



Guide du développeur

Amazon Transcribe



Amazon Transcribe: Guide du développeur

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

Les marques et la présentation commerciale d'Amazon ne peuvent être utilisées en relation avec un produit ou un service qui n'est pas d'Amazon, d'une manière susceptible de créer une confusion parmi les clients, ou d'une manière qui dénigre ou discrédite Amazon. Toutes les autres marques commerciales qui ne sont pas la propriété d'Amazon appartiennent à leurs propriétaires respectifs, qui peuvent ou non être affiliés ou connectés à Amazon, ou sponsorisés par Amazon.

Table of Contents

Qu'est-ce que Amazon Transcribe ?	1
Amazon Transcribe et éligibilité HIPAA	1
Tarification	2
Disponibilité des régions et quotas	2
Fonctionnalités disponibles	4
Langues prises en charge	7
Langages de programmation pris en charge	15
Jeu de caractères	17
Abkhaze	20
Afrikaans	23
Arabe	23
Asturien	24
Azerbaïdjanais	25
Arménien	26
Bachkir	27
Basque	29
Biélorusse	30
Bengali	31
Bosniaque	33
Bulgare	34
Catalan	35
Kurde sorani	35
Chinois simplifié	37
Chinois, traditionnel	38
Croate	39
Tchèque	39
Danois	40
Néerlandais	41
Anglais	42
Estonien	42
Farsi	43
Finnois	44
Français	45
Galicien	46

Géorgien	46
Allemand	48
Grec	48
Gujarati	50
Haoussa	52
Hébreu	52
Hindi	53
Hongrois	55
Islandais	56
Indonésien	57
Italien	57
Japonais	58
Kabyle	58
Kannada	59
Kazakh	61
Kinyarwanda	62
Coréen	63
kirghize	63
Letton	66
Lituanien	66
Luganda	67
Macédonien	68
Malais	70
Malayalam	70
Maltais	72
Marathi	73
Mari des prairies	75
Mongol	77
Norvégien (bokmål)	80
Odia/Oriya	80
Pachto	82
Polonais	84
Portugais	85
Pendjabi	86
Roumain	88
Russe	89

Serbe	90
Singhalais	92
Slovaque	94
Slovène	95
Somali	96
Espagnol	96
Soundanais	97
Swahili	98
Suédois	98
Tagalog/Philippin	99
Tamoul	99
Tatar	101
Télougou	103
Thaï	105
Turc	107
Ukrainien	108
Ouïghour	109
Ouzbek	112
Vietnamien	113
Gallois	117
Wolof	118
Zoulou	119
Comment ça marche	120
Entrées et sorties de données	121
Formats multimédias	121
Canaux audio	122
Taux d'échantillonnage	123
Sortie	123
Transcrire des numéros	126
Démarrer	132
S'inscrire à unCompte AWS	133
Installation des kitsAWS CLI de développement logiciel et	133
Configurer lesIAM informations d'identification	134
Création d'unAmazon S3 bucket	134
Création d'une stratégie IAM	135
Transcrire avec leAWS Management Console	137

Transcrire à l'aide du AWS CLI	147
Commencer un nouveau travail de transcription	147
Obtention du statut d'une tâche d'	149
Répertorier vos tâches de transcription	150
Supprimer votre tâche de transcription	151
Transcription avec les SDK AWS	151
Utilisation des AWS SDK	163
Transcription avec HTTP ou WebSockets	164
Transcription de flux en streaming	167
Bonnes pratiques	168
Streaming et résultats partiels	169
Stabilisation des résultats partiels	170
Configuration d'une transcription en streaming	175
Encodage de flux d'événements	189
Images de données	192
Mise en file d'attente des offres d'emploi	193
Activer des tâches	193
Étiquetage des ressources	198
Contrôle d'accès basé sur les étiquettes	199
Ajout de balises à vos Amazon Transcribe ressources	200
Partitionnement des locuteurs (diarisation)	204
Partitionnement des locuteurs dans une transcription par lots	205
Partitionnement des locuteurs dans une transcription en streaming	208
Exemple de sortie	211
Transcription audio multicanal	217
Utilisation de l'identification des canaux dans une transcription par lots	218
Utilisation de l'identification des chaînes dans une transcription en continu	222
Exemple de sortie	223
Identification des langues	231
Identification des langues par lots	231
Identification des langues dans un audio multilingue	232
Amélioration de la précision de l'identification des langues	233
Combinaison de l'identification des langues avec d'autres fonctionnalités Amazon Transcribe	234
Utilisation de l'identification des langues avec des transcriptions par lots	235
Identification des langues en streaming	242

Identification des langues dans un audio multilingue	243
Utilisation de l'identification des langues avec les médias en streaming	243
Transcriptions alternatives	250
Demande de transcriptions alternatives	252
Améliorer la précision de Transcription	257
Vocabulaires personnalisés	258
Tableaux de vocabulaires personnalisés et listes de vocabulaires personnalisés	260
Création d'un vocabulaire personnalisé à l'aide d'un tableau	260
Création d'un vocabulaire personnalisé à l'aide d'une liste	272
Utilisation d'un vocabulaire personnalisé	275
Modèles de langage personnalisés	282
Sources de données	282
Données d'entraînement ou de réglage	283
Création d'un modèle de langage personnalisé	284
Utilisation d'un modèle de langage personnalisé	291
Filtrer les mots	299
Création d'un filtre de vocabulaire	300
Création de filtres de vocabulaire personnalisés	301
Utiliser un filtre de glossaire personnalisé	306
Utilisation d'un filtre de vocabulaire personnalisé dans une transcription par lots	275
Utilisation d'un filtre de vocabulaire personnalisé dans une transcription en streaming	279
Détecter les discours toxiques	316
Utilisation de la détection de discours toxiques	317
Utilisation de la détection de discours toxiques dans une transcription par lots	317
Exemple de sortie	322
Rédaction de transcriptions	324
Expurgation des PII (données d'identification personnelle) dans votre tâche par lots	325
Expurgation ou identification des données d'identification personnelle (PII) dans un flux en temps réel	332
Exemple de sortie	338
Exemple de sortie rédigée (batch)	338
Exemple de sortie de streaming expurgée	341
Exemple de sortie d'identification PII	342
sous-titrage	345
Génération de fichiers de sous-titres	346
Analyse de l'audio du centre d'appels	350

Cas d'utilisation courants	350
Considérations et informations supplémentaires	352
Disponibilité des régions et quotas	353
Analyse après appel	354
Informations après appel	354
Création de catégories	358
Démarrage d'une transcription	370
Résultats d'analyse après appel	380
Activation du résumé génératif des appels	393
Analyse Call Analytics en temps réel	398
Informations en temps réel	399
Création de catégories	401
Analyses après appel avec transcriptions en temps réel	408
Démarrage d'une transcription	416
Sortie Call Analytics en temps réel	425
Transcription d'Amazon Chimeappels	431
Exemples de code	433
Actions	434
CreateVocabulary	435
DeleteMedicalTranscriptionJob	438
DeleteTranscriptionJob	441
DeleteVocabulary	445
GetTranscriptionJob	447
GetVocabulary	450
ListMedicalTranscriptionJobs	453
ListTranscriptionJobs	458
ListVocabularies	465
StartMedicalTranscriptionJob	469
StartStreamTranscriptionAsync	480
StartTranscriptionJob	484
UpdateVocabulary	504
Scénarios	507
Création et affinage d'un vocabulaire personnalisé	507
Transcription d'un fichier audio et obtention de données sur la tâche	518
Exemples de services croisés	529
Créer une application Amazon Transcribe	529

Créer une application de streaming Amazon Transcribe	530
Convertir du texte en parole, puis de nouveau en texte	530
Sécurité	532
Gestion de l'identité et des accès	533
Public ciblé	533
Authentification par des identités	534
Gestion des accès à l'aide de politiques	538
Comment Amazon Transcribe fonctionne avec IAM	541
Prévention du député confus	549
Exemples de politiques basées sur l'identité	550
Résolution des problèmes	559
Protection des données	561
Confidentialité du trafic inter-réseaux	562
Chiffrement des données	563
Refuser d'utiliser vos données pour améliorer le service	566
Surveillance Amazon Transcribe	566
Surveillance avec CloudWatch	567
Surveillance Amazon Transcribe avec CloudTrail	568
Utilisation Amazon EventBridge avec Amazon Transcribe	572
Validation de conformité	579
Résilience	581
Sécurité de l'infrastructure	581
Analyse et gestion des vulnérabilités	582
Points de terminaison d'un VPC (AWS PrivateLink)	582
Sous-réseaux partagés	585
Bonnes pratiques de sécurité	585
Amazon Transcribe Médical	587
Disponibilité et quotas par région	588
Spécialités de base	590
Transcription de termes et de mesures médicaux	591
Transcrire des numéros	593
Transcription d'une conversation médicale	595
Transcription d'un fichier audio	596
Transcription d'un flux en temps réel	601
Activation du partitionnement des haut-parleurs	604
Transcription audio multicanal	614

Transcrire une dictée médicale	622
Transcription d'un fichier audio	623
Transcription d'une dictée médicale en streaming	628
Création et utilisation de vocabulaires médicaux personnalisés	631
Création d'un fichier texte pour votre vocabulaire médical personnalisé	631
Utilisation d'un fichier texte pour créer un vocabulaire médical personnalisé	636
Transcription d'un fichier audio à l'aide d'un vocabulaire médical personnalisé	638
Transcription d'un flux en temps réel à l'aide d'un vocabulaire médical personnalisé	640
Set de personnages pour Amazon Transcribe Medical	643
Identifier le PHI dans un relevé de notes	645
Identifier les données PHI dans un fichier audio	646
Identifier les PHI dans un flux en temps réel	651
Génération de transcriptions alternatives	652
Points de terminaison d'un VPC (AWS PrivateLink)	655
Considérations pour Amazon Transcribe Points de terminaison VPC médicaux	656
Création d'un point de terminaison VPC d'interface pour Amazon Transcribe Médical	656
Création d'une politique de point de terminaison VPC pour Amazon Transcribe Streaming médical	657
Sous-réseaux partagés	658
AWS HealthScribe	659
Fichier de transcription	661
Fichier de documentation clinique	662
Démarrage d'une tâche dans AWS HealthScribe	663
Exemple de sortie	665
Chiffrement des données au repos pour AWS HealthScribe	678
Création d'une clé gérée par le client	679
Spécification d'une CMK pour AWS HealthScribe	681
Contexte de chiffrement AWS KMS	681
Historique de la documentation	682
Glossaire AWS	694
.....	dcxcv

Qu'est-ce que Amazon Transcribe ?

Amazon Transcribe est un service de reconnaissance vocale automatique qui utilise des modèles d'apprentissage automatique pour convertir le son en texte. Vous pouvez l'utiliser Amazon Transcribe comme service de transcription autonome ou pour ajouter des speech-to-text fonctionnalités à n'importe quelle application.

Avec Amazon Transcribe, vous pouvez améliorer la précision pour votre cas d'utilisation spécifique en personnalisant la langue, en filtrant le contenu pour garantir la confidentialité des clients ou une langue adaptée au public, en analysant le contenu sur plusieurs canaux audio, en partitionnant le discours de chaque locuteur, etc.

Vous pouvez transcrire du contenu multimédia en temps réel (streaming) ou des fichiers multimédia situés dans un Amazon S3 bucket (batch). Pour savoir quelles langues sont prises en charge pour chaque type de transcription, consultez le [Langues prises en charge et fonctionnalités spécifiques aux langues](#) tableau.

Rubriques

- [Amazon Transcribe et éligibilité HIPAA](#)
- [Tarification](#)
- [Disponibilité des régions et quotas](#)

Voir [Qu'est-ce que c'est Amazon Transcribe ?](#) pour une courte présentation vidéo de ce service.

Pour en savoir plus, consultez [Fonctionnement d'Amazon Transcribe](#) et [Démarrer avec Amazon Transcribe](#).

Tip

Les informations relatives à l'Amazon Transcribe API se trouvent dans la [référence de l'API](#).

Amazon Transcribe et éligibilité HIPAA

Amazon Transcribe est couvert par AWS l'éligibilité HIPAA et la BAA, qui oblige les clients de BAA à crypter tous les PHI au repos et en transit lors de leur utilisation. L'identification automatique des PHI

est disponible sans frais supplémentaires et dans toutes les régions où Amazon Transcribe elle est présente. Pour plus d'informations, consultez les rubriques [Éligibilité à la loi HIPAA et BAA](#).

Tarifification

Amazon Transcribe est un pay-as-you-go service ; le prix est basé sur les secondes d'audio transcrit, facturé sur une base mensuelle.

L'utilisation de l'appareil est facturée par incréments d'une seconde, avec un minimum de 15 secondes par incréments de 15 secondes par incréments d'une seconde. Notez que des frais supplémentaires s'appliquent pour des fonctionnalités telles que la rédaction de contenu d'informations personnelles et les modèles linguistiques personnalisés.


Pour obtenir des informations sur les coûts de chacun Région AWS, reportez-vous à la section [Amazon Transcribe Tarification](#).

Disponibilité des régions et quotas

Amazon Transcribe est pris en charge dans les versions suivantes Régions AWS :

Région	Type de transcription
af-south-1 (Le Cap)	lot, traitement par lots
ap-east-1 (Hong Kong)	lot
ap-northeast-1 (Tokyo)	lot, traitement par lots
ap-northeast-2 (Séoul)	lot, traitement par lots
ap-south-1 (Bombay)	lot, traitement par lots
ap-southeast-1 (Singapour)	lot, traitement par lots
ap-southeast-2 (Sydney)	lot, traitement par lots
ca-central-1 (Canada, Centre)	lot, traitement par lots
eu-central-1 (Francfort)	lot, traitement par lots

Région	Type de transcription
eu-north-1 (Stockholm)	lot
eu-west-1 (Irlande)	lot, traitement par lots
eu-west-2 (Londres)	lot, traitement par lots
eu-west-3 (Paris)	lot
me-south-1 (Bahreïn)	lot
sa-east-1 (São Paulo)	lot, traitement par lots
us-east-1 (Virginie du Nord)	lot, traitement par lots
us-east-2 (Ohio)	lot, traitement par lots
us-gov-east-1 (GovCloud, Est des États-Unis)	lot, traitement par lots
us-gov-west-1 (GovCloud, Ouest des États-Unis)	lot, traitement par lots
us-west-1 (San Francisco)	lot
us-west-2 (Oregon)	lot, traitement par lots

 Important

Le support régional diffère pour Amazon Transcribe [Amazon Transcribe Medical](#), et [Call Analytics](#).

Pour obtenir les points de terminaison pour chaque région prise en charge, consultez la section [Points de terminaison du service](#) dans la référence AWSgénérale.

Pour obtenir la liste des quotas relatifs à vos transcriptions, consultez les [Quotas de service](#) dans la Référence AWSgénérale. Certains quotas peuvent être modifiés sur demande. Si la colonne Ajustable contient « Oui », vous pouvez demander une augmentation. Pour ce faire, sélectionnez le lien fourni.

Amazon Transcribe features

Pour vous aider à choisir la Amazon Transcribe solution la mieux adaptée à votre cas d'utilisation, le tableau suivant propose une comparaison des fonctionnalités.

Notez que « batch » et 'post-call' font référence à la transcription d'un fichier situé dans un compartiment Amazon S3 , et que « streaming » et 'real-time' font référence à la transcription d'un média en temps réel.

Fonctionnalité	Amazon Transcribe	Amazon Transcribe Medical ¹	Amazon Transcribe Analyse des appels
----------------	-------------------	--	--

Options de configuration

Transcriptions alternatives	traitement par lots, streaming	traitement par lots, streaming	non
Identification des canaux	traitement par lots, streaming	traitement par lots, streaming	post-call, real-time
Mise en file d'attente des tâches	lot	non	post-call
Identification de la langue	traitement par lots, streaming	non	post-call
Identification multilingue	traitement par lots, streaming	non	non
Diarisation des locuteurs	traitement par lots, streaming	traitement par lots, streaming	post-call
Transcription des chiffres ²	traitement par lots, streaming	traitement par lots, streaming	post-call, real-time

Analyse des conversations

Caractéristiques des appels	non	non	post-call
---	-----	-----	-----------

Fonctionnalité	Amazon Transcribe	Amazon Transcribe Medical ¹	Amazon Transcribe Analyse des appels
Récapitulatif des appels ²	non	non	post-call
Catégorisation personnalisée	non	non	post-call
Événements de catégorie en temps réel	non	non	real-time
Détection des problèmes en temps réel ²	non	non	real-time
Sentiment des locuteurs en temps réel	non	non	real-time
Sentiment des locuteurs	non	non	post-call
Personnalisation de la langue			
Modèles de langage personnalisés ²	traitement par lots, streaming	non	post-call, real-time
Vocabulaires personnalisés	traitement par lots, streaming	traitement par lots, streaming	post-call, real-time
Organisation des ressources			
Identification	lot	lot	post-call
Données sensibles			

Fonctionnalité	Amazon Transcribe	Amazon Transcribe Medical ¹	Amazon Transcribe Analyse des appels
Identification des renseignements personnels sur la santé ²	non	traitement par lots, streaming	non
Identification des données d'identification personnelle ²	streaming	non	real-time
Expurgation de l'audio ²	non	non	post-call, real-time
Expurgation des transcriptions ²	traitement par lots, streaming	non	post-call, real-time
Filtrage du vocabulaire	traitement par lots, streaming	non	post-call, real-time
Vidéo			
Sous-titres	lot	non	non

 ¹ Amazon Transcribe Medical n'est disponible qu'en anglais américain.

² Cette fonctionnalité n'est pas disponible dans toutes les langues ; consultez le tableau [Langues prises en charge et fonctionnalités spécifiques aux langues](#) pour plus d'informations.

Langues prises en charge et fonctionnalités spécifiques aux langues

Les langues prises en charge par Amazon Transcribe sont répertoriées dans le tableau suivant ; les fonctionnalités spécifiques à chaque langue sont également répertoriées. Vérifiez que la fonctionnalité que vous souhaitez utiliser est compatible avec la langue de votre média avant de procéder à la transcription.

Pour consulter la liste complète des Amazon Transcribe fonctionnalités, reportez-vous au [résumé des fonctionnalités](#).

Dans le tableau suivant, le terme « batch » fait référence à la transcription d'un fichier multimédia situé dans un Amazon S3 bucket et le terme « streaming » à la transcription du contenu multimédia diffusé en temps réel. Pour les transcriptions Call Analytics, cela 'post-call' fait référence à la transcription d'un fichier multimédia situé dans un Amazon S3 bucket et 'real-time' à la transcription du contenu multimédia diffusé en temps réel.

Langue	Code de langue	Données en entrée	Transcrire des numéros	Acronymes	Modèles de langue personnalisés	Expurgation	Call Analytics * -
Abkhaze	ab-GE	lot	non	lot	non	non	non
Afrikaans	af-ZA	lot	non	lot	non	non	non
Arabe, Golfe	ar-AE	lot	non	non	non	non	post-call
Arabe, moderne standard	ar-SA	lot	non	non	non	non	non
Arménien	hy-AM	lot	non	lot	non	non	non
Asturien	ast-ES	lot	non	lot	non	non	non

Langue	Code de langue	Données en entrée	Transcrire des numéros	Acronymes	Modèles de langue personnalisés	Expurgation	Call Analytics * -
Azerbaïdjanais	az-AZ	lot	non	lot	non	non	non
Bachkir	ba-RU	lot	non	lot	non	non	non
Basque	eu-ES	lot	non	lot	non	non	non
Biélorusse	be-BY	lot	non	lot	non	non	non
Bengali	bn-IN	lot	non	lot	non	non	non
Bosniaque	bs-BA	lot	non	lot	non	non	non
Bulgare	bg-BG	lot	non	lot	non	non	non
Catalan	ca-ES	lot	non	lot	non	non	non
Kurde sorani, Iran	ckb-IR	lot	non	lot	non	non	non
Kurde sorani, Irak	ckb-IQ	lot	non	lot	non	non	non
Chinois simplifié	zh-CN	traitement par lots, streaming	non	non	non	non	post-call
Chinois, traditionnel	zh-TW	lot	non	non	non	non	non

Langue	Code de langue	Données en entrée	Transcrire des numéros	Acronymes	Modèles de langue personnalisés	Expurgation	Call Analytics * -
Croate	hr-HR	lot	non	lot	non	non	non
Tchèque	cs-CZ	lot	non	lot	non	non	non
Danois	da-DK	lot	non	lot	non	non	non
Néerlandais	n1-NL	lot	non	lot	non	non	non
Anglais, australien	en-AU	traitement par lots, streaming	traitement par lots, streaming	traitement par lots, streaming	traitement par lots, streaming	streaming	post-call, real-time
Anglais, britannique	en-GB	traitement par lots, streaming	traitement par lots, streaming	traitement par lots, streaming	traitement par lots, streaming	streaming	post-call, real-time
Anglais, indien	en-IN	lot	lot	lot	non	non	post-call
Anglais, irlandais	en-IE	lot	lot	lot	non	non	post-call
Anglais, Nouvelle-Zélande	en-NZ	lot	lot	lot	non	non	non
Anglais, écossais	en-AB	lot	lot	lot	non	non	post-call
Anglais, Afrique du Sud	en-ZA	lot	lot	lot	non	non	non

Langue	Code de langue	Données en entrée	Transcrire des numéros	Acronymes	Modèles de langue personnalisés	Expurgation	Call Analytics * -
Anglais, ÉTATS-UNIS	en-US	traitement par lots, streaming	traitement par lots, streaming	traitement par lots, streaming	traitement par lots, streaming	traitement par lots, streaming	post-call, real-time
Anglais, gallois	en-WL	lot	lot	lot	non	non	post-call
Estonian	et-ET	lot	non	lot	non	non	non
Farsi	fa-IR	lot	non	non	non	non	non
Finnois	fi-FI	lot	non	lot	non	non	non
Français	fr-FR	traitement par lots, streaming	non	traitement par lots, streaming	non	non	post-call, real-time
Français, canadien	fr-CA	traitement par lots, streaming	non	traitement par lots, streaming	non	non	post-call, real-time
Galicien	gl-ES	lot	non	lot	non	non	non
Géorgien	ka-GE	lot	non	lot	non	non	non
Allemand	de-DE	traitement par lots, streaming	traitement par lots, streaming	traitement par lots, streaming	traitement par lots, streaming	non	post-call, real-time
Allemand, suisse	de-CH	lot	lot	lot	non	non	post-call
Grec	el-GR	lot	non	lot	non	non	non

Langue	Code de langue	Données en entrée	Transcrire des numéros	Acronymes	Modèles de langue personnalisés	Expurgation	Call Analytics * -
Gujarati	gu-IN	lot	non	lot	non	non	non
Haoussa	ha-NG	lot	non	lot	non	non	non
Hébreu	he-IL	lot	non	non	non	non	non
Hindi, indien	hi-IN	traitement par lots, streaming	non	traitement par lots, streaming	lot	non	post-call
Hongrois	hu-HU	lot	non	lot	non	non	non
Islandais	is-IS	lot	non	lot	non	non	non
Indonésien	id-ID	lot	non	lot	non	non	non
Italien	it-IT	traitement par lots, streaming	non	traitement par lots, streaming	non	non	post-call, real-time
Japonais	ja-JP	traitement par lots, streaming	non	non	traitement par lots, streaming	non	post-call
Kabyle	kab-DZ	lot	non	lot	non	non	non
Kannada	kn-IN	lot	non	lot	non	non	non
Kazakh	kk-KZ	lot	non	lot	non	non	non
Kinyarwanda	rw-RW	lot	non	lot	non	non	non

Langue	Code de langue	Données en entrée	Transcrire des numéros	Acronymes	Modèles de langue personnalisés	Expurgation	Call Analytics * -
Coréen	ko-KR	traitement par lots, streaming	non	non	non	non	post-call
kirghize	ky-KG	lot	non	lot	non	non	non
Letton	lv-LV	lot	non	lot	non	non	non
Lituanien	lt-LT	lot	non	lot	non	non	non
Luganda	lg-IN	lot	non	lot	non	non	non
Macédonien	mk-MK	lot	non	lot	non	non	non
Malais	ms-MY	lot	non	lot	non	non	non
Malayalam	ml-IN	lot	non	lot	non	non	non
Maltais	mt-MT	lot	non	lot	non	non	non
Marathi	mr-IN	lot	non	lot	non	non	non
Mari des prairies	mhr-RU	lot	non	lot	non	non	non
Mongol	mn-MN	lot	non	lot	non	non	non
Norvégien (bokmål)	no-NO	lot	non	lot	non	non	non
Odia/Oriya	or-IN	lot	non	lot	non	non	non

Langue	Code de langue	Données en entrée	Transcrire des numéros	Acronymes	Modèles de langue personnalisés	Expurgation	Call Analytics * -
Pachto	ps-AF	lot	non	lot	non	non	non
Polonais	pl-PL	lot	non	lot	non	non	non
Portugais	pt-PT	lot	non	lot	non	non	post-call
Portugais , brésilien	pt-BR	traitement par lots, streaming	non	traitement par lots, streaming	non	non	post-call, real-time
Pendjabi	pa-IN	lot	non	lot	non	non	non
Roumain	ro-RO	lot	non	lot	non	non	non
Russe	ru-RU	lot	non	non	non	non	non
Serbe	sr-RS	lot	non	lot	non	non	non
Singhalais	si-LK	lot	non	lot	non	non	non
Slovaque	sk-SK	lot	non	lot	non	non	non
Slovène	sl-SI	lot	non	lot	non	non	non
Somali	so-SO	lot	non	lot	non	non	non
Espagnol	es-ES	lot	non	lot	non	non	post-call
Espagnol , ÉTATS-UNIS	es-US	traitement par lots, streaming	non	traitement par lots, streaming	traitement par lots, streaming	traitement par lots, streaming	post-call, real-time

Langue	Code de langue	Données en entrée	Transcrire des numéros	Acronymes	Modèles de langue personnalisés	Expurgation	Call Analytics * -
Soundanais	su-ID	lot	non	lot	non	non	non
Swahili, Kenya	sw-KE	lot	non	lot	non	non	non
Swahili, Burundi	sw-BI	lot	non	lot	non	non	non
Swahili, Rwanda	sw-RW	lot	non	lot	non	non	non
Swahili, Tanzanie	sw-TZ	lot	non	lot	non	non	non
Swahili, Ouganda	sw-UG	lot	non	lot	non	non	non
Suédois	sv-SE	lot	non	lot	non	non	non
Tagalog/Philippin	tl-PH	lot	non	lot	non	non	non
Tamoul	ta-IN	lot	non	non	non	non	non
Tatar	tt-RU	lot	non	lot	non	non	non
Télougou	te-IN	lot	non	non	non	non	non
Thaï	th-TH	traitement par lots, streaming	non	traitement par lots, streaming	non	non	non
Turc	tr-TR	lot	non	lot	non	non	non

Langue	Code de langue	Données en entrée	Transcrire des numéros	Acronymes	Modèles de langue personnalisés	Expurgation	Call Analytics * -
Ukrainien	uk-UA	lot	non	lot	non	non	non
Ouïghour	ug-CN	lot	non	lot	non	non	non
Ouzbek	uz-UZ	lot	non	lot	non	non	non
Vietnamien	vi-VN	lot	non	lot	non	non	non
Gallois	cy-WL	lot	non	lot	non	non	non
Wolof	wo-SN	lot	non	lot	non	non	non
Zoulou	zu-ZA	lot	non	lot	non	non	non

*Les informations Call Analytics suivantes ne sont prises en charge que dans certains dialectes anglais :

- [Récapitulatif des appels](#) : en- * (tous les dialectes anglais)
- [Détection des problèmes](#) : en-AU, en-GB, en-US

Langages de programmation pris en charge

Amazon Transcribe prend en charge les AWS SDK suivants :

Transcriptions par lots	Transcriptions en streaming
.NET	Le format .NET n'est pas pris en charge pour le streaming.

Transcriptions par lots	Transcriptions en streaming
AWS Interface de ligne de commande (CLI)	L'interface de ligne de commande (CLI) n'est pas prise en charge pour le streaming.
C++	C++
Go	Go
Java V2	Java V2
JavaScript	JavaScript V3
PHP V3	PHP V3
Python Boto3	SDK de streaming Python pour Amazon Transcribe
Ruby V3	Ruby V3
Rust	Rust

Pour plus d'informations sur l'utilisation des SDK avec Amazon Transcribe, reportez-vous à [Transcription avec les SDK AWS](#).

Pour plus d'informations sur tous les AWS SDK et outils de création disponibles, reportez-vous à la section [Outils sur AWS auxquels vous pouvez vous appuyer](#).

 Tip

Vous trouverez des exemples de code SDK dans les GitHub référentiels suivants :

- [AWS Exemples de code](#)
- [Amazon Transcribe Exemples](#)

Jeux de caractères pour les vocabulaires personnalisés et les filtres de vocabulaire

Pour chaque langue prise en charge par Amazon Transcribe, il existe un jeu de caractères spécifique qu'Amazon Transcribe peut reconnaître. Lorsque vous créez un vocabulaire personnalisé ou un filtre de vocabulaire, utilisez uniquement les caractères répertoriés dans le jeu de caractères de votre langue. Si vous utilisez des caractères non pris en charge, votre vocabulaire personnalisé ou votre filtre de vocabulaire échoue.

Important

Assurez-vous que votre fichier de vocabulaire personnalisé utilise uniquement les points de code Unicode pris en charge et les séquences de points de code répertoriés dans les jeux de caractères suivants.

De nombreux caractères Unicode peuvent apparaître identiques dans les polices de caractères courantes, même si elles utilisent des points de code différents. Seuls les points de code répertoriés dans le présent guide sont pris en charge. Par exemple, le mot français déjà peut être rendu à l'aide de caractères précomposés (où une valeur Unicode représente un caractère accentué) ou de caractères décomposés (où deux valeurs Unicode représentent un caractère accentué, une valeur pour le caractère de base et une autre pour l'accent).

- Version précomposée : 0064 **00E9** 006A **00E0** (rendu : déjà)
- Version décomposée : 0064 **0065 0301** 006A **0061 0300** (rendu : déjà)

Rubriques

- [Jeu de caractères abkhazes](#)
- [Jeu de caractères de l'afrikaans](#)
- [Jeu de caractères arabes](#)
- [Jeu de caractères asturiens](#)
- [Jeu de caractères azerbaïdjanais](#)
- [Jeu de caractères arméniens](#)
- [Jeu de caractères bachkirs](#)
- [Jeu de caractères basques](#)

- [Jeu de caractères biélorusses](#)
- [Jeu de caractères bengalis](#)
- [Jeu de caractères bosniens](#)
- [Jeu de caractères bulgares](#)
- [Jeu de caractères catalans](#)
- [Jeu de caractères kurdes soranis](#)
- [Jeu de caractères chinois, mandarin \(Chine continentale\), chinois simplifié](#)
- [Jeu de caractères chinois, mandarin \(Taiwan\), chinois traditionnel](#)
- [Jeu de caractères croates](#)
- [Jeu de caractères tchèques](#)
- [Jeu de caractères danois](#)
- [Jeu de caractères néerlandais](#)
- [Jeu de caractères anglais](#)
- [Jeu de caractères estoniens](#)
- [Jeu de caractères farsi](#)
- [Jeu de caractères finnois](#)
- [Jeu de caractères français](#)
- [Jeu de caractères galiciens](#)
- [Jeu de caractères géorgiens](#)
- [Jeu de caractères allemand](#)
- [Jeu de caractères grecs](#)
- [Jeu de caractères gujaratis](#)
- [Jeu de caractères hausas](#)
- [Jeu de caractères hébreu](#)
- [Jeu de caractères hindi](#)
- [Jeu de caractères hongrois](#)
- [Jeu de caractères islandais](#)
- [Jeu de caractères indonésien](#)
- [Jeu de caractères italien](#)
- [Jeu de caractères japonais](#)

- [Jeu de caractères kabyles](#)
- [Jeu de caractères kannada](#)
- [Jeu de caractères kazakhs](#)
- [Jeu de caractères kinyarwanda](#)
- [Jeu de caractères coréen](#)
- [Jeu de caractères kirghizes](#)
- [Jeu de caractères lettons](#)
- [Jeu de caractères lituaniens](#)
- [Jeu de caractères luganda](#)
- [Jeu de caractères macédoniens](#)
- [Jeu de caractères malais](#)
- [Jeu de caractères malayalam](#)
- [Jeu de caractères maltais](#)
- [Jeu de caractères marathis](#)
- [Jeu de caractères mari des prairies](#)
- [Jeu de caractères mongoles](#)
- [Jeu de caractères norvégiens \(bokmål\)](#)
- [Jeu de caractères odia/oriya](#)
- [Jeu de caractères pachtounes](#)
- [Jeu de caractères polonais](#)
- [Jeu de caractères portugais](#)
- [Jeu de caractères pendjabis](#)
- [Jeu de caractères roumains](#)
- [Jeu de caractères russe](#)
- [Jeu de caractères serbes](#)
- [Jeu de caractères singhalais](#)
- [Jeu de caractères slovaques](#)
- [Jeu de caractères slovènes](#)
- [Jeu de caractères somalis](#)
- [Jeu de caractères espagnol](#)

- [Jeu de caractères soundanais](#)
- [Jeu de caractères swahilis](#)
- [Jeu de caractères suédois](#)
- [Jeu de caractères tagalog/philippins](#)
- [Jeu de caractères tamoul](#)
- [Jeu de caractères tatars](#)
- [Jeu de caractères télougou](#)
- [Jeu de caractères thaï](#)
- [Jeu de caractères turc](#)
- [Jeu de caractères ukrainiens](#)
- [Jeu de caractères ouïghours](#)
- [Jeu de caractères ouzbeks](#)
- [Jeu de caractères vietnamien](#)
- [Jeu de caractères gallois](#)
- [Jeu de caractères wolofs](#)
- [Jeu de caractères zoulous](#)

Jeu de caractères abkhazes

Pour les vocabulaires personnalisés en abkhaze, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
a	0430	лъ	0459
б	0431	њ	045A

Caractère	Code	Caractère	Code
в	0432	ħ	045B
г	0433	ќ	045C
д	0434	#	045D
е	0435	ŷ	045E
ж	0436	ц	045F
з	0437	ѓ	0491
и	0438	ƒ	0493
й	0439	жҥ	0497
к	043A	з	0499
л	043B	қ	049B
м	043C	к	049F
н	043D	т	04A1
о	043E	ң	04A3
п	043F	н	04A5
р	0440	џ	04A9
с	0441	ç	04AB
т	0442	т	04AD
у	0443	ү	04AF
ф	0444	ұ	04B1
х	0445	х	04B3

Caractère	Code	Caractère	Code
ц	0446	ц	04B5
ч	0447	ч	04B7
ш	0448	h	04BB
щ	0449	є	04BD
ъ	044A	ё	04BF
ы	044B	#	04CA
ь	044C	ă	04D1
э	044D	ä	04D3
ю	044E	ě	04D7
я	044F	ө	04D9
#	0450	з	04E1
ë	0451	й	04E3
ђ	0452	ö	04E7
í	0453	ө	04E9
є	0454	ÿ	04EF
s	0455	ÿ	04F1
i	0456	ÿ	04F3
ï	0457	#	04F7
j	0458	Ы	04F9
#	0525		

Jeu de caractères de l'afrikaans

Pour les vocabulaires personnalisés en afrikaans, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
á	00E1	ï	00EF
è	00E8	ó	00F3
é	00E9	ô	00F4
ê	00EA	ö	00F6
ë	00EB	ú	00FA
í	00ED	û	00FB
î	00EE	ü	00FC

Jeu de caractères arabes

Pour les vocabulaires personnalisés en arabe, vous pouvez utiliser les caractères Unicode suivants dans le champ `Phrase`. Vous pouvez également utiliser le trait d'union (-) pour séparer les mots.

Caractère	Code	Caractère	Code
ء	0621	س	0633
آ	0622	ش	0634

Caractère	Code	Caractère	Code
أ	0623	ص	0635
ؤ	0624	ض	0636
إ	0625	ط	0637
ئ	0626	ظ	0638
ا	0627	ع	0639
ب	0628	غ	063A
ة	0629	ف	0641
ت	062A	ق	0642
ث	062B	ك	0643
ج	062C	ل	0644
ح	062D	م	0645
خ	062E	ن	0646
د	062F	ه	0647
ذ	0630	و	0648
ر	0631	ى	0649
ز	0632	ي	064A

Jeu de caractères asturiens

Pour les vocabulaires personnalisés en asturien, vous pouvez utiliser les caractères suivants dans le champ Phrase :

- a - z
- - (trait d'union)

- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ Phrase :

Caractère	Code	Caractère	Code
á	00E1	ñ	00F1
é	00E9	ó	00F3
í	00ED	ú	00FA
ü	00FC		

Jeu de caractères azerbaïdjanais

Pour les vocabulaires personnalisés en azerbaïdjanais, vous pouvez utiliser les caractères suivants dans le champ Phrase :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ Phrase :

Caractère	Code	Caractère	Code
ä	00E4	ğ	011F
ç	00E7	ı	0131
ö	00F6	ş	015F
ü	00FC	ə	0259
.	0307		

Jeu de caractères arméniens

Pour les vocabulaires personnalisés en arménien, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
ա	0561	օ	0574
բ	0562	յ	0575
գ	0563	և	0576
դ	0564	ւ	0577
ե	0565	ռ	0578
զ	0566	ճ	0579
է	0567	ղ	057A
ը	0568	չ	057B
թ	0569	ն	057C
ժ	056A	ու	057D
ի	056B	վ	057E
լ	056C	տ	057F
խ	056D	բ	0580
ծ	056E	ց	0581

Caractère	Code	Caractère	Code
ḷ	056F	ɫ	0582
h	0570	ɥ	0583
à	0571	f	0584
ɲ	0572	o	0585
fi	0573	ɕ	0586

Jeu de caractères bachkirs

Pour les vocabulaires personnalisés en bachkir, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
a	0430	ль	0459
б	0431	нь	045A
в	0432	һ	045B
г	0433	ќ	045C
д	0434	#	045D
е	0435	ӥ	045E
ж	0436	ц	045F

Caractère	Code	Caractère	Code
з	0437	г	0491
и	0438	ґ	0493
й	0439	ж	0497
к	043A	з	0499
л	043B	қ	049B
м	043C	к	049F
н	043D	т	04A1
о	043E	ң	04A3
п	043F	н	04A5
р	0440	р	04A9
с	0441	с	04AB
т	0442	т	04AD
у	0443	у	04AF
ф	0444	ф	04B1
х	0445	х	04B3
ц	0446	ц	04B5
ч	0447	ч	04B7
ш	0448	ш	04BB
щ	0449	щ	04BD
ъ	044A	ъ	04BF

Caractère	Code	Caractère	Code
Ы	044B	#	04CA
Ь	044C	ă	04D1
Э	044D	ä	04D3
Ю	044E	ě	04D7
Я	044F	ə	04D9
#	0450	з	04E1
ë	0451	й	04E3
ђ	0452	ö	04E7
í	0453	ө	04E9
є	0454	ÿ	04EF
s	0455	ÿ	04F1
i	0456	ý	04F3
ï	0457	#	04F7
j	0458	Ы	04F9

Jeu de caractères basques

Pour les vocabulaires personnalisés en basque, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
á	00E1	ñ	00F1
é	00E9	ó	00F3
í	00ED	ú	00FA
ü	00FC		

Jeu de caractères biélorusses

Pour les vocabulaires personnalisés en biélorusse, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
а	0430	с	0441
б	0431	т	0442
в	0432	у	0443
г	0433	ф	0444
д	0434	х	0445
е	0435	ц	0446
ж	0436	ч	0447

Caractère	Code	Caractère	Code
з	0437	ш	0448
й	0439	ы	044B
к	043A	ь	044C
л	043B	э	044D
м	043C	ю	044E
н	043D	я	044F
о	043E	ё	0451
п	043F	і	0456
р	0440	ŷ	045E

Jeu de caractères bengalis

Pour les vocabulaires personnalisés en bengali, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
ু	0981	দ	09A6
ূ	0982	ধ	09A7
ৃ	0983	ন	09A8

Caractère	Code	Caractère	Code
অ	0985	প	09AA
আ	0986	ফ	09AB
ঈ	0987	ব	09AC
ই	0988	ভ	09AD
উ	0989	ম	09AE
ঊ	098A	য	09AF
ঋ	098B	র	09B0
এ	098F	ল	09B2
ঐ	0990	শ	09B6
ও	0993	ষ	09B7
ঔ	0994	স	09B8
ক	0995	হ	09B9
খ	0996	.	09BC
গ	0997	#	09BD
ঘ	0998	†	09BE
ঙ	0999	‡	09BF
চ	099A	‡	09C0
ছ	099B	ূ	09C1
জ	099C	ূ	09C2
ঝ	099D	ূ	09C3

Caractère	Code	Caractère	Code
ঞ	099E	<	09C4
ট	099F	ৗ	09C7
ঠ	09A0	ঠ	09C8
ড	09A1	ঢ	09CB
ঢ	09A2	ঢ়	09CC
ণ	09A3	,	09CD
ত	09A4	#	09CE
থ	09A5	ঁ	09D7

Jeu de caractères bosniens

Pour les vocabulaires personnalisés en bosnien, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
ć	0107	đ	0111
č	010D	š	0161
ž	017E		

Jeu de caractères bulgares

Pour les vocabulaires personnalisés en bulgare, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
а	0430	п	043F
б	0431	р	0440
в	0432	с	0441
г	0433	т	0442
д	0434	у	0443
е	0435	ф	0444
ж	0436	х	0445
з	0437	ц	0446
и	0438	ч	0447
й	0439	ш	0448
к	043A	щ	0449
л	043B	ъ	044A
м	043C	ь	044C
н	043D	ю	044E

Caractère	Code	Caractère	Code
o	043E	я	044F

Jeu de caractères catalans

Pour les vocabulaires personnalisés en catalan, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
à	00E0	ï	00EF
ç	00E7	ò	00F2
è	00E8	ó	00F3
é	00E9	ú	00FA
í	00ED	ü	00FC
l·	0140		

Jeu de caractères kurdes soranis

Pour les vocabulaires personnalisés en kurde sorani, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)

- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ Phrase :

Caractère	Code	Caractère	Code
ئ	0626	م	0645
ا	0627	ن	0646
ب	0628	و	0648
ت	062A	پ	067E
ج	062C	چ	0686
ح	062D	ر	0695
خ	062E	ژ	0698
د	062F	ڤ	06A4
ر	0631	ک	06A9
ز	0632	گ	06AF
س	0633	ڭ	06B5
ش	0634	ھ	06BE
ع	0639	ۆ	06C6
غ	063A	ؤ	06C7
ف	0641	ى	06CC
ق	0642	ئ	06CE
ل	0644	ه	06D5

Jeu de caractères chinois, mandarin (Chine continentale), chinois simplifié

Pour les vocabulaires personnalisés en chinois (simplifié), le champ `Phrase` peut utiliser n'importe lequel des caractères répertoriés dans le fichier suivant :

- [zh-cn-character-set](#)

Le champ `SoundsLike` peut contenir les syllabes pinyin répertoriées dans le fichier suivant :

- [pinyin-character-set](#)

Lorsque vous utilisez des syllabes pinyin dans le champ `SoundsLike`, séparez-les par un trait d'union (-).

Amazon Transcribe représente les quatre tons en chinois (simplifié) en utilisant des nombres. Le tableau suivant montre comment les marques de ton sont mappées pour le mot « ma ».

Ton	Marque de ton	Numéro de ton
Ton 1	mā	ma1
Ton 2	má	ma2
Ton 3	mǎ	ma3
Ton 4	mà	ma4

Note

Pour le 5e ton (neutre), vous pouvez utiliser le ton 1, à l'exception de « er », qui doit être mappé au ton 2. Par exemple, 打转儿 sera représenté par « da3-zhuan4-er2 ».

Les vocabulaires personnalisés en chinois (simplifié) n'utilisent pas le champ `IPA`, mais vous devez tout de même inclure l'en-tête `IPA` dans la tableau de vocabulaire.

L'exemple suivant présente un fichier d'entrée au format texte. Il utilise des espaces pour aligner les colonnes. Vos fichiers d'entrée doivent utiliser des caractères TAB pour séparer les colonnes. Inclure des espaces uniquement dans la colonne `DisplayAs`.

Phrase	SoundsLike	IPA	DisplayAs
##	kang1-jian4		
##	qian3-ze2		
####	guo2-fang2-da4-chen2		
#####	shi4-jie4-bo2-lan3-hui4		###

Jeu de caractères chinois, mandarin (Taiwan), chinois traditionnel

Pour les vocabulaires personnalisés en chinois (traditionnel), le champ `Phrase` peut utiliser n'importe lequel des caractères répertoriés dans le fichier suivant :

- [zh-tw-character-set](#)

Le champ `SoundsLike` peut contenir les syllabes zhuyin répertoriées dans le fichier suivant :

- [zhuyin-character-set](#)

Lorsque vous utilisez des syllabes zhuyin dans le champ `SoundsLike`, séparez-les par un trait d'union (-).

Amazon Transcribe représente les quatre tons en chinois (traditionnel) en utilisant des nombres. Le tableau suivant montre comment les marques de tons sont mappées pour le mot ㄇㄩˋ.

Ton	Marque de ton
Ton 1	ㄇㄩˊ
Ton 2	ㄇㄩˊˊ
Ton 3	ㄇㄩˇ
Ton 4	ㄇㄩˋ

Les vocabulaires personnalisés en chinois (traditionnel) n'utilisent pas le champ IPA, mais vous devez tout de même inclure l'en-tête IPA dans la tableau de vocabulaire.

L'exemple suivant présente un fichier d'entrée au format texte. Il utilise des espaces pour aligner les colonnes. Vos fichiers d'entrée doivent utiliser des caractères TAB pour séparer les colonnes. Inclure des espaces uniquement dans la colonne DisplayAs.

Phrase	SoundsLike	IPA	DisplayAs
##	###`-##		
##	###~-##'		
####	###'-##'-##`-##~		
#####	#`-###`-##'-##~-###`	###	

Jeu de caractères croates

Pour les vocabulaires personnalisés en croate, vous pouvez utiliser les caractères suivants dans le champ Phrase :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ Phrase :

Caractère	Code	Caractère	Code
ć	0107	đ	0111
č	010D	š	0161
ž	017E		

Jeu de caractères tchèques

Pour les vocabulaires personnalisés en tchèque, vous pouvez utiliser les caractères suivants dans le champ Phrase :

- a - z

- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
á	00E1	d'	010F
é	00E9	ě	011B
í	00ED	ň	0148
ó	00F3	ř	0159
ú	00FA	š	0161
ý	00FD	t'	0165
č	010D	ů	016F
ž	017E		

Jeu de caractères danois

Pour les vocabulaires personnalisés en danois, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- A - Z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
Å	00C5	æ	00E6
Æ	00C6	é	00E9
Ø	00D8	ø	00F8
å	00E5		

Jeu de caractères néerlandais

Pour les vocabulaires personnalisés en néerlandais, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- A - Z
- ' (apostrophe)
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
à	00E0	î	00EE
á	00E1	ï	00EF
â	00E2	ñ	00F1
ä	00E4	ò	00F2
ç	00E7	ó	00F3
è	00E8	ô	00F4

Caractère	Code	Caractère	Code
é	00E9	ö	00F6
ê	00EA	ù	00F9
ë	00EB	ú	00FA
ì	00EC	û	00FB
í	00ED	ü	00FC

Jeu de caractères anglais

Pour les vocabulaires personnalisés en anglais, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- A - Z
- ' (apostrophe)
- - (trait d'union)
- . (point)

Jeu de caractères estoniens

Pour les vocabulaires personnalisés en estonien, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
ä	00E4	ü	00FC
õ	00F5	š	0161
ö	00F6	ž	017E

Jeu de caractères farsi

Pour les vocabulaires personnalisés en farsi, vous pouvez utiliser les caractères suivants dans le champ `Phrase`.

Caractère	Code	Caractère	Code
ء	0621	ظ	0638
آ	0622	ع	0639
أ	0623	غ	063A
ؤ	0624	ف	0641
ئ	0626	ق	0642
ا	0627	ل	0644
ب	0628	م	0645
ت	062A	ن	0646
ث	062B	ه	0647
ج	062C	و	0648
ح	062D	ـ	064E
خ	062E	ـ	064F
د	062F	ـ	0650

Caractère	Code	Caractère	Code
ذ	0630	س	0651
ر	0631	پ	067E
ز	0632	چ	0686
س	0633	ژ	0698
ش	0634	ک	06A9
ص	0635	گ	06AF
ض	0636	ی	06CC
ط	0637		

Jeu de caractères finnois

Pour les vocabulaires personnalisés en finnois, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
ä	00E4	ö	00F6
å	00E5	š	0161
ž	017E		

Jeu de caractères français

Pour les vocabulaires personnalisés en français, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- A - Z
- ' (apostrophe)
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
À	00C0	à	00E0
Â	00C2	â	00E2
Ç	00C7	ç	00E7
È	00C8	è	00E8
É	00C9	é	00E9
Ê	00CA	ê	00EA
Ë	00CB	ë	00EB
Î	00CE	î	00EE
Ï	00CF	ï	00EF
Ô	00D4	ô	00F4
Ö	00D6	ö	00F6
Ù	00D9	ù	00F9

Caractère	Code	Caractère	Code
Û	00DB	û	00FB
Ü	00DC	ü	00FC

Jeu de caractères galiciens

Pour les vocabulaires personnalisés en galicien, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
á	00E1	ñ	00F1
é	00E9	ó	00F3
í	00ED	ú	00FA
ü	00FC		

Jeu de caractères géorgiens

Pour les vocabulaires personnalisés en géorgien, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
ა	10D0	ბ	10E0
ბ	10D1	გ	10E1
გ	10D2	დ	10E2
დ	10D3	ე	10E3
ე	10D4	ვ	10E4
ვ	10D5	ზ	10E5
ზ	10D6	ყ	10E6
ყ	10D7	შ	10E7
შ	10D8	ჩ	10E8
ჩ	10D9	ც	10E9
ც	10DA	ძ	10EA
ძ	10DB	წ	10EB
წ	10DC	ჭ	10EC
ჭ	10DD	ხ	10ED
ხ	10DE	ძ	10EE
ძ	10DF	წ	10EF
წ	10F0		

Jeu de caractères allemand

Pour les vocabulaires personnalisés en allemand, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- A - Z
- ' (apostrophe)
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
ä	00E4	Ä	00C4
ö	00F6	Ö	00D6
ü	00FC	Ü	00DC
ß	00DF		

Jeu de caractères grecs

Pour les vocabulaires personnalisés en grec, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
ά	03AC	ν	03BD
έ	03AD	ξ	03BE
ή	03AE	ο	03BF
ί	03AF	π	03C0
ü	03B0	ρ	03C1
α	03B1	ς	03C2
β	03B2	σ	03C3
Γ	03B3	τ	03C4
δ	03B4	υ	03C5
ε	03B5	φ	03C6
ζ	03B6	χ	03C7
η	03B7	ψ	03C8
θ	03B8	ω	03C9
ι	03B9	ï	03CA
κ	03BA	ü	03CB
λ	03BB	ó	03CC
μ	03BC	ώ	03CE
ĩ	0390		

Jeu de caractères gujaratis

Pour les vocabulaires personnalisés en gujarati, vous pouvez utiliser les caractères suivants dans le champ Phrase :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ Phrase :

Caractère	Code	Caractère	Code
ઁ	0A81	ઃ	0AA6
ઃ	0A82	ઘ	0AA7
ઃ	0A83	ઙ	0AA8
અ	0A85	ય	0AAA
આ	0A86	ર	0AAB
ઇ	0A87	ભ	0AAC
ઈ	0A88	ભ	0AAD
ઉ	0A89	મ	0AAE
ઊ	0A8A	ય	0AAF
ઋ	0A8B	ર	0AB0
ઌ	0A8D	લ	0AB2
ઍ	0A8F	ળ	0AB3
ઐ	0A90	વ	0AB5
ઑ	0A91	શ	0AB6

Caractère	Code	Caractère	Code
ઓ	0A93	૫	0AB7
ઔ	0A94	સ	0AB8
ક	0A95	હ	0AB9
ખ	0A96	.	0ABC
ગ	0A97	૧	0ABE
ઘ	0A98	૨	0ABF
ઙ	0A99	૩	0AC0
ચ	0A9A	૪	0AC1
છ	0A9B	૫	0AC2
જ	0A9C	૬	0AC3
ઝ	0A9D	૭	0AC5
ઞ	0A9E	૮	0AC7
ટ	0A9F	૯	0AC8
ઠ	0AA0	૧૦	0AC9
ડ	0AA1	૧૧	0ACB
ઢ	0AA2	૧૨	0ACC
ણ	0AA3	૧૩	0ACD
ત	0AA4	૩૦	0AD0
થ	0AA5	૩૧	0AE0

Jeu de caractères hausas

Pour les vocabulaires personnalisés en hausa, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
k	0199	ḅ	0253
y	01B4	ḍ	0257
~	0303		

Jeu de caractères hébreu

Pour les vocabulaires personnalisés en hébreu, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
-	002D	◌׀	05DD
ⴕ	05D0	◌ׁ	05DE
ⴖ	05D1	◌ׂ	05DF
ⴗ	05D2	◌׃	05E0
ⴘ	05D3	◌ׄ	05E1
ⴙ	05D4	◌ׅ	05E2

Caractère	Code	Caractère	Code
ॠ	05D5	ॡ	05E3
ॢ	05D6	ॣ	05E4
।	05D7	॥	05E5
०	05D8	०	05E6
ॡ	05D9	ॢ	05E7
ॣ	05DA	।	05E8
॥	05DB	०	05E9
०	05DC	ॢ	05EA

Jeu de caractères hindi

Pour les vocabulaires personnalisés en hindi, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
-	002D	थ	0925
.	002E	द	0926
ॆ	0901	ध	0927
.	0902	न	0928
:	0903	प	092A
अ	0905	फ	092B
आ	0906	ब	092C
इ	0907	भ	092D

Caractère	Code	Caractère	Code
ई	0908	म	092E
उ	0909	य	092F
ऊ	090A	र	0930
ऋ	090B	ल	0932
ए	090F	व	0935
ऐ	0910	श	0936
ऑ	0911	ष	0937
ओ	0913	स	0938
औ	0914	ह	0939
क	0915	ट	093E
ख	0916	ठि	093F
ग	0917	ति	0940
घ	0918	ु	0941
ङ	0919	े	0942
च	091A	े	0943
छ	091B	ु	0945
ज	091C	े	0947
झ	091D	े	0948
ञ	091E	ँ	0949
ट	091F	ो	094B

Caractère	Code	Caractère	Code
ठ	0920	ौ	094C
ड	0921	ँ	094D
ढ	0922	ज़	095B
ण	0923	ड़	095C
त	0924	ढ	095D

Amazon Transcribe mappe les caractères suivants :

Caractère	Mappé avec
न (0929)	न (0928)
र (0931)	र (0930)
क (0958)	क (0915)
ख (0959)	ख (0916)
ग (095A)	ग (0917)
फ (095E)	फ (092B)
य (095F)	य (092F)

Jeu de caractères hongrois

Pour les vocabulaires personnalisés en hongrois, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
á	00E1	ö	00F6
é	00E9	ú	00FA
í	00ED	ü	00FC
ó	00F3	õ	0151
ű	0171		

Jeu de caractères islandais

Pour les vocabulaires personnalisés en islandais, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
á	00E1	ú	00FA
é	00E9	ý	00FD
ð	00F0	þ	00FE
í	00ED	æ	00E6
ó	00F3	ö	00F6

Jeu de caractères indonésien

Pour les vocabulaires personnalisés en indonésien, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- A - Z
- ' (apostrophe)
- - (trait d'union)
- . (point)

Jeu de caractères italien

Pour les vocabulaires personnalisés en italien, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- A - Z
- ' (apostrophe)
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
À	00C0	à	00E0
Ä	00C4	ä	00E4
Ç	00C7	ç	00E7
È	00C8	è	00E8
É	00C9	é	00E9

Caractère	Code	Caractère	Code
Ê	00CA	ê	00EA
Ë	00CB	ë	00EB
Ì	00CC	ì	00EC
Ò	00D2	ò	00F2
Ù	00D9	ù	00F9
Ü	00DC	ü	00FC

Jeu de caractères japonais

Pour les vocabulaires personnalisés japonais, le champ `DisplayAs` prend en charge tous les caractères hiragana, katakana et kanji, ainsi que les majuscules romaji sur toute la largeur.

