的速度是否比帳戶設定為接受請求的速度更</p>

指標	描述
	<p>快。如需詳細資訊，請參閱 <a href="#">Amazon Translate</a> 參閱 Amazon Web Services 一般參考。</p> <p>有效尺寸: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>有效尺寸:操作</p> <p>有效的統計資訊：平均、總和</p> <p>單位：計數</p>

## CloudWatch Amazon Translate 的維度

使用下列維度篩選 Amazon Translate 指標。量度會依來源語言和目標語言分組。

維度	描述
LanguagePair	將量度限制為僅包含指定語言的量度。
Operation	將量度限制為只有具有指定作業的量度。

## 使用 Amazon 監控 Amazon Translate 事 EventBridge

Amazon Translate 與 Amazon EventBridge 整合，可通知您有關影響翻譯任務和 parallel 資料資源的變更。來自 AWS 服務的事件會以近乎即時 EventBridge 的方式傳送到。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，可以自動啟動的動作包括：

- 調用函數 AWS Lambda
- 調用 AWS Systems Manager 運行命令
- 將事件轉傳至 Amazon Kinesis Data Streams

- 啟動 AWS Step Functions 狀態機
- 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針對事件做出回應的 Amazon EventBridge 規則](#)。

## Amazon Translate 事件

以下是來自 Amazon Translate 的示例事件。

### 批次翻譯工作的事件

您可以使用 Amazon 翻譯主控台或 [StartTextTranslationJob](#) 操作來執行批次翻譯任務。Amazon Translate 會在這些任務成功完成或失敗時傳送事件。這些事件類似於下列範例。

```
{
  "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 傳送事件的任務狀態。這些 jobStatus 值為：

- COMPLETED— 工作已成功完成且輸出可用。
- COMPLETED\_WITH\_ERROR— 工作已完成，但發生錯誤。可以在工作的輸出中分析錯誤。
- STOPPED— 工作已停止。
- FAILED— 工作未完成。若要取得詳細資訊，請使用 [DescribeTextTranslationJob](#) 作業。

## parallel 資料資源的事件

當您使用 Amazon Translate 建立或更新 parallel 資料資源時，它會傳送事件以指出作業是成功還是失敗。

您可以使用 Amazon Translate 主控台或[CreateParallelData](#)操作來建立 parallel 資料資源。當您這樣做時，Amazon Translate 會發送類似以下的事件。

```
{
  "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"
  }
}
```

屬status性的值為：

- ACTIVE— 作CreateParallelData 業成功，資源已準備好供您使用。
- FAILED— 作CreateParallelData業失敗。

您可以使用 Amazon Translate 主控台或[UpdateParallelData](#)操作來更新 parallel 資料資源。當您這樣做時，Amazon Translate 會發送類似以下的事件。

```
{
  "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 屬性提供 parallel 資料資源的先前版本狀態，此資源正在被更新所取代。數值為：

- ACTIVE— 已成功建立或更新先前版本。
- FAILED— 無法建立或更新先前版本。

latestUpdateAttemptStatus 屬性提供 parallel 資料資源的新版本狀態，此資源正在由更新建立。數值為：

- ACTIVE— UpdateParallelData 作業成功，且已更新的資源已準備好供您使用。
- FAILED— 作UpdateParallelData業失敗。

## Amazon Translate 的合規驗證

第三方稽核人員會評估 Amazon Translate 的安全性和合規性，作為多個 AWS 合規計劃的一部分。這些計劃包括 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](#)。

使用 Amazon Translate 時的合規責任取決於資料的敏感度、公司的合規目標以及適用的法律和法規。AWS 提供下列資源以協助遵循法規：

- [安全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。
- [建構 HIPAA 安全性與合規性白皮書 — 本白皮書](#) 說明公司如何使用建立符合 HIPAA 標準的應用 AWS 程式。
- [AWS 合規資源](#) — 此工作簿和指南集合可能適用於您的產業和所在地。
- [AWS Config](#) — 此 AWS 服務評估您的資源配置是否符合內部實踐，行業準則和法規。