Le champ `Phrase` prend en charge les caractères répertoriés dans le fichier suivant :

- [ja-jp-character-set](#)

Jeu de caractères kabyles

Pour les vocabulaires personnalisés en kabyle, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
ï	00EF	đ	1E0D

Caractère	Code	Caractère	Code
č	010D	ħ	1E25
ř	0159	ŗ	1E5B
ǧ	01E7	ş	1E63
ε	025B	ţ	1E6D
γ	0263	ẓ	1E93

Jeu de caractères kannada

Pour les vocabulaires personnalisés en kannada, vous pouvez utiliser les caractères suivants dans le champ Phrase :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ Phrase :

Caractère	Code	Caractère	Code
೦	0C82	೧	0CA7
೨	0C83	೨	0CA8
೩	0C85	೩	0CAA
೪	0C86	೪	0CAB
೫	0C87	೫	0CAC
೬	0C88	೬	0CAD
೭	0C89	೭	0CAE

Caractère	Code	Caractère	Code
ಊ	0C8A	ಯ	0CAF
ಋ	0C8B	ರ	0CB0
ಎ	0C8E	ಲ	0CB2
ಏ	0C8F	ಳ	0CB3
ಐ	0C90	ಋ	0CB5
ಒ	0C92	ಶ	0CB6
ಓ	0C93	ಷ	0CB7
ಔ	0C94	ಠ	0CB8
ಕ	0C95	ಠ	0CB9
ಖ	0C96	#	0CBC
ಗ	0C97	#	0CBD
ಘ	0C98	ಠ	0CBE
ಙ	0C99	ಠ	0CBF
ಚ	0C9A	ಠ	0CC0
ಛ	0C9B	ಠ	0CC1
ಜ	0C9C	ಠ	0CC2
ಝ	0C9D	ಠ	0CC3
ಞ	0C9E	ತಿ	0CC6
ಟ	0C9F	ತಿ	0CC7
ಠ	0CA0	ತಿ	0CC8

Caractère	Code	Caractère	Code
Ә	0CA1	Қ	0CCA
Ғ	0CA2	Қа	0CCB
Ң	0CA3	Ғ	0CCC
Ң	0CA4	Ғ	0CCD
Ғ	0CA5	Қ	0CD5
Ғ	0CA6	Қ	0CD6
Ә	0CE0		

Jeu de caractères kazakhs

Pour les vocabulaires personnalisés en kazakh, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
т	0442	ы	044B
б	0431	я	044F
о	043E	с	0441
п	043F	h	04BB
ш	0448	д	0434

Caractère	Code	Caractère	Code
и	0438	р	0440
ч	0447	г	0433
н	043D	ё	0451
қ	049B	й	0439
і	0456	ө	04E9
щ	0449	в	0432
е	0435	э	044D
ә	04D9	ң	04A3
ю	044E	л	043B
з	0437	ф	0444
х	0445	к	043A
ц	0446	у	0443
ү	04AF	ж	0436
м	043C	ғ	0493
ь	044C	а	0430
ъ	044A	ұ	04B1

Jeu de caractères kinyarwanda

Pour les vocabulaires personnalisés en kinyarwanda, vous pouvez utiliser les caractères suivants dans le champ Phrase :

- a - z
- - (trait d'union)

- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
á	00E1	ó	00F3
â	00E2	ô	00F4
ã	00E3	ú	00FA
ç	00E7	ü	00FC
è	00E8	ā	0101
é	00E9	ē	0113
ê	00EA	ī	012B
ë	00EB	ō	014D
í	00ED	ū	016B
ï	00EF	’	0301

Jeu de caractères coréen

Pour les vocabulaires personnalisés en coréen, vous pouvez utiliser n'importe quelle syllabe hangûl dans le champ `Phrase`. Pour plus d'informations, consultez [Syllabes hangûles](#) sur le site web de Wikipédia.

Jeu de caractères kirghizes

Pour les vocabulaires personnalisés en kirghize, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)

- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ Phrase :

Caractère	Code	Caractère	Code
а	0430	лъ	0459
б	0431	њ	045A
в	0432	ћ	045B
г	0433	ќ	045C
д	0434	#	045D
е	0435	ђ	045E
ж	0436	џ	045F
з	0437	ѓ	0491
и	0438	ѣ	0493
й	0439	ж	0497
к	043A	з	0499
л	043B	қ	049B
м	043C	к	049F
н	043D	т	04A1
о	043E	ң	04A3
п	043F	ф	04A5
р	0440	џ	04A9
с	0441	џ	04AB

Caractère	Code	Caractère	Code
т	0442	ṭ	04AD
у	0443	Ƴ	04AF
ф	0444	ƴ	04B1
х	0445	ɣ	04B3
ц	0446	ƶ	04B5
ч	0447	Ʒ	04B7
ш	0448	ħ	04BB
щ	0449	ɸ	04BD
ъ	044A	ɸ̣	04BF
ы	044B	#	04CA
ь	044C	ǎ	04D1
э	044D	ǎ̄	04D3
ю	044E	ě	04D7
я	044F	ə	04D9
#	0450	з	04E1
ë	0451	й	04E3
ђ	0452	ö	04E7
í	0453	ө	04E9
є	0454	ȳ	04EF
s	0455	ÿ	04F1

Caractère	Code	Caractère	Code
i	0456	ý	04F3
ï	0457	#	04F7
j	0458	ģ	04F9

Jeu de caractères lettons

Pour les vocabulaires personnalisés en letton, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
ā	0101	ķ	0137
č	010D	ļ	013C
ē	0113	ņ	0146
ģ	0123	š	0161
ī	012B	ū	016B
ž	017E		

Jeu de caractères lituaniens

Pour les vocabulaires personnalisés en lituanien, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ Phrase :

Caractère	Code	Caractère	Code
ą	0105	į	012F
č	010D	š	0161
ę	0119	ų	0173
è	0117	ū	016B
ž	017E		

Jeu de caractères luganda

Pour les vocabulaires personnalisés en luganda, vous pouvez utiliser les caractères suivants dans le champ Phrase :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ Phrase :

Caractère	Code	Caractère	Code
ÿ	00FF	ŋ	014B

Jeu de caractères macédoniens

Pour les vocabulaires personnalisés en macédonien, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
а	0430	љ	0459
б	0431	њ	045A
в	0432	ћ	045B
г	0433	ќ	045C
д	0434	#	045D
е	0435	ђ	045E
ж	0436	џ	045F
з	0437	ѓ	0491
и	0438	ѣ	0493
й	0439	ж	0497
к	043A	з	0499
л	043B	ќ	049B
м	043C	к	049F
н	043D	т	04A1

Caractère	Code	Caractère	Code
o	043E	ћ	04A3
п	043F	н	04A5
р	0440	ѝ	04A9
с	0441	џ	04AB
т	0442	ѿ	04AD
у	0443	ѣ	04AF
ф	0444	ѝ	04B1
х	0445	џ	04B3
ц	0446	ѿ	04B5
ч	0447	ѣ	04B7
ш	0448	h	04BB
щ	0449	е	04BD
ъ	044A	ѝ	04BF
ы	044B	#	04CA
ь	044C	ă	04D1
э	044D	ä	04D3
ю	044E	ě	04D7
я	044F	ə	04D9
#	0450	з	04E1
ë	0451	й	04E3

Caractère	Code	Caractère	Code
ħ	0452	ö	04E7
í	0453	ø	04E9
ε	0454	ÿ	04EF
s	0455	ÿ	04F1
i	0456	ÿ	04F3
ï	0457	#	04F7
j	0458	ÿ	04F9

Jeu de caractères malais

Pour les vocabulaires personnalisés en malais, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- A - Z
- ' (apostrophe)
- - (trait d'union)
- . (point)

Jeu de caractères malayalam

Pour les vocabulaires personnalisés en malayalam, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
o	0D02	ന	0D28
ഃ	0D03	പ	0D2A
അ	0D05	ഫ	0D2B
ആ	0D06	ബ	0D2C
ഇ	0D07	ഭ	0D2D
ഈ	0D08	മ	0D2E
ഉ	0D09	യ	0D2F
ഊ	0D0A	ര	0D30
ഋ	0D0B	ഠ	0D31
എ	0D0E	ല	0D32
ഏ	0D0F	ള	0D33
ഐ	0D10	ഴ	0D34
ഒ	0D12	വ	0D35
ഓ	0D13	ശ	0D36
ഔ	0D14	ഷ	0D37
ക	0D15	സ	0D38
ഖ	0D16	ഹ	0D39
ഗ	0D17	ഘ	0D3E
ഘ	0D18	ങ	0D3F
ങ	0D19	ച	0D40

Caractère	Code	Caractère	Code
Ⲡ	0D1A	ⲡ	0D41
ⲡ	0D1B	Ⲣ	0D42
Ⲣ	0D1C	ⲣ	0D43
ⲣ	0D1D	Ⲥ	0D46
Ⲥ	0D1E	ⲥ	0D47
ⲥ	0D1F	Ⲧ	0D48
Ⲧ	0D20	ⲧ	0D4A
ⲧ	0D21	Ⲩ	0D4B
Ⲩ	0D22	ⲩ	0D4C
ⲩ	0D23	Ⲫ	0D4D
Ⲫ	0D24	#	0D7A
ⲫ	0D25	#	0D7B
Ⲭ	0D26	#	0D7C
ⲭ	0D27	#	0D7D

Jeu de caractères maltais

Pour les vocabulaires personnalisés en maltais, vous pouvez utiliser les caractères suivants dans le champ Phrase :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
à	00E0	ù	00F9
è	00E8	ć	010B
ì	00EC	ġ	0121
ò	00F2	ħ	0127
ž	017C		

Jeu de caractères marathis

Pour les vocabulaires personnalisés en marathi, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
ॆ	0901	थ	0925
ॆ	0902	द	0926
:	0903	ध	0927
अ	0905	न	0928
आ	0906	प	092A
इ	0907	फ	092B

Caractère	Code	Caractère	Code
ई	0908	ब	092C
उ	0909	भ	092D
ऊ	090A	म	092E
ऋ	090B	य	092F
ॠ	090D	र	0930
ए	090F	ल	0932
ऐ	0910	ळ	0933
ऑ	0911	व	0935
ओ	0913	श	0936
औ	0914	ष	0937
क	0915	स	0938
ख	0916	ह	0939
ग	0917	.	093C
घ	0918	।	093E
ङ	0919	ि	093F
च	091A	ी	0940
छ	091B	ु	0941
ज	091C	ू	0942
झ	091D	े	0943
ञ	091E	ै	0945

Caractère	Code	Caractère	Code
ट	091F	˘	0947
ठ	0920	˙	0948
ड	0921	ँ	0949
ढ	0922	ं	094B
ण	0923	ँ	094C
त	0924	˘	094D
ॐ	0950		

Jeu de caractères mari des prairies

Pour les vocabulaires personnalisés en mari des prairies, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
а	0430	љ	0459
б	0431	њ	045A
в	0432	ћ	045B
г	0433	ќ	045C
д	0434	#	045D

Caractère	Code	Caractère	Code
е	0435	ÿ	045E
ж	0436	ц	045F
з	0437	г	0491
и	0438	ф	0493
й	0439	ж	0497
к	043A	з	0499
л	043B	қ	049B
м	043C	к	049F
н	043D	т	04A1
о	043E	ң	04A3
п	043F	н	04A5
р	0440	œ	04A9
с	0441	ç	04AB
т	0442	т	04AD
у	0443	ү	04AF
ф	0444	ұ	04B1
х	0445	х	04B3
ц	0446	ц	04B5
ч	0447	ч	04B7
ш	0448	h	04BB

Caractère	Code	Caractère	Code
щ	0449	е	04BD
ъ	044A	ӗ	04BF
ы	044B	#	04CA
ь	044C	ă	04D1
э	044D	ă̆	04D3
ю	044E	ě	04D7
я	044F	ө	04D9
#	0450	з	04E1
ë	0451	й	04E3
ђ	0452	ö	04E7
ѓ	0453	ө̆	04E9
є	0454	ÿ	04EF
s	0455	ÿ̆	04F1
i	0456	ÿ̇	04F3
ï	0457	#	04F7
j	0458	Ы	04F9

Jeu de caractères mongoles

Pour les vocabulaires personnalisés en mongole, vous pouvez utiliser les caractères suivants dans le champ Phrase :

- a - z
- - (trait d'union)

- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ Phrase :

Caractère	Code	Caractère	Code
а	0430	лъ	0459
б	0431	нь	045A
в	0432	ћ	045B
г	0433	ќ	045C
д	0434	#	045D
е	0435	ђ	045E
ж	0436	џ	045F
з	0437	ѓ	0491
и	0438	ѣ	0493
й	0439	ж	0497
к	043A	з	0499
л	043B	ќ	049B
м	043C	к	049F
н	043D	т	04A1
о	043E	ћ	04A3
п	043F	н	04A5
р	0440	џ	04A9
с	0441	џ	04AB

Caractère	Code	Caractère	Code
т	0442	ᠮ	04AD
у	0443	ᠮ	04AF
ф	0444	ᠮ	04B1
х	0445	ᠮ	04B3
ц	0446	ᠮ	04B5
ч	0447	ᠮ	04B7
ш	0448	h	04BB
щ	0449	ᠮ	04BD
ъ	044A	ᠮ	04BF
ы	044B	#	04CA
ь	044C	ᠮ	04D1
э	044D	ᠮ	04D3
ю	044E	ᠮ	04D7
я	044F	ᠮ	04D9
#	0450	з	04E1
ë	0451	ᠮ	04E3
ñ	0452	ö	04E7
í	0453	ᠮ	04E9
є	0454	ᠮ	04EF
s	0455	ᠮ	04F1

Caractère	Code	Caractère	Code
i	0456	ÿ	04F3
ï	0457	#	04F7
j	0458	ÿ	04F9

Jeu de caractères norvégiens (bokmål)

Pour les vocabulaires personnalisés en norvégien (bokmål), vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
å	00E5	æ	00E6
ø	00F8		

Jeu de caractères odia/oriya

Pour les vocabulaires personnalisés en odia/oriya, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
ୂ	0B01	୩	0B26
୦	0B02	୪	0B27
୧	0B03	୫	0B28
ଅ	0B05	ଢ	0B2A
ଆ	0B06	ଢ଼	0B2B
ଇ	0B07	ବ	0B2C
ଈ	0B08	ଭ	0B2D
ଉ	0B09	ଫ	0B2E
ଊ	0B0A	ଝ	0B2F
ଋ	0B0B	ର	0B30
ଏ	0B0F	ଲ	0B32
ଏଫ	0B10	ଳ	0B33
ଓ	0B13	ଶ	0B36
ଓଫ	0B14	ଷ	0B37
କ	0B15	ସ	0B38
ଖ	0B16	ହ	0B39
ଗ	0B17	.	0B3C
ଘ	0B18	।	0B3E
ଙ	0B19	୿	0B3F
ଚ	0B1A	୧	0B40

Caractère	Code	Caractère	Code
ඞ	0B1B	ඞ	0B41
ඞ	0B1C	ඞ	0B42
ඞ	0B1D	ඞ	0B43
ඞ	0B1E	ඞ	0B47
ඞ	0B1F	ඞ	0B48
ඞ	0B20	ඞ#	0B4B
ඞ	0B21	ඞ	0B4C
ඞ	0B22	ඞ	0B4D
ඞ	0B23	ඞ	0B56
ඞ	0B24	ඞ	0B5F
ඞ	0B25	ඞ	0B60
#	0B71		

Jeu de caractères pachtones

Pour les vocabulaires personnalisés en pachtone, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
آ	0622	و	0648
أ	0623	ي	064A
ؤ	0624	ء	064B
ئ	0626	ة	064C
ا	0627	هـ	064D
ب	0628	ـ	064E
ت	062A	ـ	064F
ث	062B	ـ	0650
ج	062C	ـ	0651
ح	062D	ـ	0652
خ	062E	#	0654
د	062F	ـ	0670
ذ	0630	ذ	067C
ر	0631	ر	067E
ز	0632	ز	0681
س	0633	س	0685
ش	0634	ش	0686
ص	0635	ص	0689
ض	0636	ض	0693
ط	0637	ط	0696

Caractère	Code	Caractère	Code
ظ	0638	ژ	0698
ع	0639	بن	069A
غ	063A	ک	06A9
ف	0641	گ	06AB
ق	0642	گ	06AF
ل	0644	ن	06BC
م	0645	ی	06CC
ن	0646	ی	06CD
ه	0647	ې	06D0

Jeu de caractères polonais

Pour les vocabulaires personnalisés en polonais, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
ó	00F3	ł	0142
ą	0105	ń	0144
ć	0107	ś	015B

Caractère	Code	Caractère	Code
ę	0119	ź	017A
ż	017C		

Jeu de caractères portugais

Pour les vocabulaires personnalisés en portugais, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- A - Z
- ' (apostrophe)
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
À	00C0	à	00E0
Á	00C1	á	00E1
Â	00C2	â	00E2
Ã	00C3	ã	00E3
Ä	00C4	ä	00E4
Ç	00C7	ç	00E7
È	00C8	è	00E8
É	00C9	é	00E9
Ê	00CA	ê	00EA

Caractère	Code	Caractère	Code
Ë	00CB	ë	00EB
Í	00CD	í	00ED
Ñ	00D1	ñ	00F1
Ó	00D3	ó	00F3
Ô	00D4	ô	00F4
Õ	00D5	õ	00F5
Ö	00D6	ö	00F6
Ú	00DA	ú	00FA
Ü	00DC	ü	00FC

Jeu de caractères pendjabis

Pour les vocabulaires personnalisés en pendjabi, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
ਅ	0A05	ੳ	0A27
ਆ	0A06	ਠ	0A28
ੲ	0A07	ਖ	0A2A

Caractère	Code	Caractère	Code
ਈ	0A08	ਫ	0A2B
ਉ	0A09	ਬ	0A2C
ਊ	0A0A	ਭ	0A2D
ਏ	0A0F	ਮ	0A2E
ਐ	0A10	ਯ	0A2F
ਓ	0A13	ਰ	0A30
ਔ	0A14	ਲ	0A32
ਕ	0A15	ਵ	0A35
ਖ	0A16	ਸ	0A38
ਗ	0A17	ਹ	0A39
ਘ	0A18	.	0A3C
ਙ	0A19	ਾ	0A3E
ਚ	0A1A	ਿ	0A3F
ਛ	0A1B	ੀ	0A40
ਜ	0A1C	ੁ	0A41
ਝ	0A1D	ੇ	0A42
ਞ	0A1E	ੋ	0A47
ਟ	0A1F	ੇ	0A48
ਠ	0A20	ੋ	0A4B
ਡ	0A21	ਾ	0A4C

Caractère	Code	Caractère	Code
ੳ	0A22	ੴ	0A4D
ੵ	0A23	੶	0A5C
੷	0A24	੸	0A70
੹	0A25	੺	0A71
੻	0A26	੼	0A72
੽	0A73		

Jeu de caractères roumains

Pour les vocabulaires personnalisés en roumain, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
ă	0103	#	0219
â	00E2	#	021B
î	00EE	ș	015F
ț	0163		

Jeu de caractères russe

Pour les vocabulaires personnalisés en russe, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
'	0027	п	043F
-	002D	р	0440
.	002E	с	0441
а	0430	т	0442
б	0431	у	0443
в	0432	ф	0444
г	0433	х	0445
д	0434	ц	0446
е	0435	ч	0447
ж	0436	ш	0448
з	0437	щ	0449
и	0438	ъ	044A
й	0439	ы	044B
к	043A	ь	044C
л	043B	э	044D
м	043C	ю	044E
н	043D	я	044F
о	043E	ё	0451

Jeu de caractères serbes

Pour les vocabulaires personnalisés en serbe, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
ć	0107	ı	0456
č	010D	ï	0457
đ	0111	j	0458
š	0161	љ	0459
ž	017E	њ	045A
а	0430	ћ	045B
б	0431	ќ	045C
в	0432	#	045D
г	0433	ђ	045E
д	0434	џ	045F
е	0435	ѓ	0491
ж	0436	ѣ	0493
з	0437	ж	0497
и	0438	ѝ	0499

Caractère	Code	Caractère	Code
й	0439	ќ	049B
к	043A	к	049F
л	043B	ќ	04A1
м	043C	ћ	04A3
н	043D	н	04A5
о	043E	џ	04A9
п	043F	џ	04AB
р	0440	џ	04AD
с	0441	џ	04AF
т	0442	џ	04B1
у	0443	џ	04B3
ф	0444	џ	04B5
х	0445	џ	04B7
ц	0446	h	04BB
ч	0447	е	04BD
ш	0448	е	04BF
щ	0449	#	04CA
ъ	044A	ă	04D1
ы	044B	ä	04D3
ь	044C	ě	04D7

Caractère	Code	Caractère	Code
э	044D	ө	04D9
ю	044E	з	04E1
я	044F	й	04E3
#	0450	ö	04E7
ë	0451	ө	04E9
ђ	0452	ÿ	04EF
í	0453	ÿ	04F1
є	0454	ÿ	04F3
s	0455	#	04F7
Ы	04F9		

Jeu de caractères singhalais

Pour les vocabulaires personnalisés en singhalais, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
#	0D82	#	0DAF
#	0D83	#	0DB0

Caractère	Code	Caractère	Code
#	0D85	#	0DB1
#	0D86	#	0DB3
#	0D87	#	0DB4
#	0D88	#	0DB5
#	0D89	#	0DB6
#	0D8A	#	0DB7
#	0D8B	#	0DB8
#	0D8C	#	0DB9
#	0D8D	#	0DBA
#	0D91	#	0DBB
#	0D92	#	0DBD
#	0D93	#	0DC0
#	0D94	#	0DC1
#	0D95	#	0DC2
#	0D96	#	0DC3
#	0D9A	#	0DC4
#	0D9B	#	0DC5
#	0D9C	#	0DC6
#	0D9D	#	0DCA
#	0D9E	#	0DCF

Caractère	Code	Caractère	Code
#	0D9F	#	0DD0
#	0DA0	#	0DD1
#	0DA1	#	0DD2
#	0DA2	#	0DD3
#	0DA3	#	0DD4
#	0DA4	#	0DD6
#	0DA5	#	0DD8
#	0DA7	#	0DD9
#	0DA8	##	0DDA
#	0DA9	#	0ddb
#	0DAA	##	0DDC
#	0DAB	###	0DDD
#	0DAC	##	0DDE
#	0DAD	#	0DDF
#	0DAE	#	0DF2

Jeu de caractères slovaques

Pour les vocabulaires personnalisés en slovaque, vous pouvez utiliser les caractères suivants dans le champ Phrase :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
á	00E1	ň	0148
ä	00E4	ó	00F3
č	010D	ô	00F4
d'	010F	í	0155
é	00E9	š	0161
í	00ED	t'	0165
í	013A	ú	00FA
ĭ	013E	ý	00FD
ž	017E		

Jeu de caractères slovènes

Pour les vocabulaires personnalisés en slovène, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
č	010D	š	0161
ž	017E		

Jeu de caractères somalis

Pour les vocabulaires personnalisés en somali, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
s	0073	d	0064
t	0074	a	0061
a	0061	r	0072
n	006E	d	0064

Jeu de caractères espagnol

Pour les vocabulaires personnalisés en espagnol, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- A - Z
- ' (apostrophe)
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
Á	00C1	á	00E1
É	00C9	é	00E9
Í	00CD	í	00ED
Ó	00D3	ó	0XF3
Ú	00DA	ú	00FA
Ñ	00D1	ñ	0XF1
ü	00FC		

Jeu de caractères soundanais

Pour les vocabulaires personnalisés en soundanais, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
s	0073	d	0064
t	0074	a	0061
a	0061	r	0072
n	006E	d	0064

Jeu de caractères swahilis

Pour les vocabulaires personnalisés en swahili, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
s	0073	d	0064
t	0074	a	0061
a	0061	r	0072
n	006E	d	0064

Jeu de caractères suédois

Pour les vocabulaires personnalisés en suédois, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- A - Z
- ' (apostrophe)
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
Ä	00C4	ä	00E4
Å	00C5	å	00E5
Ö	00D6	ö	00F6

Jeu de caractères tagalog/philippins

Pour les vocabulaires personnalisés en tagalog/philippin, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code		
ñ	00F1		

Jeu de caractères tamoul

Pour les vocabulaires personnalisés en tamoul, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
அ	0B85	ஈ	0BB0
ஆ	0B86	ஊ	0BB2
இ	0B87	஋	0BB5

Caractère	Code	Caractère	Code
ஈ	0B88	ழ	0BB4
உ	0B89	ள	0BB3
ஊ	0B8A	ற	0BB1
எ	0B8E	ன	0BA9
ஏ	0B8F	ஐ	0B9C
ஐ	0B90	#	0BB6
ஒ	0B92	ஷ	0BB7
ஓ	0B93	ஸ	0BB8
ஔ	0B94	ஹ	0BB9
ஃ	0B83	.	0BCD
க	0B95	ஈ	0BBE
ங	0B99	ஐ	0BBF
ச	0B9A	ஃ	0BC0
ஞ	0B9E	஁	0BC1
ட	0B9F	ஂ	0BC2
ண	0BA3	ஃ	0BC6
த	0BA4	஄	0BC7
ந	0BA8	அ	0BC8
ப	0BAA	ஆ	0BCA
ம	0BAE	இ	0BCB

Caractère	Code	Caractère	Code
ш	0BAF	ᄀ	0BCC

Jeu de caractères tatars

Pour les vocabulaires personnalisés en tatar, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
а	0430	ль	0459
б	0431	нь	045A
в	0432	ћ	045B
г	0433	ќ	045C
д	0434	#	045D
е	0435	ђ	045E
ж	0436	џ	045F
з	0437	ѓ	0491
и	0438	ѡ	0493
й	0439	жҗ	0497
к	043A	қ	0499

Caractère	Code	Caractère	Code
л	043B	қ	049B
м	043C	к	049F
н	043D	т	04A1
о	043E	ң	04A3
п	043F	ф	04A5
р	0440	р	04A9
с	0441	ç	04AB
т	0442	т	04AD
у	0443	ү	04AF
ф	0444	ү	04B1
х	0445	х	04B3
ц	0446	ц	04B5
ч	0447	ч	04B7
ш	0448	h	04BB
щ	0449	є	04BD
ъ	044A	є	04BF
ы	044B	#	04CA
ь	044C	ă	04D1
э	044D	ä	04D3
ю	044E	ě	04D7

Caractère	Code	Caractère	Code
я	044F	ə	04D9
#	0450	з	04E1
ë	0451	й	04E3
ђ	0452	ö	04E7
í	0453	ө	04E9
є	0454	ÿ	04EF
s	0455	ÿ	04F1
i	0456	ÿ	04F3
ï	0457	#	04F7
j	0458	Ы	04F9

Jeu de caractères télougou

Pour les vocabulaires personnalisés en télougou, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
-	002D	త	0C24
ఁ	0C01	ఢ	0C25
ఌ	0C02	ఢ	0C26
఍	0C03	ఢ	0C27
ఞ	0C05	ఞ	0C28
ఠ	0C06	ఠ	0C2A

Caractère	Code	Caractère	Code
ఇ	0C07	ఫ	0C2B
ఈ	0C08	బ	0C2C
ఉ	0C09	భ	0C2D
ఊ	0C0A	మ	0C2E
ఋ	0C0B	య	0C2F
ఠ	0C30	ఎ	0C0E
ఐ	0C31	ఏ	0C0F
అ	0C32	ఐ	0C10
ఆ	0C33	ఒ	0C12
ప	0C35	ఓ	0C13
ష	0C36	ఔ	0C14
స	0C37	క	0C15
స	0C38	ఖ	0C16
హ	0C39	గ	0C17
౧	0C3E	ఘ	0C18
౨	0C3F	ఙ	0C19
౩	0C40	చ	0C1A
౪	0C41	ఛ	0C1B
౫	0C42	జ	0C1C
౬	0C43	ఝ	0C1D

Caractère	Code	Caractère	Code
๒	0C44	๓	0C1E
๔	0C47	๕	0C1F
๖	0C48	๗	0C20
๘	0C4A	๙	0C21
๑๐	0C4B	๑๑	0C22
๑๒	0C4C	๑๓	0C23
๑๔	0C4D		

Jeu de caractères thaï

Pour les vocabulaires personnalisés en thaï, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
ก	0E01	ล	0E25
ข	0E02	ฬ	0E26
ฃ	0E03	ว	0E27
ค	0E04	ศ	0E28
ฅ	0E05	ษ	0E29
ฆ	0E06	ส	0E2A

Caractère	Code	Caractère	Code
ง	0E07	ห	0E2B
จ	0E08	ฬ	0E2C
ฉ	0E09	อ	0E2D
ช	0E0A	ฮ	0E2E
ซ	0E0B	ๆ	0E2F
ฌ	0E0C	๕	0E30
ญ	0E0D	๖	0E31
ฎ	0E0E	๗	0E32
ฏ	0E0F	๘	0E34
ฐ	0E10	๙	0E35
ฑ	0E11	๐	0E36
ฒ	0E12	๑	0E37
ณ	0E13	๒	0E38
ด	0E14	๓	0E39
ต	0E15	๔	0E3A
ถ	0E16	๕	0E40
ท	0E17	๖	0E41
ธ	0E18	๗	0E42
น	0E19	๘	0E43
บ	0E1A	๙	0E44

Caractère	Code	Caractère	Code
ɓ	0E1B	ɔ	0E45
ɕ	0E1C	ɔ̇	0E46
ɗ	0E1D	ɛ	0E47
ɥ	0E1E	'	0E48
ɟ	0E1F	˘	0E49
ɠ	0E20	˙	0E4A
ɡ	0E21	˚	0E4B
ɣ	0E22	˛	0E4C
ɥ	0E23	˜	0E4D
ɟ	0E24		

Jeu de caractères turc

Pour les vocabulaires personnalisés en turc, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- A - Z
- ' (apostrophe)
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
Ç	00C7	ö	00F6

Caractère	Code	Caractère	Code
Ö	00D6	û	00FB
Ü	00DC	ü	00FC
â	00E2	Ǧ	011E
ä	00E4	ǧ	011F
ç	00E7	ı	0130
è	00E8	ı	0131
é	00E9	Ş	015E
ê	00EA	ş	015F
í	00ED	š	0161
î	00EE	ž	017E
ó	00F3		

Jeu de caractères ukrainiens

Pour les vocabulaires personnalisés en ukrainien, vous pouvez utiliser les caractères suivants dans le champ Phrase :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ Phrase :

Caractère	Code	Caractère	Code
a	0430	p	0440

Caractère	Code	Caractère	Code
б	0431	с	0441
в	0432	т	0442
г	0433	у	0443
д	0434	ф	0444
е	0435	х	0445
ж	0436	ц	0446
з	0437	ч	0447
и	0438	ш	0448
й	0439	щ	0449
к	043A	ь	044C
л	043B	ю	044E
м	043C	я	044F
н	043D	ё	0454
о	043E	і	0456
п	043F	ï	0457
ғ	0491		

Jeu de caractères ouïghours

Pour les vocabulaires personnalisés en ouïghour, vous pouvez utiliser les caractères suivants dans le champ Phrase :

- a - z
- - (trait d'union)

- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ Phrase :

Caractère	Code	Caractère	Code
#	0611	و	0648
#	0613	س	0649
#	0614	ش	064A
ء	0621	=	064B
آ	0622	ا	064C
أ	0623	=	064D
ؤ	0624	ا	064E
إ	0625	ا	064F
ئ	0626	ا	0650
ا	0627	ا	0651
ب	0628	ا	0652
ة	0629	#	0653
ت	062A	#	0654
ث	062B	#	0657
ج	062C	ا	0670
ح	062D	ط	0679
خ	062E	ن	067A
د	062F	ب	067B

Caractère	Code	Caractère	Code
ذ	0630	ٹ	067C
ر	0631	ت	067D
ز	0632	پ	067E
س	0633	ث	067F
ش	0634	پ	0680
ص	0635	ح	0681
ض	0636	ج	0683
ط	0637	چ	0684
ظ	0638	خ	0685
ع	0639	ج	0686
غ	063A	چ	0687
-	0640	ڈ	0688
ف	0641	د	0689
ق	0642	د	068A
ك	0643	ذ	068C
ل	0644	د	068D
م	0645	ذ	068F
ن	0646	ڑ	0691
ه	0647	ر	0693
ر	0695		

Jeu de caractères ouzbeks

Pour les vocabulaires personnalisés en ouzbek, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
т	0442	я	044F
б	0431	с	0441
о	043E	ҳ	04B3
п	043F	д	0434
ш	0448	р	0440
и	0438	ў	045E
ч	0447	г	0433
н	043D	ё	0451
қ	049B	й	0439
е	0435	в	0432
ю	044E	э	044D
з	0437	л	043B
х	0445	ф	0444
ц	0446	к	043A

Caractère	Code	Caractère	Code
м	043C	у	0443
ь	044C	ж	0436
ъ	044A	ф	0493
а	0430		

Jeu de caractères vietnamien

Amazon Transcribe représente les six tons du vietnamien à l'aide de chiffres. Le tableau suivant montre comment les marques de ton sont mappées pour le mot « ma ».

Nom du ton	Marque de ton	Numéro de ton
ngang	ma	ma1
sắc	má	ma2
huyền	mà	ma3
hỏi	mả	ma4
ngã	mã	ma5
nặng	mạ	ma6

Pour les vocabulaires personnalisés en vietnamien, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- A - Z
- ' (apostrophe)
- - (trait d'union)
- . (point)

- & (esperluette)
- ; (point-virgule)
- _ (ligne basse)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ Phrase :

Caractère	Code	Caractère	Code
à	00E0	À	00C0
á	00E1	Á	00C1
â	00E2	Â	00C2
ã	00E3	Ã	00C3
è	00E8	È	00C8
é	00E9	É	00C9
ê	00EA	Ê	00CA
ì	00EC	Ì	00CC
í	00ED	Í	00CD
ò	00F2	Ò	00D2
ó	00F3	Ó	00D3
ô	00F4	Ô	00D4
õ	00F5	Õ	00D5
ù	00F9	Ù	00D9
ú	00FA	Ú	00DA
ý	00FD	Ý	00DD

Caractère	Code	Caractère	Code
ă	0103	Ă	0102
đ	0111	Đ	0110
ĩ	0129	Ĩ	0128
ũ	0169	Ũ	0168
ơ	01A1	Ơ	01A0
ư	01B0	Ư	01AF
ạ	1EA1	Ạ	1EA0
â	1EA3	Â	1EA2
ã	1EA5	Ã	1EA4
ă	1EA7	Ă	1EA6
ã	1EA9	Ã	1EA8
ă	1EAB	Ă	1EAA
ậ	1EAD	Ậ	1EAC
ấ	1EAF	Ấ	1EAE
ă	1EB1	Ă	1EB0
ă	1EB3	Ă	1EB2
ă	1EB5	Ă	1EB4
ă	1EB7	Ă	1EB6
ẹ	1EB9	Ẹ	1EB8
ề	1EBB	Ề	1EBA

Caractère	Code	Caractère	Code
ẽ	1EBD	Ễ	1EBC
ế	1EBF	Ế	1EBE
ề	1EC1	Ề	1EC0
ễ	1EC3	Ễ	1EC2
ễ	1EC5	Ễ	1EC4
ệ	1EC7	Ệ	1EC6
ị	1EC9	Ị	1EC8
ị	1ECB	Ị	1ECA
ọ	1ECD	Ọ	1ECC
ỏ	1ECF	Ỏ	1ECE
ố	1ED1	Ố	1ED0
ồ	1ED3	Ồ	1ED2
ỗ	1ED5	Ỗ	1ED4
ỗ	1ED7	Ỗ	1ED6
ộ	1ED9	Ộ	1ED8
ớ	1EDB	Ớ	1EDA
ờ	1EDD	Ờ	1EDC
ở	1EDF	Ở	1EDE
ỡ	1EE1	Ỡ	1EE0
ợ	1EE3	Ợ	1EE2

Caractère	Code	Caractère	Code
ұ	1EE5	Ў	1EE4
ů	1EE7	Ў	1EE6
ú	1EE9	Ў	1EE8
ù	1EEB	Ў	1EEA
ǔ	1EED	Ў	1EEC
ũ	1EEF	Ў	1EEE
ұ	1EF1	Ў	1EF0
ỳ	1EF3	Ỳ	1EF2
ʏ	1EF5	Ỳ	1EF4
ỳ	1EF7	Ỳ	1EF6
ỹ	1EF9	Ỳ	1EF8

Jeu de caractères gallois

Pour les vocabulaires personnalisés en gallois, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
à	00E0	ò	00F2

Caractère	Code	Caractère	Code
á	00E1	ó	00F3
â	00E2	ô	00F4
ä	00E4	ö	00F6
è	00E8	ù	00F9
é	00E9	ú	00FA
ê	00EA	û	00FB
ë	00EB	ü	00FC
ì	00EC	ý	00FD
í	00ED	ÿ	00FF
î	00EE	ŵ	0175
ï	00EF	ÿ	0177
ÿ	1EF3		

Jeu de caractères wolofs

Pour les vocabulaires personnalisés en wolof, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
à	00E0	ê	00EA
ã	00E3	ë	00EB
ç	00E7	ñ	00F1
è	00E8	ó	00F3
é	00E9	ô	00F4
η	014B		

Jeu de caractères zoulous

Pour les vocabulaires personnalisés en zoulou, vous pouvez utiliser les caractères suivants dans le champ `Phrase` :

- a - z
- - (trait d'union)
- . (point)

Vous pouvez également utiliser les caractères Unicode suivants dans le champ `Phrase` :

Caractère	Code	Caractère	Code
s	0073	d	0064
t	0074	a	0061
a	0061	r	0072
n	006E	d	0064

Fonctionnement d'Amazon Transcribe

Amazon Transcribe utilise des modèles d'apprentissage automatique pour convertir la parole en texte.

Outre le texte transcrit, les transcriptions contiennent des données relatives au contenu transcrit, notamment des scores de confiance et des horodatages pour chaque mot ou signe de ponctuation. Pour voir un exemple de sortie, reportez-vous à la section [Entrée et sortie de données](#). Pour obtenir la liste complète des fonctionnalités que vous pouvez appliquer à votre transcription, consultez le [résumé des fonctionnalités](#).

Les méthodes de transcription se répartissent en deux catégories principales :

- Transcriptions Batch : Transcribe fichiers multimédia qui ont été chargés dans un Amazon S3 compartiment. Vous pouvez utiliser le [AWS CLI](#) [AWS Management Console](#), et divers [AWS SDK pour les](#) transcriptions par lots.
- Transcriptions en streaming : Transcribe flux multimédia en temps réel. Vous pouvez utiliser le [AWS Management Console HTTP/2](#) et divers [AWS SDK](#) pour diffuser des transcriptions. [WebSockets](#)

Notez que la prise en charge des fonctionnalités et des langues est différente pour les transcriptions par lots et en streaming. Pour plus d'informations, reportez-vous à la [Amazon Transcribe features](#) section [Langues prises en charge](#).

Rubriques

- [Entrées et sorties de données](#)
- [Transcrire des chiffres et des signes de ponctuation](#)

 Opérations d'API pour vous aider à démarrer

Batch : [StartTranscriptionJob](#)

Diffusion en continu : [StartStreamTranscription](#), [StartStreamTranscriptionWebSocket](#)

Entrées et sorties de données

Amazon Transcribe prend des données audio, sous forme de fichier multimédia dans un Amazon S3 compartiment ou un flux multimédia, et les convertit en données texte.

Si vous transcrivez des fichiers multimédia stockés dans un Amazon S3 compartiment, vous effectuez des transcriptions par lots. Si vous transcrivez des flux multimédia, vous effectuez des transcriptions en continu. Ces deux processus ont des règles et des exigences différentes.

Avec les transcriptions par lots, vous pouvez l'utiliser [Mise en file d'attente des offres d'emploi](#) si vous n'avez pas besoin de traiter toutes vos tâches de transcription simultanément. Cela permet Amazon Transcribe de suivre vos tâches de transcription et de les traiter lorsque des créneaux sont disponibles.

Note

Amazon Transcribe peut stocker temporairement votre contenu afin d'améliorer continuellement la qualité de ses modèles d'analyse. Veuillez consulter le [FAQ Amazon Transcribe](#) pour en savoir plus. Pour demander la suppression du contenu susceptible d'avoir été stocké par Amazon Transcribe, ouvrez un dossier avec [AWS Support](#).

Rubriques

- [Formats multimédias](#)
- [Canaux audio](#)
- [Taux d'échantillonnage](#)
- [Sortie](#)

Formats multimédias

Les types de supports pris en charge diffèrent entre les transcriptions par lots et les transcriptions en streaming, bien que des formats sans perte soient recommandés pour les deux. Pour plus de détails, veuillez consulter le tableau suivant :

	Par lots	Streaming
Formats pris en charge	• AMR	• FLAC

	Par lots	Streaming
	<ul style="list-style-type: none"> • FLAC • M4A • MP3 • MP4 • Ogg • WebM • WAV 	<ul style="list-style-type: none"> • Ogg Opus • Encodage PCM
Formats recommandés	<ul style="list-style-type: none"> • FLAC • WAV avec encodage PCM 16 bits 	<ul style="list-style-type: none"> • FLAC • Audio Little-Endian 16 bits signé PCM (notez que cela n'inclut pas le format WAV)

Pour de meilleurs résultats, utilisez un format sans perte, tel que FLAC ou WAV avec un codage PCM 16 bits.

Note

Les transcriptions en streaming ne sont pas prises en charge dans toutes les langues. Reportez-vous à la colonne « Entrée de données » dans le [tableau des langues prises en charge](#) pour plus de détails.

Canaux audio

Amazon Transcribe prend en charge les médias à canal unique et double canal. Les médias comportant plus de deux chaînes ne sont actuellement pas pris en charge.

Si votre audio contient plusieurs haut-parleurs sur un canal et que vous souhaitez partitionner et étiqueter chaque haut-parleur dans votre sortie de transcription, vous pouvez utiliser [le partitionnement des haut-parleurs \(diarisation\)](#).

Si votre audio contient des voix sur deux canaux distincts, vous pouvez utiliser [l'identification des canaux](#) pour transcrire chaque canal séparément dans votre transcription.

Ces deux options produisent un seul fichier de transcription.

Note

Si vous n'activez pas [le partitionnement des haut-parleurs](#) ou [l'identification des chaînes](#), le texte de votre transcription est fourni sous la forme d'une section continue.

Taux d'échantillonnage

Pour les tâches de transcription par lots, vous pouvez choisir de fournir une fréquence d'échantillonnage, bien que ce paramètre soit facultatif. Si vous l'incluez dans votre demande, assurez-vous que la valeur que vous fournissez correspond à la fréquence d'échantillonnage réelle de votre audio. Si vous fournissez une fréquence d'échantillonnage qui ne correspond pas à votre son, votre tâche risque d'échouer.

Pour les transcriptions en streaming, vous devez inclure une fréquence d'échantillonnage dans votre demande. Comme pour les tâches de transcription par lots, assurez-vous que la valeur que vous fournissez correspond à la fréquence d'échantillonnage réelle de votre audio.

Les fréquences d'échantillonnage pour les sons de faible fidélité, tels que les enregistrements téléphoniques, utilisent généralement 8 000 Hz. Pour un son de haute fidélité, Amazon Transcribe prend en charge des valeurs comprises entre 16 000 Hz et 48 000 Hz.

Sortie

La sortie de transcription est au format JSON. La première partie de votre transcription contient la transcription elle-même sous forme de paragraphe, suivie de données supplémentaires pour chaque mot et signe de ponctuation. Les données fournies dépendent des fonctionnalités que vous incluez dans votre demande. Votre transcription contient au minimum l'heure de début, l'heure de fin et le score de confiance pour chaque mot. La [section suivante](#) présente un exemple de sortie d'une demande de transcription de base qui n'incluait aucune option ou fonctionnalité supplémentaire.

Toutes les transcriptions par lots sont stockées dans Amazon S3 des compartiments. Vous pouvez choisir d'enregistrer votre transcription dans votre propre Amazon S3 compartiment ou d'Amazon Transcribe utiliser un compartiment sécurisé par défaut. Pour en savoir plus sur la création et l'utilisation de Amazon S3 compartiments, consultez la section [Utilisation des compartiments](#).

Si vous souhaitez que votre transcription soit stockée dans un Amazon S3 compartiment qui vous appartient, spécifiez l'URI du compartiment dans votre demande de transcription. Assurez-

vous d'accorder des autorisations Amazon Transcribe d'écriture pour ce compartiment avant de commencer votre travail de transcription par lots. Si vous spécifiez votre propre compartiment, votre transcription reste dans ce compartiment jusqu'à ce que vous le supprimiez.

Si vous ne spécifiez aucun Amazon S3 compartiment, vous Amazon Transcribe utilisez un compartiment sécurisé géré par les services et vous fournit un URI temporaire que vous pouvez utiliser pour télécharger votre transcription. Notez que les URI temporaires sont valides pendant 15 minutes. Si vous obtenez une `AccessDenied` erreur lors de l'utilisation de l'URI fourni, faites une `GetTranscriptionJob` demande pour obtenir un nouvel URI temporaire pour votre transcription.

Si vous optez pour un compartiment par défaut, votre transcription est supprimée à l'expiration de votre travail (90 jours). Si vous souhaitez conserver votre relevé de notes après cette date d'expiration, vous devez le télécharger.

Les transcriptions de streaming sont renvoyées par la même méthode que celle que vous utilisez pour votre diffusion.

Tip

Si vous souhaitez convertir votre sortie JSON en turn-by-turn transcription au format Word, consultez cet [GitHub exemple \(pour Python3\)](#). Ce script fonctionne avec les transcriptions analytiques post-appel et les transcriptions par lots standard lorsque la diarisation est activée.

Exemple de sortie

Les transcriptions fournissent une transcription complète sous forme de paragraphe, suivie d'une word-for-word ventilation, qui fournit des données pour chaque mot et chaque signe de ponctuation. Cela inclut l'heure de début, l'heure de fin, un score de confiance et un type (`pronunciation` ou `punctuation`).

L'exemple suivant provient d'une simple tâche de transcription par lots qui n'incluait aucune [fonctionnalité supplémentaire](#). À chaque fonctionnalité supplémentaire que vous appliquez à votre demande de transcription, vous obtenez des données supplémentaires dans votre fichier de sortie de transcription.

Les transcriptions par lots de base contiennent deux sections principales :

1. `transcripts`: contient la transcription complète dans un seul bloc de texte.

2. `items`: contient des informations sur chaque mot et signe de ponctuation de la `transcripts` section.

Chaque fonctionnalité supplémentaire que vous incluez dans votre demande de transcription produit des informations supplémentaires dans votre transcription.

```
{
  "jobName": "my-first-transcription-job",
  "accountId": "111122223333",
  "results": {
    "transcripts": [
      {
        "transcript": "Welcome to Amazon Transcribe."
      }
    ],
    "items": [
      {
        "start_time": "0.64",
        "end_time": "1.09",
        "alternatives": [
          {
            "confidence": "1.0",
            "content": "Welcome"
          }
        ],
        "type": "pronunciation"
      },
      {
        "start_time": "1.09",
        "end_time": "1.21",
        "alternatives": [
          {
            "confidence": "1.0",
            "content": "to"
          }
        ],
        "type": "pronunciation"
      },
      {
        "start_time": "1.21",
        "end_time": "1.74",
        "alternatives": [
```

```
        {
            "confidence": "1.0",
            "content": "Amazon"
        }
    ],
    "type": "pronunciation"
},
{
    "start_time": "1.74",
    "end_time": "2.56",
    "alternatives": [
        {
            "confidence": "1.0",
            "content": "Transcribe"
        }
    ],
    "type": "pronunciation"
},
{
    "alternatives": [
        {
            "confidence": "0.0",
            "content": "."
        }
    ],
    "type": "punctuation"
}
    ]
},
"status": "COMPLETED"
}
```

Transcrire des chiffres et des signes de ponctuation

Amazon Transcribe ajoute automatiquement la ponctuation à toutes les langues prises en charge et met les mots en majuscules de manière appropriée pour les langues qui utilisent la distinction majuscules/minuscules dans leur système d'écriture.

Dans la plupart des langues, les nombres sont transcrits sous forme de mots. Toutefois, si votre média est en anglais ou en allemand, Amazon Transcribe les chiffres sont traités différemment selon le contexte dans lequel ils sont utilisés.

Par exemple, si un locuteur dit « »Meet me at eight-thirty AM on June first at one-hundred Main Street with three-dollars-and-fifty-cents and one-point-five chocolate bars, cela est transcrit comme suit :

- Dialectes anglais et allemand :Meet me at 8:30 a.m. on June 1st at 100 Main Street with \$3.50 and 1.5 chocolate bars
- Toutes les autres langues :Meet me at eight thirty a m on June first at one hundred Main Street with three dollars and fifty cents and one point five chocolate bars

Pour consulter toutes les règles associées aux numéros parlés en anglais et en allemand, consultez le tableau suivant.

Règles	Dialectes anglais (entrée audio → texte de sortie)	Dialectes allemands (entrée audio → texte de sortie)
Convertissez les nombres cardinaux supérieurs à dix en nombres.	<ul style="list-style-type: none"> • "Fifty five" → 55 • "a hundred" → 100 • "One thousand and thirty one" → 1031 • "One hundred twenty-three million four hundred fifty six thousand seven hundred eight nine" → 123,456,789 	<ul style="list-style-type: none"> • "fünfundfünfzig" → 55 • "vier tausend sechs hundert einundachtzig" → 4681 • "eine Sache" → "eine Sache"
Convertissez les nombres cardinaux suivis de « million » ou de « milliard » en chiffres suivis d'un mot quand « million » ou « milliard » n'est pas suivi d'un nombre.	<ul style="list-style-type: none"> • "one hundred million" → 100 million • "one billion" → 1 billion • "two point three million" → 2.3 million 	<ul style="list-style-type: none"> • "zehn Millionen Menschen" → 10 Millionen Menschen • "zehn Millionen fünf hundert tausend" → 10.500.000
Convertissez les nombres ordinaux supérieurs à dix en nombres.	<ul style="list-style-type: none"> • "Forty third" → 43rd • "twenty sixth avenue" → 26th avenue 	<ul style="list-style-type: none"> • "dreiundzwanzigste" → 23 • "vierzigster" → 40 • "ich war Erster" → "ich war Erster"

Règles	Dialectes anglais (entrée audio → texte de sortie)	Dialectes allemands (entrée audio → texte de sortie)
Fractions converties en leur format numérique.	<ul style="list-style-type: none"> "a quarter" → 1/4 "three sixteenths" → 3/16 "a half" → 1/2 "a hundredth" → 1/100 	<p>Les fractions ne sont pas converties en format numérique.</p> <ul style="list-style-type: none"> "ein Drittel" → "ein Drittel"
Convertissez les nombres inférieurs à dix en chiffres s'il y en a plusieurs dans une rangée.	<ul style="list-style-type: none"> "three four five" → 345 "My phone number is four two five five five five one two one two" → My phone number is 4255551212 	<ul style="list-style-type: none"> "eins zwei drei" → 123 "plus vier neun zwei vier eins" → +49241
Les mots « point » ou « point » sont affichés sous forme décimale.	<ul style="list-style-type: none"> "three hundred and three dot five" → 303.5 "three point twenty three" → 3.23 "zero point four" → 0.4 "point three" → 0.3 	<p>Les décimales sont indiquées par « , ».</p> <ul style="list-style-type: none"> "zweiundzwanzig komma drei" → 22,3
Convertir le terme « pour cent » après un nombre en signe de pourcentage (%).	<ul style="list-style-type: none"> "twenty three percent" → 23% "twenty three point four five percent" → 23.45% 	<ul style="list-style-type: none"> "fünf Prozent Hürde" → 5% Hürde "dreiundzwanzig komma vier Prozent" → 23,4%

Règles	Dialectes anglais (entrée audio → texte de sortie)	Dialectes allemands (entrée audio → texte de sortie)
<p>Convertissez des mots monétaires en symboles.</p>	<p>Convertissez les mots « dollar », « dollar américain », « dollar australien », « AUD » ou « USD » après un chiffre en signe dollar (\$) avant le chiffre.</p> <ul style="list-style-type: none"> • "one dollar and fifteen cents" → \$1.15 • "twenty three USD" → \$23 • "twenty three Australian dollars" → \$23 <p>Mots « livres », « livres britanniques » ou « GDB » après un nombre convertis en livre signe (£) avant le nombre.</p> <ul style="list-style-type: none"> • "twenty three pounds" → £23 • "I have two thousand pounds" → I have £2,000 • "five pounds thirty three pence" → £5.33 <p>Mots « roupies », « roupies indiennes » ou « INR » après un nombre convertis en signe roupie (#) avant le nombre.</p> <ul style="list-style-type: none"> • "twenty three rupees" → #23 	<p>Convertissez les mots « euro » en un signe de l'euro.</p> <ul style="list-style-type: none"> • "ein euro" → 1 € • "ein Euro vierzig" → 1,40 € • "ein Euro vierzig Cent" → 1,40 €

Règles	Dialectes anglais (entrée audio → texte de sortie)	Dialectes allemands (entrée audio → texte de sortie)
	<ul style="list-style-type: none"> "fifty rupees thirty paise" → #50.30 	
Heures converties en nombres.	<ul style="list-style-type: none"> "seven a m eastern standard time" → 7 a.m. eastern standard time "twelve thirty p m" → 12:30 p.m. 	<ul style="list-style-type: none"> "vierzehn Uhr fünfzehn" → 14:15 Uhr
Dates converties en nombres.	<ul style="list-style-type: none"> "May fifth twenty twelve" → May 5th 2012 "May five twenty twelve" → May 5 2012 "five May twenty twelve" → 5 May 2012 	<ul style="list-style-type: none"> "dritter Dezember neunzehn hundert sechundfünfzig" → 3. Dezember 1956
Séparez les plages de nombres par le mot « à ».	<ul style="list-style-type: none"> "twenty three to thirty seven" → 23 to 37 	Ne s'applique pas
Les années sont représentées par quatre chiffres ; cela n'est valable que pour les années des 20e, 21e et 22e siècles.	<ul style="list-style-type: none"> "nineteen sixty two" → 1962 « the year is twenty twelve » → l'année est2012 "twenty nineteen" → 2019 "twenty one thirty" → 2130 	Ne s'applique pas

Règles	Dialectes anglais (entrée audio → texte de sortie)	Dialectes allemands (entrée audio → texte de sortie)
Afficher les barres obliques et les tirets.	<ul style="list-style-type: none"> "fifty-five dash thirteen" → 55-13 <p>Les barres obliques ne sont pas affichées.</p> <ul style="list-style-type: none"> "fifty-five slash thirteen" → 55 slash 13 	<ul style="list-style-type: none"> "fünfundfünfzig Schrägstrich dreizehn" → 55/13 "fünfundfünfzig Strich dreizehn" → 55-13
Afficher les paragraphes numérotés.	<p>Les paragraphes numérotés ne sont pas affichés à l'aide du symbole de paragraphe (§).</p> <ul style="list-style-type: none"> "paragraph seventeen" → paragraph 17 	<ul style="list-style-type: none"> "Paragraf siebzehn" → § 17

Démarrer avec Amazon Transcribe

Avant de pouvoir créer des transcriptions, vous devez remplir quelques conditions préalables :

- [S'inscrire à un Compte AWS](#)
- [Installez les SDK AWS CLI et](#) (si vous utilisez le AWS Management Console pour vos transcriptions, vous pouvez ignorer cette étape)
- [Configurer les IAM informations d'identification](#)
- [Configurer un Amazon S3 seau](#)
- [Création d'une IAM politique](#)

Une fois ces conditions préalables, vous êtes prêt à utiliser la bonne pratique consistant à utiliser une fédération. Sélectionnez votre méthode de transcription préférée dans la liste suivante pour commencer.