- [AWS Security Hub](#)— 此 AWS 服務提供安全狀態的全面檢視，協助您檢查您 AWS 是否符合安全性產業標準和最佳做法。

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

## Amazon Translate 的彈性

AWS 全球基礎架構是圍繞區 AWS 域和可用區域建立的。AWS 區域提供多個實體分離和隔離的可用區域，這些區域透過低延遲、高輸送量和高度備援的網路連線。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。

如需區域和可用區域的相關 AWS 資訊，請參閱[AWS 全域基礎結構](#)。

## Amazon Translate 基礎設施安全

作為受管服務，Amazon Translate 受到[Amazon 網路服務：安 AWS 全流程概觀白皮書中所述的全球網路安全程序](#)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，請求必須使用存取金鑰 ID 和與 AWS Identity and Access Management (IAM) 主體相關聯的秘密存取金鑰來簽署。或者，您可以透過[AWS Security Token Service \(AWS STS\)](#) 來產生暫時安全憑證來簽署請求。

## Amazon Translate 和接口 VPC 端點 ( )AWS PrivateLink

您可以透過建立介面 VPC 端點，在 VPC 和 Amazon Translate 之間建立私有連線。介面端點採用這種技術[AWS PrivateLink](#)，可讓您在沒有網際網路閘道、NAT 裝置、VPN 連線或 AWS 直 Connect 連線的情況下私有存取 Amazon Translate API。虛擬私人雲端中的執行個體不需要公有 IP 地址即可與 Amazon Translate API 進行通訊。您的 VPC 和 Amazon Translate 之間的流量不會離開 Amazon 網絡。

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虛擬私人雲端端點 ([AWS PrivateLink](#))。

## 亞馬遜 VPC 端點翻譯的注意事項

在為 Amazon Translate 設定介面虛擬私人雲端端點之前，請務必先查看 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 創建接口 VPC 端點

您可以使用 Amazon VPC 主控台或 AWS Command Line Interface (AWS CLI) 為 Amazon Translate 服務建立 VPC 端點。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，則可以使用該區域的預設 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

您可以將端點政策附加到 VPC 端點，以控制對 Amazon 翻譯的存取。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：適用於 Amazon Translate 翻譯即時翻譯動作的 VPC 端點政策

以下是 Amazon Translate 翻譯中即時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列出之 Amazon Translate 動作的存取權。

```
{
```

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

#### 範例：Amazon Translate 翻譯批次翻譯動作的 VPC 擬私人雲端端點政策

以下是 Amazon Translate 翻譯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列出之 Amazon Translate 動作的存取權。

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

# 指引及配額

以下各節包含 Amazon 翻譯準則和配額的相關資訊。

## 主題

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

## 支援的 AWS 區域

對於列表AWS支援 Amazon 翻譯的區域，請參閱[亞馬遜翻譯端點和配額](#)在AWS一般參考。

## 合規

如需 Amazon 翻譯合規計劃的詳細資訊，請參閱[AWS合規](#)、[AWS合規計劃](#)，以及[AWS合規計劃範圍內的服務](#)。

## 限流

亞馬遜翻譯擴展以服務客戶的操作流量。如果您持續遇到調節的情況，請聯絡 [AWS Support](#)。

## 指導方針

為了持續改善其分析模型的品質，Amazon Translate 可能會儲存您的資料。若要深入瞭解，請參閱[亞馬遜翻譯問題](#)。

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

亞馬遜翻譯有下列服務準則和配額。

## 同步即時翻譯配額

描述	限制
字元編碼	UTF-8
最大輸入文字	10,000 位元組
每個文件的最大字元數	100,000
最大文件大小	100,000 位元組

## 非同步批次翻譯配額

描述	限制
字元編碼	UTF-8
每個文件的最大字元數	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
批次工作要求中的目標語言數目上限	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
並行批次轉譯工作數目上限	10
排入佇列的批次轉譯工作數目上限	1000
每秒的交易StartTextTranslationJobAPI 動作	5
每秒的交易DescribeTextTranslationJobAPI 動作	10
每秒的交易ListTextTranslationJobsAPI 動作	10
每秒的交易StopTextTranslationJobAPI 動作	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
每秒的交易ImportTerminologyAPI 動作	5
每秒的交易GetTerminologyAPI 動作	10
每秒的交易ListTerminologiesAPI 動作	10
每秒的交易DeleteTerminologyAPI 動作	5

## 並行資料配額

描述	限制
每個 AWS 區域每個 AWS 帳戶的最大平行資料資源數	1000
最大平行資料輸入檔案大小	5 GB
平行資料資源中的來源語言數目上限	1
平行資料輸入檔案中單一區段或記錄的大小上限	千字節
平行資料資源的並行建立或更新作業數目上限	1
每秒的交易CreateParallelDataAPI 動作	5
每秒的交易GetParallelDataAPI 動作	10

描述	限制
每秒的交易ListParallelDataAPI 動作	10
每秒的交易UpdateParallelDataAPI 動作	5
每秒的交易DeleteParallelDataAPI 動作	5



# Amazon Translate 的文檔歷史

下表說明此版本的 Amazon Translate 文件。

變更	描述	日期
<a href="#">新功能：簡潔</a>	Translate 現在支持簡潔的實時文本翻譯。簡潔度減少了大多數翻譯的翻譯輸出長度（與翻譯輸出相比，沒有簡潔性）。如需詳細資訊，請參閱 <a href="#">在 Amazon Translate 中使用簡潔性</a> 。	2023 年 10 月 31 日
<a href="#">文件輸入至即時翻譯的語言自動偵測</a>	當您將文件輸入即時翻譯 (主控台或 API) 時，您現在可以使用語言自動偵測功能。如需詳細資訊，請參閱 <a href="#">即時翻譯</a> 。	2023 年 8 月 3 日
<a href="#">文字 (.docx) 文件作為輸入到實時翻譯</a>	您現在可以使用 .docx 檔案 (除了文字檔案和 HTML 檔案) 做為即時翻譯 (主控台或 API) 的輸入。如需詳細資訊，請參閱 <a href="#">即時翻譯</a> 。	2023 年 7 月 17 日
<a href="#">自訂術語的增強功能</a>	Translate 現在支援自訂術語功能的增強功能，可改善翻譯流暢度和準確度。如需詳細資訊，請參閱 <a href="#">使用自訂術語自訂翻譯</a> 。	2023 年 6 月 30 日
<a href="#">文本或 HTML 文件作為輸入到實時翻譯</a>	您現在可以使用文字檔案或 HTML 檔案做為即時翻譯 (主控台或 API) 的輸入。如需詳細資訊，請參閱 <a href="#">即時翻譯</a> 。	2023 年 5 月 23 日
<a href="#">TranslateReadOnly 策略中允許的新處理行動</a>	Amazon Translate 現在允許在 TranslateReadOnly 受管政策	2023 年 5 月 23 日