- [AWS CLI](#)
- [AWS Management Console](#)
- [Kit SDK AWS](#)
- [HTTP](#)
- [WebSockets](#)

Tip

Si vous découvrez nos fonctionnalités Amazon Transcribe ou si vous souhaitez les découvrir, nous vous recommandons d'utiliser le [AWS Management Console](#). C'est également l'option la plus simple si vous souhaitez démarrer un stream à l'aide du microphone de votre ordinateur.

Étant donné que le streaming via HTTP/2 WebSockets est plus compliqué que les autres méthodes de transcription, nous vous recommandons de consulter la [Configuration d'une transcription en streaming](#) section avant de commencer à utiliser ces méthodes. Notez que nous vous recommandons vivement d'utiliser un SDK pour les transcriptions en streaming.

S'inscrire à unCompte AWS

Vous pouvez créer un compte [gratuit](#) ou un [compte payant](#). Les deux options vous donnent accès à toutesServices AWS. Le niveau gratuit comprend une période d'essai au cours de laquelle vous pouvez explorerServices AWS et estimer votre consommation. Une fois votre période d'essai expirée, vous pouvez passer à un compte payant. Les frais sont accumulés sur une pay-as-you-use base ; voir [Amazon TranscribeTarification](#) pour plus de détails.

Tip

Lorsque vous configurez votre compte, notez votreCompte AWS identifiant, car vous en avez besoin pour créerIAM des entités.

Installation des kitsAWS CLI de développement logiciel et

Pour utiliser l'Amazon TranscribeAPI, vous devez d'abord installerAWS CLI. La version actuelleAWS CLI est la version 2. Vous trouverez les instructions d'installation pour [Linux](#), [Mac](#), [Windows](#) et [Docker](#) dans le [Guide deAWS Command Line Interface l'utilisateur](#).

Une foisAWS CLI installé, vous devez le [configurer](#) pour vos informations d'identification de sécurité etRégion AWS.

Si vous souhaitez l'utiliserAmazon Transcribe avec un SDK, sélectionnez la langue de votre choix pour les instructions d'installation :

- [.NET](#)
- [C++](#)
- [Go](#)
- [Java V2](#)
- [JavaScript](#)
- [PHP V3](#)
- [AWS SDK for Python \(Boto3\)](#)(transcriptions par lots)
- [Python](#) (transcriptions en continu)
- [Ruby V3](#)
- [Rust](#) (transcriptions par lots)

- [Rust](#) (transcriptions en streaming)

Configurer les IAM informations d'identification

Lorsque vous créez un Compte AWS, vous commencez avec une seule identité de connexion disposant d'un accès complet à tous les AWS services et ressources de votre compte. Cette identité est appelée utilisateur Compte AWS racine. Elle est accessible après connexion à l'aide de l'adresse e-mail et du mot de passe utilisés pour la création du compte.

Il est vivement recommandé de ne pas utiliser l'utilisateur root pour vos tâches quotidiennes. Protégez vos informations d'identification d'utilisateur root et utilisez-les pour effectuer les tâches que seul l'utilisateur root peut effectuer.

Demandez aux utilisateurs, et notamment aux utilisateurs qui nécessitent un accès administrateur, d'appliquer la bonne pratique consistant à utiliser une fédération avec fournisseur d'identité pour accéder à en AWS utilisant des informations d'identification temporaires.

Une identité fédérée est tout utilisateur qui utilise AWS des informations d'identification fournies par le biais d'une source d'identité. Quand des identités fédérées accèdent à Comptes AWS, elles assument des rôles, ces derniers fournissant des informations d'identification temporaires.

Pour une gestion des accès centralisée, nous vous recommandons d'utiliser [AWS IAM Identity Center](#). Vous pouvez créer des utilisateurs et vous pouvez créer des utilisateurs dans IAM Identity Center. Vous pouvez également vous connecter et vous synchroniser avec un ensemble d'utilisateurs et vous synchroniser avec un ensemble d'applications pour une utilisation sur l'ensemble de vos applications et de vos applications et de vos applications Comptes AWS et de vos. Pour plus d'informations, veuillez consulter [Identity and Access Management pour Amazon Transcribe](#).

Pour en savoir plus sur les IAM meilleures pratiques, reportez-vous à [la section Meilleures pratiques de sécurité dans IAM](#).

Création d'un Amazon S3 bucket

Amazon S3 est un service de stockage d'objets sécurisé. Amazon S3 stocke vos fichiers (appelés objets) dans des conteneurs (appelés compartiments).

Pour exécuter une transcription par lots, vous devez d'abord télécharger vos fichiers multimédia dans un Amazon S3 compartiment. Si vous ne spécifiez aucun Amazon S3 compartiment pour la sortie de votre transcription, Amazon Transcribe place votre transcription dans un AWS Amazon S3

compartiment géré temporairement. La sortie de transcription AWS dans les compartiments gérés est automatiquement supprimée au bout de 90 jours.

Découvrez comment [créer votre premier compartiment S3](#) et [charger un objet dans votre compartiment](#).

Création d'une stratégie IAM

Pour gérer l'accès dans AWS, vous devez créer des stratégies et les associer à IAM des identités (utilisateurs, ordinateurs, ordinateurs, ordinateurs, ou ordinateurs AWS). Une politique définit les autorisations de l'entité à laquelle elle est attachée. Par exemple, un rôle ne peut accéder à un fichier multimédia situé dans votre Amazon S3 compartiment que si vous avez associé à ce rôle une politique lui accordant l'accès. Si vous souhaitez restreindre davantage ce rôle, vous pouvez plutôt limiter son accès à un fichier spécifique au sein d'un Amazon S3 compartiment.

Pour plus d'informations sur l'utilisation des AWS stratégies consultez :

- [Politiques et autorisations dans IAM](#)
- [Création de IAM politiques](#)
- [Comment Amazon Transcribe fonctionne avec IAM](#)

Pour des exemples de politiques que vous pouvez utiliser avec Amazon Transcribe, voir [Exemples de politiques basées sur l'identité Amazon Transcribe](#). Si vous souhaitez générer des politiques personnalisées, pensez à utiliser le [générateur AWS de politiques](#).

Vous pouvez ajouter une politique à l'aide de l'AWS Management Console AWS CLI, ou de l'AWS SDK. Pour obtenir des instructions, consultez la section [Ajouter et supprimer des autorisations IAM d'identité](#).

Les politiques ont le format suivant :

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "my-policy-name",
      "Effect": "Allow",
      "Action": [
        "service:action"
      ]
    }
  ]
}
```

```

    ],
    "Resource": [
        "amazon-resource-name"
    ]
  }
]
}

```

Les Amazon Resource Names (ARN) identifient de manière unique toutes les AWS ressources, telles qu'un Amazon S3 compartiment. Vous pouvez utiliser des ARN dans votre politique pour accorder des autorisations pour des actions spécifiques afin d'utiliser des ressources spécifiques. Par exemple, si vous souhaitez accorder un accès en lecture à un Amazon S3 compartiment et à ses sous-dossiers, vous pouvez ajouter le code suivant à la `Statement` section de votre politique de confiance :

```

{
  "Effect": "Allow",
  "Action": [
    "s3:GetObject",
    "s3:ListBucket"
  ],
  "Resource": [
    "arn:aws:s3:::DOC-EXAMPLE-BUCKET",
    "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
  ]
}

```

Voici un exemple de politique qui accorde des autorisations de lecture (GetObject, ListBucket) et d'écriture (PutObject) à un Amazon S3 compartiment et à ses sous-dossiers :DOC-EXAMPLE-BUCKET

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetObject",
        "s3:ListBucket"
      ],
      "Resource": [
        "arn:aws:s3:::DOC-EXAMPLE-BUCKET",
        "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
      ]
    }
  ]
}

```

```
    ],
  },
  {
    "Effect": "Allow",
    "Action": [
      "s3:PutObject"
    ],
    "Resource": [
      "arn:aws:s3:::DOC-EXAMPLE-BUCKET",
      "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
    ]
  }
]
```

Transcrire avec leAWS Management Console

Vous pouvez utiliser laAWS console pour les transcriptions par lots et en streaming. Si vous transcrivez un fichier multimédia situé dans unAmazon S3 compartiment, vous effectuez une transcription par lots. Si vous transcrivez un flux de données audio en temps réel, vous effectuez une transcription en continu.

Avant de commencer une transcription par lots, vous devez d'abord télécharger votre fichier multimédiaAmazon S3 dans un compartiment. Pour diffuser des transcriptions à l'aide duAWS Management Console, vous devez utiliser le microphone de votre ordinateur.

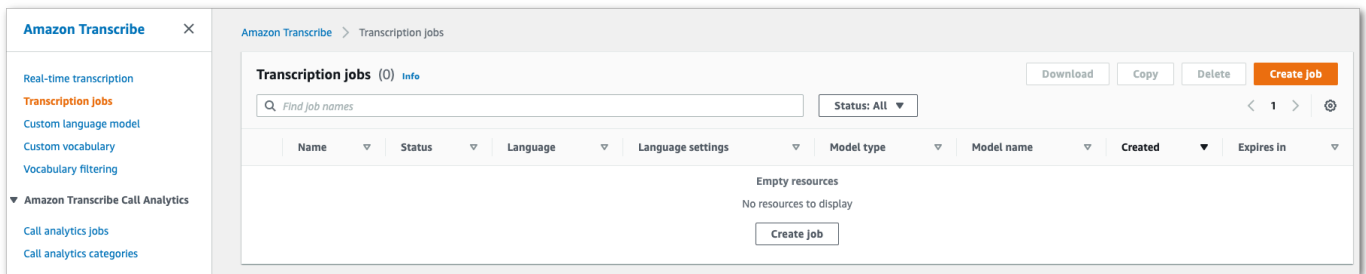
Pour consulter les formats multimédia pris en charge ainsi que les autres exigences et contraintes relatives aux supports, voir[Entrées et sorties de données](#).

Développez les sections suivantes pour une brève présentation de chaque méthode de transcription.

Transcriptions Batch

Assurez-vous d'abord d'avoir chargé le fichier multimédia que vous souhaitez transcrire dans unAmazon S3 compartiment. Si vous ne savez pas comment procéder, consultez le Guide de l'Amazon S3utilisateur : [Charger un objet dans votre compartiment](#).

1. À partir de [AWS Management Console](#), sélectionnez Tâches de transcription dans le volet de navigation de gauche. Vous accédez ainsi à la liste de vos tâches de transcription.



Sélectionnez Créer une tâche.

2. Renseignez les champs de la page Spécifier les détails de la tâche.

Specify job details [Info](#)

Job settings

Name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, . (period), _ (underscore), and - (hyphen).

Model type [Info](#)

Choose the type of model to use for the transcription job.

General model

To use a model that is not specialized for a particular use case, choose this option. Configuration options vary between languages.

Custom language model

To use a model that you trained for your specific use case, choose this option. This model has fewer configuration options than the general model.

Language settings

You can transcribe your audio file in a language that you specify or have Amazon Transcribe identify and transcribe it in the predominant language.

Specific language [Info](#)

If you know the language spoken in your source audio, choose this option to get the most accurate results. The options available for additional processing vary between languages.

Automatic language identification [Info](#)

If you don't know the language spoken in your audio files, choose this option. You have access to fewer options for additional processing than if you choose **Specific language**.

Language

Choose the language of the input audio.

► **Additional settings**

L'emplacement en entrée doit être un objet situé dans un Amazon S3 compartiment. Pour l'emplacement de sortie, vous pouvez choisir un compartiment sécurisé Amazon S3 géré par un service ou vous pouvez spécifier votre propre Amazon S3 compartiment.

Si vous choisissez un compartiment géré par les services, vous pouvez consulter un aperçu de la transcription dans le AWS Management Console et vous pouvez télécharger votre transcription depuis la page de détails du poste (voir ci-dessous).

Si vous choisissez votre propre Amazon S3 compartiment, vous ne pouvez pas voir d'aperçu dans le compartiment AWS Management Console et vous devez accéder au Amazon S3 compartiment pour télécharger votre transcription.

Input data [Info](#)

Input file location on S3
Choose an input audio or video file in Amazon S3.

Valid file formats: MP3, MP4, WAV, FLAC, AMR, OGG, and WebM.

Output data

Output data location type info [Info](#)

Service-managed S3 bucket
The output will be removed after 90 days when the job expires.

Customer specified S3 bucket
The output will not be removed from bucket even after the job expires.

Subtitle file format [Info](#)

SRT (SubRip)

VTT (WebVTT)

Tags - optional

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

No tags associated with the resource.

You can add up to 50 more tags.

Sélectionnez Next (Suivant).

- Sélectionnez les options souhaitées sur la page Configurer la tâche. Si vous souhaitez utiliser [Vocabulaires personnalisés](#) ou [Modèles de langage personnalisés](#) avec votre transcription, vous devez les créer avant de commencer votre travail de transcription.

Configure job - *optional* [Info](#)

Audio settings

Audio identification [Info](#)
Choose to split multi-channel audio into separate channels for transcription, or identify speakers in the input audio.

Alternative results [Info](#)
Enable to view more transcription results

Content removal

Content removal conceals information in the resulting transcript from your source audio file. Amazon Transcribe changes items in the transcript and does not modify the source audio.

Automatic content redaction [Info](#)
Automatic content redaction removes personally identifiable information (PII) in your transcripts. Redactions in transcripts show up as [PII].

Vocabulary filtering [Info](#)
Vocabulary filtering can remove, mask or tag specified words in the final transcript.

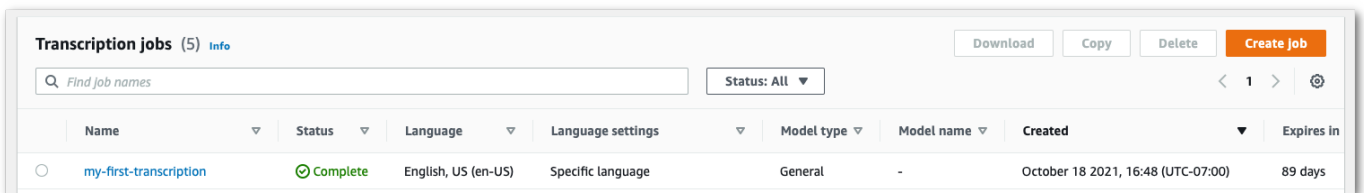
Customization

Custom vocabulary [Info](#)
A custom vocabulary improves the accuracy of recognizing words and phrases specific to your use case.

[Cancel](#) [Previous](#) [Create job](#)

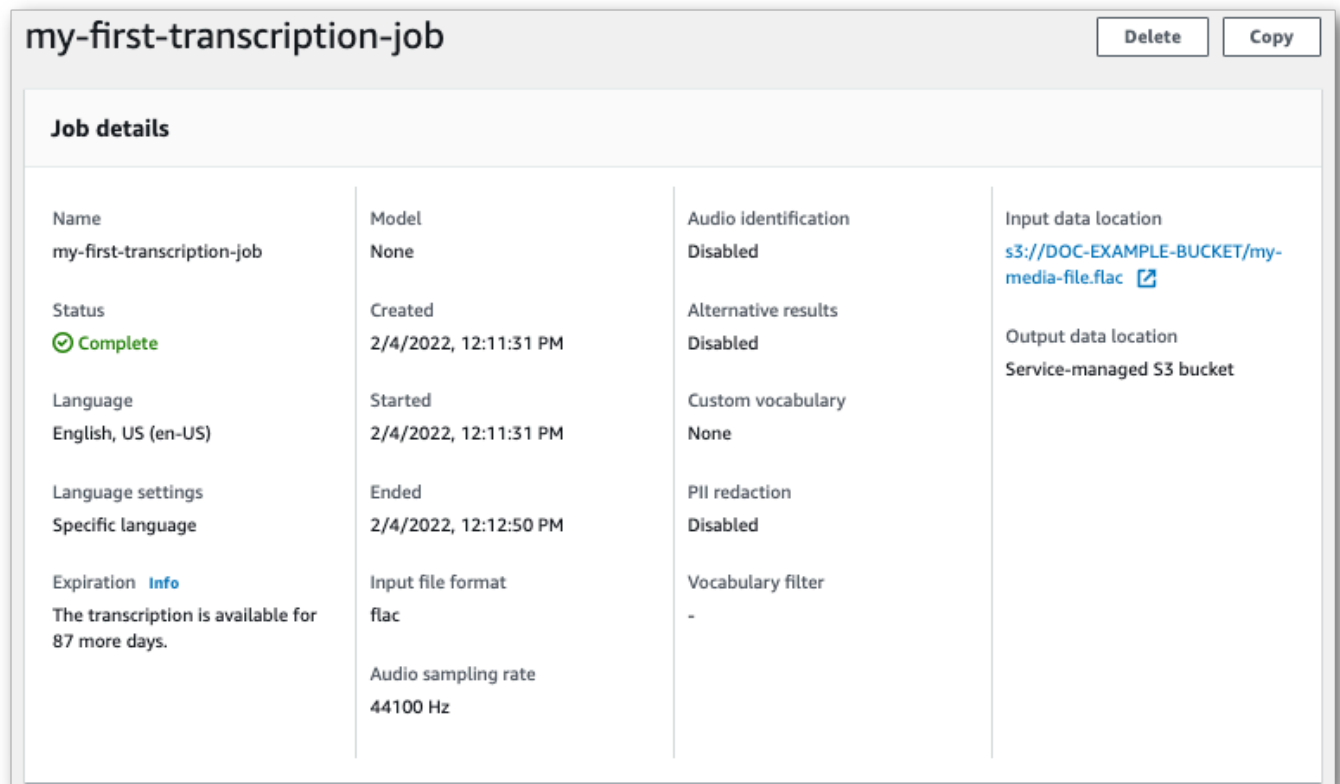
Sélectionnez Créer une tâche.

4. Vous êtes maintenant sur la page des tâches de transcription. Vous pouvez voir ici le statut du travail de transcription. Une fois terminé, sélectionnez votre transcription.



5. Vous consultez maintenant la page des détails du Job pour votre transcription. Vous pouvez voir ici toutes les options que vous avez spécifiées lors de la configuration de votre tâche de transcription.

Pour consulter votre transcription, sélectionnez le chemin de fichier lié dans la colonne de droite sous Emplacement des données de sortie. Cela vous amène au dossier de Amazon S3 sortie que vous avez spécifié. Sélectionnez votre fichier de sortie, qui possède désormais une extension .json.



6. La façon dont vous téléchargez votre transcription varie selon que vous avez choisi un Amazon S3 compartiment géré par les services ou votre propre Amazon S3 compartiment.
- Si vous avez choisi un compartiment géré par les services, vous pouvez voir un volet d'aperçu de la transcription sur la page d'informations de votre tâche de transcription, ainsi qu'un bouton de téléchargement.

The screenshot displays the Amazon Transcribe console interface for a transcription job named "my-first-transcription-job". At the top right, there are "Delete" and "Copy" buttons. The main content is divided into two sections: "Job details" and "Transcription preview".

Job details

Name my-first-transcription-job	Model None	Audio identification Disabled	Input data location s3://DOC-EXAMPLE-BUCKET/my-media-file.flac
Status ✔ Complete	Created 2/4/2022, 12:11:31 PM	Alternative results Disabled	Output data location Service-managed S3 bucket
Language English, US (en-US)	Started 2/4/2022, 12:11:31 PM	Custom vocabulary None	
Language settings Specific language	Ended 2/4/2022, 12:12:50 PM	PII redaction Disabled	
Expiration Info The transcription is available for 87 more days.	Input file format flac	Vocabulary filter -	
	Audio sampling rate 44100 Hz		

Transcription preview

You can see the first 5,000 characters of the transcription text below. To download the full text, choose [Download full transcript](#).

Text | Audio identification | Subtitles

This is a preview of the content of your transcript. If your transcript is long, you may have to scroll to see the complete preview.

Sélectionnez Télécharger et choisissez Télécharger la transcription.

- b. Si vous avez choisi votre propre Amazon S3 compartiment, aucun texte ne s'affiche dans le volet d'aperçu de la transcription sur la page d'informations de votre tâche de transcription. À la place, une boîte d'information bleue contenant un lien vers le Amazon S3 compartiment que vous avez choisi s'affiche.

my-first-transcription-job

Job details

<p>Name my-first-transcription-job</p> <p>Status ✔ Complete</p> <p>Language English, US (en-US)</p> <p>Language settings Specific language</p> <p>Expiration Info The transcription is available for 89 more days.</p>	<p>Model None</p> <p>Created 2/7/2022, 11:42:17 AM</p> <p>Started 2/7/2022, 11:42:17 AM</p> <p>Ended 2/7/2022, 11:43:37 AM</p> <p>Input file format flac</p> <p>Audio sampling rate 44100 Hz</p>	<p>Audio identification Disabled</p> <p>Alternative results Disabled</p> <p>Custom vocabulary None</p> <p>PII redaction Disabled</p> <p>Vocabulary filter -</p>	<p>Input data location s3://DOC-EXAMPLE-BUCKET/my-media-file.flac ↗</p> <p>Output data location https://s3.us-west-2.amazonaws.com/DOC-EXAMPLE-BUCKET ↗</p>
---	--	---	--

Transcription preview ▼

Select download to save a local copy of the transcription.

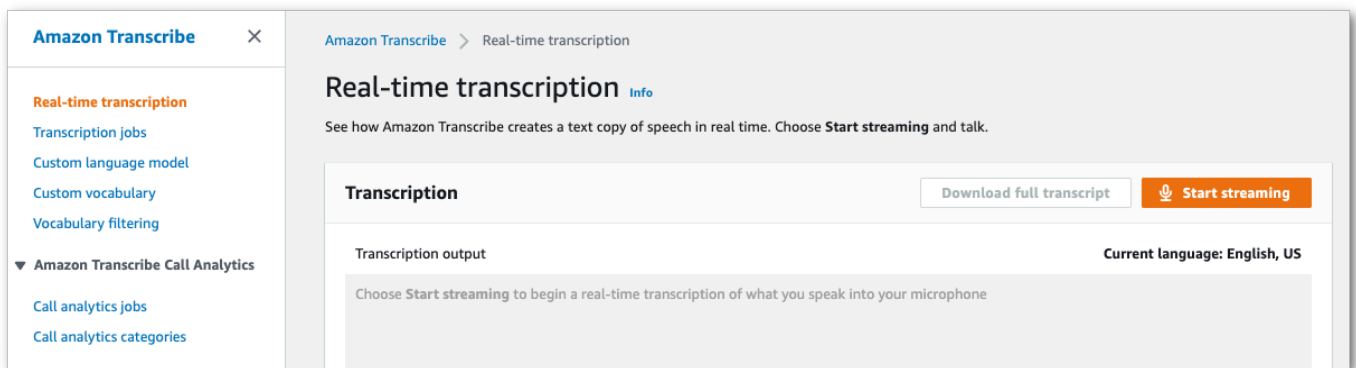
Text
Audio identification
Subtitles

i When you use your own S3 bucket for transcription output, Amazon Transcribe does not show the output in the console. You open the output file from your [S3 Bucket](#). [↗](#)

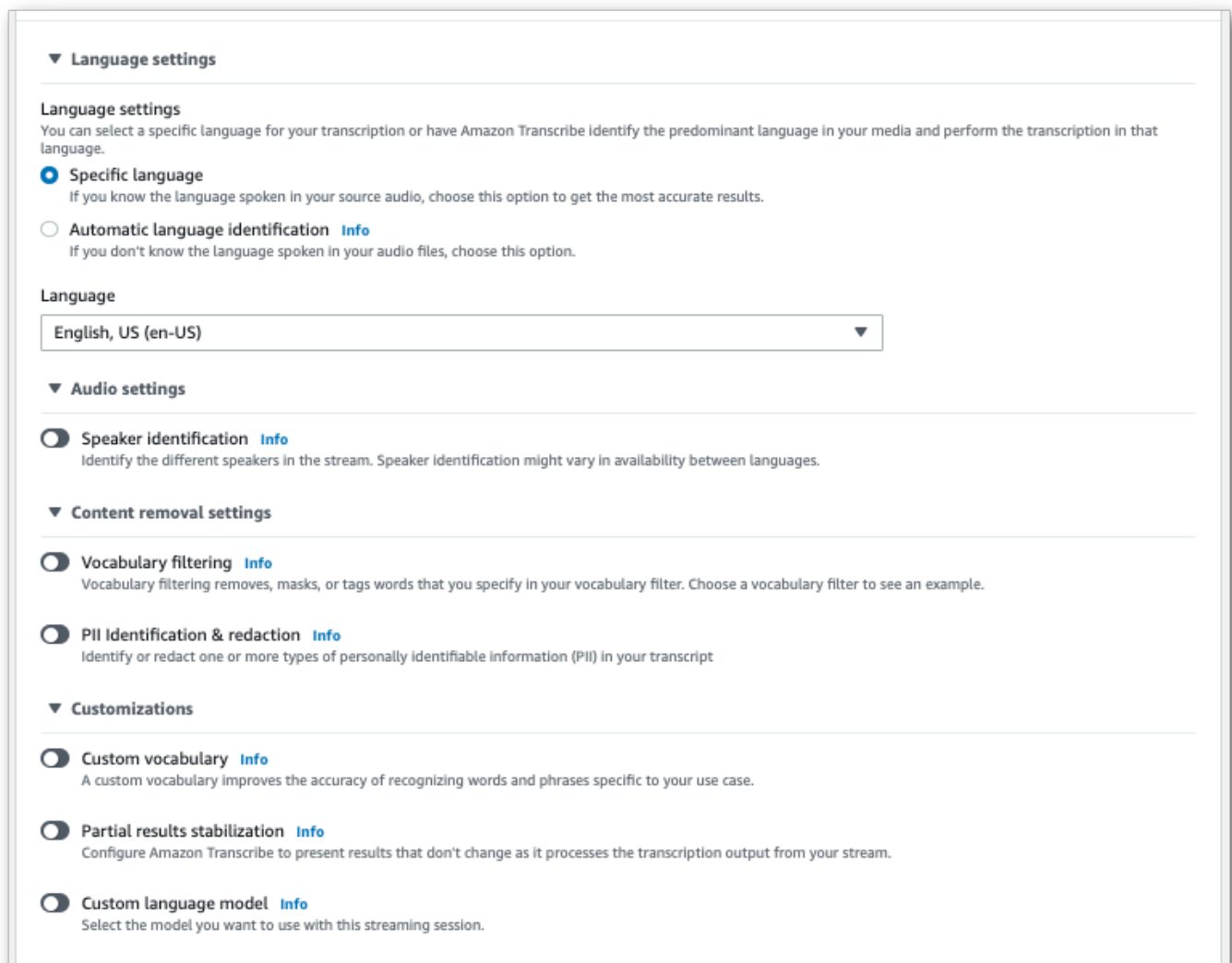
Pour accéder à votre transcription, accédez au Amazon S3 compartiment spécifié à l'aide du lien situé sous Emplacement des données de sortie dans le volet Détails de la Job ou du lien du compartiment S3 dans la zone d'information bleue du volet d'aperçu de la transcription.

Transcriptions en streaming

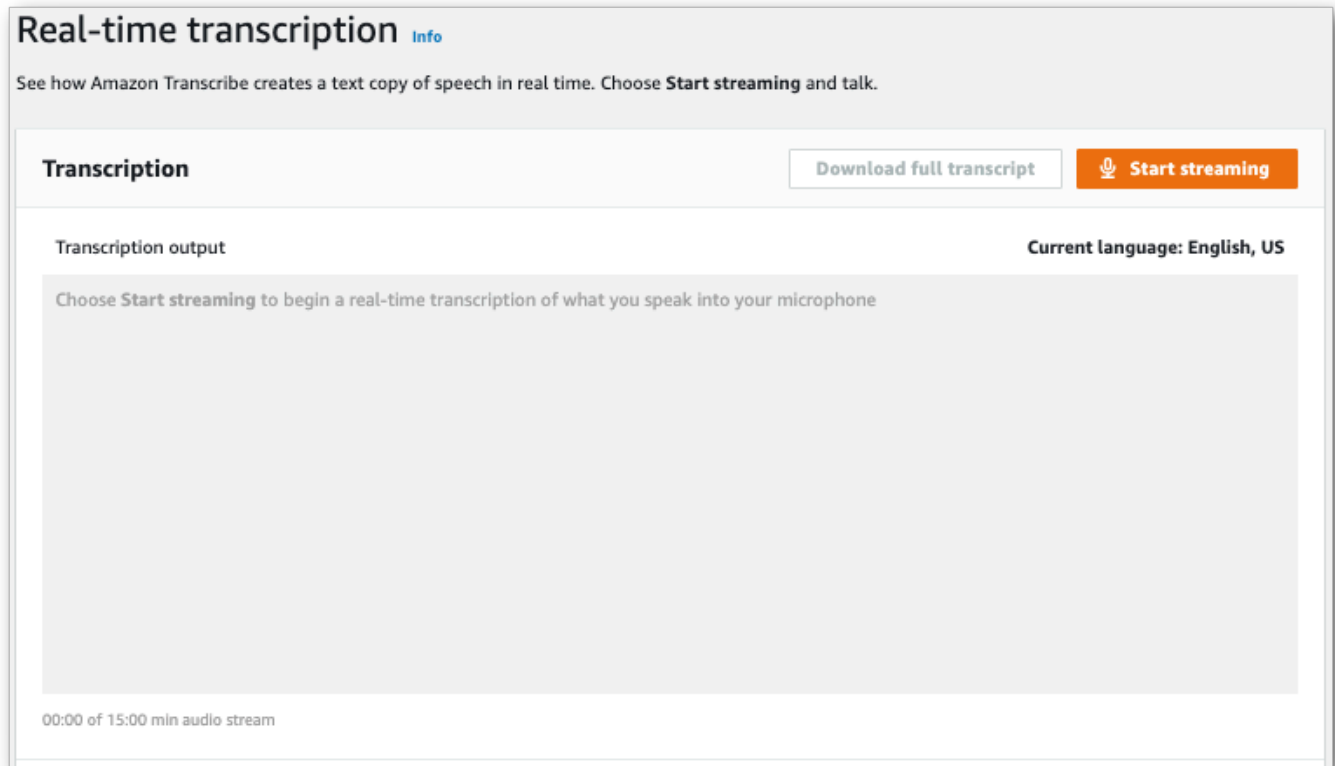
1. À partir de [AWS Management Console](#), sélectionnez Transcription en temps réel dans le volet de navigation de gauche. Vous êtes redirigé sur la page de diffusion principale, où vous pouvez sélectionner des options avant de démarrer votre diffusion.



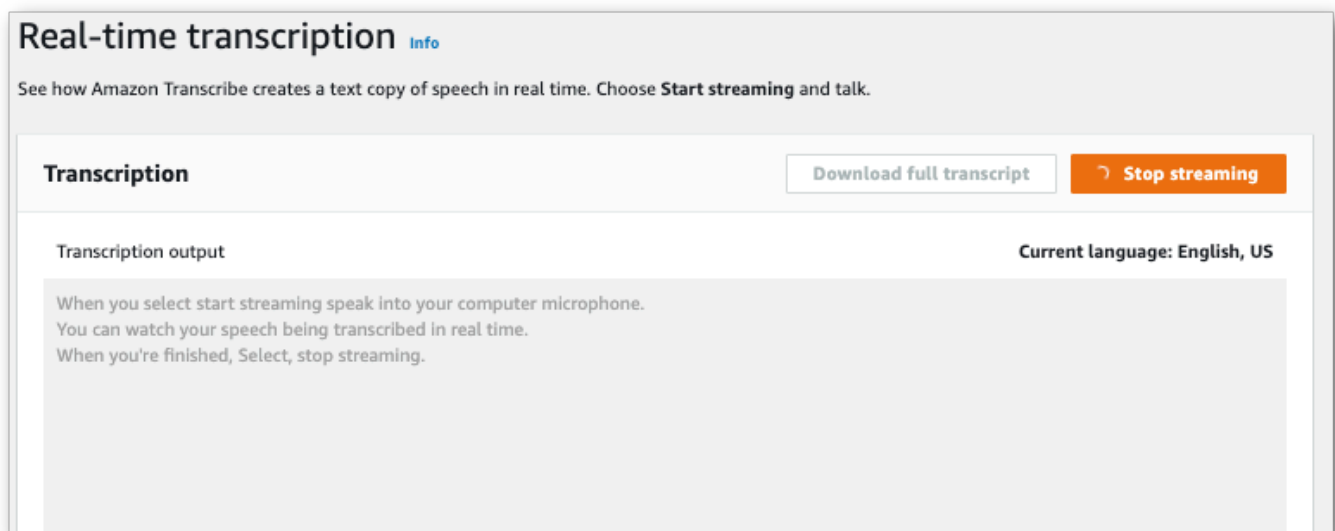
2. Sous la zone de sortie de transcription, vous avez la possibilité de sélectionner différents paramètres linguistiques et audio.



- Après avoir sélectionné les paramètres appropriés, faites défiler la page jusqu'en haut de la page et choisissez Démarrer la diffusion, puis commencez à parler dans le microphone de votre ordinateur. Vous pouvez voir votre discours transcrit en temps réel.



- Lorsque vous avez terminé, sélectionnez Arrêter la diffusion.



Vous pouvez maintenant télécharger votre transcription en sélectionnant Télécharger la transcription complète.

Transcrire à l'aide du AWS CLI

Lorsque vous utilisez le AWS CLI pour démarrer une transcription, vous pouvez exécuter toutes les commandes au niveau de la CLI. Vous pouvez également exécuter la commande que vous souhaitez utiliser, suivie du Région AWS et de l'emplacement d'un fichier JSON contenant un corps de requête. Les exemples présentés dans ce guide illustrent les deux méthodes ; toutefois, cette section se concentre sur la première méthode.

Il AWS CLI ne prend pas en charge les transcriptions en streaming.

Avant de continuer, assurez-vous d'avoir :

- Vous avez chargé votre fichier multimédia dans un Amazon S3 compartiment. Si vous ne savez pas comment créer un Amazon S3 compartiment ou charger votre fichier, reportez-vous aux sections [Créer votre premier Amazon S3 compartiment](#) et [Charger un objet dans votre compartiment](#).
- A installé le [AWS CLI](#).

Vous pouvez trouver toutes les AWS CLI commandes correspondantes Amazon Transcribe dans la [référence des AWS CLI commandes](#).

Commencer un nouveau travail de transcription

Pour démarrer une nouvelle transcription, utilisez la `start-transcription-job` commande.

1. Dans une fenêtre d'un terminal, entrez ce qui suit :

```
aws transcribe start-transcription-job \
```

Un «> » apparaît sur la ligne suivante et vous pouvez maintenant continuer à ajouter les paramètres requis, comme décrit à l'étape suivante.

Vous pouvez également omettre le «\ » et ajouter tous les paramètres, en les séparant par un espace.

2. Avec la `start-transcription-job` commande, vous devez inclure `region`, `transcription-job-name`, `media`, et `language-code` soit `identify-language`.

Si vous souhaitez spécifier un emplacement de sortie, `output-bucket-name` incluez-le dans votre demande ; si vous souhaitez spécifier un sous-dossier du compartiment de sortie spécifié, incluez-le également `output-key`.

```
aws transcribe start-transcription-job \  
  --region us-west-2 \  
  --transcription-job-name my-first-transcription-job \  
  --media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
  --language-code en-US
```

Si vous ajoutez tous les paramètres, cette requête se présente comme suit :

```
aws transcribe start-transcription-job --region us-west-2 --transcription-job-  
name my-first-transcription-job --media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-  
input-files/my-media-file.flac --language-code en-US
```

Si vous choisissez de ne pas spécifier de compartiment de sortie en utilisant `output-bucket-name`, Amazon Transcribe place votre sortie de transcription dans un compartiment géré par le service. Les transcriptions stockées dans un compartiment géré par les services expirent au bout de 90 jours.

Amazon Transcribe répond par :

```
{  
  "TranscriptionJob": {  
    "TranscriptionJobName": "my-first-transcription-job",  
    "TranscriptionJobStatus": "IN_PROGRESS",  
    "LanguageCode": "en-US",  
    "Media": {  
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-  
file.flac"  
    },  
    "StartTime": "2022-03-07T15:03:44.246000-08:00",  
    "CreationTime": "2022-03-07T15:03:44.229000-08:00"  
  }  
}
```

Votre travail de transcription est réussi si [TranscriptionJobStatus](#) vous passez de `IN_PROGRESS` à `COMPLETED`. Pour voir les mises à jour [TranscriptionJobStatus](#), utilisez

alist-transcription-job commandeget-transcription-job ou, comme indiqué dans la section suivante.

Obtention du statut d'une tâche d'.

Pour obtenir des informations sur votre tâche d'exécution, utilisez laget-transcription-job commande.

Les seuls paramètres requis pour cette commande sont l' Région AWSemplacement et le nom de la tâche.

```
aws transcribe get-transcription-job \  
--region us-west-2 \  
--transcription-job-name my-first-transcription-job
```

Amazon Transcriberépond par :

```
{  
  "TranscriptionJob": {  
    "TranscriptionJobName": "my-first-transcription-job",  
    "TranscriptionJobStatus": "COMPLETED",  
    "LanguageCode": "en-US",  
    "MediaSampleRateHertz": 48000,  
    "MediaFormat": "flac",  
    "Media": {  
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
    },  
    "Transcript": {  
      "TranscriptFileUri": "https://s3.the-URI-where-your-job-is-located.json"  
    },  
    "StartTime": "2022-03-07T15:03:44.246000-08:00",  
    "CreationTime": "2022-03-07T15:03:44.229000-08:00",  
    "CompletionTime": "2022-03-07T15:04:01.158000-08:00",  
    "Settings": {  
      "ChannelIdentification": false,  
      "ShowAlternatives": false  
    }  
  }  
}
```

Si vous avez sélectionné votre propreAmazon S3 compartiment pour la sortie de votre transcription, ce compartiment est répertorié avecTranscriptFileUri. Si vous avez sélectionné un

compartiment géré par les services, un URI temporaire est fourni ; utilisez cet URI pour télécharger votre transcription.

Note

Les URI temporaires pour les Amazon S3 compartiments gérés par les services ne sont valides que pendant 15 minutes. Si vous obtenez une `AccesDenied` erreur lors de l'utilisation de l'URI, exécutez à nouveau `get-transcription-job` demande pour obtenir un nouvel URI temporaire.

Répertorier vos tâches de transcription

Pour répertorier toutes vos tâches de transcription dans un même Région AWS fichier, utilisez la `list-transcription-jobs` commande.

Le seul paramètre requis pour cette commande est celui Région AWS dans lequel se trouvent vos tâches de transcription.

```
aws transcribe list-transcription-jobs \  
--region us-west-2
```

Amazon Transcribe répond par :

```
{  
  "NextToken": "A-very-long-string",  
  "TranscriptionJobSummaries": [  
    {  
      "TranscriptionJobName": "my-first-transcription-job",  
      "CreationTime": "2022-03-07T15:03:44.229000-08:00",  
      "StartTime": "2022-03-07T15:03:44.246000-08:00",  
      "CompletionTime": "2022-03-07T15:04:01.158000-08:00",  
      "LanguageCode": "en-US",  
      "TranscriptionJobStatus": "COMPLETED",  
      "OutputLocationType": "SERVICE_BUCKET"  
    }  
  ]  
}
```


Supprimer votre tâche de transcription

Pour supprimer votre tâche de transcription, utilisez la `delete-transcription-job` commande.

Les seuls paramètres requis pour cette commande sont l'Région AWS emplacement et le nom de la tâche.

```
aws transcribe delete-transcription-job \  
  --region us-west-2 \  
  --transcription-job-name my-first-transcription-job
```

Pour confirmer que votre demande de suppression est réussie, vous pouvez exécuter la `list-transcription-jobs` commande. Votre tâche d'action ne doit plus apparaître dans la liste.

Transcription avec les SDK AWS

Vous pouvez utiliser des kits SDK pour les transcriptions par lots et en streaming. Si vous transcrivez un fichier multimédia situé dans un Amazon S3 bucket, vous effectuez une transcription par lots. Si vous transcrivez un flux de données audio en temps réel, vous effectuez une transcription en streaming.

Pour obtenir la liste des langages de programmation que vous pouvez utiliser Amazon Transcribe, consultez [Langages de programmation pris en charge](#). Notez que les transcriptions en streaming ne sont pas prises en charge avec tous les AWS SDK. Pour consulter les formats multimédia pris en charge et les autres exigences et contraintes relatives aux médias, consultez la section [Entrées et sorties de données](#).

Pour plus d'informations sur tous les AWS SDK et outils de création disponibles, reportez-vous à la section [Outils sur AWS auxquels vous pouvez vous appuyer](#).

Tip

Pour d'autres exemples d'utilisation AWS des SDK, notamment des exemples spécifiques aux fonctionnalités, des scénarios et des exemples multiservices, reportez-vous au chapitre. [Exemples de code pour Amazon Transcribe à l'aide de kits de développement logiciel AWS](#)

Vous pouvez également trouver des exemples de code SDK dans les GitHub référentiels suivants :

- [AWS Exemples de code](#)

- [Amazon Transcribe Exemples](#)

Transcriptions par lots

Vous pouvez créer des transcriptions par lots à l'aide de l'URI d'un fichier multimédia situé dans un compartiment Amazon S3 . Si vous ne savez pas comment créer un Amazon S3 compartiment ou télécharger votre fichier, reportez-vous aux sections [Création de votre premier compartiment S3](#) et [Chargement d'un objet dans votre compartiment](#).

Java

```
import software.amazon.awssdk.auth.credentials.AwsCredentialsProvider;
import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.transcribe.TranscribeClient;
import software.amazon.awssdk.services.transcribe.model.*;
import software.amazon.awssdk.services.transcribestreaming.model.LanguageCode;

public class TranscribeDemoApp {
    private static final Region REGION = Region.US_WEST_2;
    private static TranscribeClient client;

    public static void main(String args[]) {

        client = TranscribeClient.builder()
            .credentialsProvider(getCredentials())
            .region(REGION)
            .build();

        String transcriptionJobName = "my-first-transcription-job";
        String mediaType = "flac"; // can be other types
        Media myMedia = Media.builder()
            .mediaFileUri("s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-
file.flac")
            .build();

        String outputS3BucketName = "s3://DOC-EXAMPLE-BUCKET";
        // Create the transcription job request
        StartTranscriptionJobRequest request =
        StartTranscriptionJobRequest.builder()
            .transcriptionJobName(transcriptionJobName)
```

```
        .languageCode(LanguageCode.EN_US.toString())
        .mediaSampleRateHertz(16000)
        .mediaFormat(mediaType)
        .media(myMedia)
        .outputBucketName(outputS3BucketName)
        .build();

    // send the request to start the transcription job
    StartTranscriptionJobResponse startJobResponse =
client.startTranscriptionJob(request);

    System.out.println("Created the transcription job");
    System.out.println(startJobResponse.transcriptionJob());

    // Create the get job request
    GetTranscriptionJobRequest getJobRequest =
GetTranscriptionJobRequest.builder()
        .transcriptionJobName(transcriptionJobName)
        .build();

    // send the request to get the transcription job including the job status
    GetTranscriptionJobResponse getJobResponse =
client.getTranscriptionJob(getJobRequest);

    System.out.println("Get the transcription job request");
    System.out.println(getJobResponse.transcriptionJob());
}

private static AwsCredentialsProvider getCredentials() {
    return DefaultCredentialsProvider.create();
}
}
```

JavaScript

```
const { TranscribeClient, StartTranscriptionJobCommand } = require("@aws-sdk/client-transcribe"); // CommonJS import

const region = "us-west-2";
const credentials = {
    "accessKeyId": "",
    "secretAccessKey": "",
```

```
};

const input = {
  TranscriptionJobName: "my-first-transcription-job",
  LanguageCode: "en-US",
  Media: {
    MediaFileUri: "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
  },
  OutputBucketName: "DOC-EXAMPLE-BUCKET",
};

async function startTranscriptionRequest() {
  const transcribeConfig = {
    region,
    credentials
  };
  const transcribeClient = new TranscribeClient(transcribeConfig);
  const transcribeCommand = new StartTranscriptionJobCommand(input);
  try {
    const transcribeResponse = await transcribeClient.send(transcribeCommand);
    console.log("Transcription job created, the details:");
    console.log(transcribeResponse.TranscriptionJob);
  } catch(err) {
    console.log(err);
  }
}

startTranscriptionRequest();
```

Python

```
import time
import boto3

def transcribe_file(job_name, file_uri, transcribe_client):
    transcribe_client.start_transcription_job(
        TranscriptionJobName = job_name,
        Media = {
            'MediaFileUri': file_uri
        },
        MediaFormat = 'flac',
        LanguageCode = 'en-US'
    )
```

```
max_tries = 60
while max_tries > 0:
    max_tries -= 1
    job = transcribe_client.get_transcription_job(TranscriptionJobName =
job_name)
    job_status = job['TranscriptionJob']['TranscriptionJobStatus']
    if job_status in ['COMPLETED', 'FAILED']:
        print(f"Job {job_name} is {job_status}.")
        if job_status == 'COMPLETED':
            print(
                f"Download the transcript from\n"
                f"\t{job['TranscriptionJob']['Transcript']}
['TranscriptFileUri']}.")
            break
        else:
            print(f"Waiting for {job_name}. Current status is {job_status}.")
            time.sleep(10)

def main():
    transcribe_client = boto3.client('transcribe', region_name = 'us-west-2')
    file_uri = 's3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac'
    transcribe_file('Example-job', file_uri, transcribe_client)

if __name__ == '__main__':
    main()
```

Transcriptions en streaming

Vous pouvez créer des transcriptions en streaming à l'aide d'un fichier multimédia diffusé en continu ou d'un flux multimédia en direct.

Notez que la norme n' AWS SDK for Python (Boto3) est pas prise en charge pour le Amazon Transcribe streaming. Pour démarrer une transcription en streaming à l'aide de Python, utilisez ce [SDK Python asynchrone](#) pour. Amazon Transcribe

Java

L'exemple suivant est un programme Java qui transcrit les flux de streaming audio.

Pour exécuter cet exemple, notez les éléments suivants :

- Vous devez utiliser le [kit SDK AWS pour Java 2.x](#).
- Les clients doivent utiliser Java 1.8 pour être compatible avec le [kit SDK AWS pour Java 2.x](#).
- La fréquence d'échantillonnage que vous spécifiez doit correspondre à la fréquence d'échantillonnage réelle de votre flux audio.

Voir également : [Réessayer le client pour le Amazon Transcribe streaming \(SDK Java\)](#). Ce code gère la connexion à Amazon Transcribe et tente d'envoyer des données lorsqu'il y a des erreurs sur la connexion. Par exemple, s'il y a une erreur temporaire sur le réseau, le client renvoie la demande qui a échoué.

```
public class TranscribeStreamingDemoApp {
    private static final Region REGION = Region.US_WEST_2;
    private static TranscribeStreamingAsyncClient client;

    public static void main(String args[]) throws URISyntaxException,
        ExecutionException, InterruptedException, LineUnavailableException {

        client = TranscribeStreamingAsyncClient.builder()
            .credentialsProvider(getCredentials())
            .region(REGION)
            .build();

        CompletableFuture<Void> result =
client.startStreamTranscription(getRequest(16_000),
    new AudioStreamPublisher(getStreamFromMic()),
    getResponseHandler());

        result.get();
        client.close();
    }

    private static InputStream getStreamFromMic() throws LineUnavailableException {

        // Signed PCM AudioFormat with 16,000 Hz, 16 bit sample size, mono
        int sampleRate = 16000;
        AudioFormat format = new AudioFormat(sampleRate, 16, 1, true, false);
        DataLine.Info info = new DataLine.Info(TargetDataLine.class, format);

        if (!AudioSystem.isLineSupported(info)) {
```

```
        System.out.println("Line not supported");
        System.exit(0);
    }

    TargetDataLine line = (TargetDataLine) AudioSystem.getLine(info);
    line.open(format);
    line.start();

    InputStream audioStream = new AudioInputStream(line);
    return audioStream;
}

private static AwsCredentialsProvider getCredentials() {
    return DefaultCredentialsProvider.create();
}

private static StartStreamTranscriptionRequest getRequest(Integer
mediaSampleRateHertz) {
    return StartStreamTranscriptionRequest.builder()
        .languageCode(LanguageCode.EN_US.toString())
        .mediaEncoding(MediaEncoding.PCM)
        .mediaSampleRateHertz(mediaSampleRateHertz)
        .build();
}

private static StartStreamTranscriptionResponseHandler getResponseHandler() {
    return StartStreamTranscriptionResponseHandler.builder()
        .onResponse(r -> {
            System.out.println("Received Initial response");
        })
        .onError(e -> {
            System.out.println(e.getMessage());
            StringWriter sw = new StringWriter();
            e.printStackTrace(new PrintWriter(sw));
            System.out.println("Error Occurred: " + sw.toString());
        })
        .onComplete(() -> {
            System.out.println("=== All records stream successfully ===");
        })
        .subscriber(event -> {
            List<Result> results = ((TranscriptEvent)
event).transcript().results();
            if (results.size() > 0) {
```

```
                if (!
results.get(0).alternatives().get(0).transcript().isEmpty()) {

    System.out.println(results.get(0).alternatives().get(0).transcript());
                }
            }
        })
        .build();
    }

    private InputStream getStreamFromFile(String myMediaFileName) {
        try {
            File inputFile = new
File(getClass().getClassLoader().getResource(myMediaFileName).getFile());
            InputStream audioStream = new FileInputStream(inputFile);
            return audioStream;
        } catch (FileNotFoundException e) {
            throw new RuntimeException(e);
        }
    }

    private static class AudioStreamPublisher implements Publisher<AudioStream> {
        private final InputStream inputStream;
        private static Subscription currentSubscription;

        private AudioStreamPublisher(InputStream inputStream) {
            this.inputStream = inputStream;
        }

        @Override
        public void subscribe(Subscriber<? super AudioStream> s) {

            if (this.currentSubscription == null) {
                this.currentSubscription = new SubscriptionImpl(s, inputStream);
            } else {
                this.currentSubscription.cancel();
                this.currentSubscription = new SubscriptionImpl(s, inputStream);
            }
            s.onSubscribe(currentSubscription);
        }
    }

    public static class SubscriptionImpl implements Subscription {
```



```
private static final int CHUNK_SIZE_IN_BYTES = 1024 * 1;
private final Subscriber<? super AudioStream> subscriber;
private final InputStream inputStream;
private ExecutorService executor = Executors.newFixedThreadPool(1);
private AtomicLong demand = new AtomicLong(0);

SubscriptionImpl(Subscriber<? super AudioStream> s, InputStream inputStream)
{
    this.subscriber = s;
    this.inputStream = inputStream;
}

@Override
public void request(long n) {
    if (n <= 0) {
        subscriber.onError(new IllegalArgumentException("Demand must be
positive"));
    }

    demand.getAndAdd(n);

    executor.submit(() -> {
        try {
            do {
                ByteBuffer audioBuffer = getNextEvent();
                if (audioBuffer.remaining() > 0) {
                    AudioEvent audioEvent =
audioEventFromBuffer(audioBuffer);
                    subscriber.onNext(audioEvent);
                } else {
                    subscriber.onComplete();
                    break;
                }
            } while (demand.decrementAndGet() > 0);
        } catch (Exception e) {
            subscriber.onError(e);
        }
    });
}

@Override
public void cancel() {
    executor.shutdown();
}
```

```
private ByteBuffer getNextEvent() {
    ByteBuffer audioBuffer = null;
    byte[] audioBytes = new byte[CHUNK_SIZE_IN_BYTES];

    int len = 0;
    try {
        len = inputStream.read(audioBytes);

        if (len <= 0) {
            audioBuffer = ByteBuffer.allocate(0);
        } else {
            audioBuffer = ByteBuffer.wrap(audioBytes, 0, len);
        }
    } catch (IOException e) {
        throw new UncheckedIOException(e);
    }

    return audioBuffer;
}

private AudioEvent audioEventFromBuffer(ByteBuffer bb) {
    return AudioEvent.builder()
        .audioChunk(SdkBytes.fromByteBuffer(bb))
        .build();
}
}
```

JavaScript

```
const {
    TranscribeStreamingClient,
    StartStreamTranscriptionCommand,
} = require("@aws-sdk/client-transcribe-streaming");
const { createReadStream } = require("fs");
const { join } = require("path");

const audio = createReadStream(join(__dirname, "my-media-file.flac"),
    { highWaterMark: 1024 * 16});

const LanguageCode = "en-US";
const MediaEncoding = "pcm";
```

```
const MediaSampleRateHertz = "16000";
const credentials = {
  "accessKeyId": "",
  "secretAccessKey": "",
};
async function startRequest() {
  const client = new TranscribeStreamingClient({
    region: "us-west-2",
    credentials
  });

  const params = {
    LanguageCode,
    MediaEncoding,
    MediaSampleRateHertz,
    AudioStream: (async function* () {
      for await (const chunk of audio) {
        yield {AudioEvent: {AudioChunk: chunk}};
      }
    })(),
  };
  const command = new StartStreamTranscriptionCommand(params);
  // Send transcription request
  const response = await client.send(command);
  // Start to print response
  try {
    for await (const event of response.TranscriptResultStream) {
      console.log(JSON.stringify(event));
    }
  } catch(err) {
    console.log("error")
    console.log(err)
  }
}
startRequest();
```

Python

L'exemple suivant est un programme Python qui transcrit les flux de streaming audio.