	<p>中TranslateDocument 執行此動作。如需詳細資訊，請參閱 <a href="#">AWS 受管政策：TranslateReadOnly</a>。</p>	
<a href="#">Translate 現在支援其他區域進行非同步批次處理。</a>	<p>Translate 現在支援其他區域進行非同步批次處理。如需詳細資訊，請參閱<a href="#">使用 Amazon Translate 進行非同步批次處理</a>。</p>	2023 年 3 月 28 日
<a href="#">增加即時翻譯的輸入大小</a>	<p>您現在最多可以輸入 10,000 個字元以進行即時翻譯。如需詳細資訊，請參閱 <a href="#">Amazon Translate 入門 (主控台)</a>。</p>	2022 年 12 月 16 日
<a href="#">Support 批處理模式的嵌套輸入文件夾</a>	<p>您現在可以為批次翻譯工作提供巢狀輸入資料夾。如需詳細資訊，請參閱在 Amazon Translate <a href="#">翻譯中執行批次翻譯任務</a>。</p>	2022 年 11 月 18 日
<a href="#">Support 批處理模式的自動語言檢測</a>	<p>您現在可以在批次翻譯工作中自動偵測來源語言。因此，您現在可以在批次翻譯工作中輸入不同來源語言的文件。如需詳細資訊，請參閱在 Amazon Translate <a href="#">翻譯中執行批次翻譯任務</a>。</p>	2022 年 11 月 18 日
<a href="#">Support 多種目標語言</a>	<p>您現在可以在批次轉譯工作中指定多種目標語言。如需詳細資訊，請參閱在 Amazon Translate <a href="#">翻譯中執行批次翻譯任務</a>。</p>	2022 年 10 月 10 日

[Support 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

[對其他語言的形式支持](#)

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形式等級。如需詳細資訊，請參閱 [在 Amazon Translate 中設定形式](#)。

2022 年 10 月 5 日

[單獨的 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形式層級。如需詳細資訊，請參閱 [在 Amazon Translate 中設定形式](#)。

2022 年 2 月 22 日

[新功能](#)

您現在可以在翻譯輸出中遮罩褻瀆的字詞和片語。如需詳細資訊，請參閱 [在 Amazon Translate 中遮蔽褻瀆的字詞和片語](#)。

2021 年 11 月 24 日

[AWS PrivateLink 支援](#)

您現在可以使 AWS PrivateLink 用 VPC 和 Amazon Translate 之間建立私人連線。如需詳細資訊，請參閱 [Amazon Translate 和介面 VPC 人雲端端點 \(AWS PrivateLink\)](#)。

2021 年 11 月 24 日

<a href="#">並行資料更新</a>	您現在可以建立使用 Amazon Translate 支援的任何語言的 parallel 資料資源。您不再需要使用英語作為其中一種語言。如需 parallel 資料的詳細資訊，請參閱 <a href="#">使用 parallel 資料自訂翻譯 (使用中自訂翻譯)</a> 。	2021 年 11 月 15 日
<a href="#">自訂術語方向性</a>	您現在可以建立多方向術語，其中任何語言都可以是來源語言或目標語言。如需詳細資訊，請參閱 <a href="#">建立自訂術語</a> 。	2021 年 11 月 11 日
<a href="#">新的語言</a>	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) 和旁遮普文。如需 Amazon Translate 支援的所有語言，請參閱 <a href="#">支援的語言和語言代碼</a> 。	2021 年 11 月 10 日
<a href="#">新的自訂加密設定</a>	您現在可以使用自己管理的客戶管理金鑰來加密翻譯輸出 AWS Key Management Service。如需詳細資訊，請參閱 <a href="#">執行批次轉譯工作</a>	2021 年 11 月 5 日
<a href="#">新的檔案格式支援</a>	Amazon Translate 現在支援用於非同步批次處理的 XML 本地化交換檔案格式 (XLIFF) 檔案。如需所有支援的格式，請參閱 <a href="#">支援的檔案格式</a> 。	2021 年 6 月 9 日

## [EventBridge 整合](#)

Amazon Translate 現在會將事件傳送 EventBridge 至 Amazon，以通知您有關影響翻譯任務和 parallel 資料資源的變更。如需詳細資訊，請參閱[使用 Amazon 監控 Amazon Translate 事件 EventBridge](#)。

2021 年 6 月 4 日

## [新配額](#)

Amazon Translate 現在支援多達 1000 個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