Pour exécuter cet exemple, notez les éléments suivants :

- Vous devez utiliser ce [kit SDK pour Python](#).

- La fréquence d'échantillonnage que vous spécifiez doit correspondre à la fréquence d'échantillonnage réelle de votre flux audio.

```
import asyncio
# This example uses aiofile for asynchronous file reads.
# It's not a dependency of the project but can be installed
# with `pip install aiofile`.
import aiofile

from amazon_transcribe.client import TranscribeStreamingClient
from amazon_transcribe.handlers import TranscriptResultStreamHandler
from amazon_transcribe.model import TranscriptEvent

"""
Here's an example of a custom event handler you can extend to
process the returned transcription results as needed. This
handler will simply print the text out to your interpreter.
"""

class MyEventHandler(TranscriptResultStreamHandler):
    async def handle_transcript_event(self, transcript_event: TranscriptEvent):
        # This handler can be implemented to handle transcriptions as needed.
        # Here's an example to get started.
        results = transcript_event.transcript.results
        for result in results:
            for alt in result.alternatives:
                print(alt.transcript)

async def basic_transcribe():
    # Set up our client with your chosen Region
    client = TranscribeStreamingClient(region = "us-west-2")

    # Start transcription to generate async stream
    stream = await client.start_stream_transcription(
        language_code = "en-US",
        media_sample_rate_hz = 16000,
        media_encoding = "pcm",
    )

    async def write_chunks():
        # NOTE: For pre-recorded files longer than 5 minutes, the sent audio
        # chunks should be rate limited to match the real-time bitrate of the
```

```
# audio stream to avoid signing issues.
async with aiofile.AIOFile('filepath/my-media-file.flac', 'rb') as afp:
    reader = aiofile.Reader(afp, chunk_size = 1024 * 16)
    async for chunk in reader:
        await stream.input_stream.send_audio_event(audio_chunk = chunk)
    await stream.input_stream.end_stream()

# Instantiate our handler and start processing events
handler = MyEventHandler(stream.output_stream)
await asyncio.gather(write_chunks(), handler.handle_events())

loop = asyncio.get_event_loop()
loop.run_until_complete(basic_transcribe())
loop.close()
```

C++

Reportez-vous au chapitre des exemples de code pour l'[exemple de kit SDK C++ pour le streaming](#).

Utilisation de ce service avec un AWS SDK

AWS des kits de développement logiciel (SDK) sont disponibles pour de nombreux langages de programmation populaires. Chaque SDK fournit une API, des exemples de code et de la documentation qui facilitent la création d'applications par les développeurs dans leur langage préféré.

Documentation SDK	Exemples de code
AWS SDK for C++	AWS SDK for C++ exemples de code
AWS CLI	AWS CLI exemples de code
AWS SDK for Go	AWS SDK for Go exemples de code
AWS SDK for Java	AWS SDK for Java exemples de code
AWS SDK for JavaScript	AWS SDK for JavaScript exemples de code
Kit AWS SDK pour Kotlin	Kit AWS SDK pour Kotlin exemples de code
AWS SDK for .NET	AWS SDK for .NET exemples de code

Documentation SDK	Exemples de code
AWS SDK for PHP	AWS SDK for PHP exemples de code
AWS Tools for PowerShell	Outils pour des exemples PowerShell de code
AWS SDK for Python (Boto3)	AWS SDK for Python (Boto3) exemples de code
AWS SDK for Ruby	AWS SDK for Ruby exemples de code
Kit AWS SDK pour Rust	Kit AWS SDK pour Rust exemples de code
AWS SDK pour SAP ABAP	AWS SDK pour SAP ABAP exemples de code
Kit AWS SDK pour Swift	Kit AWS SDK pour Swift exemples de code

Pour voir des exemples spécifiques à ce service, consultez [Exemples de code pour Amazon Transcribe à l'aide de kits de développement logiciel AWS](#).

Exemple de disponibilité

Vous n'avez pas trouvé ce dont vous avez besoin ? Demandez un exemple de code en utilisant le lien [Provide feedback](#) (Fournir un commentaire) en bas de cette page.

Transcription avec HTTP ou WebSockets

Amazon Transcribe prend en charge le protocole HTTP pour les transcriptions par lots (HTTP/1.1) et en streaming (HTTP/2). WebSockets sont pris en charge pour les transcriptions en streaming.

Si vous transcrivez un fichier multimédia situé dans un Amazon S3 compartiment, vous effectuez une transcription par lots. Si vous transcrivez un flux de données audio en temps réel, vous effectuez une transcription en continu.

HTTP et vous WebSockets obligent à authentifier votre demande à l'aide des en-têtes AWS Signature version 4. Reportez-vous à [la section Signature de demandes d'AWSAPI](#) pour plus d'informations.

Transcriptions Batch

Vous pouvez effectuer une requête HTTP par lots à l'aide des en-têtes suivants :

- hôte
- x-amz-target
- type de contenu
- x-amz-content-sha256
- x-amz-date
- autorisation

Voici un exemple de `StartTranscriptionJob` demande :

```
POST /transcribe HTTP/1.1
host: transcribe.us-west-2.amazonaws.com
x-amz-target: com.amazonaws.transcribe.Transcribe.StartTranscriptionJob
content-type: application/x-amz-json-1.1
x-amz-content-sha256: string
x-amz-date: YYYYMMDDTHHMMSSZ
authorization: AWS4-HMAC-SHA256 Credential=access-key/YYYYMMSS/us-west-2/transcribe/
aws4_request, SignedHeaders=content-type;host;x-amz-content-sha256;x-amz-date;x-amz-
target;x-amz-security-token, Signature=string

{
  "TranscriptionJobName": "my-first-transcription-job",
  "LanguageCode": "en-US",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
  },
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "OutputKey": "my-output-files/"
}
```

Les opérations et paramètres supplémentaires sont répertoriés dans la [référence de l'API](#) ; les paramètres communs à toutes les opérations d'AWSAPI sont répertoriés dans la section [Paramètres communs](#). Les autres éléments de signature sont détaillés dans [Éléments d'une demande deAWS signature version 4](#).

Transcriptions en streaming

Les transcriptions en streaming utilisant HTTP/2 et WebSockets sont plus complexes que l'utilisation de SDK. Nous vous recommandons de consulter [Configuration d'une transcription en streaming](#) cette section avant de configurer votre premier stream.

Pour plus d'informations sur ces méthodes, reportez-vous à [Configuration d'un flux HTTP/2](#) ou [Configuration d'un WebSocket stream](#).

Note

Nous vous recommandons vivement d'utiliser un SDK pour les transcriptions en streaming. Pour obtenir la liste des SDK pris en charge, reportez-vous à [Langages de programmation pris en charge](#).

Transcription des flux audio en streaming

Grâce au Amazon Transcribe streaming, vous pouvez produire des transcriptions en temps réel pour votre contenu multimédia. Contrairement aux transcriptions par lots, qui impliquent le téléchargement de fichiers multimédia, le streaming multimédia est diffusé Amazon Transcribe en temps réel. Amazon Transcribe renvoie ensuite une transcription, également en temps réel.

Le streaming peut inclure des médias préenregistrés (films, musique et podcasts) et des médias en temps réel (émissions d'information en direct). Les cas d'utilisation courants du streaming Amazon Transcribe incluent le sous-titrage codé en direct pour les événements sportifs et la surveillance en temps réel du son du centre d'appels.

Le contenu en streaming est diffusé sous la forme d'une série de paquets de données séquentiels, ou « blocs », qu' Amazon Transcribe transcrit instantanément. Les avantages de l'utilisation du streaming par lots incluent des speech-to-text fonctionnalités en temps réel dans vos applications et des temps de transcription plus rapides. Cependant, cette vitesse accrue peut avoir des limites en termes de précision dans certains cas.

Amazon Transcribe propose les options suivantes pour le streaming :

- [Kits SDK](#) (préférence)
- [HTTP/2](#)
- [WebSockets](#)
- [AWS Management Console](#)

Pour transcrire du son en streaming dans le AWS Management Console, parlez dans le microphone de votre ordinateur.

Tip

Pour des exemples de code SDK, reportez-vous au [référentiel AWS Samples](#) sur GitHub.

Les formats audio pris en charge pour les transcriptions en streaming sont les suivants :

- FLAC

- Audio encodé en OPUS dans un conteneur Ogg
- PCM (uniquement les formats audio Little Endian 16 bits signés, ce qui n'inclut pas WAV)

Les formats sans perte (FLAC ou PCM) sont recommandés.

Note

Les transcriptions en streaming ne sont pas prises en charge dans toutes les langues. Reportez-vous à la colonne « Entrée de données » du [tableau des langues prises en charge](#) pour plus de détails.

Pour connaître la disponibilité des transcriptions en streaming Amazon Transcribe dans la région, voir : [Amazon Transcribe Points de terminaison et quotas](#).

Bonnes pratiques

Les recommandations suivantes améliorent l'efficacité de la transcription en streaming :

- Si possible, utilisez un son codé en PCM.
- Assurez-vous que votre diffusion est aussi proche que possible du temps réel.
- La latence dépend de la taille de vos blocs audio. Si vous êtes en mesure de spécifier la taille des blocs en fonction de votre type d'audio (par exemple avec PCM), réglez chaque segment entre 50 ms et 200 ms. Vous pouvez calculer la taille du fragment audio par la formule suivante :

```
chunk_size_in_bytes = chunk_duration_in_millisecond / 1000 * audio_sample_rate * 2
```

- Utilisez une taille de bloc uniforme.
- Assurez-vous de spécifier correctement le nombre de canaux audio.
- Avec du son PCM monocanal, chaque échantillon est composé de deux octets, de sorte que chaque bloc doit être constitué d'un nombre pair d'octets.
- Avec du son PCM à deux canaux, chaque échantillon est composé de quatre octets, de sorte que chaque bloc doit être un multiple de quatre octets.
- Lorsque votre flux audio ne contient aucun signal vocal, encodez et envoyez la même quantité de silence. Par exemple, le silence pour PCM est un flux de zéro octet.

- Assurez-vous de spécifier le taux d'échantillonnage correct pour votre audio. Si possible, enregistrez à une fréquence d'échantillonnage de 16 000 Hz ; cela constitue le meilleur compromis entre la qualité et le volume de données envoyées sur le réseau. Notez que la plupart des microphones haut de gamme enregistrent à 44 100 Hz ou 48 000 Hz.

Streaming et résultats partiels

Comme le streaming fonctionne en temps réel, les transcriptions sont produites sous forme de résultats partiels. Amazon Transcribe divise le flux audio entrant en fonction de segments vocaux naturels, tels qu'un changement de haut-parleur ou une pause audio. La transcription est renvoyée à votre application sous la forme d'un flux d'événements de transcription, chaque réponse contenant plus de paroles transcrites que la précédente, jusqu'à la transcription de la totalité du segment.

Une approximation est présentée dans le bloc de code suivant. Vous pouvez suivre ce processus en action en vous connectant à la [AWS Management Console](#), en sélectionnant Transcription en temps réel et en parlant dans votre microphone. Observez le volet Sortie de transcription pendant que vous parlez.

Dans cet exemple, chaque ligne est le résultat partiel d'un segment audio.

```
The  
The Amazon.  
The Amazon is  
The Amazon is the law.  
The Amazon is the largest  
The Amazon is the largest ray  
The Amazon is the largest rain for  
The Amazon is the largest rainforest.  
The Amazon is the largest rainforest on the  
The Amazon is the largest rainforest on the planet.
```

Ces résultats partiels sont présents dans votre sortie de transcription dans les objets [Results](#). Ce bloc d'objets contient également un `IsPartial` champ. Si ce champ a la valeur `true`, cela signifie que votre segment de transcription n'est pas encore terminé. Vous pouvez voir la différence entre un segment incomplet et un segment complet ci-dessous :

```
"IsPartial": true (incomplete segment)  
  
"Transcript": "The Amazon is the largest rainforest."
```

```
"EndTime": 4.545,  
"IsPartial": true,  
"ResultId": "12345a67-8bc9-0de1-2f34-a5b678c90d12",  
"StartTime": 0.025
```

"IsPartial": false (complete segment)

```
"Transcript": "The Amazon is the largest rainforest on the planet."
```

```
"EndTime": 6.025,  
"IsPartial": false,  
"ResultId": "34567e89-0fa1-2bc3-4d56-78e90123456f",  
"StartTime": 0.025
```

Chaque mot d'un segment complet est associé à un score de confiance, qui est une valeur comprise entre 0 et 1. Une valeur plus élevée indique une plus grande probabilité que le mot soit correctement transcrit.

Tip

L'StartTime et l'EndTime d'un segment audio peuvent être utilisées pour synchroniser la sortie de transcription avec le dialogue vidéo.

Si vous exécutez une application nécessitant une faible latence, vous pouvez utiliser la [stabilisation des résultats partiels](#).

Stabilisation des résultats partiels

Amazon Transcribe commence à renvoyer les résultats de transcription dès que vous commencez à diffuser votre audio. Il renvoie ces résultats partiels de manière incrémentielle jusqu'à ce qu'il génère un résultat final au niveau d'un segment de parole naturelle. Un segment de parole naturelle est un discours continu qui contient une pause ou un changement de locuteur.

Amazon Transcribe continue de produire des résultats partiels jusqu'à ce qu'il génère le résultat de transcription final pour un segment vocal. La reconnaissance vocale étant susceptible de modifier des mots à mesure qu'elle gagne en contexte, les transcriptions en streaming peuvent légèrement changer à chaque nouvelle sortie de résultat partiel.

« Transcript » :	"EndTime":	"IsPartial":
<p>The Amazon is the largest rainforest on the planet.</p> <p>The Amazon is the largest rainforest on the planet.</p>		

Stabilisation des résultats partiels activée (haute stabilité)

<p>The</p> <p>The</p> <p>The Amazon.</p> <p>The Amazon is</p> <p>The Amazon is the large</p> <p>The Amazon is the largest</p> <p>The Amazon is the largest rainfall.</p> <p>The Amazon is the largest rain forest.</p> <p>The Amazon is the largest rain forest on</p> <p>The Amazon is the largest rain forest on the planet.</p> <p>The Amazon is the largest rain forest on the planet.</p> <p>The Amazon is the largest rain forest on the planet.</p> <p>The Amazon is the largest rain forest on the planet.</p> <p>The Amazon is the largest rain forest on the planet.</p>	<p>0.515</p> <p>1.015</p> <p>1.515</p> <p>2.015</p> <p>2.515</p> <p>3.015</p> <p>3.515</p> <p>4.015</p> <p>4.515</p> <p>5.015</p> <p>5.515</p> <p>6.015</p> <p>6.335</p> <p>6.335</p>	<p>true</p> <p>true</p> <p>true</p> <p>true</p> <p>true</p> <p>true</p> <p>true</p> <p>true</p> <p>true</p> <p>true</p> <p>true</p> <p>true</p> <p>false</p>
--	---	--

Lorsque vous activez la stabilisation des résultats partiels, Amazon Transcribe utilise un `Stable` champ pour indiquer si un élément est stable, le terme « élément » faisant référence à un mot transcrit ou à un signe de ponctuation. Les valeurs pour `Stable` sont `true` ou `false`. Les éléments marqués `false` (non stables) sont plus susceptibles de changer à mesure que votre segment est transcrit. À l'inverse, les éléments marqués `true` (stables) ne changeront pas.

Vous pouvez choisir de rendre les mots instables afin que vos sous-titres s'alignent sur le discours. Même si les sous-titres changent légèrement à mesure que du contexte est ajouté, l'expérience utilisateur est améliorée par rapport aux rafales de texte périodiques, qui peuvent ou non correspondre à la parole.

Vous pouvez également choisir d'afficher les mots instables dans un format différent, par exemple en italique, pour indiquer aux utilisateurs que ces mots peuvent changer. L'affichage des résultats partiels limite la quantité de texte affichée à un moment donné. Cela peut être important lorsque vous êtes confronté à des contraintes d'espace, comme dans le cas des sous-titres vidéo.

Approfondissez vos connaissances avec le blog AWS Machine Learning

Pour en savoir plus sur l'amélioration de la précision grâce aux transcriptions en temps réel, consultez les sections suivantes :

- [Améliorez l'expérience de transcription en streaming grâce à la stabilisation Amazon Transcribe partielle des résultats](#)
- ["What was that?" Increasing subtitle accuracy for live broadcasts using Amazon Transcribe](#)

Exemple de sortie de stabilisation des résultats partiels

L'exemple de sortie suivant montre des indicateurs `Stable` pour un segment incomplet (`"IsPartial": true`). Vous pouvez constater que les mots « to » et « Amazon » ne sont pas stables et peuvent donc changer avant que le segment ne soit finalisé.

```
"Transcript": {
  "Results": [
    {
      "Alternatives": [
        {
          "Items": [
            {
              "Content": "Welcome",
```

```
        "EndTime": 2.4225,
        "Stable": true,
        "StartTime": 1.65,
        "Type": "pronunciation",
        "VocabularyFilterMatch": false
    },
    {
        "Content": "to",
        "EndTime": 2.8325,
        "Stable": false,
        "StartTime": 2.4225,
        "Type": "pronunciation",
        "VocabularyFilterMatch": false
    },
    {
        "Content": "Amazon",
        "EndTime": 3.635,
        "Stable": false,
        "StartTime": 2.8325,
        "Type": "pronunciation",
        "VocabularyFilterMatch": false
    },
    {
        "Content": ".",
        "EndTime": 3.635,
        "Stable": false,
        "StartTime": 3.635,
        "Type": "punctuation",
        "VocabularyFilterMatch": false
    }
    ],
    "Transcript": "Welcome to Amazon."
}
],
"EndTime": 4.165,
"IsPartial": true,
"ResultId": "12345a67-8bc9-0de1-2f34-a5b678c90d12",
"StartTime": 1.65
}
]
```


Configuration d'une transcription en streaming

Cette section développe la section principale consacrée au [streaming](#). Il est destiné à fournir des informations aux utilisateurs qui souhaitent configurer leur flux avec HTTP/2 ou WebSockets directement, plutôt qu'avec un AWS SDK. Les informations de cette section peuvent également être utilisées pour créer votre propre kit SDK.

Important

Nous vous recommandons vivement d'utiliser des SDK plutôt que d'utiliser HTTP/2 directement. Les kits SDK constituent la méthode la plus simple et la plus fiable pour transcrire des flux de données. Pour démarrer le streaming à l'aide d'un AWS SDK, consultez [Transcription avec les SDK AWS](#).

Configuration d'un flux HTTP/2

Les composants clés d'un [protocole HTTP/2 pour le](#) streaming de demandes de transcription Amazon Transcribe sont les suivants :

- Une image d'en-tête. Il contient les en-têtes HTTP/2 de votre demande et une signature dans l'en-tête d'autorisation qui Amazon Transcribe sert de signature initiale pour signer les trames de données.
- Une ou plusieurs trames de message dans [encodage de flux d'événements](#) qui contiennent des métadonnées et des octets audio bruts.
- Une image de fin. Il s'agit d'un message signé dans un [encodage de flux d'événements](#) avec un corps vide.

Note

Amazon Transcribe ne prend en charge qu'un seul flux par session HTTP/2. Si vous essayez d'utiliser plusieurs flux, votre demande de transcription échouera.

1. Associez la politique suivante au IAM rôle qui fait la demande. Voir [Ajouter IAM des politiques](#) pour plus d'informations.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "my-transcribe-http2-policy",
      "Effect": "Allow",
      "Action": "transcribe:StartStreamTranscription",
      "Resource": "*"
    }
  ]
}
```

2. Pour démarrer la session, envoyez une requête HTTP/2 à Amazon Transcribe.

```
POST /stream-transcription HTTP/2
host: transcribestreaming.us-west-2.amazonaws.com
X-Amz-Target: com.amazonaws.transcribe.Transcribe.StartStreamTranscription
Content-Type: application/vnd.amazon.eventstream
X-Amz-Content-Sha256: string
X-Amz-Date: YYYYMMDDTHHMMSSZ
Authorization: AWS4-HMAC-SHA256 Credential=access-key/YYYYMMDD/us-west-2/transcribe/aws4_request, SignedHeaders=content-type;host;x-amz-content-sha256;x-amz-date;x-amz-target;x-amz-security-token, Signature=string
x-amzn-transcribe-language-code: en-US
x-amzn-transcribe-media-encoding: flac
x-amzn-transcribe-sample-rate: 16000
transfer-encoding: chunked
```

Les opérations et paramètres supplémentaires sont répertoriés dans la [Référence d'API](#). Les paramètres communs à toutes les opérations d'API AWS sont répertoriés dans la section [Paramètres courants](#).

Amazon Transcribe envoie la réponse suivante :

```
HTTP/2.0 200
x-amzn-transcribe-language-code: en-US
x-amzn-transcribe-media-encoding: flac
x-amzn-transcribe-sample-rate: 16000
x-amzn-request-id: 8a08df7d-5998-48bf-a303-484355b4ab4e
x-amzn-transcribe-session-id: b4526fcf-5eee-4361-8192-d1cb9e9d6887
content-type: application/json
```

3. Créez un événement audio contenant vos données audio. Combinez les en-têtes décrits dans le tableau suivant avec une partie des octets audio dans un message encodé en événement. Pour créer la charge utile du message d'événement, utilisez une mémoire tampon en octet brut.

Longueur en octets du nom de l'en-tête	Nom de l'en-tête (chaîne)	Type de valeur de l'en-tête	Longueur en octets de la chaîne de valeur	Valeur de chaîne (UTF-8)
13	:content-type	7	24	application/octet-stream
11	:event-type	7	10	AudioEvent
13	:message-type	7	5	event

Les données binaires de cet exemple de demande sont encodées en base64. Dans une requête réelle, les données sont des octets bruts.

```
:content-type: "application/vnd.amazon.eventstream"
:event-type: "AudioEvent"
:message-type: "event"
Uk1GRjzxpQBxQVZfZm10IBAAAAABAAEAgD4AAAB9AAACABAAZGF0YVtWpQAAAAAAAAAAAAAAAAAAD//wIA/
f8EAA==
```

4. Créez un message audio contenant vos données audio.
 - a. L'image de données de votre message audio contient des en-têtes encodés en événement comprenant la date actuelle et une signature pour la partie audio et l'événement audio.

Longueur en octets du nom de l'en-tête	Nom de l'en-tête (chaîne)	Type de valeur de l'en-tête	Longueur en octets de la chaîne de valeur	Valeur
16	:chunk-signature	6	varie	signature générée

Longueur en octets du nom de l'en-tête	Nom de l'en-tête (chaîne)	Type de valeur de l'en-tête	Longueur en octets de la chaîne de valeur	Valeur
5	:date	8	8	timestamp

Les données binaires de cette demande sont encodées en base64. Dans une requête réelle, les données sont des octets bruts.

```
:date: 2019-01-29T01:56:17.291Z
:chunk-signature: signature
```

```
AAAA0gAAAIKVoRFcTTcjb250ZW50LXR5cGUHABhhcHBsaWNhdG1vbi9vY3R1dC1zdHJ1YW0L0mV2ZW50LXR5cGUHAAPBdWRpb0V2ZW50DTptZXNzYwdlLXR5cGUHAAVldmVudAxDb256ZW50LVR5cGUHABphcHBsaWNhdG1vbi94LWFtei1qc29uLTEuMVJJRkY88T0AV0FWRWZtdCAQAAAAAQABAIA
+AAAAfQAAAQAGRhdGFU8D0AAAAA
AAAAAAAAAAAA//8CAP3/BAC7QLFf
```

- b. Créez une chaîne à signer, comme indiqué à la section [Créer une chaîne à signer pour Signature Version 4](#). Votre chaîne suit le format suivant :

```
String stringToSign =
"AWS4-HMAC-SHA256" +
"\n" +
DateTime +
"\n" +
Keypath +
"\n" +
Hex(priorSignature) +
"\n" +
HexHash(nonSignatureHeaders) +
"\n" +
HexHash(payload);
```

- *DateTime*: date et heure de création de la signature. Le format est YYYYMMDDTHHMMSSZ, où YYYY=année, MM=mois, DD=jour, HH=heure, MM=minute, SS=secondes, et 'T' et 'Z' sont des caractères fixes. Pour plus d'informations, consultez la section [Gestion des dates dans Signature Version 4](#).

- Keypath : portée de la signature au format `date/region/service/aws4_request`. Par exemple, `20220127/us-west-2/transcribe/aws4_request`.
 - Hex : fonction qui encode l'entrée sous forme de représentation hexadécimale.
 - priorSignature : signature pour l'image précédente. Pour la première image de données, utilisez la signature de l'image d'en-tête.
 - HexHash: fonction qui crée d'abord un hachage SHA-256 de son entrée, puis utilise la fonction Hex pour coder le hachage.
 - non SignatureHeaders : L' DateTime en-tête est codé sous forme de chaîne.
 - payload : mémoire tampon en octets contenant les données d'événement audio.
- c. Dérivez une clé de signature de votre clé d'accès AWS secrète et utilisez-la pour signer `lestringToSign`. La clé dérivée est spécifique à la date, au service et à la Région AWS, pour un niveau de protection plus élevé. Pour plus d'informations, consultez la section [Calculer la signature pour Signature Version 4 AWS](#).

Assurez-vous de mettre en œuvre la fonction `GetSignatureKey` pour dériver votre clé de signature. Si vous n'avez pas encore dérivé de clé de signature, reportez-vous à la section [Exemples de dérivation d'une clé de signature pour Signature Version 4](#).

```
String signature = HMACSHA256(derivedSigningKey, stringToSign);
```

- HMACSHA256 : fonction qui crée une signature à l'aide de la fonction de hachage SHA-256.
- dérivée `SigningKey` : la clé de signature de la version 4 de Signature.
- `string ToSign` : chaîne que vous avez calculée pour le bloc de données.

Une fois que vous avez calculé la signature pour l'image de données, construisez une mémoire tampon en octets contenant la date, la signature et la charge utile de l'événement audio. Envoyez le tableau en octets à Amazon Transcribe pour la transcription.

5. Pour indiquer que le flux audio est complet, envoyez une image de fin (une image de données vide) ne contenant que la date et une signature. L'image de fin se construit de la même façon qu'une image de données.

Amazon Transcribe répond par un flux d'événements de transcription, envoyé à votre application. Cette réponse est encodée en flux d'événements. Elle contient le préambule standard et les en-têtes suivants :

Longueur en octets du nom de l'en-tête	Nom de l'en-tête (chaîne)	Type de valeur de l'en-tête	Longueur en octets de la chaîne de valeur	Valeur de chaîne (UTF-8)
13	:content-type	7	16	application/json
11	:event-type	7	15	TranscriptEvent
13	:message-type	7	5	event

Les événements sont envoyés en octet brut. Dans cet exemple, les octets sont encodés en base64.

```
AAAAUwAAAEP1RHpYBTpkYXR1CAAAAWiXUkMLEDpjaHVuay1zaWduYXR1cmUGACct6Zy+uymwEK2Srlp/
zVBI
5eGn83jdBwCaRUBJA+eaDafqjqI=
```

Pour voir les résultats de la transcription, décidez les octets bruts à l'aide de l'encodage de flux d'événements :

```
:content-type: "application/vnd.amazon.eventstream"
:event-type: "TranscriptEvent"
:message-type: "event"

{
  "Transcript":
    {
      "Results":
        [
          results
        ]
    }
}
```

6. Pour mettre fin à votre flux, envoyez un événement audio vide à Amazon Transcribe. Créez l'événement audio comme n'importe quel événement, mais avec une charge utile vide. Signez l'événement et incluez la signature dans l'en-tête :chunk-signature, comme suit :

```
:date: 2019-01-29T01:56:17.291Z
:chunk-signature: signature
```

Gestion des erreurs de streaming HTTP/2

Si une erreur se produit lors du traitement de votre flux multimédia, Amazon Transcribe envoie une réponse d'exception. La réponse est encodée en flux d'événement.

Elle contient le préambule standard et les en-têtes suivants :

Longueur en octets du nom de l'en-tête	Nom de l'en-tête (chaîne)	Type de valeur de l'en-tête	Longueur en octets de la chaîne de valeur	Valeur de chaîne (UTF-8)
13	:content-type	7	16	application/json
11	:event-type	7	19	BadRequestException
13	:message-type	7	9	exception

Lorsque la réponse d'exception est décodée, elle contient les informations suivantes :

```
:content-type: "application/vnd.amazon.eventstream"
:event-type: "BadRequestException"
:message-type: "exception"
```

Exception message

Configuration d'un WebSocket stream

Les composants clés d'un [WebSocketprotocole](#) de diffusion en continu des demandes de transcription Amazon Transcribe sont les suivants :

- La demande de mise à niveau. Il contient les paramètres de requête pour votre demande, ainsi qu'une signature qui Amazon Transcribe sert de signature initiale pour signer les blocs de données.

- Une ou plusieurs trames de message dans [encodage de flux d'événements](#) qui contiennent des métadonnées et des octets audio bruts.
- Une image de fin. Il s'agit d'un message signé dans un [encodage de flux d'événements](#) avec un corps vide.

Note

Amazon Transcribe ne prend en charge qu'un seul flux par WebSocket session. Si vous essayez d'utiliser plusieurs flux, votre demande de transcription échouera.

1. Associez la politique suivante au IAM rôle qui fait la demande. Voir [Ajouter IAM des politiques](#) pour plus d'informations.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "my-transcribe-websocket-policy",
      "Effect": "Allow",
      "Action": "transcribe:StartStreamTranscriptionWebSocket",
      "Resource": "*"
    }
  ]
}
```

2. Pour démarrer la session, créez une URL présignée au format suivant. Les sauts de ligne ont été ajoutés pour faciliter la lecture.

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/stream-transcription-
websocket?
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=access-key%2FYYYYMMDD%2Fus-west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=YYYYMMDDTHHMMSSZ
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=string
&X-Amz-SignedHeaders=content-type%3Bhost%3Bx-amz-date
&language-code=en-US
&media-encoding=flac
```



```
&sample-rate=16000
```

Note

La valeur maximale pour X-Amz-Expires est de 300 (5 minutes).

Les opérations et paramètres supplémentaires sont répertoriés dans la [Référence d'API](#). Les paramètres communs à toutes les opérations d'API AWS sont répertoriés dans la section [Paramètres courants](#).

Pour construire l'URL de votre demande et créer la [signature Signature Version 4](#), reportez-vous aux étapes suivantes. Les exemples sont en pseudo-code.

- a. Créez une demande canonique. Une demande canonique est une chaîne qui inclut des informations de votre demande dans un format normalisé. Cela garantit que lors de la AWS réception de la demande, elle peut calculer la même signature que celle que vous avez créée pour votre URL. Pour plus d'informations, consultez la section [Créer une demande canonique pour Signature Version 4](#).

```
# HTTP verb
method = "GET"
# Service name
service = "transcribe"
# Region
region = "us-west-2"
# Amazon Transcribe streaming endpoint
endpoint = "wss://transcribestreaming.us-west-2.amazonaws.com:8443"
# Host
host = "transcribestreaming.us-west-2.amazonaws.com:8443"
# Date and time of request
amz-date = YYYYMMDDTHHMMSSZ
# Date without time for credential scope
datestamp = YYYYMMDD
```

- b. Créez un URI canonique qui est la partie de l'URI comprise entre le domaine et la chaîne de requête.

```
canonical_uri = "/stream-transcription-websocket"
```

- c. Créez les en-têtes canoniques et les en-têtes signés. Notez la barre oblique `\n` dans les en-têtes canoniques.
- Ajoutez le nom d'en-tête en minuscules suivi du signe deux points (:).
 - Ajoutez une liste de valeurs séparées par des virgules pour cet en-tête. Ne triez pas les valeurs dans les en-têtes ayant plusieurs valeurs.
 - Ajoutez une nouvelle ligne (`\n`).

```
canonical_headers = "host:" + host + "\n"
signed_headers = "host"
```

- d. Associez l'algorithme à l'algorithme de hachage. Utilisez SHA-256.

```
algorithm = "AWS4-HMAC-SHA256"
```

- e. Créez la portée des informations d'identification qui déterminera la clé dérivée de la date, de la Région AWS et du service. Par exemple, `20220127/us-west-2/transcribe/aws4_request`.

```
credential_scope = datestamp + "/" + region + "/" + service + "/" +
    "aws4_request"
```

- f. Créez la chaîne de requête canonique. Les valeurs des chaîne de requête doivent être encodées en URI et triées par nom.
- Triez les noms de paramètre selon le point de code de caractère dans l'ordre croissant. Les paramètres avec des noms en double doivent être triés par valeur. Par exemple, un nom de paramètre qui commence par la lettre majuscule F précède un nom de paramètre qui commence par la lettre minuscule b.
 - N'encodez pas de l'URI les caractères autorisés que RFC 3986 définit : A à Z, a à z, 0 à 9, le trait d'union (-), le trait de soulignement (_), le point final (.) et le tilde (~).
 - Encodez de pourcentage tous les autres caractères avec `%XY`, où X et Y représentent les caractères hexadécimaux (0 à 9 et les lettres majuscules A à F). Par exemple, le caractère espace doit être encodé sous la forme `%20` (sans inclure « + », comme le font certains schémas d'encodage) et les caractères UTF-8 étendus doivent être sous la forme `%XY%ZA%BC`.
 - Encodez deux fois tous les caractères égaux à (=) dans les valeurs de paramètre.

```
canonical_querystring = "X-Amz-Algorithm=" + algorithm
canonical_querystring += "&X-Amz-Credential=" + URI-encode(access key + "/" +
  credential_scope)
canonical_querystring += "&X-Amz-Date=" + amz_date
canonical_querystring += "&X-Amz-Expires=300"
canonical_querystring += "&X-Amz-Security-Token=" + token
canonical_querystring += "&X-Amz-SignedHeaders=" + signed_headers
canonical_querystring += "&language-code=en-US&media-encoding=flac&sample-
rate=16000"
```

- g. Créez un hachage de la charge utile. Pour une demande GET, la charge utile est une chaîne vide.

```
payload_hash = HashSHA256(("").Encode("utf-8")).HexDigest()
```

- h. Combinez les éléments suivants pour créer la demande canonique.

```
canonical_request = method + '\n'
  + canonical_uri + '\n'
  + canonical_querystring + '\n'
  + canonical_headers + '\n'
  + signed_headers + '\n'
  + payload_hash
```

3. Créez la chaîne à signer qui contient les informations des métadonnées sur votre demande. Vous utilisez la chaîne à signer dans l'étape suivante lorsque vous calculez la signature de la demande. Pour plus d'informations, consultez la section [Créer une chaîne à signer pour Signature Version 4](#).

```
string_to_sign=algorithm + "\n"
  + amz_date + "\n"
  + credential_scope + "\n"
  + HashSHA256(canonical_request.Encode("utf-8")).HexDigest()
```

4. Calculez la signature. Pour ce faire, dérivez une clé de signature de votre clé d'accès AWS secrète. La clé dérivée est spécifique à la date, au service et à la Région AWS, pour un niveau de protection plus élevé. Utilisez cette clé dérivée pour signer la demande. Pour plus d'informations, voir [Calculer la signature pour la version 4 de la AWS signature](#).

Assurez-vous de mettre en œuvre la fonction `GetSignatureKey` pour dériver votre clé de signature. Si vous n'avez pas encore dérivé de clé de signature, reportez-vous à la section [Exemples de dérivation d'une clé de signature pour Signature Version 4](#).

```
#Create the signing key
signing_key = GetSignatureKey(secret_key, timestamp, region, service)

# Sign the string_to_sign using the signing key
signature = HMAC.new(signing_key, (string_to_sign).Encode("utf-8"),
    Sha256()).HexDigest
```

La fonction `HMAC(key, data)` représente une fonction HMAC-SHA256 qui renvoie les résultats au format binaire.

5. Ajoutez les informations de signature à la demande et créez l'URL de demande.

Une fois que vous avez calculé la signature, vous l'ajoutez à la chaîne de requête. Pour plus d'informations, consultez la section [Ajout de la signature à la demande](#).

Commencez par ajouter les informations d'authentification à la chaîne de requête.

```
canonical_querystring += "&X-Amz-Signature=" + signature
```

Ensuite, créez l'URL pour la demande.

```
request_url = endpoint + canonical_uri + "?" + canonical_querystring
```

Utilisez l'URL de demande associée à votre WebSocket bibliothèque pour envoyer la demande à Amazon Transcribe.

6. La demande Amazon Transcribe doit inclure les en-têtes suivants. Ces en-têtes sont généralement gérés par votre bibliothèque WebSocket cliente.

```
Host: transcribestreaming.us-west-2.amazonaws.com:8443
Connection: Upgrade
Upgrade: websocket
Origin: URI-of-WebSocket-client
Sec-WebSocket-Version: 13
Sec-WebSocket-Key: randomly-generated-string
```

7. Lorsqu'il Amazon Transcribe reçoit votre WebSocket demande, il répond par une réponse de WebSocket mise à niveau. En général, votre WebSocket bibliothèque gère cette réponse et met en place un socket pour les communications avec Amazon Transcribe.

Voici la réponse de Amazon Transcribe. Les sauts de ligne ont été ajoutés pour faciliter la lecture.

```
HTTP/1.1 101 WebSocket Protocol Handshake

Connection: upgrade
Upgrade: websocket
websocket-origin: wss://transcribestreaming.us-west-2.amazonaws.com:8443
websocket-location: transcribestreaming.us-west-2.amazonaws.com:8443/stream-
transcription-websocket?
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-west-2%2Ftranscribe
%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Signature=Signature Version 4 signature
&X-Amz-SignedHeaders=host
&language-code=en-US
&session-id=String
&media-encoding=flac
&sample-rate=16000
x-amzn-RequestId: RequestId
Strict-Transport-Security: max-age=31536000
sec-websocket-accept: hash-of-the-Sec-WebSocket-Key-header
```

8. Faites votre demande de WebSocket streaming.

Une fois la WebSocket connexion établie, le client peut commencer à envoyer une séquence de trames audio, chacune étant codée à l'aide du [codage du flux d'événements](#).

Chaque trame de données contient trois en-têtes de données combinés à un fragment d'octets audio bruts. Le tableau suivant décrit ces en-têtes.

Longueur en octets du nom de l'en-tête	Nom de l'en-tête (chaîne)	Type de valeur de l'en-tête	Longueur en octets de la chaîne de valeur	Valeur de chaîne (UTF-8)
13	:content-type	7	24	application/octet-stream
11	:event-type	7	10	AudioEvent
13	:message-type	7	5	event

9. Pour mettre fin au flux de données, envoyez un fragment audio vide dans un message codé de flux d'événement.

La réponse contient des octets bruts codés de flux d'événement dans la charge utile. Elle contient le préambule standard et les en-têtes suivants :

Longueur en octets du nom de l'en-tête	Nom de l'en-tête (chaîne)	Type de valeur de l'en-tête	Longueur en octets de la chaîne de valeur	Valeur de chaîne (UTF-8)
13	:content-type	7	16	application/json
11	:event-type	7	15	TranscriptEvent
13	:message-type	7	5	event

Lorsque vous décidez la réponse binaire, vous obtenez une structure JSON avec les résultats de la transcription.

Gestion des erreurs WebSocket de streaming

Si une exception survient lors du traitement de votre demande, Amazon Transcribe répond par une WebSocket trame de terminal contenant une réponse codée dans le flux d'événements. Cette réponse contient la description des en-têtes dans le tableau suivant, et le corps de la réponse

contient un message d'erreur descriptif. Après avoir envoyé la réponse d'exception, Amazon Transcribe envoie un cadre fermé.

Longueur en octets du nom de l'en-tête	Nom de l'en-tête (chaîne)	Type de valeur de l'en-tête	Longueur en octets de la chaîne de valeur	Valeur de chaîne (UTF-8)
13	:content-type	7	16	application/json
15	:exception-type	7	varie	varie, voir ci-dessous
13	:message-type	7	9	exception

L'en-tête `exception-type` contient l'une des valeurs suivantes.

- **BadRequestException** : une erreur client s'est produite lors de la création du flux, ou une erreur s'est produite lors de la diffusion de données. Assurez-vous que votre client est prêt à accepter les données et réessayez votre demande.
- **InternalFailureException**: Amazon Transcribe a eu un problème lors de la poignée de main avec le client. Réessayez votre demande.
- **LimitExceededException** : le client a dépassé la limite de flux simultanés. Pour plus d'informations, consultez [Limites Amazon Transcribe](#). Réduisez le nombre de flux que vous transcrivez.
- **UnrecognizedClientException**: La demande de WebSocket mise à niveau a été signée avec une clé d'accès ou une clé secrète incorrecte. Assurez-vous que vous créez correctement la clé d'accès et renouvelez votre demande.

Amazon Transcribe peut également renvoyer toutes les erreurs de service courantes. Pour obtenir une liste, consultez [Erreurs courantes](#).

Encodage de flux d'événements

Amazon Transcribe utilise un format appelé encodage du flux d'événements pour les transcriptions en continu.

L'encodage de flux d'événements permet une communication bidirectionnelle entre un client et un serveur. Les trames de données envoyées au service de Amazon Transcribe streaming sont codées dans ce format. La réponse de utilise Amazon Transcribe également ce codage.

Chaque message est constitué de deux sections : le préambule et les données. Le préambule est composé des éléments suivants :

1. La longueur totale du message en octets
2. La longueur en octets de tous les en-têtes combinés

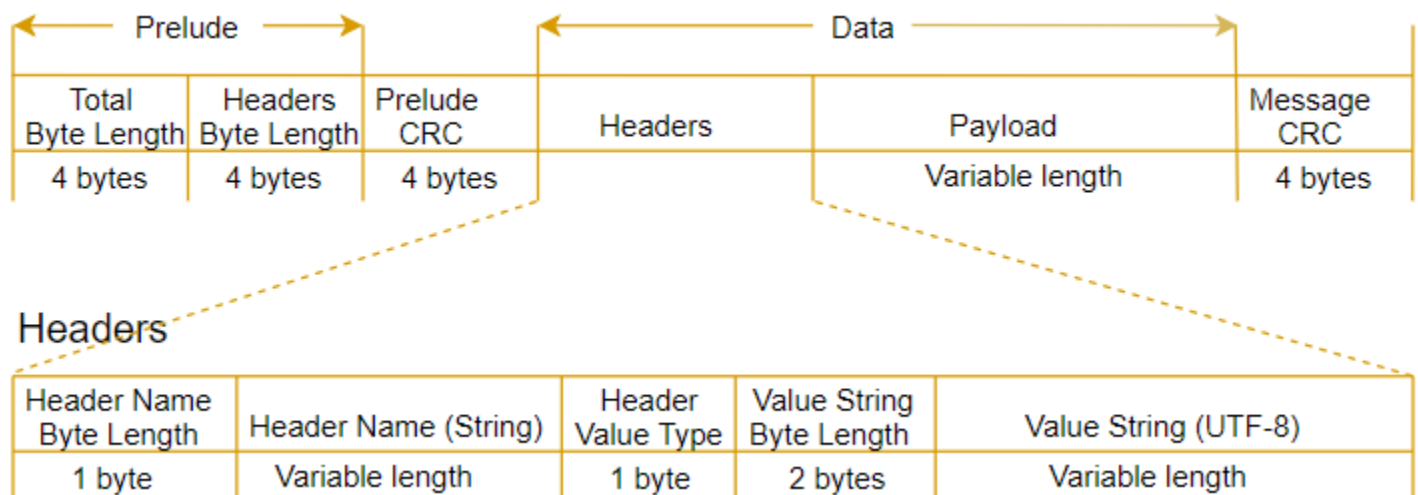
La section des données comprend :

1. En-têtes
2. Charge utile

Chaque section se termine par un total de contrôle CRC (Cyclic Redundancy Check) entier à 4 octets de poids fort. Le total de contrôle CRC du message correspond à la fois à la section prélude et à la section de données. Amazon Transcribe utilise CRC32 (souvent appelé GZIP CRC32) pour calculer les deux CRC. Pour plus d'informations sur CRC32, consultez la rubrique [Spécification du format de fichiers GZIP version 4.3](#).

La surcharge totale de message, incluant le préambule et les deux totaux de contrôle, s'élève à 16 octets.

Le schéma suivant montre les composants qui constituent un message et un en-tête. Il y a plusieurs en-têtes par message.



Chaque message comporte les composants suivants :

- **Préambule** : se compose de deux champs de 4 octets, pour un total fixe de 8 octets.
 - Premier champ de 4 octets : entier de poids fort de l'ensemble du message, y compris ce champ de 4 octets.
 - Deuxième champ de 4 octets : entier de poids fort de la partie des en-têtes du message, sans le champ des en-têtes.
- **CRC du préambule** : le total de contrôle CRC de 4 octets de la partie préambule du message, sans le CRC. Le préambule possède un CRC distinct du CRC du message. Cela permet de Amazon Transcribe détecter immédiatement les informations corrompues sur la longueur des octets sans provoquer d'erreurs, telles que des dépassements de mémoire tampon.
- **En-têtes** : métadonnées qui décrivent le message (par exemple, type du message, de contenu, etc.). Les messages comportent plusieurs en-têtes, qui sont des paires clé-valeur, dans lesquels la clé est une chaîne UTF-8. Les en-têtes peuvent être insérés dans n'importe quel ordre dans la partie en-têtes du message, et chaque en-tête ne peut apparaître qu'une seule fois.
- **Charge utile** : contenu audio qui doit être transcrit.
- **CRC du message** : total de contrôle CRC de 4 octets du début du message au début du total de contrôle. En d'autres termes, il s'agit de l'ensemble des éléments du message, à l'exception du CRC.

Le cadre d'en-tête est le cadre d'autorisation pour la transcription en streaming. Amazon Transcribe utilise la valeur de l'en-tête d'autorisation comme point de départ pour générer une chaîne d'en-têtes d'autorisation pour les trames de données de la demande.

Chaque en-tête contient les composants suivants ; il existe plusieurs en-têtes par image.

- **Longueur en octet du nom de l'en-tête** : longueur en octet du nom de l'en-tête.
- **Nom de l'en-tête** : nom de l'en-tête, qui indique son type. Pour connaître les valeurs valides, consultez les descriptions d'image suivantes :
- **Type de valeur d'en-tête** : nombre indiquant la valeur d'en-tête. La liste suivante indique les valeurs possibles pour l'en-tête et ce qu'elles indiquent.
 - 0 – TRUE
 - 1 – FALSE
 - 2 – BYTE
 - 3 – SHORT

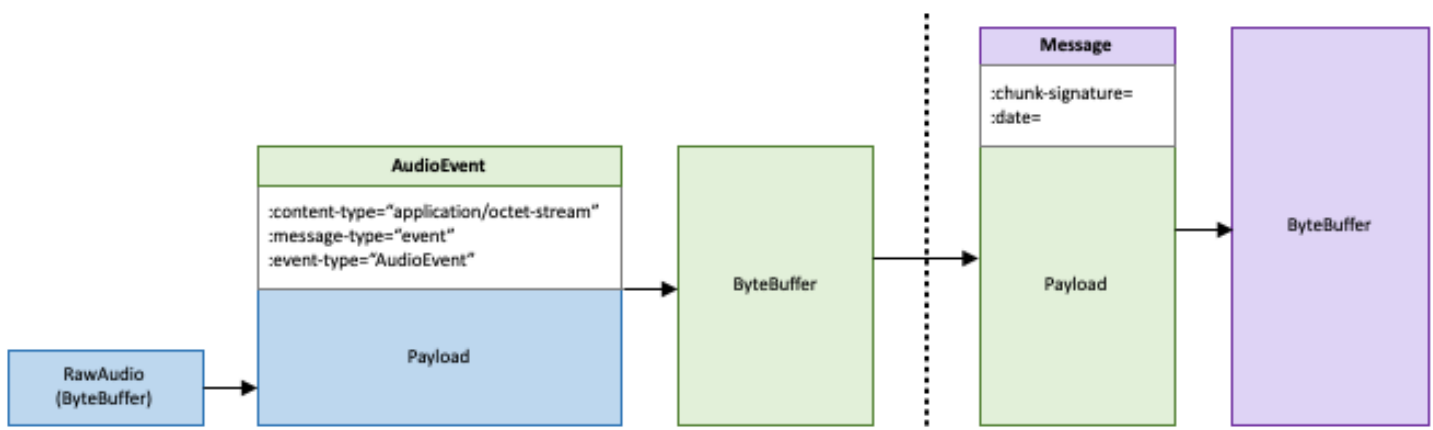
- 4 – INTEGER
 - 5 – LONG
 - 6 – BYTE ARRAY
 - 7 – STRING
 - 8 – TIMESTAMP
 - 9 – UUID
- Longueur en octet de la chaîne de valeur : longueur en octet de la chaîne de valeur d'en-tête.
 - Valeur de l'en-tête : valeur de la chaîne de l'en-tête. Les valeurs valides de ce champ dépendent du type d'en-tête. Pour plus d'informations, consultez les sections [Configuration d'un flux HTTP/2](#) ou [Configuration d'un WebSocket stream](#).

Images de données

Chaque requête en streaming contient une ou plusieurs images de données. La création d'une image de données se fait en deux étapes :

1. Combiner des données audio brutes aux métadonnées pour créer la charge utile de votre requête.
2. Combiner la charge utile à une signature pour former le message d'événement envoyé à Amazon Transcribe.

Le schéma suivant illustre leur fonctionnement.



Mise en file d'attente des offres d'emploi

Grâce à la mise en file d'attente des tâches, vous pouvez soumettre plus de demandes de transcription que celles qui peuvent être traitées simultanément. Sans mise en file d'attente des tâches, une fois que vous avez atteint le quota de demandes simultanées autorisées, vous devez attendre qu'une ou plusieurs demandes soient terminées avant d'en soumettre une nouvelle.

La mise en file d'attente des tâches est facultative pour les demandes de transcription. La mise en file d'attente des tâches est automatiquement activée pour les demandes d'analyse post-appel.

Si vous activez la mise en file d'attente des tâches, Amazon Transcribe crée une file d'attente contenant toutes les demandes dépassant votre limite. Dès qu'une demande est terminée, une nouvelle demande est retirée de votre file d'attente et traitée. Les demandes en file d'attente sont traitées dans un ordre FIFO (premier entré, premier sorti).

Vous pouvez ajouter jusqu'à 10 tâches sur simple demande de tâches à la demande de tâches. Si vous dépassez cette limite, vous obtenez une `LimitExceededConcurrentJobException` erreur. Pour maintenir des performances optimales, Amazon Transcribe n'utilise que 90 % de votre quota (soit un ratio de bande passante de 0,9) pour traiter les tâches en file d'attente. Notez que ce sont des valeurs par lots qui peuvent être augmentés sur simple demande.

Tip

Vous trouverez une liste des limites et des quotas par défaut pour les Amazon Transcribe ressources dans la [Référence AWS générale](#). Certaines de tâches par lots peuvent être augmentés sur simple demande.

Si vous activez la mise en file d'attente des tâches sans dépasser le quota de demandes simultanées, toutes les demandes sont traitées simultanément.

Activer les tâches

Vous pouvez activer la mise en file d'attente des tâches à l'aide AWS des kits SDK AWS Management Console AWS CLI, ou. Consultez les exemples suivants ; consultez les exemples suivants :

AWS Management Console

1. Connectez-vous à [AWS Management Console](#).

2. Dans le volet de navigation, choisissez Tâches de transcription, puis sélectionnez Créer une tâche (en haut à droite). La page Spécifier les détails de la tâche s'ouvre.
3. Dans la zone Paramètres de la Job, vous trouverez un panneau de paramètres supplémentaires. Si vous agrandissez ce panneau, vous pouvez sélectionner la case Ajouter à la file d'attente des tâches pour activer la mise en file d'attente des tâches.

Specify job details [Info](#)

Job settings

Name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, . (period), _ (underscore), and - (hyphen).

Language settings

You can transcribe your audio file in a language that you specify or have Amazon Transcribe identify and transcribe it in the predominant language.

Specific language [Info](#)

If you know the language spoken in your source audio, choose this option to get the most accurate results. The options available for additional processing vary between languages.

Automatic language identification [Info](#)

If you don't know the language spoken in your audio files, choose this option. You have access to fewer options for additional processing than if you choose **Specific language**.

Language

Choose the language of the input audio.

Model type [Info](#)

Choose the type of model to use for the transcription job.

General model

To use a model that is not specialized for a particular use case, choose this option. Configuration options vary between languages.

Custom language model

To use a model that you trained for your specific use case, choose this option. This model has fewer configuration options than the general model.

▼ **Additional settings**

Job queue - optional [Info](#)

Enables you to submit jobs beyond the limit for concurrent jobs (100). You must specify access permissions to the resources that job queuing uses.

Add to job queue

4. Remplissez tous les autres champs que vous souhaitez inclure sur la page Spécifier les détails du poste, puis sélectionnez Suivant. Vous accédez alors à la page Configurer la tâche - facultative.
5. Sélectionnez Créer une tâche pour exécuter votre tâche de transcription.

AWS CLI

Cet exemple utilise la [start-transcription-job](#) commande et le `job-execution-settings` paramètre avec le `AllowDeferredExecution` sous-paramètre. Notez que lorsque vous incluez `AllowDeferredExecution` dans votre demande, vous devez également inclure `DataAccessRoleArn`.

Pour plus d'informations, consultez [StartTranscriptionJob](#) et [JobExecutionSettings](#).

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--transcription-job-name my-first-transcription-job \  
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
--output-bucket-name DOC-EXAMPLE-BUCKET \  
--output-key my-output-files/ \  
--language-code en-US \  
--job-execution-settings  
  AllowDeferredExecution=true,DataAccessRoleArn=arn:aws:iam::111122223333:role/  
ExampleRole
```

Voici un autre exemple d'utilisation de la [start-transcription-job](#) commande et d'un corps de requête qui permet la mise en file d'attente.

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--cli-input-json file://my-first-queueing-request.json
```

Le fichier `my-first-queueing-request.json` contient le corps de requête suivant.

```
{  
  "TranscriptionJobName": "my-first-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
  },  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
```

```
"OutputKey": "my-output-files/",
"LanguageCode": "en-US",
"JobExecutionSettings": {
    "AllowDeferredExecution": true,
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/ExampleRole"
}
}
```

AWS SDK for Python (Boto3)

Cet exemple utilise le AWS SDK for Python (Boto3) pour activer la mise en file d'attente des tâches à l'aide de l'`AllowDeferredExecution` argument de la méthode [start_transcription_job](#). Notez que lorsque vous incluez `AllowDeferredExecution` dans votre demande, vous devez également inclure `DataAccessRoleArn`. Pour plus d'informations, consultez [StartTranscriptionJob](#) et [JobExecutionSettings](#).

Pour des exemples supplémentaires d'utilisation AWS des SDK, y compris des exemples spécifiques à des fonctionnalités, des scénarios et des exemples interservices, consultez le [Exemples de code pour Amazon Transcribe à l'aide de kits de développement logiciel AWS](#) chapitre.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-queueing-request"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
transcribe.start_transcription_job(
    TranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'my-output-files/',
    LanguageCode = 'en-US',
    JobExecutionSettings = {
        'AllowDeferredExecution': True,
        'DataAccessRoleArn': 'arn:aws:iam::111122223333:role/ExampleRole'
    }
)

while True:
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)
```

```
if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:
    break
print("Not ready yet...")
time.sleep(5)
print(status)
```

Vous pouvez consulter la progression d'une tâche en file d'attente via leAWS Management Console ou en soumettant une [GetTranscriptionJob](#) demande. Lorsqu'une tâche est en file d'attente, elle l'StatusestQUEUED. Le statut passe àIN_PROGRESS une fois que le traitement de votre tâche commence, puis passe àCOMPLETED ouFAILED lorsque le traitement est terminé.

Étiquetage des ressources

Une balise est une étiquette de métadonnées personnalisée que vous pouvez ajouter à une ressource pour faciliter l'identification, l'organisation et la recherche dans une recherche. Les balises sont composées de deux parties distinctes : une clé de balise et une valeur de balise. C'est ce que l'on appelle une paire clé:valeur.

Une clé de balise représente généralement une catégorie plus vaste, tandis qu'une valeur de balise représente un sous-ensemble de cette catégorie. Par exemple, vous pourriez avoir la balise `KEY=color` et la balise `value=blue`, qui produiraient la paire clé:valeur `Color:Blue`. Notez que vous pouvez définir la valeur d'une balise sur une chaîne vide, mais vous ne pouvez pas définir la valeur d'une balise sur null. Omettre la valeur de balise équivaut à l'utilisation d'une chaîne vide.

Tip

AWS Billing and Cost Management vous pouvez utiliser des balises pour séparer vos factures en catégories dynamiques. Par exemple, si vous ajoutez des balises pour représenter les différents services de votre entreprise, par exemple `Department:Sales` ou `Department:Legal`, vous pouvez obtenir la répartition des coûts par département.

Dans Amazon Transcribe, vous pouvez baliser les ressources suivantes :

- Emplois de transcription
- Emplois de transcription médicale
- Tâches de transcription post-appel liées à l'analyse des appels
- Vocabulaires personnalisés
- Vocabulaires médicaux personnalisés
- Filtres de vocabulaire personnalisés
- Modèles linguistiques personnalisés

Les clés de balise peuvent comporter jusqu'à 128 caractères et les valeurs de balise peuvent comporter jusqu'à 256 caractères. Amazon Transcribe prend en charge jusqu'à 50 balises par ressource. Pour une ressource donnée, chaque clé de balise doit être unique et comporter une seule valeur. Notez que vos balises ne peuvent pas commencer par `aws:AWS` car ce préfixe est réservé

aux balises générées par le système. Vous ne pouvez pas ajouter, modifier ou supprimer des balises, et elles ne sont pas prises en compte dans votre `tags-per-resource` limite.

Opérations d'API spécifiques au balisage des ressources

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

Pour utiliser les API de balisage, vous devez inclure un nom Amazon Resource Name (ARN) avec votre demande. Les ARN ont le format `arn:partition:service:region:account-id:resource-type/resource-id`. Par exemple, l'ARN associé à une tâche de transcription peut ressembler à `:arn:aws:transcribe:us-west-2:111122223333:transcription-job/my-transcription-job-name`.

Pour en savoir plus sur le balisage, y compris les meilleures pratiques, consultez la section [AWS Ressources sur le balisage](#).

Contrôle d'accès basé sur les étiquettes

Vous pouvez utiliser des balises pour contrôler l'accès dans votre Comptes AWS. Pour le contrôle d'accès basé sur des balises, vous devez fournir les informations des balises dans l'élément de condition d'une IAM stratégie. Vous pouvez ensuite utiliser les balises et la clé de condition de balise associée pour contrôler l'accès à :

- **Ressources** : Contrôler l'accès à vos Amazon Transcribe ressources en fonction des balises que vous avez attribuées à ces ressources.
 - Utilisez la clé `aws:ResourceTag/key-name` condition pour spécifier quelle paire clé:valeur de balise doit être attachée à la ressource.
- **Demandes** : Contrôler les balises qui peuvent être transmises dans une demande.
 - Utilisez la clé `aws:RequestTag/key-name` condition pour spécifier quelles balises peuvent être ajoutées, modifiées ou supprimées d'un IAM utilisateur ou rôle.
- **Processus d'autorisation** : contrôlez l'accès basé sur des balises pour n'importe quelle partie de votre processus d'autorisation.
 - Utilisez la clé `aws:TagKeys/` condition pour contrôler si des clés de balise spécifiques peuvent être utilisées sur une ressource, dans une demande ou par un mandataire. Dans ce cas, la valeur de la clé n'importe pas.

Pour un exemple de stratégie de contrôle d'accès basée sur des balises, consultez [Affichage des tâches de transcription basées sur les balises](#).

Pour plus d'informations sur le contrôle d'accès basé sur des balises, consultez [Contrôle de l'accès aux AWS ressources à l'aide de balises](#).

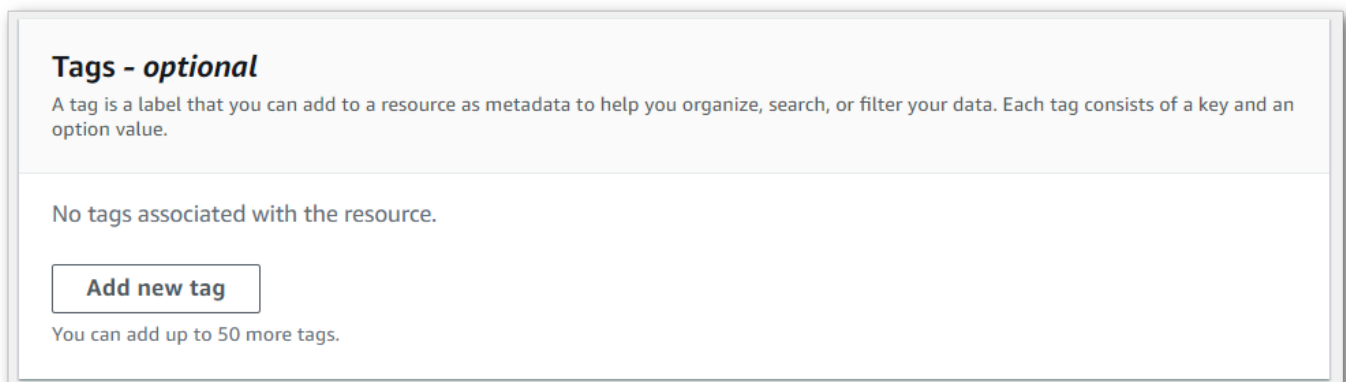
Ajout de balises à vos Amazon Transcribe ressources

Vous pouvez ajouter des balises avant ou après l'exécution de votre Amazon Transcribe tâche. À l'aide des API `Create*` et `Start*` existantes, vous pouvez ajouter des balises à votre demande de transcription.

Vous pouvez ajouter, modifier ou supprimer des balises à l'aide des kits de développement logiciel AWS Management Console AWS CLI, ou AWS des kits SDK ; consultez les exemples suivants :

AWS Management Console

1. Connectez-vous à [AWS Management Console](#).
2. Dans le volet de navigation, choisissez Tâches de transcription, puis sélectionnez Créer une tâche (en haut à droite). La page Spécifier les détails de la tâche s'ouvre.
3. Accédez au bas de la page Spécifier les détails du poste pour trouver la case Étiquettes - facultative et sélectionnez Ajouter une nouvelle étiquette.



4. Entrez les informations pour le champ Clé et, éventuellement, pour le champ Valeur.

Tags - optional

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

Key

Value - optional

You can add up to 49 more tags.

5. Remplissez tous les autres champs que vous souhaitez inclure sur la page Spécifier les détails du poste, puis sélectionnez Suivant. Vous accédez alors à la page Configurer la tâche - facultative.

Sélectionnez Créer une tâche pour exécuter votre tâche de transcription.

6. Vous pouvez consulter les balises associées à une tâche de transcription en accédant à la page Tâches de transcription, en sélectionnant une tâche de transcription et en faisant défiler la page d'informations de cette tâche jusqu'en bas. Si vous souhaitez modifier vos balises, vous pouvez le faire en sélectionnant Gérer les balises.

Tags (2)		<input type="button" value="Manage Tags"/>
Key	Value	
color	blue	

AWS CLI

Cet exemple utilise la [start-transcription-job](#) commande et le Tags paramètre. Pour plus d'informations, consultez [StartTranscriptionJob](#) et [Tag](#).

```
aws transcribe start-transcription-job \
--region us-west-2 \
--transcription-job-name my-first-transcription-job \
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \
--output-bucket-name DOC-EXAMPLE-BUCKET \
--output-key my-output-files/ \
```

```
--language-code en-US \  
--tags Key=color,Value=blue Key=shape,Value=square
```

Voici un autre exemple utilisant la [start-transcription-job](#) commande et un corps de requête qui ajoute des balises à cette tâche.

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--cli-input-json file://filepath/my-first-tagging-job.json
```

Le fichier `my-first-tagging-job.json` contient le corps de requête suivant.

```
{  
  "TranscriptionJobName": "my-first-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
  },  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "OutputKey": "my-output-files/",  
  "LanguageCode": "en-US",  
  "Tags": [  
    {  
      "Key": "color",  
      "Value": "blue"  
    },  
    {  
      "Key": "shape",  
      "Value": "square"  
    }  
  ]  
}
```

AWS SDK for Python (Boto3)

L'exemple suivant utilise le AWS SDK for Python (Boto3) pour ajouter une balise à l'aide de l'`Tags` argument de la méthode [start_transcription_job](#). Pour plus d'informations, consultez [StartTranscriptionJob](#) et [Tag](#).

Pour des exemples supplémentaires d'utilisation AWS des SDK, y compris des exemples spécifiques à des fonctionnalités, des scénarios et des exemples interservices, consultez le [Exemples de code pour Amazon Transcribe à l'aide de kits de développement logiciel AWS](#) chapitre.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
transcribe.start_transcription_job(
    TranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'my-output-files/',
    LanguageCode = 'en-US',
    Tags = [
        {
            'Key': 'color',
            'Value': 'blue'
        }
    ]
)

while True:
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)
    if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Partitionnement des locuteurs (diarisation)

Grâce à la diarisation des haut-parleurs, vous pouvez faire la distinction entre les différents haut-parleurs dans votre sortie de transcription. Amazon Transcribe peut différencier un maximum de 30 locuteurs uniques et étiqueter le texte de chaque locuteur avec une valeur unique (spk_0 jusqu'à spk_9).

Outre les [sections de transcription standard](#) (transcripts et items), les demandes pour lesquelles le partitionnement des locuteurs est activé incluent une section `speaker_labels`. Cette section est regroupée par locuteur et contient des informations sur chaque énoncé, notamment l'étiquette du locuteur et des horodatages.

```
"speaker_labels": {
  "channel_label": "ch_0",
  "speakers": 2,
  "segments": [
    {
      "start_time": "4.87",
      "speaker_label": "spk_0",
      "end_time": "6.88",
      "items": [
        {
          "start_time": "4.87",
          "speaker_label": "spk_0",
          "end_time": "5.02"
        }
      ],
      ...
    }
  ]
}
```

Pour consulter un exemple de transcription complet avec partitionnement des locuteurs (pour deux locuteurs), consultez la section [Exemple de sortie de diarisation \(batch\)](#).

Partitionnement des locuteurs dans une transcription par lots

Pour partitionner des locuteurs dans une transcription par lots, consultez les exemples suivants :

AWS Management Console

1. Connectez-vous à la [AWS Management Console](#).
2. Dans le volet de navigation, choisissez Tâches de transcription, puis sélectionnez Créer une tâche (en haut à droite). La page Spécifier les détails de la tâche s'ouvre.

Specify job details Info

Job settings

Name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, . (period), _ (underscore), and - (hyphen).

Model type Info

Choose the type of model to use for the transcription job.

General model
To use a model that is not specialized for a particular use case, choose this option. Configuration options vary between languages.

Custom language model
To use a model that you trained for your specific use case, choose this option. This model has fewer configuration options than the general model.

Language settings

You can transcribe your audio file in a language that you specify or have Amazon Transcribe identify and transcribe it in the predominant language.

Specific language Info
If you know the language spoken in your source audio, choose this option to get the most accurate results. The options available for additional processing vary between languages.

Automatic language identification Info
If you don't know the language spoken in your audio files, choose this option. You have access to fewer options for additional processing than if you choose **Specific language**.

Language

Choose the language of the input audio.

► **Additional settings**

3. Renseignez les champs que vous souhaitez inclure sur la page Spécifier les détails de la tâche, puis sélectionnez Suivant. Vous accédez alors à la page Configurer la tâche - facultatif.

Dans le volet Paramètres audio, sélectionnez Partitionnement des locuteurs (sous l'en-tête « Type d'identification audio »). Vous pouvez éventuellement spécifier le nombre de locuteurs que vous souhaitez partitionner, jusqu'à un maximum de 10.

Audio settings

Audio identification [Info](#)
Choose to split multi-channel audio into separate channels for transcription, or partition speakers in the input audio.

Audio identification type

Channel identification

Speaker partitioning

Maximum number of speakers
Providing the number of speakers can increase the accuracy of your results.

The maximum number of speakers is 10.

Alternative results [Info](#)
Enable to view more transcription results

4. Sélectionnez Créer une tâche pour exécuter votre tâche de transcription.

AWS CLI

Cet exemple utilise le [start-transcription-job](#). Pour plus d'informations, consultez [StartTranscriptionJob](#).

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--transcription-job-name my-first-transcription-job \  
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
--output-bucket-name DOC-EXAMPLE-BUCKET \  
--output-key my-output-files/ \  
--language-code en-US \  
--show-speaker-labels TRUE \  

```



```
--max-speaker-labels 3
```

Voici un autre exemple d'utilisation de la [start-transcription-job](#) commande et d'un corps de requête qui permet le partitionnement des haut-parleurs avec cette tâche.

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--cli-input-json file://my-first-transcription-job.json
```

Le fichier my-first-transcription-job.json contient le corps de requête suivant.

```
{  
  "TranscriptionJobName": "my-first-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
  },  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "OutputKey": "my-output-files/",  
  "LanguageCode": "en-US",  
  "ShowSpeakerLabels": 'TRUE',  
  "MaxSpeakerLabels": 3  
}
```

AWS SDK for Python (Boto3)

Cet exemple utilise le AWS SDK for Python (Boto3) pour identifier les canaux à l'aide de la méthode [start_transcription_job](#). Pour plus d'informations, consultez [StartTranscriptionJob](#).

```
from __future__ import print_function  
import time  
import boto3  
transcribe = boto3.client('transcribe', 'us-west-2')  
job_name = "my-first-transcription-job"  
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
transcribe.start_transcription_job(  
    TranscriptionJobName = job_name,  
    Media = {  
        'MediaFileUri': job_uri  
    },  
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',  
    OutputKey = 'my-output-files/',
```

```
LanguageCode = 'en-US',
Settings = {
    'ShowSpeakerLabels': True,
    'MaxSpeakerLabels': 3
}
)

while True:
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)
    if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Partitionnement des locuteurs dans une transcription en streaming


Pour partitionner des locuteurs dans une transcription en streaming, consultez les exemples suivants :

Transcriptions en streaming

1. Connectez-vous à la [AWS Management Console](#).
2. Dans le panneau de navigation, choisissez Transcription en temps réel. Faites défiler jusqu'à Paramètres audio et développez ce champ s'il est réduit.

Real-time transcription [Info](#)

See how Amazon Transcribe creates a text copy of speech in real time. Choose **Start streaming** and talk.

Transcription Download full transcript  Start streaming

Transcription output Current language: English, US

Choose **Start streaming** to begin a real-time transcription of what you speak into your microphone

00:00 of 15:00 min audio stream

- ▶ **Language settings**
- ▼ **Audio settings**
- Speaker partitioning** [Info](#)
Partition the different speakers in the stream. Speaker partitioning might vary in availability between languages.
- ▶ **Content removal settings**
- ▶ **Customizations**

3. Activez Partitionnement des locuteurs.

- ▶ **Language settings**
- ▼ **Audio settings**
- Speaker partitioning** [Info](#)
Partition the different speakers in the stream. Speaker partitioning might vary in availability between languages.
- ▶ **Content removal settings**
- ▶ **Customizations**

4. Vous êtes prêt à transcrire votre flux. Sélectionnez Démarrer le streaming et commencez à parler. Pour mettre fin à votre dictée, sélectionnez Arrêter le streaming.

Flux HTTP/2

Cet exemple crée une requête HTTP/2 qui partitionne les locuteurs dans votre sortie de transcription. Pour plus d'informations sur l'utilisation du streaming HTTP/2 avec Amazon Transcribe, consultez [Configuration d'un flux HTTP/2](#). Pour plus de détails sur les paramètres et les en-têtes spécifiques à Amazon Transcribe, voir [StartStreamTranscription](#).

```
POST /stream-transcription HTTP/2
host: transcribestreaming.us-west-2.amazonaws.com
X-Amz-Target: com.amazonaws.transcribe.Transcribe.StartStreamTranscription
Content-Type: application/vnd.amazon.eventstream
X-Amz-Content-Sha256: string
X-Amz-Date: 20220208T235959Z
Authorization: AWS4-HMAC-SHA256 Credential=access-key/20220208/us-west-2/transcribe/
aws4_request, SignedHeaders=content-type;host;x-amz-content-sha256;x-amz-date;x-amz-
target;x-amz-security-token, Signature=string
x-amzn-transcribe-language-code: en-US
x-amzn-transcribe-media-encoding: flac
x-amzn-transcribe-sample-rate: 16000
x-amzn-transcribe-show-speaker-label: true
transfer-encoding: chunked
```

Les définitions des paramètres se trouvent dans la [référence d'API](#) ; les paramètres communs à toutes les opérations d' AWS API sont répertoriés dans la section [Paramètres communs](#).

WebSocket flux

Cet exemple crée une URL présignée qui sépare les locuteurs dans votre sortie de transcription. Les sauts de ligne ont été ajoutés pour faciliter la lecture. Pour plus d'informations sur l'utilisation WebSocket des flux avec Amazon Transcribe, consultez [Configuration d'un WebSocket stream](#). Pour plus de détails sur les paramètres, consultez la section [StartStreamTranscription](#).

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/stream-transcription-
websocket?
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-
west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=string
&X-Amz-SignedHeaders=content-type%3Bhost%3Bx-amz-date
```

```
&language-code=en-US
&specialty=PRIMARYCARE
&type=DICTION
&media-encoding=flac
&sample-rate=16000
&show-speaker-label=true
```

Les définitions des paramètres se trouvent dans la [référence d'API](#) ; les paramètres communs à toutes les opérations d' AWS API sont répertoriés dans la section [Paramètres communs](#).

Exemple de sortie de diarisation (batch)

Voici un exemple de sortie pour une transcription par lots avec la diarisation activée.

```
{
  "jobName": "my-first-transcription-job",
  "accountId": "111122223333",
  "results": {
    "transcripts": [
      {
        "transcript": "I've been on hold for an hour. Sorry about that."
      }
    ],
    "speaker_labels": {
      "channel_label": "ch_0",
      "speakers": 2,
      "segments": [
        {
          "start_time": "4.87",
          "speaker_label": "spk_0",
          "end_time": "6.88",
          "items": [
            {
              "start_time": "4.87",
              "speaker_label": "spk_0",
              "end_time": "5.02"
            },
            {
              "start_time": "5.02",
              "speaker_label": "spk_0",
              "end_time": "5.17"
            }
          ]
        }
      ]
    }
  }
}
```

```
        "start_time": "5.17",
        "speaker_label": "spk_0",
        "end_time": "5.29"
    },
    {
        "start_time": "5.29",
        "speaker_label": "spk_0",
        "end_time": "5.64"
    },
    {
        "start_time": "5.64",
        "speaker_label": "spk_0",
        "end_time": "5.84"
    },
    {
        "start_time": "6.11",
        "speaker_label": "spk_0",
        "end_time": "6.26"
    },
    {
        "start_time": "6.26",
        "speaker_label": "spk_0",
        "end_time": "6.88"
    }
]
},
{
    "start_time": "8.49",
    "speaker_label": "spk_1",
    "end_time": "9.24",
    "items": [
        {
            "start_time": "8.49",
            "speaker_label": "spk_1",
            "end_time": "8.88"
        },
        {
            "start_time": "8.88",
            "speaker_label": "spk_1",
            "end_time": "9.05"
        },
        {
            "start_time": "9.05",
            "speaker_label": "spk_1",
```

```
                "end_time": "9.24"
            }
        ]
    }
},
"items": [
    {
        "start_time": "4.87",
        "speaker_label": "spk_0",
        "end_time": "5.02",
        "alternatives": [
            {
                "confidence": "1.0",
                "content": "I've"
            }
        ],
        "type": "pronunciation"
    },
    {
        "start_time": "5.02",
        "speaker_label": "spk_0",
        "end_time": "5.17",
        "alternatives": [
            {
                "confidence": "1.0",
                "content": "been"
            }
        ],
        "type": "pronunciation"
    },
    {
        "start_time": "5.17",
        "speaker_label": "spk_0",
        "end_time": "5.29",
        "alternatives": [
            {
                "confidence": "1.0",
                "content": "on"
            }
        ],
        "type": "pronunciation"
    },
    {
```

```
    "start_time": "5.29",
    "speaker_label": "spk_0",
    "end_time": "5.64",
    "alternatives": [
      {
        "confidence": "1.0",
        "content": "hold"
      }
    ],
    "type": "pronunciation"
  },
  {
    "start_time": "5.64",
    "speaker_label": "spk_0",
    "end_time": "5.84",
    "alternatives": [
      {
        "confidence": "1.0",
        "content": "for"
      }
    ],
    "type": "pronunciation"
  },
  {
    "start_time": "6.11",
    "speaker_label": "spk_0",
    "end_time": "6.26",
    "alternatives": [
      {
        "confidence": "1.0",
        "content": "an"
      }
    ],
    "type": "pronunciation"
  },
  {
    "start_time": "6.26",
    "speaker_label": "spk_0",
    "end_time": "6.88",
    "alternatives": [
      {
        "confidence": "1.0",
        "content": "hour"
      }
    ]
  }
```



```
    ],
    "type": "pronunciation"
  },
  {
    "speaker_label": "spk_0",
    "alternatives": [
      {
        "confidence": "0.0",
        "content": "."
      }
    ],
    "type": "punctuation"
  },
  {
    "start_time": "8.49",
    "speaker_label": "spk_1",
    "end_time": "8.88",
    "alternatives": [
      {
        "confidence": "1.0",
        "content": "Sorry"
      }
    ],
    "type": "pronunciation"
  },
  {
    "start_time": "8.88",
    "speaker_label": "spk_1",
    "end_time": "9.05",
    "alternatives": [
      {
        "confidence": "0.902",
        "content": "about"
      }
    ],
    "type": "pronunciation"
  },
  {
    "start_time": "9.05",
    "speaker_label": "spk_1",
    "end_time": "9.24",
    "alternatives": [
      {
        "confidence": "1.0",
```

```
        "content": "that"
      }
    ],
    "type": "pronunciation"
  },
  {
    "speaker_label": "spk_1",
    "alternatives": [
      {
        "confidence": "0.0",
        "content": "."
      }
    ],
    "type": "punctuation"
  }
]
},
"status": "COMPLETED"
}
```

Transcription audio multicanal

Si votre audio comporte deux canaux, vous pouvez utiliser l'identification des canaux pour transcrire le discours de chaque canal séparément. Amazon Transcribe ne prend actuellement pas en charge le son avec plus de deux canaux.

Dans votre transcription, les chaînes se voient attribuer les libellés `ch_0` et `ch_1`.

Outre les [sections de transcription standard](#) (`transcriptsetitems`), les demandes pour lesquelles l'identification des chaînes est activée incluent une `channel_labels` section. Cette section contient chaque énoncé ou signe de ponctuation, groupé par canal, ainsi que le libellé, les horodatages et le score de confiance associés à ce canal.

```
"channel_labels": {
  "channels": [
    {
      "channel_label": "ch_0",
      "items": [
        {
          "channel_label": "ch_0",
          "start_time": "4.86",
          "end_time": "5.01",
          "alternatives": [
            {
              "confidence": "1.0",
              "content": "I've"
            }
          ],
          "type": "pronunciation"
        },
        ...
      ],
      "channel_label": "ch_1",
      "items": [
        {
          "channel_label": "ch_1",
          "start_time": "8.5",
          "end_time": "8.89",
          "alternatives": [
            {
              "confidence": "1.0",
              "content": "Sorry"
            }
          ]
        }
      ]
    }
  ]
}
```

```
        }
      ],
      "type": "pronunciation"
    },
    ...
    "number_of_channels": 2
  },
```

Notez que si une personne sur une chaîne parle en même temps qu'une autre personne sur une chaîne distincte, les horodatages de chaque chaîne se chevauchent pendant que les personnes parlent l'une sur l'autre.

Pour consulter un exemple de transcription complet avec identification des chaînes, consultez [Exemple de sortie d'identification de canal \(batch\)](#).

Utilisation de l'identification des canaux dans une transcription par lots

Pour identifier les canaux dans une transcription par lots, vous pouvez utiliser les AWS Management Console, AWS CLI, ou AWS SDK ; consultez les exemples suivants :

AWS Management Console

1. Connectez-vous à [AWS Management Console](#).
2. Dans le volet de navigation, choisissez Tâches de transcription, puis sélectionnez Créer une tâche (en haut à droite). La page Spécifier les détails de la tâche s'ouvre.

Specify job details [Info](#)

Job settings

Name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, . (period), _ (underscore), and - (hyphen).

Model type [Info](#)
Choose the type of model to use for the transcription job.

General model
To use a model that is not specialized for a particular use case, choose this option. Configuration options vary between languages.

Custom language model
To use a model that you trained for your specific use case, choose this option. This model has fewer configuration options than the general model.

Language settings
You can transcribe your audio file in a language that you specify or have Amazon Transcribe identify and transcribe it in the predominant language.

Specific language [Info](#)
If you know the language spoken in your source audio, choose this option to get the most accurate results. The options available for additional processing vary between languages.

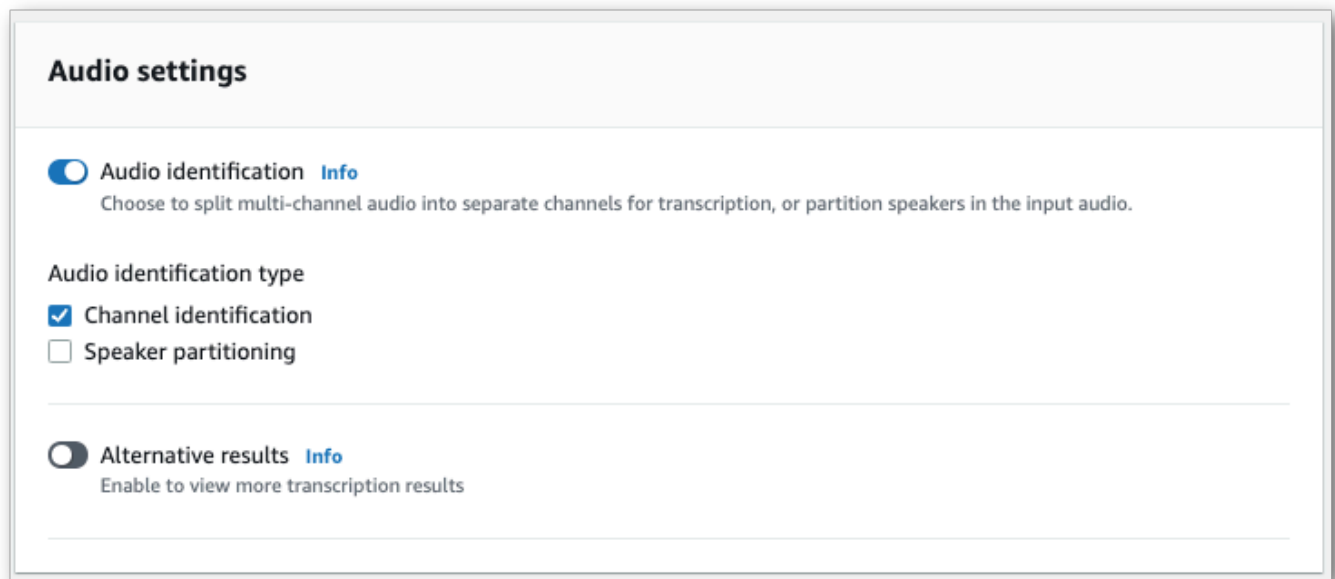
Automatic language identification [Info](#)
If you don't know the language spoken in your audio files, choose this option. You have access to fewer options for additional processing than if you choose **Specific language**.

Language
Choose the language of the input audio.

► **Additional settings**

3. Remplissez tous les champs que vous souhaitez inclure sur la page Spécifier les détails du poste, puis sélectionnez Suivant. Vous accédez alors à la page Configurer la tâche - facultative.

Dans le panneau des paramètres audio, sélectionnez Identification du canal (sous la rubrique « Type d'identification audio »).



4. Sélectionnez Créer une tâche pour exécuter votre tâche de transcription.

AWS CLI

Cet exemple utilise le [start-transcription-job](#). Pour plus d'informations, veuillez consulter [StartTranscriptionJob](#).

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--transcription-job-name my-first-transcription-job \  
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
--output-bucket-name DOC-EXAMPLE-BUCKET \  
--output-key my-output-files/ \  
--language-code en-US \  
--settings ChannelIdentification=true
```

Voici un autre exemple d'utilisation de la [start-transcription-job](#) commande et d'un corps de requête qui permet d'identifier un canal avec cette tâche.

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--cli-input-json file://my-first-transcription-job.json
```

Le fichier `my-first-transcription-job.json` contient le corps de requête suivant.

```
{
  "TranscriptionJobName": "my-first-transcription-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
  },
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "OutputKey": "my-output-files/",
  "LanguageCode": "en-US",
  "Settings": {
    "ChannelIdentification": true
  }
}
```

AWS SDK for Python (Boto3)

Cet exemple utilise le AWS SDK for Python (Boto3) pour identifier les canaux à l'aide de la méthode [start_transcription_job](#). Pour plus d'informations, consultez [StartTranscriptionJob](#).

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
transcribe.start_transcription_job(
    TranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'my-output-files/',
    LanguageCode = 'en-US',
    Settings = {
        'ChannelIdentification': True
    }
)

while True:
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)
    if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
```

```
print(status)
```

Utilisation de l'identification des chaînes dans une transcription en continu

Pour identifier les chaînes dans une transcription en continu, vous pouvez utiliser HTTP/2 ou WebSockets voir les exemples suivants :

flux HTTP/2

Cet exemple crée une requête HTTP/2 qui sépare les canaux dans votre sortie de transcription. Pour plus d'informations sur l'utilisation du streaming HTTP/2 avec Amazon Transcribe, consultez [Configuration d'un flux HTTP/2](#). Pour plus de détails sur les paramètres et les en-têtes spécifiques à Amazon Transcribe, consultez [StartStreamTranscription](#).

```
POST /stream-transcription HTTP/2
host: transcribestreaming.us-west-2.amazonaws.com
X-Amz-Target: com.amazonaws.transcribe.Transcribe.StartStreamTranscription
Content-Type: application/vnd.amazon.eventstream
X-Amz-Content-Sha256: string
X-Amz-Date: 20220208T235959Z
Authorization: AWS4-HMAC-SHA256 Credential=access-key/20220208/us-west-2/transcribe/
aws4_request, SignedHeaders=content-type;host;x-amz-content-sha256;x-amz-date;x-amz-
target;x-amz-security-token, Signature=string
x-amzn-transcribe-language-code: en-US
x-amzn-transcribe-media-encoding: flac
x-amzn-transcribe-sample-rate: 16000
x-amzn-channel-identification: TRUE
transfer-encoding: chunked
```

Les définitions de paramètres se trouvent dans la [référence de l'API](#) ; les paramètres communs à toutes les opérations d'AWSAPI sont répertoriés dans la section [Paramètres communs](#).

WebSocket ruisseau

Cet exemple crée une URL présignée qui sépare les canaux dans votre sortie de transcription. Les sauts de ligne ont été ajoutés pour faciliter la lecture. Pour plus d'informations sur l'utilisation WebSocket des flux avec Amazon Transcribe, consultez [Configuration d'un WebSocket stream](#). Pour plus de détails sur les paramètres, reportez-vous à la section [StartStreamTranscription](#).


```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/stream-transcription-
websocket?
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-
west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=string
&X-Amz-SignedHeaders=content-type%3Bhost%3Bx-amz-date
&language-code=en-US
&specialty=PRIMARYCARE
&type=DICTATION
&media-encoding=flac
&sample-rate=16000
&channel-identification=TRUE
```

Les définitions de paramètres se trouvent dans la [référence de l'API](#) ; les paramètres communs à toutes les opérations d'AWSAPI sont répertoriés dans la section [Paramètres communs](#).

Exemple de sortie d'identification de canal (batch)

Voici un exemple de sortie pour une transcription par lots avec l'identification des canaux activée.

```
{
  "jobName": "my-first-transcription-job",
  "accountId": "111122223333",
  "results": {
    "transcripts": [
      {
        "transcript": "I've been on hold for an hour. Sorry about that."
      }
    ],
    "channel_labels": {
      "channels": [
        {
          "channel_label": "ch_0",
          "items": [
            {
              "channel_label": "ch_0",
              "start_time": "4.86",
              "end_time": "5.01",
            }
          ]
        }
      ]
    }
  }
}
```

```
        "alternatives": [  
            {  
                "confidence": "1.0",  
                "content": "I've"  
            }  
        ],  
        "type": "pronunciation"  
    },  
    {  
        "channel_label": "ch_0",  
        "start_time": "5.01",  
        "end_time": "5.16",  
        "alternatives": [  
            {  
                "confidence": "1.0",  
                "content": "been"  
            }  
        ],  
        "type": "pronunciation"  
    },  
    {  
        "channel_label": "ch_0",  
        "start_time": "5.16",  
        "end_time": "5.28",  
        "alternatives": [  
            {  
                "confidence": "1.0",  
                "content": "on"  
            }  
        ],  
        "type": "pronunciation"  
    },  
    {  
        "channel_label": "ch_0",  
        "start_time": "5.28",  
        "end_time": "5.62",  
        "alternatives": [  
            {  
                "confidence": "1.0",  
                "content": "hold"  
            }  
        ],  
        "type": "pronunciation"  
    },  
    ],  
    "type": "pronunciation"  
}
```

```
{
  "channel_label": "ch_0",
  "start_time": "5.62",
  "end_time": "5.83",
  "alternatives": [
    {
      "confidence": "1.0",
      "content": "for"
    }
  ],
  "type": "pronunciation"
},
{
  "channel_label": "ch_0",
  "start_time": "6.1",
  "end_time": "6.25",
  "alternatives": [
    {
      "confidence": "1.0",
      "content": "an"
    }
  ],
  "type": "pronunciation"
},
{
  "channel_label": "ch_0",
  "start_time": "6.25",
  "end_time": "6.87",
  "alternatives": [
    {
      "confidence": "1.0",
      "content": "hour"
    }
  ],
  "type": "pronunciation"
},
{
  "channel_label": "ch_0",
  "language_code": "en-US",
  "alternatives": [
    {
      "confidence": "0.0",
      "content": "."
    }
  ]
}
```

```
        ],
        "type": "punctuation"
    }
]
},
{
  "channel_label": "ch_1",
  "items": [
    {
      "channel_label": "ch_1",
      "start_time": "8.5",
      "end_time": "8.89",
      "alternatives": [
        {
          "confidence": "1.0",
          "content": "Sorry"
        }
      ],
      "type": "pronunciation"
    },
    {
      "channel_label": "ch_1",
      "start_time": "8.89",
      "end_time": "9.06",
      "alternatives": [
        {
          "confidence": "0.9176",
          "content": "about"
        }
      ],
      "type": "pronunciation"
    },
    {
      "channel_label": "ch_1",
      "start_time": "9.06",
      "end_time": "9.25",
      "alternatives": [
        {
          "confidence": "1.0",
          "content": "that"
        }
      ],
      "type": "pronunciation"
    }
  ],
}
```

```
        {
            "channel_label": "ch_1",
            "alternatives": [
                {
                    "confidence": "0.0",
                    "content": "."
                }
            ],
            "type": "punctuation"
        }
    ]
},
"number_of_channels": 2
},
"items": [
    {
        "channel_label": "ch_0",
        "start_time": "4.86",
        "end_time": "5.01",
        "alternatives": [
            {
                "confidence": "1.0",
                "content": "I've"
            }
        ],
        "type": "pronunciation"
    },
    {
        "channel_label": "ch_0",
        "start_time": "5.01",
        "end_time": "5.16",
        "alternatives": [
            {
                "confidence": "1.0",
                "content": "been"
            }
        ],
        "type": "pronunciation"
    },
    {
        "channel_label": "ch_0",
        "start_time": "5.16",
        "end_time": "5.28",
```

```
        "alternatives": [
            {
                "confidence": "1.0",
                "content": "on"
            }
        ],
        "type": "pronunciation"
    },
    {
        "channel_label": "ch_0",
        "start_time": "5.28",
        "end_time": "5.62",
        "alternatives": [
            {
                "confidence": "1.0",
                "content": "hold"
            }
        ],
        "type": "pronunciation"
    },
    {
        "channel_label": "ch_0",
        "start_time": "5.62",
        "end_time": "5.83",
        "alternatives": [
            {
                "confidence": "1.0",
                "content": "for"
            }
        ],
        "type": "pronunciation"
    },
    {
        "channel_label": "ch_0",
        "start_time": "6.1",
        "end_time": "6.25",
        "alternatives": [
            {
                "confidence": "1.0",
                "content": "an"
            }
        ],
        "type": "pronunciation"
    },
    },
```

```
{
  "channel_label": "ch_0",
  "start_time": "6.25",
  "end_time": "6.87",
  "alternatives": [
    {
      "confidence": "1.0",
      "content": "hour"
    }
  ],
  "type": "pronunciation"
},
{
  "channel_label": "ch_0",
  "alternatives": [
    {
      "confidence": "0.0",
      "content": "."
    }
  ],
  "type": "punctuation"
},
{
  "channel_label": "ch_1",
  "start_time": "8.5",
  "end_time": "8.89",
  "alternatives": [
    {
      "confidence": "1.0",
      "content": "Sorry"
    }
  ],
  "type": "pronunciation"
},
{
  "channel_label": "ch_1",
  "start_time": "8.89",
  "end_time": "9.06",
  "alternatives": [
    {
      "confidence": "0.9176",
      "content": "about"
    }
  ],
}
```

```
        "type": "pronunciation"
    },
    {
        "channel_label": "ch_1",
        "start_time": "9.06",
        "end_time": "9.25",
        "alternatives": [
            {
                "confidence": "1.0",
                "content": "that"
            }
        ],
        "type": "pronunciation"
    },
    {
        "channel_label": "ch_1",
        "alternatives": [
            {
                "confidence": "0.0",
                "content": "."
            }
        ],
        "type": "punctuation"
    }
]
},
"status": "COMPLETED"
}
```


Identification des langues dominantes dans vos médias

Amazon Transcribe est capable d'identifier automatiquement les langues parlées dans vos médias sans que vous ayez à spécifier de code de langue.

L'[identification des langues par lots](#) permet d'identifier la langue dominante parlée dans votre fichier multimédia ou, si votre média contient plusieurs langues, elle peut identifier toutes les langues parlées. Pour améliorer la précision de l'identification des langues, vous pouvez éventuellement fournir une liste d'au moins deux langues que vous pensez être présentes dans votre média.

L'[identification de la langue de diffusion](#) peut identifier une langue par canal (deux canaux au maximum sont pris en charge) ou, si le flux contient plusieurs langues, elle peut identifier toutes les langues parlées. Les demandes de streaming doivent comporter au moins deux options de langue supplémentaires dans votre demande. La sélection d'options de langue permet une identification plus rapide des langues. Plus Amazon Transcribe est en mesure d'identifier la langue rapidement, moins il y a de perte de données dans les premières secondes de votre flux.

Important

Les transcriptions par lots et en streaming prennent en charge différentes langues. Reportez-vous à la colonne Entrée de données du [tableau des langues prises en charge](#) pour plus de détails. Notez que le suédois et le vietnamien ne sont actuellement pas pris en charge pour l'identification des langues.

Pour en savoir plus sur la surveillance et les événements associés à l'identification des langues, consultez la section [Événements d'identification des langues](#).

Identification des langues avec des tâches de transcription par lots

Utilisez l'identification des langues par lots pour identifier automatiquement la ou les langues de votre fichier multimédia.

Si votre fichier multimédia ne contient qu'une seule langue, vous pouvez activer l'[identification d'une langue unique](#), qui identifie la langue dominante parlée dans votre fichier multimédia et crée votre transcription en utilisant uniquement cette langue.

Si votre fichier multimédia contient plusieurs langues, vous pouvez activer l'[identification multilingue](#), qui identifie toutes les langues parlées dans votre fichier multimédia et crée votre transcription en utilisant chaque langue identifiée. Notez qu'une transcription multilingue est produite. Vous pouvez utiliser d'autres services Amazon Translate, tels que la traduction de votre relevé de notes.

Reportez-vous au tableau des [langues prises en charge](#) pour obtenir la liste complète des langues prises en charge et des codes de langue associés.

Pour de meilleurs résultats, assurez-vous que votre fichier multimédia contient au moins 30 secondes de parole.

Pour des exemples d'utilisation avec le SDK AWS Management Console AWS CLI,, et AWS Python, consultez [Utilisation de l'identification des langues avec des transcriptions par lots](#).

Identification des langues dans un audio multilingue

L'identification multilingue est destinée aux fichiers multimédias multilingues. Elle fournit une transcription qui reflète toutes les [langues prises en charge](#) parlées dans votre média. Cela signifie que si les locuteurs changent de langue en cours de conversation, ou si chaque participant parle une langue différente, votre sortie de transcription détecte et transcrit correctement chaque langue. Par exemple, si votre média contient un locuteur bilingue qui alterne entre l'anglais américain (en-US) et l'hindi (hi-IN), l'identification multilingue peut identifier et transcrire l'anglais américain comme en-US et l'hindi parlé comme hi-IN.

Cela diffère de l'identification dans d'une seule langue, où une seule langue dominante est utilisée pour créer une transcription. Dans ce cas, toute langue parlée qui n'est pas la langue dominante est incorrectement transcrite.

Note

L'expurgation et les modèles de langue personnalisés ne sont actuellement pas pris en charge avec l'identification multilingue.

Note

Les langues suivantes sont actuellement prises en charge avec l'identification multilingue : en-AB, en-AU, en-GB, en-IE, en-NZ, en-US, en-WL, en-ZA, es-ES, es-US, fr-CA, fr-FR, zh-

CN, zh-TW, pt-BR, pt-PT, de-CH, de-DE, af-ZA, ar-AE, da-DK, he-IL, hi-in, ID-ID, Fa-ir, It-it, Ja-JP, Ko-KR, MS-My, NL-NL, Ru-ru, Ta-in, Te-in, Th-TH, TR-TR

Les transcriptions multilingues fournissent un récapitulatif des langues détectées et le temps total pendant lequel chaque langue est parlée dans vos médias. Voici un exemple :

```
"results": {
  "transcripts": [
    {
      "transcript": "welcome to Amazon transcribe. ## ## ##### ### #### ####
## #### ### #####"
    }
  ],
  ...
  "language_codes": [
    {
      "language_code": "en-US",
      "duration_in_seconds": 2.45
    },
    {
      "language_code": "hi-IN",
      "duration_in_seconds": 5.325
    },
    {
      "language_code": "ja-JP",
      "duration_in_seconds": 4.15
    }
  ]
}
```

Amélioration de la précision de l'identification des langues

L'identification des langues vous permet d'inclure la liste des langues que vous pensez être présentes dans vos médias. L'inclusion des options de langue (`LanguageOptions`) limite Amazon Transcribe l'utilisation des seules langues que vous spécifiez lorsque vous associez votre audio à la bonne langue, ce qui peut accélérer l'identification de la langue et améliorer la précision associée à l'attribution du dialecte linguistique approprié.

Si vous choisissez d'inclure des codes de langue, vous devez en inclure au moins deux. Il n'y a aucune limite quant au nombre de codes de langue que vous pouvez inclure, mais nous vous recommandons d'en utiliser entre deux et cinq pour une efficacité et une précision optimales.

Note

Si vous incluez des codes de langue dans votre demande et qu'aucun des codes de langue que vous fournissez ne correspond à la langue ou aux langues identifiées dans votre audio, Amazon Transcribe sélectionne la langue la plus proche parmi les codes de langue que vous avez spécifiés. Une transcription dans cette langue est ensuite générée. Par exemple, si votre contenu multimédia est en anglais américain (en-US) et que vous fournissez les codes de langue zh-CN, et fr-FR et de-DE, Amazon Transcribe il est probable que votre contenu corresponde à l'allemand (de-DE) et produira une transcription en allemand. La non-concordance entre les codes de langue et les langues parlées peut entraîner une transcription inexacte. Nous vous recommandons donc de faire preuve de prudence lorsque vous incluez des codes de langue.

Combinaison de l'identification des langues avec d'autres fonctionnalités Amazon Transcribe

Vous pouvez utiliser l'identification des langues par lots en combinaison avec toute autre fonctionnalité Amazon Transcribe. Si vous associez l'identification des langues à d'autres fonctionnalités, vous êtes limité aux langues prises en charge par ces fonctionnalités. Par exemple, si vous utilisez l'identification de la langue pour la rédaction de contenu, vous êtes limité à l'anglais américain (en-US) ou à l'espagnol américain (es-US), car il s'agit de la seule langue disponible pour la rédaction. Pour plus d'informations, consultez la section [Langues prises en charge et fonctionnalités spécifiques aux langues](#).

Important

Si vous utilisez l'identification automatique de la langue avec la rédaction de contenu activée et que votre fichier audio contient des langues autres que l'anglais américain (en-US) ou l'espagnol américain (es-US), seul le contenu en anglais américain ou en espagnol américain est expurgé dans votre transcription. Les autres langues ne peuvent pas être expurgées et il n'y a aucun avertissement ni échec de la tâche.

Modèles de langue personnalisés, vocabulaires personnalisés et filtres de vocabulaires personnalisés

Si vous souhaitez ajouter un ou plusieurs modèles de langue personnalisés, vocabulaires personnalisés ou filtres de vocabulaires personnalisés à votre demande d'identification des langues, vous devez inclure le paramètre [LanguageIdSettings](#). Vous pouvez ensuite spécifier un code de langue avec un modèle de langue personnalisé, un vocabulaire personnalisé et un filtre de vocabulaire personnalisé correspondants. Notez que l'identification multilingue ne prend pas en charge les modèles de langue personnalisés.

Il est recommandé d'inclure `LanguageOptions` lorsque vous utilisez [LanguageIdSettings](#) pour vous assurer que le bon dialecte linguistique est identifié. Par exemple, si vous spécifiez un vocabulaire en-US personnalisé, mais que vous Amazon Transcribe déterminez que c'est la langue parlée dans vos médias en-AU, votre vocabulaire personnalisé n'est pas appliqué à votre transcription. Si vous incluez `LanguageOptions` et spécifiez en-US comme seul dialecte de langue anglaise, votre vocabulaire personnalisé est appliqué à votre transcription.

Pour des exemples de [LanguageIdSettings](#) dans une demande, reportez-vous à l'option 2 des volets déroulants de l'AWS CLI et des kits SDK AWS de la section [Utilisation de l'identification des langues avec des transcriptions par lots](#).

Utilisation de l'identification des langues avec des transcriptions par lots

Vous pouvez utiliser l'identification automatique des langues dans une tâche de transcription par lots en utilisant la AWS Management Console, l'AWS CLI ou des kits SDK AWS . Consultez les exemples suivants :

AWS Management Console

1. Connectez-vous à la [AWS Management Console](#).
2. Dans le volet de navigation, choisissez Tâches de transcription, puis sélectionnez Créer une tâche (en haut à droite). La page Spécifier les détails de la tâche s'ouvre.
3. Dans le volet Paramètres de la tâche, recherchez la section Paramètres de langue et sélectionnez Identification automatique des langues ou Identification automatique de plusieurs langues.

Vous pouvez sélectionner plusieurs options de langue (dans la liste déroulante Sélectionner les langues) si vous connaissez les langues présentes dans votre fichier audio. La sélection d'options de langue peut améliorer la précision, mais n'est pas obligatoire.

Specify job details [Info](#)

Job settings

Name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, . (period), _ (underscore), and - (hyphen).

Language settings

You can transcribe your audio file in a language that you specify or have Amazon Transcribe identify and transcribe it in the predominant language.

- Specific language** [Info](#)
If you know the language spoken in your source audio, choose this option to get the most accurate results. The options available for additional processing vary between languages.
- Automatic language identification** [Info](#)
If you don't know the language spoken in your audio files, choose this option. You have access to fewer options for additional processing than if you choose **Specific language**.
- Automatic multiple languages identification** [Info](#)
If there are multiple languages spoken in your audio files and you're not sure what these languages are, choose this option. This selection provides limited additional processing options compared to **Specific language**.

Language options for automatic language identification - *optional*

To improve accuracy, choose at least two languages spoken the most often in your audio library. Amazon Transcribe chooses from one of the languages you've specified to transcribe each audio file. Leave this field empty if you're unsure about which languages to select.

- English, US (en-US)
- English, AU (en-AU)
- English, UK (en-GB)
- Hindi, IN (hi-IN)
- Spanish, US (es-US)

4. Renseignez les autres champs que vous souhaitez inclure sur la page Spécifier les détails de la tâche, puis sélectionnez Suivant. Vous accédez alors à la page Configurer la tâche - facultatif.

Configure job - *optional* [Info](#)

Audio settings

Audio identification [Info](#)
Choose to split multi-channel audio into separate channels for transcription, or identify speakers in the input audio.

Alternative results [Info](#)
Enable to view more transcription results

Content removal

Content removal conceals information in the resulting transcript from your source audio file. Amazon Transcribe changes items in the transcript and does not modify the source audio.

PII redaction [Info](#)
Label the type of PII and also mask the content with the PII entity type in the transcription output. For example, (123) 456-7890 will be masked as [PHONE].

Vocabulary filtering [Info](#)
Vocabulary filtering can remove, mask or tag specified words in the final transcript.

Customization

Custom vocabulary [Info](#)
A custom vocabulary improves the accuracy of recognizing words and phrases specific to your use case.

[Cancel](#) [Previous](#) [Create Job](#)

5. Sélectionnez Créer une tâche pour exécuter votre tâche de transcription.

AWS CLI

Cet exemple utilise la [start-transcription-job](#) commande et le `IdentifyLanguage` paramètre. Pour plus d'informations, consultez [StartTranscriptionJob](#) et [LanguageIdSettings](#).

Option 1 : Sans le paramètre `language-id-settings`. Utilisez cette option si vous n'incluez pas de modèle de langue personnalisé, de vocabulaire personnalisé ou de filtre de vocabulaire personnalisé dans votre demande. `language-options` est facultative, mais recommandée.

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--transcription-job-name my-first-transcription-job \  
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
--output-bucket-name DOC-EXAMPLE-BUCKET \  
--output-key my-output-files/ \  
--identify-language \ (or --identify-multiple-languages) \  
--language-options "en-US" "hi-IN"
```

Option 2 : Avec le paramètre `language-id-settings`. Utilisez cette option si vous incluez un modèle de langue personnalisé, un vocabulaire personnalisé ou un filtre de vocabulaire personnalisé dans votre demande.

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--transcription-job-name my-first-transcription-job \  
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
--output-bucket-name DOC-EXAMPLE-BUCKET \  
--output-key my-output-files/ \  
--identify-language \ (or --identify-multiple-languages) \  
--language-options "en-US" "hi-IN" \  
--language-id-settings en-US=VocabularyName=my-en-US-vocabulary,en-  
US=VocabularyFilterName=my-en-US-vocabulary-filter,en-US=LanguageModelName=my-en-US-  
language-model,hi-IN=VocabularyName=my-hi-IN-vocabulary,hi-IN=VocabularyFilterName=my-  
hi-IN-vocabulary-filter
```

Voici un autre exemple d'utilisation de la [start-transcription-job](#) commande et d'un corps de requête identifiant la langue.

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--cli-input-json file://filepath/my-first-language-id-job.json
```

Le fichier `my-first-language-id-job.json` contient le corps de requête suivant.

Option 1 : Sans le paramètre `LanguageIdSettings`. Utilisez cette option si vous n'incluez pas de modèle de langue personnalisé, de vocabulaire personnalisé ou de filtre de vocabulaire personnalisé dans votre demande. `LanguageOptions` est facultative, mais recommandée.

```
{
  "TranscriptionJobName": "my-first-transcription-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
  },
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "OutputKey": "my-output-files/",
  "IdentifyLanguage": true, (or "IdentifyMultipleLanguages": true),
  "LanguageOptions": [
    "en-US", "hi-IN"
  ]
}
```

Option 2 : Avec le paramètre `LanguageIdSettings`. Utilisez cette option si vous incluez un modèle de langue personnalisé, un vocabulaire personnalisé ou un filtre de vocabulaire personnalisé dans votre demande.

```
{
  "TranscriptionJobName": "my-first-transcription-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
  },
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "OutputKey": "my-output-files/",
  "IdentifyLanguage": true, (or "IdentifyMultipleLanguages": true)
  "LanguageOptions": [
    "en-US", "hi-IN"
  ],
  "LanguageIdSettings": {
    "en-US" : {
      "LanguageModelName": "my-en-US-language-model",
      "VocabularyFilterName": "my-en-US-vocabulary-filter",
      "VocabularyName": "my-en-US-vocabulary"
    },
    "hi-IN": {
      "VocabularyName": "my-hi-IN-vocabulary",
      "VocabularyFilterName": "my-hi-IN-vocabulary-filter"
    }
  }
}
```

```
}  
}
```

AWS SDK for Python (Boto3)

Cet exemple utilise le AWS SDK for Python (Boto3) pour identifier la langue de votre fichier à l'aide de l'IdentifyLanguage argument de la méthode [start_transcription_job](#). Pour plus d'informations, consultez [StartTranscriptionJob](#) et [LanguageIdSettings](#).

Pour d'autres exemples d'utilisation AWS des SDK, notamment des exemples spécifiques aux fonctionnalités, des scénarios et des exemples multiservices, reportez-vous au chapitre. [Exemples de code pour Amazon Transcribe à l'aide de kits de développement logiciel AWS](#)

Option 1 : Sans le paramètre LanguageIdSettings. Utilisez cette option si vous n'incluez pas de modèle de langue personnalisé, de vocabulaire personnalisé ou de filtre de vocabulaire personnalisé dans votre demande. LanguageOptions est facultative, mais recommandée.