## [名額增加](#)

parallel 資料輸入檔案的大小上限從 1 MB 增加到 5 MB。如需所有 Amazon Translate 配額，請參閱[指導方針和限制](#)。

2021 年 3 月 31 日

## [新的語言](#)

Amazon Translate 現在支援下列語言：亞美尼亞語、加泰羅尼亞語、波斯語、菲律賓他加祿語、古吉拉特語、海地克里奧爾語、冰島語、卡納達語、哈薩克語、立陶宛語、馬來亞拉姆語、馬耳他語、蒙古語、僧伽羅語、泰盧固語、烏茲別克和威爾士語。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 parallel 資料 (包含來源文字及其翻譯範例所組成) 來自訂批次轉譯工作。使用 parallel 資料的工作稱為「使用中自訂翻譯」工作。在這些任務期間，Amazon Translate 會調整翻譯輸出，以反映 parallel 資料中的範例。如需詳細資訊，請參閱[使用 parallel 資料自訂翻譯 \(使用中自訂翻譯\)](#)。

2020 年 11 月 23 日

## [新的檔案格式支援](#)

Amazon Translate 現在支援下列辦公室開啟 XML 檔案格式作為非同步批次處理的輸入：Word 文件 (.docx)、PowerPoint 簡報 (.pptx)、Excel 活頁簿 (.xlsx)。如需詳細資訊，請參閱[開始批次轉譯工作](#)。

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) 區域的非同步批次處理。如需提供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 新增了對亞太區域 (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 支援的 AWS 區域完整清單，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 為翻譯增加了新語言：南非荷蘭語，阿爾巴尼亞語，阿姆哈拉語，阿塞拜疆語，孟加拉語，波斯尼亞語，保加利亞語，加拿大法語，克羅地亞語，達里語，愛沙尼亞語，格魯吉亞語，豪薩語，拉脫維亞語，帕什圖語，塞爾維亞語，斯洛伐克語，斯洛文尼亞語，索馬里語，斯瓦希里語，菲律賓語和泰米爾語。如需 Amazon 翻譯可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 新增翻譯語言：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 翻譯可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 合規性](#)。如需詳細資訊，請參閱 [合規](#)。

2019 年 7 月 31 日

<a href="#">新功能</a>	Amazon Translate 新增 <a href="#">SOC 合規性</a> 。如需詳細資訊，請參閱 <a href="#">合規</a> 。	2019 年 5 月 30 日
<a href="#">新區域</a>	Amazon Translate 新增了對亞太區域 (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 支援的 AWS 區域完整清單，請參閱 Amazon Web Services 一般參考中的 <a href="#">AWS 區域表或 AWS 區域和端點</a> 。	2019 年 5 月 8 日
<a href="#">新的語言</a>	Amazon Translate 為翻譯增加了新的翻譯語言：印地文、馬來文、挪威文和波斯文。如需 Amazon 翻譯可直接翻譯的語言組合清單，請參閱 <a href="#">支援的語言</a> 。	2019 年 5 月 6 日
<a href="#">新區域</a>	Amazon Translate 新增了對歐洲 (法蘭克福) 和亞太區域 (首爾) 區域的支援。如需 Amazon Translate 支援的 AWS 區域完整清單，請參閱 Amazon Web Services 一般參考中的 <a href="#">AWS 區域表或 AWS 區域和端點</a> 。	2019 年 2 月 28 日
<a href="#">新功能</a>	Amazon Translate 新增 <a href="#">PCI 合規性</a> 。如需詳細資訊，請參閱 <a href="#">合規</a> 。	2018 年 12 月 12 日



## [新功能](#)

Amazon Translate 新增了四個新的 API 和自訂術語功能，可讓您更好地控制翻譯。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 現在可以翻譯下列語言的文件：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 透過大幅減少不支援的語言配對數量，持續改善直接翻譯。如需 Amazon 翻譯可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 翻譯在支援的語言之間新增了直接翻譯，而不是英文。如需 Amazon 翻譯可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 新增了[HIPAA](#) 合規性。如需詳細資訊，請參閱[合規](#)。

2018 年 10 月 25 日

## [新功能](#)

Amazon Translate 新增多種翻譯語言：中文 (繁體)、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 新增了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 參考現在是一個單獨的文檔。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。