```
from __future__ import print_function  
import time  
import boto3  
transcribe = boto3.client('transcribe', 'us-west-2')  
job_name = "my-first-transcription-job"  
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
transcribe.start_transcription_job(  
    TranscriptionJobName = job_name,  
    Media = {  
        'MediaFileUri': job_uri  
    },  
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',  
    OutputKey = 'my-output-files/',  
    MediaFormat = 'flac',  
    IdentifyLanguage = True, (or IdentifyMultipleLanguages = True),  
    LanguageOptions = [  
        'en-US', 'hi-IN'  
    ]  
)  
  
while True:  
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)  
    if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:  
        break  
    print("Not ready yet...")
```

```
time.sleep(5)
print(status)
```

Option 2 : Avec le paramètre `LanguageIdSettings`. Utilisez cette option si vous incluez un modèle de langue personnalisé, un vocabulaire personnalisé ou un filtre de vocabulaire personnalisé dans votre demande.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe')
job_name = "my-first-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
transcribe.start_transcription_job(
    TranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'my-output-files/',
    MediaFormat='flac',
    IdentifyLanguage=True, (or IdentifyMultipleLanguages=True)
    LanguageOptions = [
        'en-US', 'hi-IN'
    ],
    LanguageIdSettings={
        'en-US': {
            'VocabularyName': 'my-en-US-vocabulary',
            'VocabularyFilterName': 'my-en-US-vocabulary-filter',
            'LanguageModelName': 'my-en-US-language-model'
        },
        'hi-IN': {
            'VocabularyName': 'my-hi-IN-vocabulary',
            'VocabularyFilterName': 'my-hi-IN-vocabulary-filter'
        }
    }
)

while True:
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)
    if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:
        break
    print("Not ready yet...")
```

```
time.sleep(5)
print(status)
```

Identification des langues avec des transcriptions en streaming

L'identification des langues en streaming permet d'identifier la langue dominante parlée dans votre flux multimédia. Amazon Transcribe a besoin d'au moins trois secondes d'élocution pour identifier la langue.

Si votre flux ne contient qu'une seule langue, vous pouvez activer l'identification d'une langue unique, qui identifie la langue dominante parlée dans votre fichier multimédia et crée une transcription en utilisant uniquement cette langue.

Si le flux contient plusieurs langues, vous pouvez activer l'identification multilingue, qui identifie toutes les langues parlées dans votre flux et crée votre transcription en utilisant chaque langue identifiée. Notez qu'une transcription multilingue est produite. Vous pouvez utiliser d'autres services, tels qu'Amazon Transcribe, pour traduire votre transcription.

Pour utiliser l'identification des langues en streaming, vous devez fournir au moins deux codes de langue et vous ne pouvez sélectionner qu'un seul dialecte par langue par flux. Cela signifie que vous ne pouvez pas sélectionner en-US et en-AU comme options de langue pour la même transcription.

Vous pouvez également sélectionner une langue préférée parmi les codes de langue que vous fournissez. L'ajout d'une langue préférée peut accélérer le processus d'identification des langues, ce qui est utile pour les clips audio courts.

Important

Si aucun des codes de langue que vous fournissez ne correspond à la langue ou aux langues identifiées dans votre audio, Amazon Transcribe sélectionne la langue la plus proche parmi les codes de langue que vous avez spécifiés. Une transcription dans cette langue est ensuite générée. Par exemple, si votre média est en anglais américain (en-US) et que vous fournissez à Amazon Transcribe les codes de langue zh-CN, fr-FR et de-DE, Amazon Transcribe est susceptible de faire correspondre votre média à l'allemand (de-DE) et de produire une transcription en allemand. La non-concordance entre les codes de langue et les langues parlées peut entraîner une transcription inexacte. Nous vous recommandons donc de faire preuve de prudence lorsque vous incluez des codes de langue.

Si votre média contient deux canaux, Amazon Transcribe peut identifier la langue dominante parlée dans chaque canal. Dans ce cas, définissez le paramètre [ChannelIdentification](#) sur `true` et chaque canal est transcrit séparément. Notez que la valeur par défaut de ce paramètre est `false`. Si vous ne le modifiez pas, seul le premier canal est transcrit et une seule langue est identifiée.

L'identification des langues en streaming ne peut pas être combinée à des modèles de langue personnalisés ou à l'expurgation. Si vous associez l'identification des langues à d'autres fonctionnalités, vous êtes limité aux langues prises en charge par ces fonctionnalités, ainsi qu'aux transcriptions en streaming. Reportez-vous à la section [Langues prises en charge](#).

Note

PCM et FLAC sont les seuls formats audio pris en charge pour l'identification des langues en streaming.

Identification des langues dans un audio multilingue

L'identification multilingue est destinée aux flux multilingues. Elle fournit une transcription qui reflète toutes les langues prises en charge parlées dans le flux. Cela signifie que si les locuteurs changent de langue en cours de conversation, ou si chaque participant parle une langue différente, votre sortie de transcription détecte et transcrit correctement chaque langue.

Par exemple, si le flux contient un locuteur bilingue qui alterne entre l'anglais américain (`en-US`) et l'hindi (`hi-IN`), l'identification multilingue peut identifier et transcrire l'anglais américain en tant que `en-US` et l'hindi parlé en tant que `hi-IN`. Cela diffère de l'identification dans d'une seule langue, où une seule langue dominante est utilisée pour créer une transcription. Dans ce cas, toute langue parlée qui n'est pas la langue dominante est incorrectement transcrite.

Note

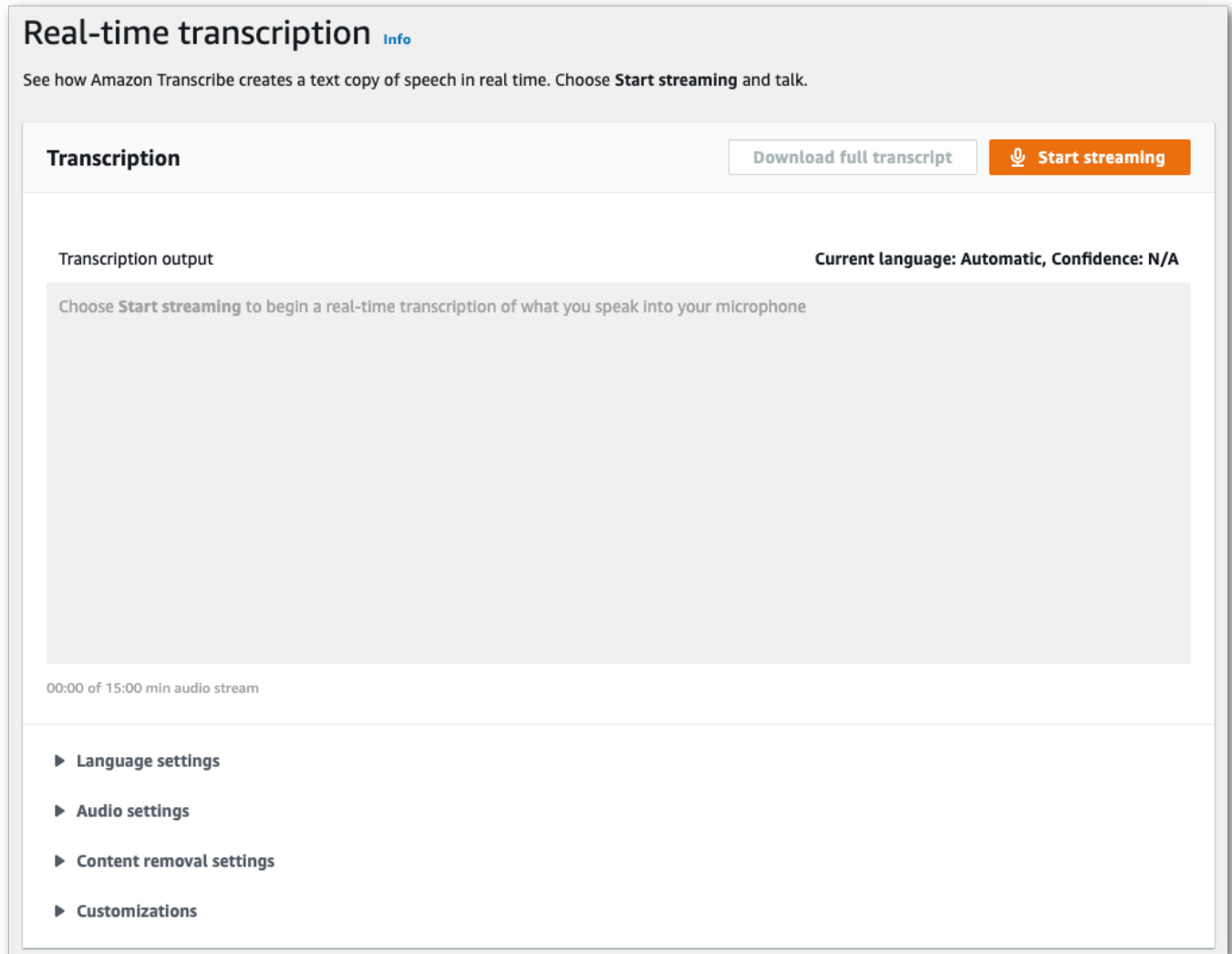
L'expurgation et les modèles de langue personnalisés ne sont actuellement pas pris en charge avec l'identification multilingue.

Utilisation de l'identification des langues avec les médias en streaming

Vous pouvez utiliser l'identification automatique des langues dans une transcription en streaming en utilisant la AWS Management Console, HTTP/2 ou WebSockets. Consultez les exemples suivants :

AWS Management Console

1. Connectez-vous à la [AWS Management Console](#).
2. Dans le panneau de navigation, choisissez Transcription en temps réel. Faites défiler jusqu'à Paramètres de langue et développez ce champ s'il est réduit.



3. Sélectionnez Identification automatique de la langue ou Identification automatique de plusieurs langues.

▼ Language settings

Language settings

You can select a specific language for your transcription or have Amazon Transcribe identify the predominant language in your media and perform the transcription in that language.

Specific language

If you know the language spoken in your source audio, choose this option to get the most accurate results.

Automatic language identification [Info](#)

If you don't know the language spoken in your audio files, choose this option.

Automatic multiple languages identification [Info](#)

If there are multiple languages spoken in your audio files and you're not sure what these languages are, choose this option. This selection provides limited additional processing options compared to **Specific language**.

Language options for automatic language identification

To improve language identification accuracy, select a minimum of 2 language options.

Choose language(s) ▼

Preferred language - *optional*

Specify one preferred language from your previous selection.

Choose language ▼

▶ **Audio settings**

▶ **Content removal settings**

▶ **Customizations**

4. Fournissez au moins deux codes de langue pour votre transcription. Notez que vous ne pouvez fournir qu'un seul dialecte par langue. Par exemple, vous ne pouvez pas sélectionner à la fois en-US et fr-CA comme options de langue pour la même transcription.

▼ **Language settings**

Language settings
You can select a specific language for your transcription or have Amazon Transcribe identify the predominant language in your media and perform the transcription in that language.

Specific language
If you know the language spoken in your source audio, choose this option to get the most accurate results.

Automatic language identification [Info](#)
If you don't know the language spoken in your audio files, choose this option.

Automatic multiple languages identification [Info](#)
If there are multiple languages spoken in your audio files and you're not sure what these languages are, choose this option. This selection provides limited additional processing options compared to **Specific language**.

Language options for automatic language identification
To improve language identification accuracy, select a minimum of 2 language options.

Choose language(s) ▼

English, US (en-US) ✕ French, CA (fr-CA) ✕

Preferred language - optional
Specify one preferred language from your previous selection.

Choose language ▲

Q

None
English, US (en-US)
French, CA (fr-CA)

5. (Facultatif) Dans le sous-ensemble de langues que vous avez sélectionné à l'étape précédente, vous pouvez choisir une langue préférée pour votre transcription.

▼ **Language settings**

Language settings
 You can select a specific language for your transcription or have Amazon Transcribe identify the predominant language in your media and perform the transcription in that language.

Specific language
 If you know the language spoken in your source audio, choose this option to get the most accurate results.

Automatic language identification [Info](#)
 If you don't know the language spoken in your audio files, choose this option.

Language options for automatic language identification
 To improve language identification accuracy, select a minimum of 2 language options.

Choose language(s) ▼

English, US (en-US) X French, CA (fr-CA) X

Preferred language - optional
 Specify one preferred language from your previous selection.

Choose language ▲

None

English, US (en-US)

French, CA (fr-CA)

► **Customizations**

- Vous êtes prêt à transcrire votre flux. Sélectionnez Démarrer le streaming et commencez à parler. Pour mettre fin à votre dictée, sélectionnez Arrêter le streaming.

Flux HTTP/2

Cet exemple crée une requête HTTP/2 avec l'identification des langues activée. Pour plus d'informations sur l'utilisation du streaming HTTP/2 avec Amazon Transcribe, consultez la section [Configuration d'un flux HTTP/2](#). Pour plus de détails sur les paramètres et les en-têtes spécifiques à Amazon Transcribe, consultez la section [StartStreamTranscription](#).

```
POST /stream-transcription HTTP/2
host: transcribestreaming.us-west-2.amazonaws.com
X-Amz-Target: com.amazonaws.transcribe.Transcribe.StartStreamTranscription
Content-Type: application/vnd.amazon.eventstream
X-Amz-Content-Sha256: string
X-Amz-Date: 20220208T235959Z
Authorization: AWS4-HMAC-SHA256 Credential=access-key/20220208/us-west-2/transcribe/
aws4_request, SignedHeaders=content-type;host;x-amz-content-sha256;x-amz-date;x-amz-
target;x-amz-security-token, Signature=string
x-amzn-transcribe-media-encoding: flac
x-amzn-transcribe-sample-rate: 16000
```

```
x-amzn-transcribe-identify-language: true
x-amzn-transcribe-language-options: en-US,de-DE
x-amzn-transcribe-preferred-language: en-US
transfer-encoding: chunked
```

Cet exemple crée une requête HTTP/2 avec l'identification de plusieurs langues activée. Pour plus d'informations sur l'utilisation du streaming HTTP/2 avec Amazon Transcribe, consultez la section [Configuration d'un flux HTTP/2](#). Pour plus de détails sur les paramètres et les en-têtes spécifiques à Amazon Transcribe, consultez la section [StartStreamTranscription](#).

```
POST /stream-transcription HTTP/2
host: transcribestreaming.us-west-2.amazonaws.com
X-Amz-Target: com.amazonaws.transcribe.Transcribe.StartStreamTranscription
Content-Type: application/vnd.amazon.eventstream
X-Amz-Content-Sha256: string
X-Amz-Date: 20220208T235959Z
Authorization: AWS4-HMAC-SHA256 Credential=access-key/20220208/us-west-2/transcribe/
aws4_request, SignedHeaders=content-type;host;x-amz-content-sha256;x-amz-date;x-amz-
target;x-amz-security-token, Signature=string
x-amzn-transcribe-media-encoding: flac
x-amzn-transcribe-sample-rate: 16000
x-amzn-transcribe-identify-multiple-languages: true
x-amzn-transcribe-language-options: en-US,de-DE
x-amzn-transcribe-preferred-language: en-US
transfer-encoding: chunked
```

Si vous utilisez `identify-language` ou `identify-multiple-languages` dans la demande, vous devez également inclure `language-options`. Vous ne pouvez pas utiliser à la fois `language-code` et `identify-language` dans la même demande.

Les définitions des paramètres se trouvent dans la [Référence de l'API](#) ; les paramètres communs à toutes les opérations d'API AWS sont répertoriés dans la section [Paramètres courants](#).

Flux WebSocket

Cet exemple crée une URL présignée qui utilise l'identification des langues dans un flux WebSocket. Les sauts de ligne ont été ajoutés pour faciliter la lecture. Pour plus d'informations sur l'utilisation de flux WebSocket avec Amazon Transcribe, consultez la section [Configuration d'un WebSocket stream](#). Pour plus de détails sur les paramètres, consultez [StartStreamTranscription](#).

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/stream-transcription-
websocket?
```

```
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-
west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=string
&X-Amz-SignedHeaders=content-type%3Bhost%3Bx-amz-date
&media-encoding=flac
&sample-rate=16000
&identify-language=true
&language-options=en-US,de-DE
&preferred-language=en-US
```

Cet exemple crée une URL présignée qui utilise l'identification de plusieurs langues dans un flux WebSocket. Les sauts de ligne ont été ajoutés pour faciliter la lecture. Pour plus d'informations sur l'utilisation de flux WebSocket avec Amazon Transcribe, consultez la section [Configuration d'un WebSocket stream](#). Pour plus de détails sur les paramètres, consultez [StartStreamTranscription](#).

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/stream-transcription-
websocket?
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-
west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=string
&X-Amz-SignedHeaders=content-type%3Bhost%3Bx-amz-date
&media-encoding=flac
&sample-rate=16000
&identify-multiple-languages=true
&language-options=en-US,de-DE
&preferred-language=en-US
```

Si vous utilisez `identify-language` ou `identify-multiple-languages` dans la demande, vous devez également inclure `language-options`. Vous ne pouvez pas utiliser à la fois `language-code` et `identify-language` dans la même demande.

Les définitions des paramètres se trouvent dans la [Référence de l'API](#) ; les paramètres communs à toutes les opérations d'API AWS sont répertoriés dans la section [Paramètres courants](#).

Transcriptions alternatives

Lors de la Amazon Transcribe transcription audio, il crée différentes versions de la même transcription et attribue un score de confiance à chaque version. Dans une transcription classique, vous n'obtenez que la version ayant le score de confiance le plus élevé.

Si vous activez les transcriptions alternatives, Amazon Transcribe renvoie d'autres versions de votre transcription présentant des niveaux de confiance inférieurs. Vous pouvez choisir de renvoyer jusqu'à 10 transcriptions alternatives. Si vous spécifiez un nombre d'alternatives supérieur à celui Amazon Transcribe indiqué, seul le nombre réel d'alternatives est renvoyé.

Toutes les alternatives se trouvent dans le même fichier de sortie de transcription et sont présentées au niveau du segment. Les segments sont des pauses naturelles du discours, telles qu'un changement de haut-parleur ou une pause dans le son.

Les transcriptions alternatives ne sont disponibles que pour les transcriptions par lots.

Votre sortie de transcription est structurée comme suit : Les ellipses (. . .) dans les exemples de code indiquent où le contenu a été supprimé par souci de concision.

1. Transcription finale complète pour un segment donné.

```
"results": {
  "language_code": "en-US",
  "transcripts": [
    {
      "transcript": "The amazon is the largest rainforest on the planet."
    }
  ],

```

2. Un score de confiance pour chaque mot de la transcript section précédente.

```
"items": [
  {
    "start_time": "1.15",
    "end_time": "1.35",
    "alternatives": [
      {
        "confidence": "1.0",
        "content": "The"
      }
    ]
  }

```

```

    ],
    "type": "pronunciation"
  },
  {
    "start_time": "1.35",
    "end_time": "2.05",
    "alternatives": [
      {
        "confidence": "1.0",
        "content": "amazon"
      }
    ],
    "type": "pronunciation"
  },

```

3. Vos transcriptions alternatives se trouvent dans la `segments` partie de votre sortie de transcription. Les alternatives pour chaque segment sont classées par score de confiance décroissant.

```

"segments": [
  {
    "start_time": "1.04",
    "end_time": "5.065",
    "alternatives": [
      ...
      "transcript": "The amazon is the largest rain forest on the
planet.",
      "items": [
        {
          "start_time": "1.15",
          "confidence": "1.0",
          "end_time": "1.35",
          "type": "pronunciation",
          "content": "The"
        },
        ...
        {
          "start_time": "3.06",
          "confidence": "0.0037",
          "end_time": "3.38",
          "type": "pronunciation",
          "content": "rain"
        }
      ],

```

```
{
  "start_time": "3.38",
  "confidence": "0.0037",
  "end_time": "3.96",
  "type": "pronunciation",
  "content": "forest"
},
```

4. Un statut à la fin de la sortie de votre transcription.

```
"status": "COMPLETED"
}
```

Demande de transcriptions alternatives

Vous pouvez demander des transcriptions alternatives à l'aide des AWS Management Console, AWS CLI, ou AWS SDK ; consultez les exemples suivants :

AWS Management Console

1. Connectez-vous à [AWS Management Console](#).
2. Dans le volet de navigation, choisissez Tâches de transcription, puis sélectionnez Créer une tâche (en haut à droite). La page Spécifier les détails de la tâche s'ouvre.

Specify job details [Info](#)

Job settings

Name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, . (period), _ (underscore), and - (hyphen).

Model type [Info](#)

Choose the type of model to use for the transcription job.

General model
To use a model that is not specialized for a particular use case, choose this option. Configuration options vary between languages.

Custom language model
To use a model that you trained for your specific use case, choose this option. This model has fewer configuration options than the general model.

Language settings

You can transcribe your audio file in a language that you specify or have Amazon Transcribe identify and transcribe it in the predominant language.

Specific language [Info](#)
If you know the language spoken in your source audio, choose this option to get the most accurate results. The options available for additional processing vary between languages.

Automatic language identification [Info](#)
If you don't know the language spoken in your audio files, choose this option. You have access to fewer options for additional processing than if you choose **Specific language**.

Language

Choose the language of the input audio.

► **Additional settings**

3. Remplissez tous les champs que vous souhaitez inclure sur la page Spécifier les détails du poste, puis sélectionnez Suivant. Vous accédez alors à la page Configurer la tâche - facultative.

Sélectionnez Résultats alternatifs et spécifiez le nombre maximum de résultats de transcription alternatifs que vous souhaitez inclure dans votre transcription.

Configure job - optional [Info](#)

Audio settings

Audio identification [Info](#)
Choose to split multi-channel audio into separate channels for transcription, or identify speakers in the input audio.

Alternative results [Info](#)
Enable to view more transcription results

Maximum alternatives
Provide the number of alternative transcription to provide in the text output.

The maximum number of alternative results is 10.

4. Sélectionnez Créer une tâche pour exécuter votre tâche de transcription.

AWS CLI

Cet exemple utilise la [start-transcription-job](#) commande et le `ShowAlternatives` paramètre. Pour plus d'informations, consultez [StartTranscriptionJob](#) et [ShowAlternatives](#).

Notez que si vous incluez `ShowAlternatives=true` dans votre demande, vous devez également inclure `MaxAlternatives`.

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--transcription-job-name my-first-transcription-job \  
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
--output-bucket-name DOC-EXAMPLE-BUCKET \  
--output-key my-output-files/ \  
--language-code en-US \  
--settings ShowAlternatives=true,MaxAlternatives=4
```

Voici un autre exemple d'utilisation de la [start-transcription-job](#) commande et d'un corps de requête qui inclut des transcriptions alternatives.

```
aws transcribe start-transcription-job \  

```



```
--region us-west-2 \  
--cli-input-json file://filepath/my-first-alt-transcription-job.json
```

Le fichier `my-first-alt-transcription-job.json` contient le corps de requête suivant.

```
{  
  "TranscriptionJobName": "my-first-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
  },  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "OutputKey": "my-output-files/",  
  "LanguageCode": "en-US",  
  "Settings": {  
    "ShowAlternatives": true,  
    "MaxAlternatives": 4  
  }  
}
```

AWS SDK for Python (Boto3)

L'exemple suivant utilise le AWS SDK for Python (Boto3) pour demander des transcriptions alternatives en utilisant l'`ShowAlternatives` argument de la méthode [start_transcription_job](#). Pour plus d'informations, consultez [StartTranscriptionJob](#) et [ShowAlternatives](#).

Pour des exemples supplémentaires d'utilisation AWS des SDK, y compris des exemples spécifiques à des fonctionnalités, des scénarios et des exemples interservices, consultez le [Exemples de code pour Amazon Transcribe à l'aide de kits de développement logiciel AWS](#) chapitre.

Notez que si vous incluez '`ShowAlternatives`':`True` dans votre demande, vous devez également inclure `MaxAlternatives`.

```
from __future__ import print_function  
import time  
import boto3  
transcribe = boto3.client('transcribe', 'us-west-2')  
job_name = "my-first-transcription-job"  
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
transcribe.start_transcription_job(  
    TranscriptionJobName = job_name,  
    Media = {  
        'MediaFileUri': job_uri
```

```
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'my-output-files/',
    LanguageCode = 'en-US',
    Settings = {
        'ShowAlternatives':True,
        'MaxAlternatives':4
    }
)

while True:
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)
    if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Améliorer la précision de la transcription grâce à des vocabulaires et des modèles linguistiques personnalisés

Si votre média contient des termes spécifiques à un domaine ou non standard, tels que des noms de marque, des acronymes, des termes techniques et du jargon, Amazon Transcribe il est possible que ces termes ne soient pas correctement capturés dans votre sortie de transcription.

Pour corriger les inexactitudes de transcription et personnaliser votre sortie en fonction de votre cas d'utilisation spécifique, vous pouvez créer [Vocabulaires personnalisés](#) et [Modèles de langage personnalisés](#).

- [Vocabulaires personnalisés](#) sont conçus pour améliorer et améliorer à la fois la reconnaissance et la mise en forme de mots spécifiques dans tous les contextes. Cela implique de fournir Amazon Transcribe des mots et, éventuellement, des formulaires de prononciation et d'affichage.

Si certains termes ne Amazon Transcribe sont pas correctement affichés dans vos transcriptions, vous pouvez créer un fichier de vocabulaire personnalisé qui indique Amazon Transcribe comment vous souhaitez que ces termes soient affichés. Cette approche spécifique aux mots est la plus appropriée pour corriger des termes tels que les noms de marque et les acronymes.

- [Modèles de langage personnalisés](#) sont conçus pour saisir le contexte associé aux termes. Cela implique de Amazon Transcribe fournir un volume important de données textuelles spécifiques au domaine.

Si le rendu des termes techniques n'Amazon Transcribe est pas correct ou si vous utilisez un homophone incorrect dans vos transcriptions, vous pouvez créer un modèle linguistique personnalisé qui enseigne la langue spécifique à Amazon Transcribe votre domaine. Par exemple, un modèle linguistique personnalisé peut apprendre quand utiliser « floe » (banquise) par rapport à « flow » (flux linéaire).

Cette approche contextuelle est la plus appropriée pour transcrire de grands volumes de discours spécifiques à un domaine. Les modèles linguistiques personnalisés peuvent apporter des améliorations significatives en termes de précision par rapport aux seuls vocabulaires personnalisés. Lorsque vous utilisez des transcriptions par lots, vous pouvez inclure à la fois un modèle linguistique personnalisé et un vocabulaire personnalisé dans votre demande.

i Tip

Pour obtenir une précision de transcription optimale, utilisez des vocabulaires personnalisés en conjonction avec vos modèles linguistiques personnalisés.

Pour une démonstration vidéo expliquant comment créer un vocabulaire personnalisé à l'aide du AWS Management Console, voir [Utilisation d'un vocabulaire personnalisé](#).

Pour une démonstration vidéo sur la création et l'utilisation de modèles linguistiques personnalisés, voir [Utilisation de modèles linguistiques personnalisés \(CLM\) pour améliorer la précision de la transcription](#).

i Approfondissez vos connaissances grâce au blog sur AWS Machine Learning

Vocabulaires Custom

- [Transcriptions en direct des courses de F1 avec Amazon Transcribe](#)

Modèles linguistiques personnalisés :

- [Création de modèles linguistiques personnalisés pour améliorer les speech-to-text performances Amazon Transcribe](#)
- [Améliorez la précision de la transcription des cours en classe grâce à des modèles linguistiques personnalisés pour Amazon Transcribe](#)

Vocabulaires personnalisés

Utilisez des vocabulaires personnalisés pour améliorer la précision de la transcription d'un ou de plusieurs mots spécifiques. Il s'agit généralement de termes spécifiques à un domaine, tels que des noms de marque et des acronymes, des noms propres et des mots qu' Amazon Transcribe ne rend pas correctement.

Les vocabulaires personnalisés peuvent être utilisés avec toutes les langues prises en charge. Notez que seuls les caractères répertoriés dans le [jeu de caractères](#) de votre langue peuvent être utilisés dans un vocabulaire personnalisé.

⚠ Important

Vous êtes responsable de l'intégrité de vos propres données lorsque vous les utilisez Amazon Transcribe. N'entrez pas d'informations confidentielles, personnelles (PII) ou protégées sur la santé (PHI) dans un vocabulaire personnalisé.

Considérations à prendre en compte lors de la création d'un vocabulaire personnalisé :

- Vous pouvez avoir jusqu'à 100 fichiers de vocabulaire personnalisés par Compte AWS
- La taille de chaque vocabulaire personnalisé est limitée à 50 Ko.
- Si vous utilisez l'API pour créer votre vocabulaire personnalisé, votre fichier de vocabulaire doit être au format texte (*.txt). Si vous utilisez le AWS Management Console, votre fichier de vocabulaire peut être au format texte (*.txt) ou au format valeur séparée par des virgules (*.csv).
- Chaque entrée d'un vocabulaire personnalisé ne peut pas dépasser 256 caractères.
- Pour utiliser un vocabulaire personnalisé, celui-ci doit avoir été créé en même temps Région AWS que votre transcription.

ℹ Tip

Vous pouvez tester votre vocabulaire personnalisé à l'aide du AWS Management Console. Une fois que votre vocabulaire personnalisé est prêt à être utilisé, connectez-vous au AWS Management Console, sélectionnez Transcription en temps réel, faites défiler l'écran jusqu'à Personnalisations, activez Vocabulaire personnalisé et sélectionnez votre vocabulaire personnalisé dans la liste déroulante. Sélectionnez ensuite Démarrer le streaming. Prononcez quelques mots de votre vocabulaire personnalisé dans votre micro pour vérifier s'ils s'affichent correctement.

Tableaux de vocabulaires personnalisés et listes de vocabulaires personnalisés

Important

Les vocabulaires personnalisés sous forme de liste sont obsolètes. Si vous créez un nouveau vocabulaire personnalisé, utilisez le [format de tableau](#).

Les tableaux vous offrent davantage d'options et de contrôle sur l'entrée et la sortie des mots de votre vocabulaire personnalisé. Avec les tableaux, vous devez définir plusieurs catégories (Phrase and DisplayAs), ce qui vous permet d'affiner vos résultats.

Les listes ne comportent pas d'options supplémentaires. Vous pouvez donc uniquement saisir les entrées telles que vous souhaitez qu'elles apparaissent dans votre transcription, en remplaçant tous les espaces par des tirets.

Les AWS SDK et AWS Management Console AWS CLI, utilisent tous des tables de vocabulaire personnalisées de la même manière ; les listes sont utilisées différemment pour chaque méthode et peuvent donc nécessiter un formatage supplémentaire pour une utilisation réussie entre les méthodes.

Pour plus d'informations, consultez [Création d'un vocabulaire personnalisé à l'aide d'un tableau](#) et [Création d'un vocabulaire personnalisé à l'aide d'une liste](#).

Pour aller un peu plus loin et apprendre à utiliser Amazon Augmented AI avec des vocabulaires personnalisés, consultez la section [Start building a human review along with Amazon Transcribe](#).

Opérations d'API spécifiques aux vocabulaires personnalisés

[CreateVocabulary](#), [DeleteVocabulary](#), [GetVocabulary](#), [ListVocabularies](#), [UpdateVocabulary](#)

Création d'un vocabulaire personnalisé à l'aide d'un tableau

L'utilisation d'un tableau est la méthode préférée pour créer votre vocabulaire personnalisé. Les tableaux de vocabulaires doivent comporter quatre colonnes (Phrase, SoundsLike, IPA, and DisplayAs), qui peuvent être incluses dans n'importe quel ordre :

Expression	SoundsLike	IPA	DisplayAs
<p>Obligatoire. Chaque ligne de votre tableau doit contenir une entrée dans cette colonne.</p> <p>N'utilisez pas d'espaces dans cette colonne.</p> <p>Si votre entrée contient plusieurs mots, séparez-les par un trait d'union (-). Par exemple, Andorra-la-Vella ou Los-Angeles .</p> <p>Pour les acronymes, les lettres prononcées doivent être séparées par un point. Le point final doit également être prononcé. Si votre acronyme est au pluriel, vous devez utiliser un trait d'union entre l'acronyme et le « s ». Par exemple, « CLI » devient C.L.I. (et non C.L.I) et « ABCs » devient A.B.C.-s (et non A.B.C-s).</p>	<p>SoundsLike n'est plus pris en charge pour le vocabulaire personnalisé. Veuillez laisser la colonne vide.</p> <p>Toutes les valeurs de cette colonne seront ignorées.</p> <p>Nous supprimerons le support de cette colonne à l'avenir.</p>	<p>IPAn'est plus pris en charge pour le vocabulaire personnalisé. Veuillez laisser la colonne vide.</p> <p>Toutes les valeurs de cette colonne seront ignorées.</p> <p>Nous supprimerons le support de cette colonne à l'avenir.</p>	<p>Facultatif. Les lignes de cette colonne peuvent être laissées vides.</p> <p>Vous pouvez utiliser des espaces dans cette colonne.</p> <p>Définit l'apparence que vous souhaitez donner à votre entrée dans votre sortie de transcription. Par exemple, Andorra-la-Vella dans la colonne Phrase devient Andorra la Vella dans la colonne DisplayAs .</p> <p>Si une ligne de cette colonne est vide, Amazon Transcribe utilise le contenu de la Phrase colonne pour déterminer le résultat.</p> <p>Vous pouvez inclure des chiffres (0-9) dans cette colonne.</p>

Expression	SoundsLike	IPA	DisplayAs
Si votre expression est composée à la fois d'un mot et d'un acronyme, ces deux éléments doivent être séparés par un trait d'union. Par exemple, « DynamoDB » devient Dynamo-D.B.			
N'incluez pas de chiffres dans cette colonne ; les chiffres doivent être épelés. Par exemple, « VX02Q » devient V.X.-zero-two-Q.			

Points à prendre en compte lors de la création de votre tableau :

- Votre tableau doit contenir les quatre en-têtes (Phrase, SoundsLike, IPA, and DisplayAs) de colonne. La Phrase colonne doit contenir une entrée sur chaque ligne. La possibilité de fournir des entrées de prononciation via IPA et n'SoundsLike est plus prise en charge et vous pouvez laisser la colonne vide. Toutes les valeurs de ces colonnes seront ignorées.
- Chaque colonne doit être délimitée par des tabulations ou des virgules (,) ; cela s'applique à chaque ligne de votre fichier de vocabulaire personnalisé. Si une ligne contient des colonnes vides, vous devez tout de même inclure un délimiteur (tabulation ou virgule) pour chaque colonne.
- Les espaces ne sont autorisés que dans les colonnes IPA et DisplayAs. N'utilisez pas d'espaces pour séparer les colonnes.
- IPA et SoundsLike sont plus pris en charge pour le vocabulaire personnalisé. Veuillez laisser la colonne vide. Toutes les valeurs de ces colonnes seront ignorées. Nous supprimerons le support de cette colonne à l'avenir.

- La colonne `DisplayAs` prend en charge les symboles et les caractères spéciaux (par exemple, `C+`). Toutes les autres colonnes prennent en charge les caractères répertoriés sur la page de [jeu de caractères](#) de votre langue.
- Si vous souhaitez inclure des chiffres dans la colonne `Phrase`, vous devez les épeler. Les chiffres (0-9) ne sont pris en charge que dans la colonne `DisplayAs`.
- Vous devez enregistrer votre tableau sous forme de fichier texte brut (*.txt) au format LF. Si vous utilisez un autre format, par exemple CRLF, votre vocabulaire personnalisé ne peut pas être traité.
- Vous devez télécharger votre fichier de vocabulaire personnalisé dans un Amazon S3 bucket et le traiter à l'aide de celui-ci [CreateVocabulary](#) avant de pouvoir l'inclure dans une demande de transcription. Reportez-vous à [Création de tableaux de vocabulaires personnalisés](#) pour obtenir des instructions.

Note

Saisissez des acronymes, ou d'autres termes dont les lettres doivent être prononcées individuellement, sous forme de lettres séparées par des points (**A . B . C .**). Pour saisir le pluriel d'un acronyme, tel que « ABCs », séparez le « s » de l'acronyme par un tiret (**A . B . C . -s**). Vous pouvez utiliser des majuscules ou des minuscules pour saisir un acronyme. Les acronymes ne sont pas pris en charge dans toutes les langues ; consultez la section [Langues prises en charge et fonctionnalités spécifiques aux langues](#).

Voici un exemple de tableau de vocabulaire personnalisé (où **[TAB]** représente un caractère de tabulation) :

```
Phrase[TAB]SoundsLike[TAB]IPA[TAB]DisplayAs
Los-Angeles[TAB][TAB][TAB]Los Angeles
Eva-Maria[TAB][TAB][TAB]
A.B.C.-s[TAB][TAB][TAB]ABCs
Amazon-dot-com[TAB][TAB][TAB]Amazon.com
C.L.I.[TAB][TAB][TAB]CLI
Andorra-la-Vella[TAB][TAB][TAB]Andorra la Vella
Dynamo-D.B.[TAB][TAB][TAB]DynamoDB
V.X.-zero-two[TAB][TAB][TAB]VX02
V.X.-zero-two-Q.[TAB][TAB][TAB]VX02Q
```

Pour plus de clarté visuelle, voici le même tableau avec des colonnes alignées. N'ajoutez pas d'espaces entre les colonnes de votre tableau de vocabulaire personnalisé ; votre tableau doit sembler mal aligné comme dans l'exemple précédent.

Phrase	[TAB]SoundsLike	[TAB]IPA	[TAB]DisplayAs
Los-Angeles	[TAB]	[TAB]	[TAB]Los Angeles
Eva-Maria	[TAB]	[TAB]	[TAB]
A.B.C.-s	[TAB]	[TAB]	[TAB]ABCs
amazon-dot-com	[TAB]	[TAB]	[TAB]amazon.com
C.L.I.	[TAB]	[TAB]	[TAB]CLI
Andorra-la-Vella	[TAB]	[TAB]	[TAB]Andorra la Vella
Dynamo-D.B.	[TAB]	[TAB]	[TAB]DynamoDB
V.X.-zero-two	[TAB]	[TAB]	[TAB]VX02
V.X.-zero-two-Q.	[TAB]	[TAB]	[TAB]VX02Q

Création de tableaux de vocabulaires personnalisés

Pour traiter une table de vocabulaire personnalisée à utiliser avec Amazon Transcribe, consultez les exemples suivants :

AWS Management Console

1. Connectez-vous à la [AWS Management Console](#).
2. Dans le panneau de navigation, choisissez Vocabulaire personnalisé. La page Vocabulaire personnalisé s'ouvre. Elle vous permet de consulter les vocabulaires existants ou d'en créer un nouveau.
3. Sélectionnez Créer du vocabulaire.

The screenshot shows the 'Custom vocabulary' page in the Amazon Transcribe console. At the top, there's a breadcrumb 'Amazon Transcribe > Custom vocabulary' and a title 'Custom vocabulary' with an 'Info' link. Below the title is a sub-header 'Overview' and a brief instruction: 'Use custom vocabularies to improve transcription accuracy. Learn more'. The main content is divided into two steps: '1. Create custom vocabulary' and '2. Apply to Real-time or batch transcription'. Step 1 includes an icon of a document with a pencil and text explaining that users can create a vocabulary by uploading a file or adding phrases, and that templates (.csv, .txt) are available. Step 2 includes an icon of a document with a play button and text explaining that the created vocabulary can be applied to real-time or batch transcription jobs. Below this is a 'Manage vocabularies' section with a search bar, a 'Filter by: All' dropdown, and a table with columns for Name, Language, Last modified, and Status. The table is currently empty, showing 'Empty resources' and 'No resources to display', with a 'Create vocabulary' button at the bottom.

Vous accédez alors à la page Créer du vocabulaire. Entrez un nom pour votre nouveau vocabulaire personnalisé.

Trois possibilités s'offrent à vous :

- a. Téléchargez un fichier txt ou csv depuis votre ordinateur.

Vous pouvez créer votre vocabulaire personnalisé à partir de zéro ou télécharger un modèle pour vous aider à démarrer. Votre vocabulaire est ensuite renseigné automatiquement dans le volet Afficher et modifier le vocabulaire.

Create vocabulary [Info](#)

Vocabulary settings

Name

Vocabulary names can be up to 200 characters in length. Allowed characters: a-z, A-Z, 0-9, periods (.), dashes (-), and underscores (_).

Language

 ▼

Create and import vocabulary [Info](#)

Vocabulary input source

- File upload**
Upload a vocabulary table from your computer.
- S3 location**
Import a vocabulary table from an S3 location.
- Create vocabulary on console**
Manually create a vocabulary table on the console.

Download vocabulary template – *Optional*

Download and complete a custom vocabulary template in your preferred format.

 **Download template** ▼

Import from file

 **Choose File**

File format: txt, csv, maximum size 50 KB.

- b. Importez un fichier txt ou csv à partir d'un Amazon S3 emplacement.

Vous pouvez créer votre vocabulaire personnalisé à partir de zéro ou télécharger un modèle pour vous aider à démarrer. Téléchargez votre fichier de vocabulaire terminé dans un compartiment Amazon S3 et spécifiez son URI dans votre demande. Votre vocabulaire est ensuite renseigné automatiquement dans le volet Afficher et modifier le vocabulaire.

Create and import vocabulary [Info](#)

Vocabulary input source

File upload
Upload a vocabulary table from your computer.

S3 location
Import a vocabulary table from an S3 location.

Create vocabulary on console
Manually create a vocabulary table on the console.

Download vocabulary template – Optional
Download and complete a custom vocabulary template in your preferred format.

[Download template](#) ▼

Import from S3
Provide a path to the S3 location where your vocabulary file is stored. To find a path, go to [Amazon S3](#).

Resource URI

[View](#) [Browse S3](#)

- c. Créez manuellement votre vocabulaire dans la console.

Accédez au volet Afficher et modifier le vocabulaire et sélectionnez Ajouter 10 lignes. Vous pouvez désormais saisir des termes manuellement.

Create and import vocabulary [Info](#)

Vocabulary input source

File upload
Upload a vocabulary table from your computer.

S3 location
Import a vocabulary table from an S3 location.

Create vocabulary on console
Manually create a vocabulary table on the console.

View and edit vocabulary (0) [Reset vocabulary](#) [Delete](#) [Download latest vocabulary](#) ▼

[Show all](#) ▼ < 1 >

Phrase ↗	SoundsLike (optional) ↗	IPA (optional) ↗	DisplayAs (optional) ↗
No rows added yet			

[Add 10 rows](#)

4. Vous pouvez modifier votre vocabulaire dans le volet Afficher et modifier le vocabulaire. Pour apporter des modifications, cliquez sur l'entrée que vous souhaitez modifier.

View and edit vocabulary - new (10) [Info](#) Reset vocabulary Delete Download latest ▼

Q Filter Phrase, SoundsLike, IPA or DisplayAs Show all ▼ < 1 >

<input type="checkbox"/>	Phrase ✎ ▼	SoundsLike - optional ✎ ▼	IPA - optional ✎ ▼	DisplayAs - optional ✎ ▼
<input type="checkbox"/>	Amazon-E.-C.-two	-	-	Amazon EC2
<input type="checkbox"/>	Amazon-S.-three	-	-	Amazon S3
<input type="checkbox"/>	Amazon-elasticache	-	-	Amazon ElastiCache
<input type="checkbox"/>	Amazon-sagemaker	-	-	Amazon SageMaker
<input type="checkbox"/>	A.-W.-S.-iam	-	-	AWS IAM
<input type="checkbox"/>	A.-W.-S.-I.-o.-T.	-	-	AWS IoT
<input type="checkbox"/>	A.-W.-S.-W.-A.-F.	-	-	AWS WAF
<input type="checkbox"/>	c.-plus-plus	-	-	C++
<input type="checkbox"/>	nice-d.-c.-v.	-	-	NICE DCV
<input type="checkbox"/>	w.-w.-w.-dot-amazon-dot-com	-	-	www.amazon.com

Add row

Si vous faites une erreur, vous recevez un message d'erreur détaillé afin que vous puissiez corriger tout problème avant de traiter votre vocabulaire. Notez que si vous ne corrigez pas toutes les erreurs avant de sélectionner Créer du vocabulaire, votre demande de vocabulaire échoue.

View and edit vocabulary - new (4) [Info](#) Reset vocabulary Delete Download latest ▼

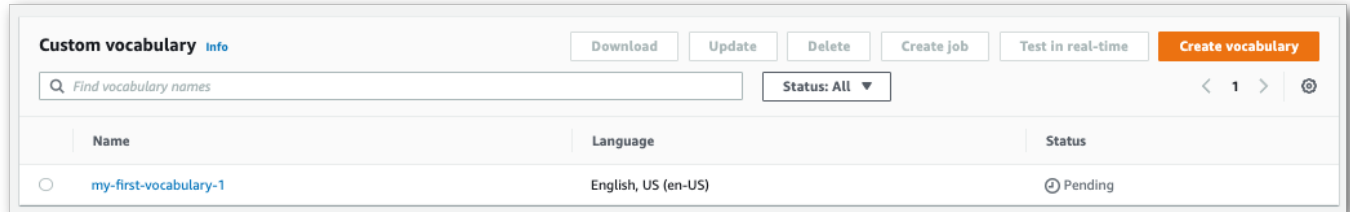
Q Filter Phrase, SoundsLike, IPA or DisplayAs Show all ▼ < 1 >

<input type="checkbox"/>	Phrase ✎ ▼	SoundsLike - optional ✎ ▼	IPA - optional ✎ ▼	DisplayAs - optional ✎ ▼
<input type="checkbox"/>	<input type="text" value="Amazon-E.-C. two"/> ✕ ✓ ⚠ Phrase contains unsupported characters (" "). Phrase contains a formatting error.	-	-	Amazon EC2
<input type="checkbox"/>	Amazon-S.-three	-	-	Amazon S3
<input type="checkbox"/>	c.-plus-plus	-	-	C++
<input type="checkbox"/>	w.-w.-w.-dot-amazon-dot-com	-	-	www.amazon.com

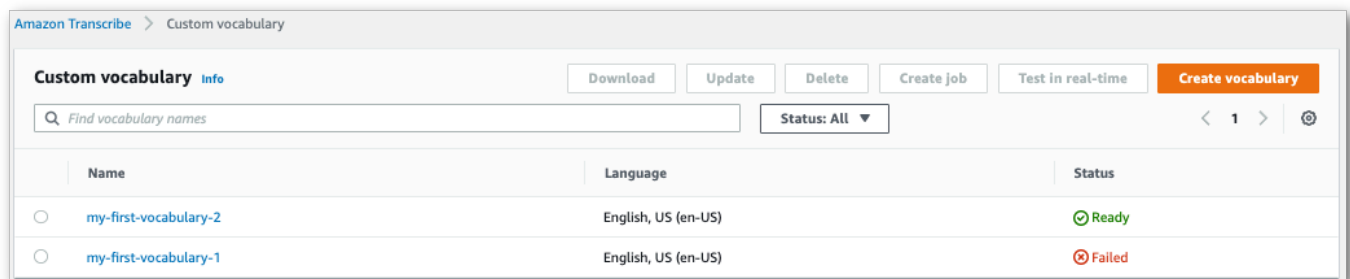
Add row

Cochez la case (✓) pour enregistrer vos modifications ou cliquez sur le « X » pour les ignorer.

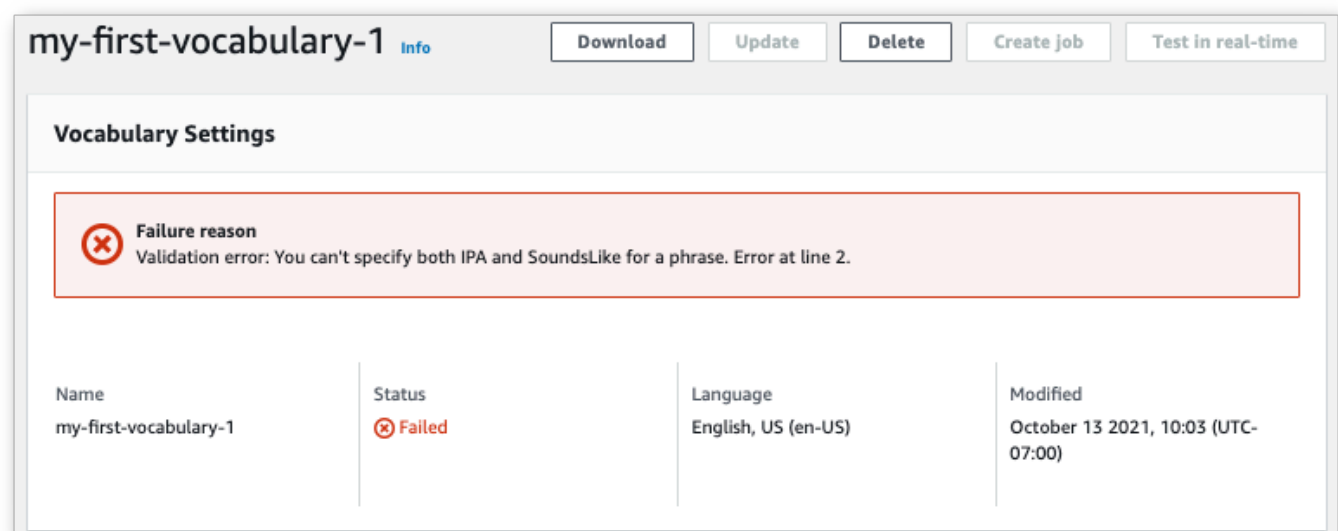
5. Vous pouvez également ajouter des balises à votre vocabulaire personnalisé. Une fois que vous avez rempli tous les champs et que vous êtes satisfait de votre vocabulaire, sélectionnez Créer du vocabulaire en bas de la page. Vous revenez alors à la page Vocabulaire personnalisé où vous pouvez voir le statut de votre vocabulaire personnalisé. Lorsque le statut passe de « En attente » à « Prêt », votre vocabulaire personnalisé peut être utilisé avec une transcription.



6. Si le statut passe à « Échec », sélectionnez le nom de votre vocabulaire personnalisé pour accéder à sa page d'informations.



En haut de cette page, une bannière Raison de l'échec fournit des informations sur la raison pour laquelle votre vocabulaire personnalisé a échoué. Corrigez l'erreur dans votre fichier texte, puis réessayez.



AWS CLI

Cet exemple utilise la commande [create-vocabulary](#) avec un fichier de vocabulaire personnalisé sous forme de tableau. Pour plus d'informations, consultez [CreateVocabulary](#).

Pour utiliser un vocabulaire personnalisé existant dans une tâche de transcription, VocabularyName définissez-le dans le [Settings](#) champ lorsque vous appelez l'[StartTranscriptionJob](#) opération ou, dans le AWS Management Console, choisissez le vocabulaire personnalisé dans la liste déroulante.

```
aws transcribe create-vocabulary \  
--vocabulary-name my-first-vocabulary \  
--vocabulary-file-uri s3://DOC-EXAMPLE-BUCKET/my-vocabularies/my-vocabulary-file.txt \  
--language-code en-US
```

Voici un autre exemple d'utilisation de la commande [create-vocabulary](#) et d'un corps de requête qui crée votre vocabulaire personnalisé.

```
aws transcribe create-vocabulary \  
--cli-input-json file://filepath/my-first-vocab-table.json
```

Le fichier my-first-vocab-table.json contient le corps de requête suivant.

```
{  
  "VocabularyName": "my-first-vocabulary",  
  "VocabularyFileUri": "s3://DOC-EXAMPLE-BUCKET/my-vocabularies/my-vocabulary-  
table.txt",  
  "LanguageCode": "en-US"  
}
```

Lorsque VocabularyState passe de PENDING à READY, votre vocabulaire personnalisé est prêt à être utilisé avec une transcription. Pour afficher le statut actuel de votre vocabulaire personnalisé, exécutez :

```
aws transcribe get-vocabulary \  
--vocabulary-name my-first-vocabulary
```


AWS SDK for Python (Boto3)

Cet exemple utilise le AWS SDK for Python (Boto3) pour créer un vocabulaire personnalisé à partir d'un tableau à l'aide de la méthode [create_vocabulary](#). Pour plus d'informations, consultez [CreateVocabulary](#).

Pour utiliser un vocabulaire personnalisé existant dans une tâche de transcription, VocabularyName définissez-le dans le [Settings](#) champ lorsque vous appelez l'[StartTranscriptionJob](#) opération ou, dans le AWS Management Console, choisissez le vocabulaire personnalisé dans la liste déroulante.

Pour d'autres exemples d'utilisation AWS des SDK, notamment des exemples spécifiques aux fonctionnalités, des scénarios et des exemples multiservices, reportez-vous au chapitre. [Exemples de code pour Amazon Transcribe à l'aide de kits de développement logiciel AWS](#)

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
vocab_name = "my-first-vocabulary"
response = transcribe.create_vocabulary(
    LanguageCode = 'en-US',
    VocabularyName = vocab_name,
    VocabularyFileUri = 's3://DOC-EXAMPLE-BUCKET/my-vocabularies/my-vocabulary-
table.txt'
)

while True:
    status = transcribe.get_vocabulary(VocabularyName = vocab_name)
    if status['VocabularyState'] in ['READY', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Note

Si vous créez un nouveau Amazon S3 compartiment pour vos fichiers de vocabulaire personnalisés, assurez-vous que le IAM rôle à l'origine de la [CreateVocabulary](#) demande est autorisé à accéder à ce compartiment. Si le rôle ne dispose pas des autorisations appropriées, votre demande échoue. Vous pouvez éventuellement spécifier un IAM rôle dans

vosre demande en incluant le `DataAccessRoleArn` paramètre. Pour plus d'informations sur IAM les rôles et les politiques dans Amazon Transcribe, voir [Exemples de politiques basées sur l'identité Amazon Transcribe](#).

Création d'un vocabulaire personnalisé à l'aide d'une liste

⚠ Important

Les vocabulaires personnalisés sous forme de liste sont désormais déconseillés. Par conséquent, si vous créez un nouveau vocabulaire personnalisé, nous vous recommandons vivement d'utiliser le format de tableau.

Vous pouvez créer des vocabulaires personnalisés à partir de listes à l'aide des kits de AWS développement logiciel (SDK) AWS Management Console AWS CLI, ou.

- AWS Management Console: vous devez créer et charger un fichier texte contenant votre vocabulaire personnalisé. Vous pouvez utiliser des entrées séparées par des lignes ou des virgules. Notez que votre liste doit être enregistrée sous forme de fichier texte (*.txt) au LF format. Si vous utilisez un autre format, tel que CRLF votre vocabulaire personnalisé n'est pas accepté par Amazon Transcribe.
- AWS CLI et AWSSDK : vous devez inclure votre vocabulaire personnalisé sous forme d'entrées séparées par des virgules dans votre appel d'API à l'aide de l'indicateur. [Phrases](#)

Si une entrée contient plusieurs mots, vous devez mettre un trait d'union pour chaque mot. Par exemple, vous incluez « Los Angeles » en tant que **Los-Angeles** et « Andorre-la-Vieille » en tant que **Andorra-la-Vella**

Voici des exemples des deux formats de liste valides. Reportez-vous à [Création de listes de vocabulaire personnalisées](#) pour des exemples spécifiques à la méthode.

- Entrées séparées par des virgules :

```
Los-Angeles,CLI,Eva-Maria,ABCs,Andorra-la-Vella
```

- Entrées séparées par des lignes :

```
Los-Angeles
CLI
Eva-Maria
ABCs
Andorra-la-Vella
```

⚠ Important

Vous ne pouvez utiliser que des caractères compatibles avec votre langue. Reportez-vous au jeu de [caractères](#) de votre langue pour plus de détails.

Les listes de vocabulaire personnalisées ne sont pas prises en charge avec [CreateMedicalVocabulary](#) cette opération. Si vous créez un vocabulaire médical personnalisé, vous devez utiliser un format de tableau ; reportez-vous [Création d'un vocabulaire personnalisé à l'aide d'un tableau](#) aux instructions.

Création de listes de vocabulaire personnalisées

Pour traiter une liste de vocabulaire personnalisée à utiliser avec Amazon Transcribe, consultez les exemples suivants :

AWS CLI

Cet exemple utilise la commande [create-vocabulary](#) avec un fichier de vocabulaire personnalisé au format liste. Pour plus d'informations, veuillez consulter [CreateVocabulary](#).

```
aws transcribe create-vocabulary \  
--vocabulary-name my-first-vocabulary \  
--language-code en-US \  
--phrases {CLI,Eva-Maria,ABCs}
```

Voici un autre exemple utilisant la commande [create-vocabulary](#) et un corps de requête qui crée votre vocabulaire personnalisé.

```
aws transcribe create-vocabulary \  
--cli-input-json file://filepath/my-first-vocab-list.json
```

Le fichier `my-first-vocab-list.json` contient le corps de requête suivant.

```
{
  "VocabularyName": "my-first-vocabulary",
  "LanguageCode": "en-US",
  "Phrases": [
    "CLI", "Eva-Maria", "ABCs"
  ]
}
```

Une `VocabularyState` fois que vous `PENDING` passez de `àREADY`, votre vocabulaire personnalisé est prêt à être utilisé avec une transcription. Pour afficher l'état actuel de votre vocabulaire personnalisé, exécutez :

```
aws transcribe get-vocabulary \
--vocabulary-name my-first-vocabulary
```

AWS SDK for Python (Boto3)

Cet exemple utilise le AWS SDK for Python (Boto3) pour créer un vocabulaire personnalisé à partir d'une liste à l'aide de la méthode [create_vocabulary](#). Pour plus d'informations, veuillez consulter [CreateVocabulary](#).

Pour des exemples supplémentaires utilisant les AWS kits SDK, notamment des exemples spécifiques aux fonctionnalités, des scénarios et des exemples interservices, consultez le chapitre. [Exemples de code pour Amazon Transcribe à l'aide de kits de développement logiciel AWS](#)

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
vocab_name = "my-first-vocabulary"
response = transcribe.create_vocabulary(
    LanguageCode = 'en-US',
    VocabularyName = vocab_name,
    Phrases = [
        'CLI', 'Eva-Maria', 'ABCs'
    ]
)

while True:
    status = transcribe.get_vocabulary(VocabularyName = vocab_name)
    if status['VocabularyState'] in ['READY', 'FAILED']:
```

```
    break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Note

Si vous créez un nouveau Amazon S3 bucket pour vos fichiers de vocabulaire personnalisés, assurez-vous que le IAM rôle à l'origine de la [CreateVocabulary](#) demande est autorisé à accéder à ce bucket. Si le rôle ne dispose pas des autorisations appropriées, votre demande échoue. Vous pouvez éventuellement spécifier un IAM rôle dans votre demande en incluant le `DataAccessRoleArn` paramètre. Pour plus d'informations sur IAM les rôles et les politiques dans Amazon Transcribe, consultez [Exemples de politiques basées sur l'identité Amazon Transcribe](#).

Utilisation d'un vocabulaire personnalisé

Une fois votre vocabulaire personnalisé créé, vous pouvez l'inclure dans vos demandes de transcription. Reportez-vous aux sections suivantes pour obtenir des exemples.

La langue du vocabulaire personnalisé que vous incluez dans votre demande doit correspondre au code de langue que vous avez spécifié pour votre média. Si les langues ne correspondent pas, votre vocabulaire personnalisé n'est pas appliqué à votre transcription et il n'y a aucun avertissement ni erreur.

Utilisation d'un vocabulaire personnalisé dans une transcription par lots

Pour utiliser un vocabulaire personnalisé avec une transcription par lots, consultez les exemples suivants :

AWS Management Console

1. Connectez-vous à [AWS Management Console](#).
2. Dans le volet de navigation, choisissez Tâches de transcription, puis sélectionnez Créer une tâche (en haut à droite). La page Spécifier les détails de la tâche s'ouvre.

Specify job details [Info](#)

Job settings

Name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, . (period), _ (underscore), and - (hyphen).

Model type [Info](#)

Choose the type of model to use for the transcription job.

General model

To use a model that is not specialized for a particular use case, choose this option. Configuration options vary between languages.

Custom language model

To use a model that you trained for your specific use case, choose this option. This model has fewer configuration options than the general model.

Language settings

You can transcribe your audio file in a language that you specify or have Amazon Transcribe identify and transcribe it in the predominant language.

Specific language [Info](#)

If you know the language spoken in your source audio, choose this option to get the most accurate results. The options available for additional processing vary between languages.

Automatic language identification [Info](#)

If you don't know the language spoken in your audio files, choose this option. You have access to fewer options for additional processing than if you choose **Specific language**.

Language

Choose the language of the input audio.

▶ **Additional settings**

Nommez votre tâche et spécifiez votre support d'entrée. Incluez éventuellement d'autres champs, puis choisissez Suivant.

3. Au bas de la page Configurer la tâche, dans le panneau Personnalisation, activez Vocabulaire personnalisé.

Configure job - *optional* [Info](#)

Audio settings

Audio identification [Info](#)
Choose to split multi-channel audio into separate channels for transcription, or identify speakers in the input audio.

Alternative results [Info](#)
Enable to view more transcription results

Content removal

Content removal conceals information in the resulting transcript from your source audio file. Amazon Transcribe changes items in the transcript and does not modify the source audio.

PII redaction [Info](#)
Label the type of PII and also mask the content with the PII entity type in the transcription output. For example, (123) 456-7890 will be masked as [PHONE].

Vocabulary filtering [Info](#)
Vocabulary filtering can remove, mask or tag specified words in the final transcript.

Customization

Custom vocabulary [Info](#)
A custom vocabulary improves the accuracy of recognizing words and phrases specific to your use case.

Vocabulary selection
The vocabularies shown here are based on your language settings. You can choose up to one vocabulary per language. You can also [create a new vocabulary](#). [↗](#)

Choose a vocabulary ▼

Cancel Previous **Create job**

4. Sélectionnez votre vocabulaire personnalisé dans le menu déroulant.

Sélectionnez Créer une tâche pour exécuter votre tâche de transcription.

AWS CLI

Cet exemple utilise la [start-transcription-job](#) commande et le `Settings` paramètre avec le `VocabularyName` sous-paramètre. Pour plus d'informations, consultez [StartTranscriptionJob](#) et [Settings](#).

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--transcription-job-name my-first-transcription-job \  
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
--output-bucket-name DOC-EXAMPLE-BUCKET \  
--output-key my-output-files/ \  
--language-code en-US \  
--settings VocabularyName=my-first-vocabulary
```

Voici un autre exemple d'utilisation de la [start-transcription-job](#) commande et d'un corps de requête qui inclut votre vocabulaire personnalisé associé à cette tâche.

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--cli-input-json file://my-first-vocabulary-job.json
```

Le fichier `my-first-vocabulary-job.json` contient le corps de requête suivant.

```
{  
  "TranscriptionJobName": "my-first-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
  },  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "OutputKey": "my-output-files/",  
  "LanguageCode": "en-US",  
  "Settings": {  
    "VocabularyName": "my-first-vocabulary"  
  }  
}
```

AWS SDK for Python (Boto3)

Cet exemple utilise le AWS SDK for Python (Boto3) pour inclure un vocabulaire personnalisé à l'aide de l'`Settings` argument de la méthode [start_transcription_job](#). Pour plus d'informations, consultez [StartTranscriptionJob](#) et [Settings](#).

Pour des exemples supplémentaires d'utilisation AWS des SDK, y compris des exemples spécifiques à des fonctionnalités, des scénarios et des exemples interservices, consultez le [Exemples de code pour Amazon Transcribe à l'aide de kits de développement logiciel AWS](#) chapitre.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
transcribe.start_transcription_job(
    TranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'my-output-files/',
    LanguageCode = 'en-US',
    Settings = {
        'VocabularyName': 'my-first-vocabulary'
    }
)

while True:
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)
    if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Utiliser un vocabulaire personnalisé dans une transcription en streaming

Pour utiliser un vocabulaire personnalisé avec une transcription en continu, consultez les exemples suivants :

AWS Management Console

1. Connectez-vous au [AWS Management Console](#).
2. Dans le panneau de navigation, choisissez Transcription en temps réel. Faites défiler la page jusqu'à Personnalisations et agrandissez ce champ s'il est réduit.

Real-time transcription [Info](#)

See how Amazon Transcribe creates a text copy of speech in real time. Choose **Start streaming** and talk.

Transcription [Download full transcript](#) [Start streaming](#)

Transcription output Current language: English, US

Choose **Start streaming** to begin a real-time transcription of what you speak into your microphone

00:00 of 15:00 min audio stream

- ▶ Language settings
- ▶ Audio settings
- ▶ Content removal settings
- ▶ Customizations

3. Activez Vocabulaire personnalisé et sélectionnez un vocabulaire personnalisé dans le menu déroulant.

▼ Customizations

Custom vocabulary [Info](#)
A custom vocabulary improves the accuracy of recognizing words and phrases specific to your use case.

Vocabulary selection
The vocabularies shown here are based on your language settings. You can choose up to one vocabulary per language. You can also [create a new vocabulary](#).

Choose a vocabulary ▼

Incluez les autres paramètres que vous souhaitez appliquer à votre stream.

4. Vous êtes maintenant prêt à transcrire votre stream. Sélectionnez Démarrer la diffusion et commencez à parler. Pour terminer votre dictée, sélectionnez Arrêter la diffusion.

Streaming HTTP/2

Cet exemple crée une requête HTTP/2 qui inclut votre vocabulaire personnalisé. Pour plus d'informations sur l'utilisation du streaming HTTP/2 avec Amazon Transcribe, consultez [Configuration d'un flux HTTP/2](#). Pour plus de détails sur les paramètres et les en-têtes spécifiques à Amazon Transcribe, consultez [StartStreamTranscription](#).

```
POST /stream-transcription HTTP/2
host: transcribestreaming.us-west-2.amazonaws.com
X-Amz-Target: com.amazonaws.transcribe.Transcribe.StartStreamTranscription
Content-Type: application/vnd.amazon.eventstream
X-Amz-Content-Sha256: string
X-Amz-Date: 20220208T235959Z
Authorization: AWS4-HMAC-SHA256 Credential=access-key/20220208/us-west-2/transcribe/
aws4_request, SignedHeaders=content-type;host;x-amz-content-sha256;x-amz-date;x-amz-
target;x-amz-security-token, Signature=string
x-amzn-transcribe-language-code: en-US
x-amzn-transcribe-media-encoding: flac
x-amzn-transcribe-sample-rate: 16000
x-amzn-transcribe-vocabulary-name: my-first-vocabulary
transfer-encoding: chunked
```

Les définitions de paramètres se trouvent dans la [référence de l'API](#) ; les paramètres communs à toutes les opérations d'AWSAPI sont répertoriés dans la section [Paramètres communs](#).

WebSocket ruisseau

Cet exemple crée une URL présignée qui applique votre vocabulaire personnalisé à un WebSocket flux. Les sauts de ligne ont été ajoutés pour faciliter la lecture. Pour plus d'informations sur l'utilisation WebSocket des flux avec Amazon Transcribe, consultez [Configuration d'un WebSocket stream](#). Pour plus de détails sur les paramètres, reportez-vous à la section [StartStreamTranscription](#).

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/stream-transcription-
websocket?
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-
west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=string
&X-Amz-SignedHeaders=content-type%3Bhost%3Bx-amz-date
```

```
&language-code=en-US  
&media-encoding=flac  
&sample-rate=16000  
&vocabulary-name=my-first-vocabulary
```

Les définitions de paramètres se trouvent dans la [référence de l'API](#) ; les paramètres communs à toutes les opérations d'AWSAPI sont répertoriés dans la section [Paramètres communs](#).

Modèles de langage personnalisés

Les modèles linguistiques personnalisés sont conçus pour améliorer la précision de la transcription des discours spécifiques à un domaine. Cela inclut tout contenu en dehors de ce que vous entendriez dans les conversations normales de tous les jours. Par exemple, si vous transcrivez les actes d'une conférence scientifique, il est peu probable qu'une transcription standard reconnaisse la plupart des termes scientifiques utilisés par les présentateurs. Dans ce cas, vous pouvez créer un modèle de langage personnalisé pour reconnaître les termes spécialisés utilisés dans votre discipline.

Contrairement aux vocabulaires personnalisés, qui améliorent la reconnaissance d'un mot en fournissant des indices (tels que des prononciations), les modèles de langage personnalisés apprennent le contexte associé à un mot donné. Cela inclut comment et quand un mot est utilisé, ainsi que la relation qu'un mot entretient avec d'autres mots. Par exemple, si vous entraînez votre modèle à l'aide d'articles de recherche sur la climatologie, il se peut que votre modèle apprenne que « banquise » est une paire de mots plus probable que « flux de glace ».

Pour consulter les langues prises en charge pour les modèles de langage personnalisés, reportez-vous à [Langues prises en charge et fonctionnalités spécifiques aux langues](#). Notez que si vous incluez un modèle de langage personnalisé dans votre demande, vous ne pouvez pas activer l'identification de la langue (vous devez spécifier un code de langue).

 Opérations d'API spécifiques aux modèles de langage personnalisés

[CreateLanguageModel](#), [DeleteLanguageModel](#), [DescribeLanguageModel](#),
[ListLanguageModels](#)

Sources de données

Vous pouvez utiliser n'importe quel type de données texte pour entraîner votre modèle. Toutefois, plus le contenu de votre texte est proche de votre contenu audio, plus votre modèle est précis. Il

est donc important de choisir des données textuelles qui utilisent les mêmes termes dans le même contexte que votre fichier audio.

Les meilleures données pour l'entraînement d'un modèle sont des transcriptions précises. Ces données sont considérées comme des données internes au domaine. Les données textuelles du domaine ont exactement les mêmes termes, la même utilisation et le même contexte que l'audio que vous souhaitez transcrire.

Si vous ne disposez pas de transcriptions précises, utilisez des articles de journaux, des rapports techniques, des livres blancs, des actes de conférence, des manuels d'instructions, des articles de presse, du contenu de sites Web et tout autre texte contenant les termes souhaités utilisés dans un contexte similaire à celui de votre audio. Ces données sont considérées comme liées au domaine.

La création d'un modèle de langage personnalisé robuste peut nécessiter une quantité importante de données textuelles, qui doivent contenir les termes prononcés dans votre fichier audio. Vous pouvez fournir jusqu'à 2 Go de données texte pour entraîner votre modèle. C'est ce que l'on appelle les données d'entraînement. Si vous n'avez pas (ou peu) de transcriptions dans le domaine, vous pouvez éventuellement fournir jusqu'à 200 Mo de données texte pour ajuster votre modèle. C'est ce que l'on appelle les données de réglage.

Données d'entraînement ou de réglage

Les données d'apprentissage ont pour but d'apprendre à Amazon Transcribe à reconnaître de nouveaux termes et à connaître le contexte dans lequel ces termes sont utilisés. La création d'un modèle robuste Amazon Transcribe peut nécessiter un volume important de données textuelles pertinentes. Il est fortement recommandé de fournir autant de données d'entraînement que possible, jusqu'à la limite de 2 Go.

L'objectif du réglage des données est d'aider à affiner et à optimiser les relations contextuelles apprises à partir de vos données d'entraînement. Les données de réglage ne sont pas requises pour la création d'un modèle de langue personnalisé.

C'est à vous de décider de la meilleure façon de sélectionner l'entraînement et, éventuellement, de régler les données. Chaque cas est unique et dépend du type et de la quantité de données dont vous disposez. Les données de réglage sont recommandées lorsque vous ne disposez pas de données d'entraînement dans le domaine.

Si vous choisissez d'inclure les deux types de données, ne superposez pas vos données d'entraînement et de réglage ; les données d'entraînement et de réglage doivent être uniques. Les

données qui se chevauchent peuvent biaiser et fausser votre modèle de langage personnalisé, ce qui a une incidence sur sa précision.

D'une manière générale, nous vous recommandons d'utiliser un texte précis et propre au domaine comme données d'entraînement dans la mesure du possible. Voici quelques scénarios généraux, classés par ordre de préférence :

- Si vous disposez de plus de 10 000 mots de texte de transcription précis et intégré au domaine, utilisez-le comme données d'apprentissage. Dans ce cas, il n'est pas nécessaire d'inclure des données de réglage. C'est le modèle idéal pour la formation d'un modèle de langage personnalisé.
- Si votre texte de transcription intégré au domaine contient moins de 10 000 mots et que vous n'obtenez pas les résultats souhaités, pensez à compléter vos données d'entraînement avec des textes écrits liés au domaine, tels que des rapports techniques. Dans ce cas, réservez une petite partie (10 à 25 %) de vos données de transcription internes au domaine à utiliser comme données de réglage.
- Si vous n'avez pas de texte de transcription intégré au domaine, téléchargez tout le texte lié à votre domaine sous forme de données d'apprentissage. Dans ce cas, le texte de type transcription est préférable au texte écrit. Il s'agit du scénario le moins efficace pour former un modèle de langage personnalisé.

Lorsque vous êtes prêt à créer votre modèle, reportez-vous à [Création d'un modèle de langage personnalisé](#).

Création d'un modèle de langage personnalisé

Avant de créer votre modèle de langage personnalisé, vous devez :

- Préparation de vos données. Les données doivent être enregistrées au format texte brut et ne peuvent contenir aucun caractère spécial.
- Chargez vos données dans un Amazon S3 bucket. Il est recommandé de créer des dossiers distincts pour les données d'entraînement et de réglage.
- Assurez-vous d'Amazon Transcribe avoir accès à votre Amazon S3 bucket. Vous devez spécifier un IAM rôle doté d'autorisations d'accès pour utiliser vos données.

Préparation de vos données

Vous pouvez compiler toutes vos données dans un seul fichier ou les enregistrer sous forme de fichiers multiples. Notez que si vous choisissez d'inclure des données de réglage, celles-ci doivent être enregistrées dans un fichier distinct de vos données d'entraînement.

Peu importe le nombre de fichiers texte que vous utilisez pour vos données d'entraînement ou de réglage. Le téléchargement d'un fichier de 100 000 mots produit le même résultat que le téléchargement de 10 fichiers de 10 000 mots. Préparez vos données de texte de la manière qui vous convient le mieux.

Assurez-vous que tous vos fichiers de données répondent aux critères suivants :

- Ils sont tous rédigés dans la même langue que le modèle que vous souhaitez créer. Par exemple, si vous souhaitez créer un modèle de langue personnalisé qui transcrit le son en anglais américain (en-US), toutes vos données de texte doivent être en anglais américain.
- Ils sont au format texte brut avec un encodage UTF-8.
- Ils ne contiennent aucun caractère spécial ni aucune mise en forme, tels que des balises HTML.
- Leur taille totale combinée maximale est de 2 Go pour les données d'entraînement et de 200 Mo pour les données de réglage.

Si l'un de ces critères n'est pas satisfait, votre modèle échoue.

Chargement de vos données

Avant de charger vos données, créez un nouveau dossier pour vos données d'entraînement. Si vous utilisez des données de réglage, créez un autre dossier distinct.

Les URI de vos compartiments peuvent ressembler à :

- `s3://DOC-EXAMPLE-BUCKET/my-model-training-data/`
- `s3://DOC-EXAMPLE-BUCKET/my-model-tuning-data/`

Chargez vos données d'entraînement et de réglage dans les compartiments appropriés.

Vous pourrez ajouter d'autres données à ces compartiments ultérieurement. Toutefois, si vous le faites, vous devez recréer votre modèle avec les nouvelles données. Les modèles existants ne peuvent pas être mis à jour avec de nouvelles données.

Autoriser l'accès à vos données

Pour créer un modèle de langage personnalisé, vous devez spécifier un IAM rôle autorisé à accéder à votre Amazon S3 bucket. Si vous ne possédez pas encore de rôle ayant accès au Amazon S3 bucket dans lequel vous avez placé vos données d'entraînement, vous devez en créer un. Une fois que vous avez créé un rôle, vous pouvez attacher une politique pour attacher des autorisations à ce rôle. N'attachez pas de politique à un utilisateur.

Pour obtenir des exemples de politiques, consultez [Exemples de politiques basées sur l'identité Amazon Transcribe](#).

Pour savoir comment créer une nouvelle IAM identité, voir [IAM identités \(utilisateurs, groupes d'utilisateurs et rôles\)](#).

Pour en savoir plus sur les stratégies, consultez :

- [Politiques et autorisations dans IAM](#)
- [Création de IAM politiques](#)
- [Gestion de l'accès pour les ressources AWS](#)

Création de votre modèle de langue personnalisé

Lorsque vous créez votre modèle de langue personnalisé, vous devez choisir un modèle de base. Il existe deux options de modèle de base :

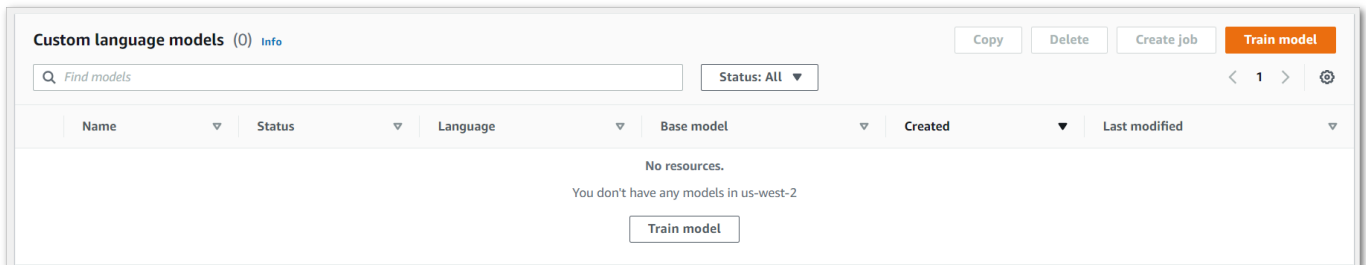
- **NarrowBand**: utilisez cette option pour les fichiers audio dont la fréquence d'échantillonnage est inférieure à 16 000 Hz. Ce type de modèle est généralement utilisé pour les conversations téléphoniques enregistrées à 8 000 Hz.
- **WideBand**: utilisez cette option pour le son dont la fréquence d'échantillonnage est supérieure ou égale à 16 000 Hz.

Vous pouvez créer des modèles de langage personnalisés à l'AWS Management Console aide des kits de AWS développement logiciel (SDK) ou des kits de développement logiciel (SDK). Consultez les exemples suivants : AWS CLI

AWS Management Console

1. Connectez-vous à [AWS Management Console](#).

2. Dans le volet de navigation, choisissez un modèle de langage personnalisé. Cela ouvre la page Modèles de langage personnalisés où vous pouvez consulter les modèles de langage personnalisés existants ou former un nouveau modèle de langage personnalisé.
3. Pour entraîner un nouveau modèle, sélectionnez Train model.



Vous accédez alors à la page du modèle de train. Ajoutez un nom, spécifiez la langue et choisissez le modèle de base que vous souhaitez pour votre modèle. Ajoutez ensuite le parcours à votre entraînement et, éventuellement, vos données de réglage. Vous devez inclure un IAM rôle autorisé à accéder à vos données.

Train model [Info](#)

Model settings

Name

The name can be up to 200 characters long. Valid characters:A-Z,a-z,0-9,and _ - (hyphen).

Language

Choose the language of your model.

 ▼

Base model [Info](#)

Choose the base model that you want to use to create your custom language model. Choose the model based on the sample rate of your source audio.

Narrow band

For audio that has a sample rate less than 16 KHz. Typically, this is 8 KHz audio from telephone conversations.

Wide band

For audio that has a sample rate of 16 KHz or greater. Typically, this is 16 KHz audio from media sources.

Training data [Info](#)

Training data location on S3

Type or paste the S3 prefix for the text files that you want to use as training data, or browse to find the files that have matching S3 prefixes.

The file format must be plain text in the language that you have selected for the model. The maximum file size is 2 GB.

Tuning data - optional [Info](#)

Tuning data location on S3

Type or paste the S3 prefix for the text files that you want to use as tuning data, or browse to find the files that have matching S3 prefixes.

The file format must be plain text in the language that you have selected for the model. The maximum file size is 200 MB.

Access permissions

IAM role [Info](#)

Use an existing IAM role

Create an IAM role

By choosing **Train model** you are authorizing creation of this role.

Role name

A role that grants access to the S3 input locations.

 ▼

4. Une fois que vous avez rempli tous les champs, sélectionnez **Modèle de train** au bas de la page.

AWS CLI

Cet exemple utilise la [create-language-model](#) commande. Pour plus d'informations, consultez [CreateLanguageModel](#) et [LanguageModel](#).

```
aws transcribe create-language-model \  
--base-model-name NarrowBand \  
--model-name my-first-language-model \  
--input-data-config S3Uri=s3://DOC-EXAMPLE-BUCKET/my-clm-training-  
data/,TuningDataS3Uri=s3://DOC-EXAMPLE-BUCKET/my-clm-tuning-  
data/,DataAccessRoleArn=arn:aws:iam::111122223333:role/ExampleRole \  
--language-code en-US
```

Voici un autre exemple utilisant la [create-language-model](#) commande et un corps de requête qui crée votre modèle de langage personnalisé.

```
aws transcribe create-language-model \  
--cli-input-json file://filepath/my-first-language-model.json
```

Le fichier `my-first-language-model.json` contient le corps de requête suivant.

```
{  
  "BaseModelName": "NarrowBand",  
  "ModelName": "my-first-language-model",  
  "InputDataConfig": {  
    "S3Uri": "s3://DOC-EXAMPLE-BUCKET/my-clm-training-data",  
    "TuningDataS3Uri": "s3://DOC-EXAMPLE-BUCKET/my-clm-tuning-data",  
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/ExampleRole"  
  },  
  "LanguageCode": "en-US"  
}
```

AWS SDK for Python (Boto3)

Cet exemple utilise le AWS SDK for Python (Boto3) pour créer une CLM à l'aide de la méthode [create_language_model](#). Pour plus d'informations, consultez [CreateLanguageModel](#) et [LanguageModel](#).

Pour des exemples supplémentaires utilisant les AWS kits SDK, notamment des exemples spécifiques aux fonctionnalités, des scénarios et des exemples interservices, consultez le chapitre. [Exemples de code pour Amazon Transcribe à l'aide de kits de développement logiciel AWS](#)

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
model_name = 'my-first-language-model',
transcribe.create_language_model(
    LanguageCode = 'en-US',
    BaseModelName = 'NarrowBand',
    ModelName = model_name,
    InputDataConfig = {
        'S3Uri': 's3://DOC-EXAMPLE-BUCKET/my-clm-training-data/',
        'TuningDataS3Uri': 's3://DOC-EXAMPLE-BUCKET/my-clm-tuning-data/',
        'DataAccessRoleArn': 'arn:aws:iam::111122223333:role/ExampleRole'
    }
)

while True:
    status = transcribe.get_language_model(ModelName = model_name)
    if status['LanguageModel']['ModelStatus'] in ['COMPLETED', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Mettre à jour votre modèle de langue personnalisé

Amazon Transcribe met continuellement à jour les modèles de base disponibles pour les modèles linguistiques personnalisés. Pour bénéficier de ces mises à jour, nous vous recommandons de former de nouveaux modèles linguistiques personnalisés tous les 6 à 12 mois.

Pour savoir si votre modèle de langage personnalisé utilise le dernier modèle de base, exécutez une [DescribeLanguageModel](#) requête à l'aide du SDK AWS CLI ou d'un AWS SDK, puis recherchez le `UpgradeAvailability` champ dans votre réponse.

Si `UpgradeAvailability` tel est `true` le cas, votre modèle n'exécute pas la dernière version du modèle de base. Pour utiliser le dernier modèle de base dans un modèle de langue personnalisé, vous devez créer un nouveau modèle de langue personnalisé. Les modèles linguistiques personnalisés ne peuvent pas être mis à niveau.

Utilisation d'un modèle de langage personnalisé

Une fois que vous avez créé votre modèle linguistique personnalisé, vous pouvez l'inclure dans vos demandes de transcription. Reportez-vous aux sections suivantes pour obtenir des exemples.

La langue du modèle que vous incluez dans votre demande doit correspondre au code de langue que vous avez spécifié pour votre média. Si les langues ne correspondent pas, votre modèle linguistique personnalisé n'est pas appliqué à votre transcription et il n'y a aucun avertissement ni erreur.

Utilisation d'un modèle linguistique personnalisé dans une transcription par lots

Pour utiliser un modèle linguistique personnalisé avec une transcription par lots, consultez les exemples suivants :

AWS Management Console

1. Connectez-vous à [AWS Management Console](#).
2. Dans le volet de navigation, choisissez Tâches de transcription, puis sélectionnez Créer une tâche (en haut à droite). La page Spécifier les détails de la tâche s'ouvre.
3. Dans le panneau Paramètres de la Job, sous Type de modèle, sélectionnez la case Modèle de langue personnalisé.

Job settings

Name

MyTranscriptionJob

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, . (period), _ (underscore), and - (hyphen).

Model type [Info](#)

Choose the type of model to use for the transcription job.

General model
To use a model that is not specialized for a particular use case, choose this option. Configuration options vary between languages.

Custom language model
To use a model that you trained for your specific use case, choose this option. This model has fewer configuration options than the general model.

Language

Choose the language of the input audio.

English, US (en-US) ▼

Custom model selection

Choose an existing model or [create a new one.](#) [↗](#)

Choose model ▼

► **Additional settings**

Vous devez également sélectionner une langue de saisie dans le menu déroulant.

Job settings

Name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, . (period), _ (underscore), and - (hyphen).

Model type [Info](#)

Choose the type of model to use for the transcription job.

General model

To use a model that is not specialized for a particular use case, choose this option. Configuration options vary between languages.

Custom language model

To use a model that you trained for your specific use case, choose this option. This model has fewer configuration options than the general model.

Language

Choose the language of the input audio.

English, US (en-US) ▲

English, US (en-US)

English, AU (en-AU)

English, UK (en-GB)

Hindi, IN (hi-IN)

Spanish, US (es-US)

4. Sous Sélection de modèles personnalisés, sélectionnez un modèle linguistique personnalisé existant dans le menu déroulant ou créez-en un nouveau.

Ajoutez l'Amazon S3 emplacement de votre fichier d'entrée dans le panneau Données d'entrée.

5. Sélectionnez Suivant pour accéder à des options de configuration supplémentaires.

Sélectionnez Créer une tâche pour exécuter votre tâche de transcription.

AWS CLI

Cet exemple utilise la [start-transcription-job](#) commande et le `ModelSettings` paramètre avec le `VocabularyName` sous-paramètre. Pour plus d'informations, consultez [StartTranscriptionJob](#) et [ModelSettings](#).

```
aws transcribe start-transcription-job \
```

```
--region us-west-2 \  
--transcription-job-name my-first-transcription-job \  
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
--output-bucket-name DOC-EXAMPLE-BUCKET \  
--output-key my-output-files/ \  
--language-code en-US \  
--model-settings LanguageModelName=my-first-language-model
```

Voici un autre exemple d'utilisation de la [start-transcription-job](#) commande et d'un corps de requête qui inclut votre modèle de langage personnalisé avec cette tâche.

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--cli-input-json file://my-first-model-job.json
```

Le fichier `my-first-model-job.json` contient le corps de requête suivant.

```
{  
  "TranscriptionJobName": "my-first-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
  },  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "OutputKey": "my-output-files/",  
  "LanguageCode": "en-US",  
  "ModelSettings": {  
    "LanguageModelName": "my-first-language-model"  
  }  
}
```

AWS SDK for Python (Boto3)

Cet exemple utilise le AWS SDK for Python (Boto3) pour inclure un modèle de langage personnalisé à l'aide de l'`ModelSettings` argument de la méthode [start_transcription_job](#). Pour plus d'informations, consultez [StartTranscriptionJob](#) et [ModelSettings](#).

Pour des exemples supplémentaires d'utilisation AWS des SDK, y compris des exemples spécifiques à des fonctionnalités, des scénarios et des exemples interservices, consultez le [Exemples de code pour Amazon Transcribe à l'aide de kits de développement logiciel AWS](#) chapitre.

```
from __future__ import print_function
```



```
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
transcribe.start_transcription_job(
    TranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'my-output-files/',
    LanguageCode = 'en-US',
    ModelSettings = {
        'LanguageModelName': 'my-first-language-model'
    }
)

while True:
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)
    if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Utilisation d'un modèle linguistique personnalisé dans une transcription en continu


Pour utiliser un modèle linguistique personnalisé avec une transcription en continu, consultez les exemples suivants :

AWS Management Console

1. Connectez-vous au [AWS Management Console](#).
2. Dans le panneau de navigation, choisissez Transcription en temps réel. Faites défiler la page jusqu'à Personnalisations et agrandissez ce champ s'il est réduit.

Real-time transcription [Info](#)

See how Amazon Transcribe creates a text copy of speech in real time. Choose **Start streaming** and talk.

Transcription Download full transcript  **Start streaming**

Transcription output Current language: English, US

Choose **Start streaming** to begin a real-time transcription of what you speak into your microphone

00:00 of 15:00 min audio stream

- ▶ **Language settings**
- ▶ **Audio settings**
- ▶ **Content removal settings**
- ▶ **Customizations**


3. Activez **Modèle de langage personnalisé** et sélectionnez un modèle dans le menu déroulant.

▼ **Customizations**

Custom vocabulary [Info](#)
A custom vocabulary improves the accuracy of recognizing words and phrases specific to your use case.

Partial results stabilization [Info](#)
Configure Amazon Transcribe to present results that don't change as it processes the transcription output from your stream.

Custom language model [Info](#)
Select the model you want to use with this streaming session.

Custom model selection [Info](#)
The models shown here are based on your language settings. You can choose up to one model per language. You can also [create a new model](#). 

Choose model ▼

Incluez les autres paramètres que vous souhaitez appliquer à votre flux.

4. Vous êtes maintenant prêt à transcrire votre stream. Sélectionnez **Démarrer la diffusion** et commencez à parler. Pour mettre fin à votre dictée, sélectionnez **Arrêter la diffusion**.

flux HTTP/2

Cet exemple crée une requête HTTP/2 qui inclut votre modèle de langage personnalisé. Pour plus d'informations sur l'utilisation du streaming HTTP/2 avec Amazon Transcribe, consultez [Configuration d'un flux HTTP/2](#). Pour plus de détails sur les paramètres et les en-têtes spécifiques à Amazon Transcribe, consultez [StartStreamTranscription](#).

```
POST /stream-transcription HTTP/2
host: transcribestreaming.us-west-2.amazonaws.com
X-Amz-Target: com.amazonaws.transcribe.Transcribe.StartStreamTranscription
Content-Type: application/vnd.amazon.eventstream
X-Amz-Content-Sha256: string
X-Amz-Date: 20220208T235959Z
Authorization: AWS4-HMAC-SHA256 Credential=access-key/20220208/us-west-2/transcribe/
aws4_request, SignedHeaders=content-type;host;x-amz-content-sha256;x-amz-date;x-amz-
target;x-amz-security-token, Signature=string
x-amzn-transcribe-language-code: en-US
x-amzn-transcribe-media-encoding: flac
x-amzn-transcribe-sample-rate: 16000
x-amzn-transcribe-language-model-name: my-first-language-model
transfer-encoding: chunked
```

Les définitions de paramètres se trouvent dans la [référence de l'API](#) ; les paramètres communs à toutes les opérations d'AWSAPI sont répertoriés dans la section [Paramètres communs](#).

WebSocket ruisseau

Cet exemple crée une URL présignée qui applique votre modèle de langage personnalisé à un WebSocket flux. Les sauts de ligne ont été ajoutés pour faciliter la lecture. Pour plus d'informations sur l'utilisation WebSocket des flux avec Amazon Transcribe, consultez [Configuration d'un WebSocket stream](#). Pour plus de détails sur les paramètres, reportez-vous à la section [StartStreamTranscription](#).

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/stream-transcription-
websocket?
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-
west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=string
```

```
&X-Amz-SignedHeaders=content-type%3Bhost%3Bx-amz-date  
&language-code=en-US  
&media-encoding=flac  
&sample-rate=16000  
&language-model-name=my-first-language-model
```

Les définitions de paramètres se trouvent dans la [référence de l'API](#) ; les paramètres communs à toutes les opérations d'AWSAPI sont répertoriés dans la section [Paramètres communs](#).

Utilisation de filtres de vocabulaire personnalisés pour supprimer, masquer ou signaler des mots

Un filtre de vocabulaire personnalisé est un fichier texte qui contient une liste personnalisée de mots individuels que vous souhaitez modifier dans votre sortie de transcription.

Un cas d'utilisation courant est la suppression de termes offensants ou profanes ; toutefois, les filtres de vocabulaire personnalisés sont entièrement personnalisés, vous pouvez donc sélectionner les mots de votre choix. Par exemple, si vous avez un nouveau produit sur le point d'être lancé, vous pouvez masquer le nom du produit dans les transcriptions des réunions. Dans ce cas, vous gardez les parties prenantes up-to-date tout en gardant le nom du produit secret jusqu'au lancement.

Le filtrage du vocabulaire comporte trois méthodes d'affichage : `mask`, `remove`, et `tag`. Reportez-vous aux exemples suivants pour voir comment chacun fonctionne.

- **Masque** : remplace les mots spécifiés par trois astérisques (`***`).

```
"transcript": "You can specify a list of *** or *** words, and *** *** removes them from transcripts automatically."
```

- **Supprimer** : Supprime les mots spécifiés, ne laissant rien à leur place.

```
"transcript": "You can specify a list of or words, and removes them from transcripts automatically."
```

- **Tag** : ajoute une balise (`"vocabularyFilterMatch": true`) à chaque mot spécifié, mais ne modifie pas le mot lui-même. Le balisage permet de remplacer et de modifier rapidement les transcriptions.

```
"transcript": "You can specify a list of profane or offensive words, and amazon transcribe removes them from transcripts automatically."
```

```
...
```

```
  "alternatives": [  
    {  
      "confidence": "1.0",  
      "content": "profane"  
    }  
  ],  
  "type": "pronunciation",
```

```
"vocabularyFilterMatch": true
```

Lorsque vous soumettez une demande de transcription, vous pouvez spécifier un filtre de vocabulaire personnalisé et la méthode de filtrage que vous souhaitez appliquer. Amazon Transcribe modifie ensuite les correspondances de mots exactes lorsqu'elles apparaissent dans votre transcription, en fonction de la méthode de filtrage que vous avez spécifiée.

Des filtres de vocabulaire personnalisés peuvent être appliqués aux demandes de transcription par lots et en continu. Pour savoir comment créer un filtre de vocabulaire personnalisé, consultez [Création d'un filtre de vocabulaire](#). Pour savoir comment appliquer votre filtre de vocabulaire personnalisé, consultez [Utiliser un filtre de glossaire personnalisé](#).

Note

Amazon Transcribe masque automatiquement les termes sensibles à la race, mais vous pouvez désactiver ce filtre par défaut en contactant le [support AWS technique](#).

Pour une présentation vidéo du filtrage du vocabulaire, voir [Utilisation des filtres de vocabulaire](#).

Opérations d'API spécifiques au filtrage du vocabulaire

[CreateVocabularyFilter](#), [DeleteVocabularyFilter](#), [GetVocabularyFilter](#), [ListVocabularyFilters](#), [UpdateVocabularyFilter](#)

Création d'un filtre de vocabulaire

Il existe deux options pour créer un filtre de vocabulaire personnalisé :

1. Enregistrez une liste de mots séparés par des lignes sous forme de fichier texte brut avec un codage UTF-8.
 - Vous pouvez utiliser cette approche avec les [AWS Management Console](#), [AWS CLI](#), ou [AWS les SDK](#).
 - Si vous utilisez le [AWS Management Console](#), vous pouvez fournir un chemin local ou un [Amazon S3 URI](#) pour votre fichier de vocabulaire personnalisé.

- Si vous utilisez les AWS kits SDK AWS CLI ou, vous devez télécharger votre fichier de vocabulaire personnalisé dans un Amazon S3 compartiment et inclure l'Amazon S3URI dans votre demande.
2. Incluez une liste de mots séparés par des virgules directement dans votre demande d'API.
- Vous pouvez utiliser cette approche avec les AWS CLI ou AWS les SDK utilisant le [Words](#) paramètre.

Pour des exemples de chaque méthode, reportez-vous à [Création de filtres de vocabulaire personnalisés](#)

Points à prendre en compte lors de la création de votre filtre de vocabulaire personnalisé :

- Les mots ne sont pas sensibles à la casse. Par exemple, les termes « malédiction » et « malédiction » sont traités de la même manière.
- Seules les correspondances de mots exactes sont filtrées. Par exemple, si votre filtre inclut « jurer » mais que votre contenu multimédia contient le mot « jure » ou « jure », ceux-ci ne sont pas filtrés. Seules les instances de « jure » sont filtrées. Vous devez donc inclure toutes les variantes des mots que vous souhaitez filtrer.
- Les filtres ne s'appliquent pas aux mots contenus dans d'autres mots. Par exemple, si un filtre de vocabulaire personnalisé contient « marin » mais pas « sous-marin », le terme « sous-marin » n'est pas modifié dans la transcription.
- Chaque entrée ne peut contenir qu'un seul mot (sans espaces).
- Si vous enregistrez votre filtre de vocabulaire personnalisé sous forme de fichier texte, il doit être au format texte brut avec un codage UTF-8.
- Vous pouvez disposer d'un maximum de 100 filtres AWS de vocabulaire personnalisés par filtre, chacun pouvant atteindre 50 Ko.
- Vous ne pouvez utiliser que des caractères compatibles avec votre langue. Reportez-vous au jeu de [caractères](#) de votre langue pour plus de détails.

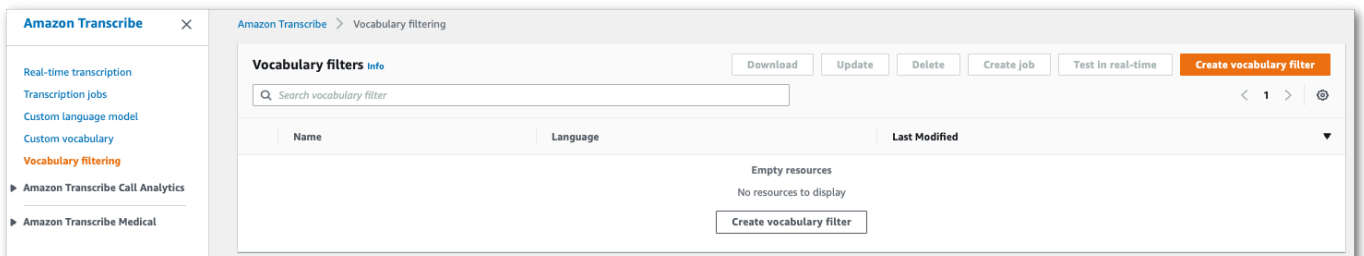
Création de filtres de vocabulaire personnalisés

Pour traiter un filtre de vocabulaire personnalisé à utiliser avec Amazon Transcribe, consultez les exemples suivants :

AWS Management Console

Avant de continuer, enregistrez votre filtre de vocabulaire personnalisé sous forme de fichier texte (*.txt). Vous pouvez éventuellement télécharger votre fichier dans un Amazon S3 compartiment.

1. Connectez-vous à [AWS Management Console](#).
2. Dans le panneau de navigation, choisissez Filtrage du vocabulaire. Cela ouvre la page des filtres de vocabulaire dans laquelle vous pouvez consulter les filtres de vocabulaire personnalisés existants ou en créer un nouveau.
3. Sélectionnez Créer un filtre de vocabulaire.



Vous accédez alors à la page de filtre de création de vocabulaire. Entrez un nom pour votre nouveau filtre de vocabulaire personnalisé.

Sélectionnez l'option Téléchargement de fichier ou localisation S3 sous Source d'entrée de vocabulaire. Spécifiez ensuite l'emplacement de votre fichier de vocabulaire personnalisé.

Create vocabulary filter [Info](#)

Vocabulary filtering settings

Name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9 and - (hyphen).

Language

Vocabulary input source [Info](#)

File upload

S3 location

Vocabulary filter file location on S3

Provide a path to the S3 location where your vocabulary filter file is stored. To find a path, go to [Amazon S3](#)

File format: txt, maximum size 50 KB.

Tags - optional

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

No tags associated with the resource.

You can add up to 50 more tags.

- Vous pouvez éventuellement ajouter des balises à votre filtre de vocabulaire personnalisé. Une fois que vous avez rempli tous les champs, sélectionnez Créer un filtre de vocabulaire au bas de la page. Si aucune erreur ne se produit lors du traitement de votre fichier, vous êtes redirigé vers la page des filtres de vocabulaire.

Votre filtre de vocabulaire personnalisé est maintenant prêt à être utilisé.

AWS CLI

Cet exemple utilise la [create-vocabulary-filter](#) commande pour transformer une liste de mots en un filtre de vocabulaire personnalisé utilisable. Pour plus d'informations, veuillez consulter [CreateVocabularyFilter](#).

Option 1 : Vous pouvez inclure votre liste de mots dans votre demande à l'aide du `words` paramètre.

```
aws transcribe create-vocabulary-filter \  
--vocabulary-filter-name my-first-vocabulary-filter \  
--language-code en-US \  
--words profane,offensive,Amazon,Transcribe
```

Option 2 : Vous pouvez enregistrer votre liste de mots sous forme de fichier texte et la télécharger dans un Amazon S3 compartiment, puis inclure l'URI du fichier dans votre demande à l'aide du `vocabulary-filter-file-uri` paramètre.

```
aws transcribe create-vocabulary-filter \  
--vocabulary-filter-name my-first-vocabulary-filter \  
--language-code en-US \  
--vocabulary-filter-file-uri s3://DOC-EXAMPLE-BUCKET/my-vocabulary-filters/my-vocabulary-filter.txt
```

Voici un autre exemple d'utilisation de la [create-vocabulary-filter](#) commande et d'un corps de requête qui crée votre filtre de vocabulaire personnalisé.

```
aws transcribe create-vocabulary-filter \  
--cli-input-json file://filepath/my-first-vocab-filter.json
```

Le fichier `my-first-vocab-filter.json` contient le corps de requête suivant.

Option 1 : Vous pouvez inclure votre liste de mots dans votre demande à l'aide du `Words` paramètre.

```
{  
  "VocabularyFilterName": "my-first-vocabulary-filter",  
  "LanguageCode": "en-US",  
  "Words": [  
    "profane", "offensive", "Amazon", "Transcribe"  
  ]  
}
```

Option 2 : Vous pouvez enregistrer votre liste de mots sous forme de fichier texte et la télécharger dans un Amazon S3 compartiment, puis inclure l'URI du fichier dans votre demande à l'aide du `VocabularyFilterFileUri` paramètre.

```
{  
  "VocabularyFilterName": "my-first-vocabulary-filter",  
  "LanguageCode": "en-US",
```

```
"VocabularyFilterFileUri": "s3://DOC-EXAMPLE-BUCKET/my-vocabulary-filters/my-
vocabulary-filter.txt"
}
```

Note

Si vous l'incluez `VocabularyFilterFileUri` dans votre demande, vous ne pouvez pas utiliser `Words` ; vous devez choisir l'un ou l'autre.

AWS SDK for Python (Boto3)

Cet exemple utilise le [AWS SDK for Python \(Boto3\)](#) pour créer un filtre de vocabulaire personnalisé à l'aide de la méthode [create_vocabulary_filter](#). Pour plus d'informations, veuillez consulter [CreateVocabularyFilter](#).

Pour des exemples supplémentaires d'utilisation AWS des SDK, y compris des exemples spécifiques à des fonctionnalités, des scénarios et des exemples interservices, consultez le [Exemples de code pour Amazon Transcribe à l'aide de kits de développement logiciel AWS](#) chapitre.

Option 1 : Vous pouvez inclure votre liste de mots dans votre demande à l'aide du `Words` paramètre.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
vocab_name = "my-first-vocabulary-filter"
response = transcribe.create_vocabulary_filter(
    LanguageCode = 'en-US',
    VocabularyFilterName = vocab_name,
    Words = [
        'profane', 'offensive', 'Amazon', 'Transcribe'
    ]
)
```

Option 2 : Vous pouvez enregistrer votre liste de mots sous forme de fichier texte et la télécharger dans un [Amazon S3](#) compartiment, puis inclure l'URI du fichier dans votre demande à l'aide du `VocabularyFilterFileUri` paramètre.

```
from __future__ import print_function
```

```
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
vocab_name = "my-first-vocabulary-filter"
response = transcribe.create_vocabulary_filter(
    LanguageCode = 'en-US',
    VocabularyFilterName = vocab_name,
    VocabularyFilterFileUri = 's3://DOC-EXAMPLE-BUCKET/my-vocabulary-filters/my-
vocabulary-filter.txt'
)
```

Note

Si vous l'incluez `VocabularyFilterFileUri` dans votre demande, vous ne pouvez pas utiliser `Words` ; vous devez choisir l'un ou l'autre.

Note

Si vous créez un nouveau Amazon S3 compartiment pour vos fichiers de filtre de vocabulaire personnalisé, assurez-vous que le IAM rôle à l'origine de la [CreateVocabularyFilter](#) demande dispose des autorisations nécessaires pour accéder à ce compartiment. Si le rôle ne dispose pas des autorisations appropriées, votre demande échoue. Vous pouvez éventuellement spécifier un IAM rôle dans votre demande en incluant le `DataAccessRoleArn` paramètre. Pour plus d'informations sur IAM les rôles et les politiques dans Amazon Transcribe, consultez [Exemples de politiques basées sur l'identité Amazon Transcribe](#).

Utiliser un filtre de glossaire personnalisé

Une fois votre filtre de vocabulaire personnalisé créé, vous pouvez l'inclure dans vos demandes de transcription. Reportez-vous aux sections suivantes pour obtenir des exemples.

La langue du filtre de vocabulaire personnalisé que vous incluez dans votre demande doit correspondre au code de langue que vous avez spécifié pour votre média. Si vous utilisez l'identification de la langue et que vous spécifiez plusieurs options linguistiques, vous pouvez inclure un filtre de vocabulaire personnalisé par langue spécifiée. Si les langues de vos filtres de vocabulaire

personnalisés ne correspondent pas à la langue identifiée dans votre audio, vos filtres ne sont pas appliqués à votre transcription et il n'y a aucun avertissement ni erreur.

Utilisation d'un filtre de vocabulaire personnalisé dans une transcription par lots

Pour utiliser un filtre de vocabulaire personnalisé avec une transcription par lots, consultez les exemples suivants :

AWS Management Console

1. Connectez-vous à [AWS Management Console](#).
2. Dans le volet de navigation, choisissez Tâches de transcription, puis sélectionnez Créer une tâche (en haut à droite). La page Spécifier les détails de la tâche s'ouvre.

Specify job details [Info](#)

Job settings

Name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, . (period), _ (underscore), and - (hyphen).

Model type [Info](#)

Choose the type of model to use for the transcription job.

General model
To use a model that is not specialized for a particular use case, choose this option. Configuration options vary between languages.

Custom language model
To use a model that you trained for your specific use case, choose this option. This model has fewer configuration options than the general model.

Language settings

You can transcribe your audio file in a language that you specify or have Amazon Transcribe identify and transcribe it in the predominant language.

Specific language [Info](#)
If you know the language spoken in your source audio, choose this option to get the most accurate results. The options available for additional processing vary between languages.

Automatic language identification [Info](#)
If you don't know the language spoken in your audio files, choose this option. You have access to fewer options for additional processing than if you choose **Specific language**.

Language

Choose the language of the input audio.

► **Additional settings**

Donnez un nom à votre tâche et spécifiez votre support d'entrée. Incluez éventuellement d'autres champs, puis choisissez Suivant.

3. Sur la page Configurer la tâche, dans le panneau de suppression de contenu, activez le filtrage du vocabulaire.

Configure job - optional [Info](#)

Audio settings

Audio identification [Info](#)
Choose to split multi-channel audio into separate channels for transcription, or identify speakers in the input audio.

Alternative results [Info](#)
Enable to view more transcription results

Content removal

Content removal conceals information in the resulting transcript from your source audio file. Amazon Transcribe changes items in the transcript and does not modify the source audio.

PII redaction [Info](#)
Label the type of PII and also mask the content with the PII entity type in the transcription output. For example, (123) 456-7890 will be masked as [PHONE].

Vocabulary filtering [Info](#)
Vocabulary filtering can remove, mask or tag specified words in the final transcript.

Filter selection
The vocabulary filters shown here are based on your language settings. You can choose up to one vocabulary filter per language. You can also [create a new vocabulary filter](#). [↗](#)

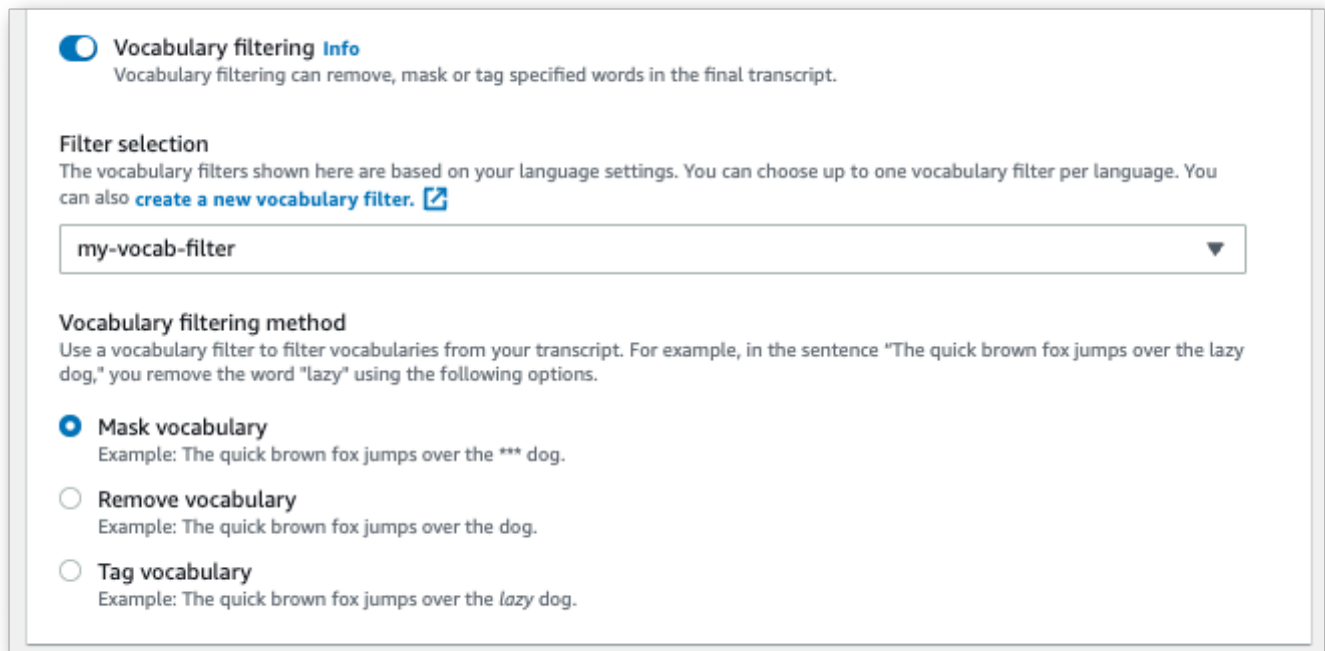
Choose a vocabulary filter ▼

Customization

Custom vocabulary [Info](#)
A custom vocabulary improves the accuracy of recognizing words and phrases specific to your use case.

Cancel Previous **Create job**

4. Sélectionnez votre filtre de vocabulaire personnalisé dans le menu déroulant et spécifiez la méthode de filtrage.



5. Sélectionnez Créer une tâche pour exécuter votre tâche de transcription.

AWS CLI

Cet exemple utilise la [start-transcription-job](#) commande et le `Settings` paramètre avec les `VocabularyFilterMethod` sous-paramètres `VocabularyFilterName` et. Pour plus d'informations, consultez [StartTranscriptionJob](#) et [Settings](#).

```
aws transcribe start-transcription-job \
--region us-west-2 \
--transcription-job-name my-first-transcription-job \
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \
--output-bucket-name DOC-EXAMPLE-BUCKET \
--output-key my-output-files/ \
--language-code en-US \
--settings VocabularyFilterName=my-first-vocabulary-filter,VocabularyFilterMethod=mask
```

Voici un autre exemple d'utilisation de la [start-transcription-job](#) commande et d'un corps de requête qui inclut votre filtre de vocabulaire personnalisé associé à cette tâche.

```
aws transcribe start-transcription-job \
--region us-west-2 \
--cli-input-json file://my-first-vocabulary-filter-job.json
```


Le fichier `my-first-vocabulary-filter-job.json` contient le corps de requête suivant.

```
{
  "TranscriptionJobName": "my-first-transcription-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
  },
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "OutputKey": "my-output-files/",
  "LanguageCode": "en-US",
  "Settings": {
    "VocabularyFilterName": "my-first-vocabulary-filter",
    "VocabularyFilterMethod": "mask"
  }
}
```

AWS SDK for Python (Boto3)

Cet exemple utilise le [AWS SDK for Python \(Boto3\)](#) pour inclure un filtre de vocabulaire personnalisé utilisant l'`Settings` argument de la méthode [start_transcription_job](#). Pour plus d'informations, consultez [StartTranscriptionJob](#) et [Settings](#).

Pour des exemples supplémentaires d'utilisation AWS des SDK, y compris des exemples spécifiques à des fonctionnalités, des scénarios et des exemples interservices, consultez le [Exemples de code pour Amazon Transcribe à l'aide de kits de développement logiciel AWS](#) chapitre.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
transcribe.start_transcription_job(
    TranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'my-output-files/',
    LanguageCode = 'en-US',
    Settings = {
        'VocabularyFilterName': 'my-first-vocabulary-filter',
```

```
        'VocabularyFilterMethod': 'mask'
    }
)

while True:
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)
    if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Utilisation d'un filtre de vocabulaire personnalisé dans une transcription en streaming


Pour utiliser un filtre de vocabulaire personnalisé avec une transcription en continu, consultez les exemples suivants :

AWS Management Console

1. Connectez-vous au [AWS Management Console](#).
2. Dans le panneau de navigation, choisissez Transcription en temps réel. Faites défiler l'écran vers le bas jusqu'à Paramètres de suppression de contenu et agrandissez ce champ s'il est réduit.

Real-time transcription [Info](#)

See how Amazon Transcribe creates a text copy of speech in real time. Choose **Start streaming** and talk.

Transcription Download full transcript  **Start streaming**

Transcription output Current language: English, US

Choose **Start streaming** to begin a real-time transcription of what you speak into your microphone

00:00 of 15:00 min audio stream

- ▶ **Language settings**
- ▶ **Audio settings**
- ▶ **Content removal settings**
- ▶ **Customizations**

3. Activez le filtrage du vocabulaire. Sélectionnez un filtre de glossaire personnalisé dans le menu déroulant et spécifiez la méthode de filtrage.

▼ Content removal settings

Vocabulary filtering [Info](#)
Vocabulary filtering removes, masks, or tags words that you specify in your vocabulary filter. Choose a vocabulary filter to see an example.

Filter selection
The vocabulary filters shown here are based on your language settings. You can choose up to one vocabulary filter per language. You can also [create a new vocabulary filter](#).

my-vocab-filter ▼

Vocabulary filtering method [Info](#)
Use a vocabulary filter to filter vocabularies from your transcript. For example, in the sentence "The quick brown fox jumps over the lazy dog," you remove the word "lazy" using the following options.

- Mask vocabulary**
Example: The quick brown fox jumps over the *** dog.
- Remove vocabulary**
Example: The quick brown fox jumps over the dog.
- Tag vocabulary**
Example: The quick brown fox jumps over the *lazy* dog.

Ajoutez les autres paramètres que vous souhaitez appliquer à votre flux.

4. Voici le moment pour transcrire votre stream. Sélectionnez Démarrer la diffusion et commencez à parler. Pour mettre fin à votre dictée, sélectionnez Arrêter la diffusion.

Stream HTTP/2

Cet exemple crée une requête HTTP/2 qui inclut votre filtre de vocabulaire personnalisé et votre méthode de filtrage. Pour plus d'informations sur l'utilisation du streaming HTTP/2 avec Amazon Transcribe, consultez [Configuration d'un flux HTTP/2](#). Pour plus de détails sur les paramètres et les en-têtes spécifiques à Amazon Transcribe, consultez [StartStreamTranscription](#).

```
POST /stream-transcription HTTP/2
host: transcribestreaming.us-west-2.amazonaws.com
X-Amz-Target: com.amazonaws.transcribe.Transcribe.StartStreamTranscription
Content-Type: application/vnd.amazon.eventstream
X-Amz-Content-Sha256: string
X-Amz-Date: 20220208T235959Z
Authorization: AWS4-HMAC-SHA256 Credential=access-key/20220208/us-west-2/transcribe/
aws4_request, SignedHeaders=content-type;host;x-amz-content-sha256;x-amz-date;x-amz-
target;x-amz-security-token, Signature=string
x-amzn-transcribe-language-code: en-US
x-amzn-transcribe-media-encoding: flac
x-amzn-transcribe-sample-rate: 16000
x-amzn-transcribe-vocabulary-filter-name: my-first-vocabulary-filter
x-amzn-transcribe-vocabulary-filter-method: mask
transfer-encoding: chunked
```

Les définitions de paramètres se trouvent dans la [référence de l'API](#) ; les paramètres communs à toutes les opérations de l'AWSAPI sont répertoriés dans la section [Paramètres communs](#).

WebSocket ruisseau

Cet exemple crée une URL présignée qui applique votre filtre de vocabulaire personnalisé à un WebSocket flux. Les sauts de ligne ont été ajoutés pour faciliter la lecture. Pour plus d'informations sur l'utilisation WebSocket des flux avec Amazon Transcribe, consultez [Configuration d'un WebSocket stream](#). Pour plus de détails sur les paramètres, reportez-vous à la section [StartStreamTranscription](#).

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/stream-transcription-
websocket?
&X-Amz-Algorithm=AWS4-HMAC-SHA256
```

```
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-  
west-2%2Ftranscribe%2Faws4_request  
&X-Amz-Date=20220208T235959Z  
&X-Amz-Expires=300  
&X-Amz-Security-Token=security-token  
&X-Amz-Signature=string  
&X-Amz-SignedHeaders=content-type%3Bhost%3Bx-amz-date  
&language-code=en-US  
&media-encoding=flac  
&sample-rate=16000  
&vocabulary-filter-name=my-first-vocabulary-filter  
&vocabulary-filter-method=mask
```

Les définitions de paramètres se trouvent dans la [référence de l'API](#) ; les paramètres communs à toutes les opérations de l'AWSAPI sont répertoriés dans la section [Paramètres communs](#).

Détecter les discours toxiques

La détection des discours toxiques est conçue pour aider à modérer les plateformes de réseaux sociaux qui impliquent peer-to-peer dialogue, tel que les jeux en ligne et les plateformes de chat social. L'utilisation d'un discours toxique peut être profondément préjudiciable aux individus, aux groupes de pairs et aux communautés. Le signalement du langage injurieux aide les organisations à garder les conversations courtoises et à maintenir un environnement en ligne sûr et inclusif permettant aux utilisateurs de créer, de partager et de participer librement.

Amazon TranscribeLa détection de la toxicité utilise à la fois des signaux audio et textuels pour identifier et classer les contenus toxiques basés sur la voix en sept catégories, notamment le harcèlement sexuel, les discours de haine, les menaces, les abus, les blasphèmes, les insultes et les images graphiques. Outre le texte, Amazon TranscribeLa détection de la toxicité utilise des indices vocaux, tels que les tonalités et la tonalité, pour identifier les intentions toxiques présentes dans le discours. Il s'agit d'une amélioration par rapport aux systèmes de modération de contenu standard qui sont conçus pour se concentrer uniquement sur des termes spécifiques, sans tenir compte de l'intention.

Amazon Transcribesignale et classe les propos toxiques, ce qui minimise le volume de données devant être traitées manuellement. Cela permet aux modérateurs de contenu de gérer rapidement et efficacement le discours sur leurs plateformes.

Les catégories de discours toxiques incluent :

- **Blasphème:** Discours qui contient des mots, des phrases ou des acronymes impolis, vulgaires ou offensants.
- **Discours de haine:** Discours qui critique, insulte, dénonce ou déshumanise une personne ou un groupe sur la base d'une identité (telle que la race, l'origine ethnique, le sexe, la religion, l'orientation sexuelle, les capacités et l'origine nationale).
- **Sexuel:** Discours qui indique un intérêt, une activité ou une excitation sexuels en utilisant des références directes ou indirectes à des parties du corps, à des traits physiques ou à des relations sexuelles.
- **Insultes:** Discours comportant un langage dégradant, humiliant, moqueur, insultant ou dénigrant. Ce type de langage est également qualifié d'intimidation.
- **Violence ou menace:** Discours qui inclut des menaces visant à infliger de la douleur, des blessures ou de l'hostilité à une personne ou à un groupe.

- **Graphique:** Discours qui utilise des images visuellement descriptives et désagréablement vives. Ce type de langage est souvent intentionnellement verbeux pour amplifier l'inconfort du destinataire.
- **Harcèlement ou abus:** Discours destiné à affecter le bien-être psychologique du destinataire, y compris en termes humiliants et objectivants. Ce type de langage est également qualifié de harcèlement.

La détection de la toxicité analyse les segments de parole (le discours entre les pauses naturelles) et attribue des scores de confiance à ces segments. Les scores de confiance sont des valeurs comprises entre 0 et 1. Un score de confiance plus élevé indique une plus grande probabilité que le contenu soit un discours toxique dans la catégorie associée. Vous pouvez utiliser ces scores de confiance pour définir le seuil de détection de toxicité approprié pour votre cas d'utilisation.

Note

La détection de la toxicité n'est disponible que pour les transcriptions par lots en anglais américain(en-US).

Afficher [exemple de sortie](#) au format JSON.

Utilisation de la détection de discours toxiques

Utilisation de la détection de discours toxiques dans une transcription par lots

Pour utiliser la détection de discours toxiques avec une transcription par lots, consultez les exemples suivants :

AWS Management Console

1. Connectez-vous à [AWS Management Console](#).
2. Dans le volet de navigation, choisissez **Tâches de transcription**, puis sélectionnez **Créer un emploi**(en haut à droite). Cela ouvre le **Spécifier les détails de la tâche**page.

Specify job details [Info](#)

Job settings

Name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, . (period), _ (underscore), and - (hyphen).

Model type [Info](#)

Choose the type of model to use for the transcription job.

General model

To use a model that is not specialized for a particular use case, choose this option. Configuration options vary between languages.

Custom language model

To use a model that you trained for your specific use case, choose this option. This model has fewer configuration options than the general model.

Language settings

You can transcribe your audio file in a language that you specify or have Amazon Transcribe identify and transcribe it in the predominant language.

Specific language [Info](#)

If you know the language spoken in your source audio, choose this option to get the most accurate results. The options available for additional processing vary between languages.

Automatic language identification [Info](#)

If you don't know the language spoken in your audio files, choose this option. You have access to fewer options for additional processing than if you choose **Specific language**.

Language

Choose the language of the input audio.

3. Sur leSpécifier les détails de la tâchepage, vous pouvez également activer la rédaction des informations personnelles si vous le souhaitez. Notez que les autres options répertoriées ne sont pas prises en charge avec la détection de la toxicité. Sélectionnez Suivant. Cela vous amène auConfigurer la tâche : facultatifpage. Dans leRéglages audiopanneau, sélectionnezDétection de toxicité.

Audio settings

Audio identification [Info](#)
Choose to split multi-channel audio into separate channels for transcription, or partition speakers in the input audio.

Alternative results [Info](#)
Enable to view more transcription results

Toxicity detection [Info](#)
Flag toxic speech in your transcription output

Content removal

Content removal conceals information in the resulting transcript from your source audio file. Amazon Transcribe changes items in the transcript and does not modify the source audio.

PII redaction [Info](#)
Label the type of PII and also mask the content with the PII entity type in the transcription output. For example, (123) 456-7890 will be masked as [PHONE].

Vocabulary filtering [Info](#)
Vocabulary filtering can remove, mask or tag specified words in the final transcript.

Customization

Custom vocabulary [Info](#)
A custom vocabulary improves the accuracy of recognizing words and phrases specific to your use case.

[Cancel](#) [Previous](#) [Create job](#)

4. Sélectionnez **Créer un emploi** pour exécuter votre travail de transcription.
5. Une fois votre travail de transcription terminé, vous pouvez télécharger votre transcription à partir du **Télécharger** menu déroulant sur la page détaillée de la tâche de transcription.

AWS CLI

Cet exemple utilise le [start-transcription-job](#) commande et `ToxicityDetection` paramètre. Pour plus d'informations, consultez [StartTranscriptionJob](#) et [ToxicityDetection](#).

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--transcription-job-name my-first-transcription-job \  
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
--output-bucket-name DOC-EXAMPLE-BUCKET \  
--output-key my-output-files/ \  
--language-code en-US \  
--toxicity-detection ToxicityCategories=ALL
```

Voici un autre exemple utilisant le [start-transcription-job](#) commande, et un organisme de demande qui inclut la détection de la toxicité.

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--cli-input-json file://filepath/my-first-toxicity-job.json
```

Le dossier `my-first-toxicity-job.json` contient le corps de requête suivant.

```
{  
  "TranscriptionJobName": "my-first-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
  },  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "OutputKey": "my-output-files/",  
  "LanguageCode": "en-US",  
  "ToxicityDetection": [  
    {  
      "ToxicityCategories": [ "ALL" ]  
    }  
  ]  
}
```

AWS SDK for Python (Boto3)

Cet exemple utilise le [AWS SDK for Python \(Boto3\)](#) pour activer [ToxicityDetection](#) pour le [démarrage transcription_job](#) méthode. Pour plus d'informations, consultez [StartTranscriptionJob](#) et [ToxicityDetection](#).

Pour des exemples supplémentaires utilisant le [AWS Les kits SDK](#), y compris des exemples spécifiques aux fonctionnalités, des scénarios et des exemples interservices, se réfèrent au [Exemples de code pour Amazon Transcribe à l'aide de kits de développement logiciel AWS](#) chapitre.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
transcribe.start_transcription_job(
    TranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'my-output-files/',
    LanguageCode = 'en-US',
    ToxicityDetection = [
        {
            'ToxicityCategories': ['ALL']
        }
    ]
)

while True:
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)
    if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Exemple de sortie

Les discours toxiques sont étiquetés et catégorisés dans votre sortie de transcription. Chaque instance de discours toxique est classée et un score de confiance (une valeur comprise entre 0 et 1) lui est attribué. Une valeur de confiance plus élevée indique une plus grande probabilité que le contenu soit un discours toxique relevant de la catégorie spécifiée.

Exemple de sortie (JSON)

Voici un exemple de sortie au format JSON montrant un discours toxique classé avec les scores de confiance associés.

```
{
  "jobName": "my-toxicity-job",
  "accountId": "111122223333",
  "results": {
    "transcripts": [...],
    "items": [...],
    "toxicity_detection": [
      {
        "text": "What the * are you doing man? That's why I didn't want to play
with your * . man it was a no, no I'm not calming down * man. I well I spent I spent
too much * money on this game.",
        "toxicity": 0.7638,
        "categories": {
          "profanity": 0.9913,
          "hate_speech": 0.0382,
          "sexual": 0.0016,
          "insult": 0.6572,
          "violence_or_threat": 0.0024,
          "graphic": 0.0013,
          "harassment_or_abuse": 0.0249
        },
        "start_time": 8.92,
        "end_time": 21.45
      },
      Items removed for brevity
      {
        "text": "What? Who? What the * did you just say to me? What's your
address? What is your * address? I will pull up right now on your * * man. Take your *
back to , tired of this **.",
        "toxicity": 0.9816,
```

```
    "categories": {
      "profanity": 0.9865,
      "hate_speech": 0.9123,
      "sexual": 0.0037,
      "insult": 0.5447,
      "violence_or_threat": 0.5078,
      "graphic": 0.0037,
      "harassment_or_abuse": 0.0613
    },
    "start_time": 43.459,
    "end_time": 54.639
  },
]
},
...
"status": "COMPLETED"
}
```

La suppression ou l'identification personnelle des données d'identification personnelle.

La rédaction est utilisée pour masquer ou supprimer du contenu sensible, sous la forme d'informations personnelles identifiables (PII), de vos transcriptions. Les types d'informations personnelles que Amazon Transcribe vous pouvez supprimer varient entre les transcriptions par lots et les transcriptions en continu. Pour consulter la liste des informations personnelles pour chaque méthode de transcription, reportez-vous à [Expurgation des PII \(données d'identification personnelle\) dans votre tâche par lots](#) et [Expurgation ou identification des données d'identification personnelle \(PII\) dans un flux en temps réel](#). Avec les transcriptions en streaming, vous avez également la possibilité de signaler les informations personnelles sans les modifier ; reportez-vous à la section [Exemple de sortie d'identification PII](#) pour un exemple de sortie.

Lorsque la rédaction est activée, vous avez la possibilité de générer uniquement une transcription expurgée ou à la fois une transcription expurgée et une transcription non expurgée. Si vous choisissez de générer uniquement une transcription expurgée, notez que votre média est le seul endroit où la conversation complète est stockée. Si vous supprimez votre média d'origine, il n'y aura aucune trace des informations personnelles non expurgées. Pour cette raison, il peut être prudent de générer une transcription non expurgée en plus d'une transcription expurgée.

Pour en savoir plus sur la rédaction d'informations d'identification personnelle à l'aide de transcriptions par lots, consultez : [Expurgation des PII \(données d'identification personnelle\) dans votre tâche par lots](#).

Pour en savoir plus sur la rédaction ou l'identification des informations personnelles à l'aide de transcriptions en streaming, consultez : [Expurgation ou identification des données d'identification personnelle \(PII\) dans un flux en temps réel](#).

Important

La fonction de rédaction est conçue pour identifier et supprimer les données sensibles. Toutefois, en raison de la nature prédictive de l'apprentissage automatique, il est possible que vous ne puissiez pas identifier et supprimer toutes les instances de données sensibles de votre transcription. Nous vous recommandons vivement de consulter toute sortie rédigée pour vous assurer qu'elle répond à vos besoins.

La fonction de données personnelles relatives à la santé, telles que la loi américaine HIPAA (Health Insurance Portability and Accountability Act) de 1996, la loi américaine HIPAA (Health Insurance Portability and Accountability Act) de 1996 la loi américaine HIPAA.

Pour une présentation vidéo de la fonction Amazon Transcribe de rédaction, voir [Utiliser la rédaction de contenu pour identifier et rédiger des informations d'identification personnelle](#).

Expurgation des PII (données d'identification personnelle) dans votre tâche par lots

Lorsque vous supprimez des informations personnelles identifiables (PII) d'une transcription lors d'un travail de transcription par lots, remplacez Amazon Transcribe chaque instance identifiée de PII par le corps du texte principal de votre transcription. [PII] Vous pouvez également afficher le type de PII qui est expurgé dans la word-for-word partie de la sortie de transcription. Pour un exemple de sortie, consultez la section [Exemple de sortie rédigée \(batch\)](#).

La rédaction avec transcriptions par lots est disponible en anglais américain (en-US) et en espagnol américain (es-US). L'expurgation n'est pas compatible avec l'[identification des langues](#).

Les transcriptions expurgées et non expurgées sont stockées dans le même compartiment de sortie. Amazon S3 Amazon Transcribe les stocke dans un compartiment que vous spécifiez ou dans le Amazon S3 compartiment par défaut géré par le service.

Les types de données personnelles que Amazon Transcribe vous pouvez reconnaître pour les transcriptions par lots

Type de PII	Description
ADDRESS	Adresse physique, telle que 100 Main Street, Anytown, États-Unis ou Suite n°12, bâtiment 123. Une adresse peut inclure une rue, un bâtiment, un emplacement, une ville, un état, un pays, un comté, un code postal, une circonscription, un quartier, etc.
ALL	Expurgez ou identifiez tous les types de PII répertoriés dans ce tableau.

Type de PII	Description
BANK_ACCOUNT_NUMBER	Numéro du compte bancaire aux États-Unis. Il s'agit généralement d'un numéro composé de 10 à 12 chiffres, mais Amazon Transcribe reconnaît également les numéros de compte bancaire lorsque seuls les 4 derniers chiffres sont présents.
BANK_ROUTING	Numéro d'acheminement du compte bancaire aux États-Unis. Il s'agit généralement d'un numéro à 9 chiffres, mais Amazon Transcribe reconnaît également les numéros de routage lorsque seuls les 4 derniers chiffres sont présents.
CREDIT_DEBIT_CVV	Code de vérification de carte (CVV) à 3 chiffres présent sur les cartes de crédit et de débit VISA et Discover. MasterCard Sur les cartes de crédit ou de débit American Express, il s'agit d'un code numérique à 4 chiffres.
CREDIT_DEBIT_EXPIRY	Date d'expiration d'une carte de crédit ou de débit. Ce numéro comporte généralement 4 chiffres et est au format mois/année ou MM/AA. Par exemple, Amazon Transcribe peut reconnaître des dates d'expiration telles que le 21/01, le 01/2021 et le 1er janvier 2021.
CREDIT_DEBIT_NUMBER	Numéro de carte de crédit ou de débit. La longueur de ces numéros peut varier de 13 à 16 chiffres, mais ils reconnaissent Amazon Transcribe également les numéros de carte de crédit ou de débit lorsque seuls les 4 derniers chiffres sont présents.
EMAIL	Adresse e-mail, telle que efua.owusu@email.com.

Type de PII	Description
NAME	Nom d'une personne. Ce type d'entité n'inclut pas les titres, tels que M., Mme, Miss ou Dr. Il Amazon Transcribe n'applique pas ce type d'entité aux noms qui font partie d'organisations ou d'adresses. Par exemple, Amazon Transcribe reconnaît l'organisation John Doe en tant qu'organisation et Jane Doe Street en tant qu'adresse.
PHONE	Numéro de téléphone. Ce type d'entité inclut également les numéros de télécopieur et de pager.
PIN	Numéro d'identification personnel (PIN) à 4 chiffres qui permet à une personne d'accéder aux informations de son compte bancaire.
SSN	Un numéro de sécurité sociale (SSN) est un numéro à 9 chiffres délivré aux citoyens américains, aux résidents permanents et aux résidents actifs temporaires. Amazon Transcribe reconnaît également les numéros de sécurité sociale lorsque seuls les 4 derniers chiffres sont présents.

Vous pouvez démarrer une tâche de transcription par lots à l'aide du AWS Management Console ou du AWS SDK. AWS CLI

AWS Management Console

1. Connectez-vous à la [AWS Management Console](#).
2. Dans le volet de navigation, choisissez Tâches de transcription, puis sélectionnez Créer une tâche (en haut à droite). La page Spécifier les détails de la tâche s'ouvre.

- Après avoir rempli les champs souhaités sur la page Spécifier les détails de la tâche, sélectionnez Suivant pour accéder à la page Configurer la tâche - facultatif. Vous y trouverez le volet Suppression de contenu avec le bouton à bascule Expurgation des PII.

Configure job - *optional* [Info](#)

Audio settings

- Audio identification** [Info](#)
Choose to split multi-channel audio into separate channels for transcription, or identify speakers in the input audio.
- Alternative results** [Info](#)
Enable to view more transcription results

Content removal

Content removal conceals information in the resulting transcript from your source audio file. Amazon Transcribe changes items in the transcript and does not modify the source audio.

- PII redaction** [Info](#)
Label the type of PII and also mask the content with the PII entity type in the transcription output. For example, (123) 456-7890 will be masked as [PHONE].

- Une fois que vous avez sélectionné Expurgation des PII, vous pouvez sélectionner tous les types de PII que vous souhaitez expurger. Vous pouvez également choisir d'avoir une transcription non expurgée si vous sélectionnez la zone Inclure la transcription non expurgée dans la sortie de la tâche.

Content removal

Content removal conceals information in the resulting transcript from your source audio file. Amazon Transcribe changes items in the transcript and does not modify the source audio.

PII redaction [Info](#)
 Label the type of PII and also mask the content with the PII entity type in the transcription output. For example, (123) 456-7890 will be masked as [PHONE].

Include unredacted transcript in job output
 Returns unredacted version of the transcript in addition to the redacted version.

Select PII entity types (11 of 11 selected)

Select All

Financial (6 of 6 selected)

<input checked="" type="checkbox"/> BANK_ACCOUNT_NUMBER	<input checked="" type="checkbox"/> BANK_ROUTING	<input checked="" type="checkbox"/> CREDIT_DEBIT_NUMBER
<input checked="" type="checkbox"/> CREDIT_DEBIT_CVV	<input checked="" type="checkbox"/> CREDIT_DEBIT_EXPIRY	<input checked="" type="checkbox"/> PIN

Personal (5 of 5 selected)

<input checked="" type="checkbox"/> NAME	<input checked="" type="checkbox"/> ADDRESS	<input checked="" type="checkbox"/> PHONE
<input checked="" type="checkbox"/> EMAIL	<input checked="" type="checkbox"/> SSN	

Vocabulary filtering [Info](#)
 Vocabulary filtering can remove, mask or tag specified words in the final transcript.

5. Sélectionnez Créer une tâche pour exécuter votre tâche de transcription.

AWS CLI

Cet exemple utilise la [start-transcription-job](#) commande et le `content-redaction` paramètre. Pour plus d'informations, consultez [StartTranscriptionJob](#) et [ContentRedaction](#).

```
aws transcribe start-transcription-job \
--region us-west-2 \
--transcription-job-name my-first-transcription-job \
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \
--output-bucket-name DOC-EXAMPLE-BUCKET \
--output-key my-output-files/ \
--language-code en-US \
```

```
--content-redaction
RedactionType=PII,RedactionOutput=redacted,PiiEntityTypes=NAME,ADDRESS,BANK_ACCOUNT_NUMBER
```

Voici un autre exemple d'utilisation de [start-transcription-job](#) cette méthode, et le corps de la demande expurge les informations personnelles pour cette tâche.

```
aws transcribe start-transcription-job \
--region us-west-2 \
--cli-input-json file://filepath/my-first-redaction-job.json
```

Le fichier `my-first-redaction-job.json` contient le corps de requête suivant.

```
{
  "TranscriptionJobName": "my-first-transcription-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
  },
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "OutputKey": "my-output-files/",
  "LanguageCode": "en-US",
  "ContentRedaction": {
    "RedactionOutput": "redacted",
    "RedactionType": "PII",
    "PiiEntityTypes": [
      "NAME",
      "ADDRESS",
      "BANK_ACCOUNT_NUMBER"
    ]
  }
}
```

AWS SDK for Python (Boto3)

Cet exemple utilise le AWS SDK for Python (Boto3) pour supprimer du contenu à l'aide de l'`ContentRedaction` argument de la méthode [start_transcription_job](#). Pour plus d'informations, consultez [StartTranscriptionJob](#) et [ContentRedaction](#).

Pour d'autres exemples d'utilisation AWS des SDK, notamment des exemples spécifiques aux fonctionnalités, des scénarios et des exemples multiservices, reportez-vous au chapitre. [Exemples de code pour Amazon Transcribe à l'aide de kits de développement logiciel AWS](#)

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
transcribe.start_transcription_job(
    TranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'my-output-files/',
    LanguageCode = 'en-US',
    ContentRedaction = {
        'RedactionOutput': 'redacted',
        'RedactionType': 'PII',
        'PiiEntityTypes': [
            'NAME', 'ADDRESS', 'BANK_ACCOUNT_NUMBER'
        ]
    }
)

while True:
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)
    if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Note

La rédaction d'informations personnelles pour les tâches par lots n'est prise en charge que dans les pays suivants Régions AWS : Asie-Pacifique (Hong Kong), Asie-Pacifique (Mumbai), Asie-Pacifique (Séoul), Asie-Pacifique (Sydney), Asie-Pacifique (Tokyo), (États-Unis ouest), Canada GovCloud (centre), UE (Francfort), UE (Irlande), UE (Londres), UE (Paris), Moyen-Orient (Bahreïn), Amérique du Sud (Sao Paulo), États-Unis Est (Virginie du Nord), USA Est (Ohio), USA Ouest (Oregon) et USA Ouest (Californie du Nord).

Expurgation ou identification des données d'identification personnelle (PII) dans un flux en temps réel

Lorsque vous expurgez des données d'identification personnelle (PII) d'une transcription en streaming, Amazon Transcribe remplace chaque instance identifiée de PII par [PII] dans votre transcription.

L'identification des PII est une autre option disponible pour les transcriptions en streaming. Lorsque vous activez l'identification des informations personnelles, les Amazon Transcribe étiquette dans vos résultats de transcription sous un `Entities` objet. Pour un exemple de sortie, consultez les sections [Exemple de sortie de streaming expurgée](#) et [Exemple de sortie d'identification PII](#).

La rédaction et l'identification des informations personnelles à l'aide de transcriptions en streaming sont disponibles dans les dialectes anglais suivants : australien (en-AU), britannique (), américain (en-GB) et espagnol (dialecte américainen-US) (). es-US

L'identification et l'expurgation des PII pour les tâches en streaming ne sont effectuées qu'après transcription complète des segments audio.

Les types de données personnelles que Amazon Transcribe vous pouvez reconnaître pour les transcriptions en streaming

Type de PII	Description
ADDRESS	Adresse physique, telle que 100 Main Street, Anytown, États-Unis ou Suite n°12, bâtiment 123. Une adresse peut inclure une rue, un bâtiment, un emplacement, une ville, un état, un pays, un comté, un code postal, une circonscription, un quartier, etc.
ALL	Expurgez ou identifiez tous les types de PII répertoriés dans ce tableau.
BANK_ACCOUNT_NUMBER	Numéro du compte bancaire aux États-Unis. Il s'agit généralement d'un numéro composé de 10 à 12 chiffres, mais Amazon Transcribe reconnaît également les numéros de compte

Type de PII	Description
	bancaire lorsque seuls les 4 derniers chiffres sont présents.
BANK_ROUTING	Numéro d'acheminement du compte bancaire aux États-Unis. Il s'agit généralement d'un numéro à 9 chiffres, mais Amazon Transcribe reconnaît également les numéros de routage lorsque seuls les 4 derniers chiffres sont présents.
CREDIT_DEBIT_CVV	Code de vérification de carte (CVV) à 3 chiffres présent sur les cartes de crédit et de débit VISA et Discover. MasterCard Sur les cartes de crédit ou de débit American Express, il s'agit d'un code numérique à 4 chiffres.
CREDIT_DEBIT_EXPIRY	Date d'expiration d'une carte de crédit ou de débit. Ce numéro comporte généralement 4 chiffres et est au format mois/année ou MM/AA. Par exemple, Amazon Transcribe peut reconnaître des dates d'expiration telles que le 21/01, le 01/2021 et le 1er janvier 2021.
CREDIT_DEBIT_NUMBER	Numéro de carte de crédit ou de débit. La longueur de ces numéros peut varier de 13 à 16 chiffres, mais ils reconnaissent Amazon Transcribe également les numéros de carte de crédit ou de débit lorsque seuls les 4 derniers chiffres sont présents.
EMAIL	Adresse e-mail, telle que efua.owusu@email.com.

Type de PII	Description
NAME	Nom d'une personne. Ce type d'entité n'inclut pas les titres, tels que M., Mme, Miss ou Dr. Il Amazon Transcribe n'applique pas ce type d'entité aux noms qui font partie d'organisations ou d'adresses. Par exemple, Amazon Transcribe reconnaît l'organisation John Doe en tant qu'organisation et Jane Doe Street en tant qu'adresse.
PHONE	Numéro de téléphone. Ce type d'entité inclut également les numéros de télécopieur et de pager.
PIN	Numéro d'identification personnel (PIN) à 4 chiffres qui permet à une personne d'accéder aux informations de son compte bancaire.
SSN	Un numéro de sécurité sociale (SSN) est un numéro à 9 chiffres délivré aux citoyens américains, aux résidents permanents et aux résidents actifs temporaires. Amazon Transcribe reconnaît également les numéros de sécurité sociale lorsque seuls les 4 derniers chiffres sont présents.

Vous pouvez démarrer une transcription en streaming en utilisant le AWS Management Console WebSocket, ou HTTP/2.

AWS Management Console

1. Connectez-vous au [AWS Management Console](#).
2. Dans le panneau de navigation, choisissez Transcription en temps réel. Faites défiler jusqu'à Paramètres de suppression de contenu et développez ce champ s'il est réduit.

Transcription [Download full transcript](#) [Start streaming](#)

Transcription output Current language: English, US

Choose **Start streaming** to begin a real-time transcription of what you speak into your microphone

00:00 of 15:00 min audio stream

- ▶ Language settings
- ▶ Audio settings
- ▶ Content removal settings
- ▶ Customizations

3. Activez l'identification et l'expurgation des PII.

▼ Content removal settings

Vocabulary filtering [Info](#)
Vocabulary filtering removes, masks, or tags words that you specify in your vocabulary filter. Choose a vocabulary filter to see an example.

PII Identification & redaction [Info](#)
Identify or redact one or more types of personally identifiable information (PII) in your transcript

4. Sélectionnez Identification uniquement ou Identification et expurgation, puis sélectionnez les types d'entités de PII que vous souhaitez identifier ou expurger dans votre transcription.

▼ **Content removal settings**

Vocabulary filtering [Info](#)
Vocabulary filtering removes, masks, or tags words that you specify in your vocabulary filter. Choose a vocabulary filter to see an example.

PII Identification & redaction [Info](#)
Identify or redact one or more types of personally identifiable information (PII) in your transcript

Select PII detection type

Identification only
Label the type of PII identified but not redact it in the transcription output

Identification & redaction
Label the type of PII and also mask the content with the PII entity type in the transcription output. For example, (123)456-7890 will be masked as [PHONE]

Select PII entity types (22 of 22 selected)

Select All

Financial (6 of 6 selected)

<input checked="" type="checkbox"/> BANK_ACCOUNT_NUMBER	<input checked="" type="checkbox"/> BANK_ROUTING	<input checked="" type="checkbox"/> CREDIT_DEBIT_NUMBER
<input checked="" type="checkbox"/> CREDIT_DEBIT_CVV	<input checked="" type="checkbox"/> CREDIT_DEBIT_EXPIRY	<input checked="" type="checkbox"/> PIN

Personal (8 of 8 selected)

<input checked="" type="checkbox"/> NAME	<input checked="" type="checkbox"/> ADDRESS	<input checked="" type="checkbox"/> PHONE
<input checked="" type="checkbox"/> EMAIL	<input checked="" type="checkbox"/> SSN	<input checked="" type="checkbox"/> PASSPORT_NUMBER
<input checked="" type="checkbox"/> DRIVER_ID	<input checked="" type="checkbox"/> AGE	

Digital footprint (7 of 7 selected)

<input checked="" type="checkbox"/> URL	<input checked="" type="checkbox"/> USERNAME	<input checked="" type="checkbox"/> PASSWORD
<input checked="" type="checkbox"/> AWS_ACCESS_KEY	<input checked="" type="checkbox"/> AWS_SECRET_KEY	<input checked="" type="checkbox"/> IP_ADDRESS
<input checked="" type="checkbox"/> MAC_ADDRESS		

Other (1 of 1 selected)

DATE_TIME

► **Customizations**

- Vous êtes prêt à transcrire votre flux. Sélectionnez Démarrer le streaming et commencez à parler. Pour mettre fin à votre dictée, sélectionnez Arrêter le streaming.

WebSocket flux

Cet exemple crée une URL présignée qui utilise la redaction des informations personnelles (ou identification des informations personnelles) dans un flux. WebSocket Les sauts de ligne ont été ajoutés pour faciliter la lecture. Pour plus d'informations sur l'utilisation WebSocket des flux avec Amazon Transcribe, consultez [Configuration d'un WebSocket stream](#). Pour plus de détails sur les paramètres, consultez la section [StartStreamTranscription](#).

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/stream-transcription-
websocket?
&X-Amz-Algorithm=AWS4-HMAC-SHA256
```

```
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-
west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=string
&X-Amz-SignedHeaders=content-type%3Bhost%3Bx-amz-date
&language-code=en-US
&media-encoding=flac
&sample-rate=16000
&pii-entity-types=NAME,ADDRESS
&content-redaction-type=PII (or &content-identification-type=PII)
```

Vous ne pouvez pas utiliser à la fois `content-identification-type` et `content-redaction-type` dans la même demande.

Les définitions des paramètres se trouvent dans la [référence d'API](#) ; les paramètres communs à toutes les opérations d' AWS API sont répertoriés dans la section [Paramètres communs](#).

Flux HTTP/2

Cet exemple crée une requête HTTP/2 avec l'identification ou l'expurgation des PII activée. Pour plus d'informations sur l'utilisation du streaming HTTP/2 avec Amazon Transcribe, consultez [Configuration d'un flux HTTP/2](#). Pour plus de détails sur les paramètres et les en-têtes spécifiques à Amazon Transcribe, voir [StartStreamTranscription](#).

```
POST /stream-transcription HTTP/2
host: transcribestreaming.us-west-2.amazonaws.com
X-Amz-Target: com.amazonaws.transcribe.Transcribe.StartStreamTranscription
Content-Type: application/vnd.amazon.eventstream
X-Amz-Content-Sha256: string
X-Amz-Date: 20220208T235959Z
Authorization: AWS4-HMAC-SHA256 Credential=access-key/20220208/us-west-2/transcribe/
aws4_request, SignedHeaders=content-type;host;x-amz-content-sha256;x-amz-date;x-amz-
target;x-amz-security-token, Signature=string
x-amzn-transcribe-language-code: en-US
x-amzn-transcribe-media-encoding: flac
x-amzn-transcribe-sample-rate: 16000
x-amzn-transcribe-content-identification-type: PII (or x-amzn-transcribe-content-
redaction-type: PII)
x-amzn-transcribe-pii-entity-types: NAME,ADDRESS
transfer-encoding: chunked
```

Vous ne pouvez pas utiliser à la fois `content-identification-type` et `content-redaction-type` dans la même demande.

Les définitions des paramètres se trouvent dans la [référence d'API](#) ; les paramètres communs à toutes les opérations d' AWS API sont répertoriés dans la section [Paramètres communs](#).

Note

La rédaction d'informations personnelles pour le streaming n'est prise en charge que dans les pays suivants Régions AWS : Asie-Pacifique (Séoul), Asie-Pacifique (Sydney), Asie-Pacifique (Tokyo), Canada (centre), UE (Francfort), UE (Irlande), UE (Londres), États-Unis est (Virginie du Nord), États-Unis est (Ohio) et États-Unis ouest (Oregon).

Exemple de sortie de rédaction et d'identification d'informations personnelles

Les exemples suivants montrent la sortie expurgée de tâches par lots et de streaming, ainsi que l'identification des informations personnelles issues d'une tâche de streaming.

Les tâches de transcription utilisant la rédaction de contenu génèrent deux types de confiance valeurs. La confiance ASR (Automatic Speech Recognition) indique les éléments avec le type `type_punctuation` ou `pronunciation`, ou qui correspondent à un énoncé spécifique. Dans la sortie de transcription suivante, le mot `Good` possède une confiance de `1.0`. Cette valeur de confiance indique qu'Amazon Transcribe est sûr à 100 % que le mot prononcé dans cette transcription est « Bien ». La valeur de confiance d'une balise `[PII]` indique le niveau de confiance selon lequel le discours marqué pour la rédaction correspond vraiment à des informations personnelles identifiables. Dans la sortie de transcription suivante, la confiance de `0.9999` indique qu'il Amazon Transcribe est certain à 99,99 % que l'entité qu'il a expurgée dans la transcription est une PII.

Exemple de sortie rédigée (batch)

```
{
  "jobName": "my-first-transcription-job",
  "accountId": "111122223333",
  "isRedacted": true,
  "results": {
    "transcripts": [
      {
```

```

    "transcript": "Good morning, everybody. My name is [PII], and today I
feel like
my Social
is [PII].
personal
    "transcript": "Good morning, everybody. My name is [PII], and today I
    sharing a whole lot of personal information with you. Let's start with
    Security number [PII]. My credit card number is [PII] and my C V V code
    I hope that Amazon Transcribe is doing a good job at redacting that
    information away. Let's check."
    }
  ],
  "items": [
    {
      "start_time": "2.86",
      "end_time": "3.35",
      "alternatives": [
        {
          "confidence": "1.0",
          "content": "Good"
        }
      ],
      "type": "pronunciation"
    },
    Items removed for brevity
    {
      "start_time": "5.56",
      "end_time": "6.25",
      "alternatives": [
        {
          "content": "[PII]",
          "redactions": [
            {
              "confidence": "0.9999",
              "type": "NAME",
              "category": "PII"
            }
          ]
        }
      ],
      "type": "pronunciation"
    },
    Items removed for brevity
  ],
},
},

```

```
"status": "COMPLETED"
}
```

Voici la transcription non expurgée à des fins de comparaison :

```
{
  "jobName": "job id",
  "accountId": "111122223333",
  "isRedacted": false,
  "results": {
    "transcripts": [
      {
        "transcript": "Good morning, everybody. My name is Mike, and today I
feel like
my Social
job
at redacting that personal information away. Let's check."
      }
    ],
    "items": [
      {
        "start_time": "2.86",
        "end_time": "3.35",
        "alternatives": [
          {
            "confidence": "1.0",
            "content": "Good"
          }
        ],
        "type": "pronunciation"
      },
      Items removed for brevity
      {
        "start_time": "5.56",
        "end_time": "6.25",
        "alternatives": [
          {
            "confidence": "0.9999",
            "content": "Mike",
          }
        ]
      }
    ]
  }
}
```

```

        ],
        "type": "pronunciation"
    },
    Items removed for brevity
],
},
"status": "COMPLETED"
}

```

Exemple de sortie de streaming expurgée

```

{
  "TranscriptResultStream": {
    "TranscriptEvent": {
      "Transcript": {
        "Results": [
          {
            "Alternatives": [
              {
                "Transcript": "my name is [NAME]",
                "Items": [
                  {
                    "Content": "my",
                    "EndTime": 0.3799375,
                    "StartTime": 0.0299375,
                    "Type": "pronunciation"
                  },
                  {
                    "Content": "name",
                    "EndTime": 0.5899375,
                    "StartTime": 0.3899375,
                    "Type": "pronunciation"
                  },
                  {
                    "Content": "is",
                    "EndTime": 0.7899375,
                    "StartTime": 0.5999375,
                    "Type": "pronunciation"
                  },
                  {
                    "Content": "[NAME]",
                    "EndTime": 1.0199375,
                    "StartTime": 0.7999375,

```



```
        "Content": "my",
        "EndTime": 0.3799375,
        "StartTime": 0.0299375,
        "Type": "pronunciation"
    },
    {
        "Content": "name",
        "EndTime": 0.5899375,
        "StartTime": 0.3899375,
        "Type": "pronunciation"
    },
    {
        "Content": "is",
        "EndTime": 0.7899375,
        "StartTime": 0.5999375,
        "Type": "pronunciation"
    },
    {
        "Content": "mike",
        "EndTime": 0.9199375,
        "StartTime": 0.7999375,
        "Type": "pronunciation"
    }
],
"Entities": [
    {
        "Content": "mike",
        "Category": "PII",
        "Type": "NAME",
        "StartTime" : 0.7999375,
        "EndTime" : 1.0199375,
        "Confidence": 0.9989
    }
]
},
"EndTime": 1.02,
"IsPartial": false,
"ResultId": "12345a67-8bc9-0de1-2f34-a5b678c90d12",
"StartTime": 0.0199375
}
]
```

```
}  
  }  
}
```

sous-titrage

Amazon Transcribe prend en charge les sorties WebVTT (*.vtt) et SubRip (*.srt) à utiliser comme sous-titres vidéo. Vous pouvez sélectionner un ou les deux types de fichiers lors de la configuration de votre tâche de transcription vidéo par lots. Lorsque vous utilisez la fonction de sous-titres, les fichiers de sous-titres que vous avez sélectionnés et un fichier de transcription normal (contenant des informations supplémentaires) sont produits. Les fichiers de sous-titres et de transcription sont envoyés vers la même destination.

Les sous-titres s'affichent en même temps que le texte est prononcé et restent visibles jusqu'à ce qu'il y ait une pause naturelle ou que l'orateur finisse de parler. Notez que si vous activez les sous-titres dans votre demande de transcription et que votre fichier audio ne contient aucune parole, aucun fichier de sous-titres n'est créé.

Important

Amazon Transcribe utilise un index de départ par défaut de 0 pour la sortie des sous-titres, qui diffère de la valeur la plus couramment utilisée de 1. Si vous avez besoin d'un index de départ de 1, vous pouvez le spécifier dans le AWS Management Console ou dans votre demande d'API à l'aide du [OutputStartIndex](#) paramètre.

L'utilisation d'un index de départ incorrect peut entraîner des erreurs de compatibilité avec d'autres services. Assurez-vous donc de vérifier l'index de départ dont vous avez besoin avant de créer vos sous-titres. Si vous ne savez pas quelle valeur utiliser, nous vous recommandons de choisir 1. Reportez-vous à [Subtitles](#) pour plus d'informations.

Fonctionnalités prises en charge avec les sous-titres :

- Rédaction du contenu : tout contenu expurgé est reflété sous la forme « PII » à la fois dans vos fichiers de sortie de sous-titres et de transcription standard. Le son n'est pas modifié.
- Filtres de vocabulaire : les fichiers de sous-titres sont générés à partir du fichier de transcription, de sorte que tous les mots que vous filtrez dans votre sortie de transcription standard sont également filtrés dans vos sous-titres. Le contenu filtré s'affiche sous forme d'espaces blancs ou *** dans vos fichiers de transcription et de sous-titres. Le son n'est pas modifié.

- Diarisation du locuteur : s'il y a plusieurs locuteurs dans un segment de sous-titre donné, des tirets sont utilisés pour distinguer chaque locuteur. Cela s'applique à la fois au WebVTT et aux SubRip formats ; par exemple :
 - -- Texte prononcé par la personne 1
 - -- Texte prononcé par la personne 2

Les fichiers de sous-titres sont stockés au même Amazon S3 emplacement que la sortie de votre transcription.

Consultez [Amazon Transcribe Video Snacks : Création de sous-titres vidéo sans écrire de code](#) pour découvrir comment créer des sous-titres en vidéo.

Génération de fichiers de sous-titres

Vous pouvez créer des fichiers de sous-titres à l'aide des AWSKits SDK AWS Management Console AWS CLI, ou ; consultez les exemples suivants :

AWS Management Console

1. Connectez-vous à [AWS Management Console](#).
2. Dans le volet de navigation, choisissez Tâches de transcription, puis sélectionnez Créer une tâche (en haut à droite). Cela ouvre la page Spécifier les détails de la tâche. Les options de sous-titres se trouvent dans le panneau Données de sortie.
3. Sélectionnez les formats souhaités pour vos fichiers de sous-titres, puis choisissez une valeur pour votre index de départ. Notez que la Amazon Transcribe valeur par défaut est 0, mais 1 qu'elle est plus largement utilisée. Si vous ne savez pas quelle valeur utiliser, nous vous recommandons de choisir 1, car cela peut améliorer la compatibilité avec d'autres services.

Output data

Output data location type info [Info](#)

- Service-managed S3 bucket
The output will be removed after 90 days when the job expires.
- Customer specified S3 bucket
The output will not be removed from bucket even after the job expires.

Subtitle file format [Info](#)

- SRT (SubRip)
- VTT (WebVTT)

Specify the start index

0

- Renseignez tous les autres champs que vous souhaitez inclure sur la page Spécifier les détails de la tâche, puis sélectionnez Suivant. Cela vous amène à la page Configurer la tâche - optionnelle.
- Sélectionnez Créer une tâche pour exécuter votre tâche de transcription.

AWS CLI

Cet exemple utilise la [start-transcription-job](#) commande et le Subtitles paramètre. Pour plus d'informations, consultez [StartTranscriptionJob](#) et [Subtitles](#).

```
aws transcribe start-transcription-job \
--region us-west-2 \
--transcription-job-name my-first-transcription-job \
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \
--output-bucket-name DOC-EXAMPLE-BUCKET \
--output-key my-output-files/ \
--language-code en-US \
--subtitles Formats=vtt,srt,OutputStartIndex=1
```

Voici un autre exemple utilisant la [start-transcription-job](#) commande et un corps de requête qui ajoute des sous-titres à cette tâche.

```
aws transcribe start-transcription-job \
--region us-west-2 \
```

```
--cli-input-json file://my-first-subtitle-job.json
```

Le fichier my-first-subtitle-job.json contient le corps de requête suivant.

```
{
  "TranscriptionJobName": "my-first-transcription-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
  },
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "OutputKey": "my-output-files/",
  "LanguageCode": "en-US",
  "Subtitles": {
    "Formats": [
      "vtt", "srt"
    ],
    "OutputStartIndex": 1
  }
}
```

AWS SDK for Python (Boto3)

Cet exemple utilise le AWS SDK for Python (Boto3) pour ajouter des sous-titres à l'aide de l'`Subtitles` argument de la méthode [start_transcription_job](#). Pour plus d'informations, consultez [StartTranscriptionJob](#) et [Subtitles](#).

Pour des exemples supplémentaires utilisant les AWS kits SDK, notamment des exemples spécifiques aux fonctionnalités, des scénarios et des exemples interservices, consultez le chapitre [Exemples de code pour Amazon Transcribe à l'aide de kits de développement logiciel AWS](#)

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
transcribe.start_transcription_job(
    TranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
```

```
OutputKey = 'my-output-files/',
LanguageCode = 'en-US',
Subtitles = {
    'Formats': [
        'vtt', 'srt'
    ],
    'OutputStartIndex': 1
}
)

while True:
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)
    if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Analyse de l'audio du centre d'appels avec Call Analytics

Utilisez Amazon Transcribe Call Analytics pour mieux comprendre les interactions client-agent. Call Analytics est spécialement conçu pour le son des centres d'appels et vous fournit automatiquement des données précieuses relatives à chaque appel et à chaque participant. Vous pouvez également affiner les données à des moments précis de l'appel. Par exemple, vous pouvez comparer le sentiment du client dans les premières secondes d'un appel au dernier quart de l'appel pour voir si votre agent a fourni une expérience positive. D'autres exemples de cas d'utilisation sont répertoriés dans la [section suivante](#).

Call Analytics est disponible pour les transcriptions après appel et en temps réel. Si vous transcrivez un fichier situé dans un Amazon S3 bucket, vous effectuez une transcription après un appel. Si vous transcrivez un flux audio, vous effectuez une transcription en temps réel. Ces deux méthodes de transcription proposent des informations et des fonctionnalités Call Analytics différentes. Pour plus d'informations sur chacune de ces méthodes, consultez les sections [Analyse après appel](#) et [Analyse Call Analytics en temps réel](#).

Grâce aux transcriptions Call Analytics en temps réel, vous pouvez également inclure des [analyses après appel](#) dans votre demande. Votre transcription d'analyse post-appel est stockée dans le Amazon S3 compartiment que vous spécifiez dans votre demande. Pour plus d'informations, consultez la section [Analyses après appel avec transcriptions en temps réel](#).

Opérations d'API spécifiques à Call Analytics

Après appel : [CreateCallAnalyticsCategory](#), [DeleteCallAnalyticsCategory](#), [DeleteCallAnalyticsJob](#), [GetCallAnalyticsCategory](#), [GetCallAnalyticsJob](#), [ListCallAnalyticsCategories](#), [ListCallAnalyticsJobs](#), [StartCallAnalyticsJob](#), [UpdateCallAnalyticsCategory](#)

En temps réel : [StartCallAnalyticsStreamTranscription](#), [StartCallAnalyticsStreamTranscriptionWebSocket](#)

Cas d'utilisation courants

Transcriptions après appel :

- Surveiller la fréquence des problèmes au fil du temps : utilisez la [catégorisation des appels](#) pour identifier les mots clés récurrents dans vos transcriptions.

- Obtenir des informations sur l'expérience de votre service client : utilisez les [caractéristiques de l'appel](#) (temps hors conversation, temps de conversation, interruptions, volume de la voix, vitesse de conversation) et l'analyse des sentiments pour déterminer si les problèmes des clients sont résolus de manière appropriée pendant l'appel.
- Garantir la conformité réglementaire ou le respect de la politique de l'entreprise : définissez des [mots clés et des phrases](#) pour les messages d'accueil spécifiques à l'entreprise ou les clauses de non-responsabilité afin de vérifier que vos agents respectent les exigences réglementaires.
- Améliorer le traitement des données personnelles des clients : utilisez l'[expurgation des données d'identification personnelle \(PII\)](#) dans votre sortie de transcription ou votre fichier audio pour protéger la confidentialité des clients.
- Améliorer la formation du personnel : utilisez des critères (sentiment, temps hors conversation, interruptions, vitesse de conversation) pour signaler les transcriptions qui peuvent être utilisées comme exemples d'interactions positives ou négatives avec les clients.
- Mesurer l'efficacité du personnel dans la création d'une expérience client positive : utilisez l'[analyse des sentiments](#) pour déterminer si vos agents sont capables de transformer un sentiment négatif en un sentiment positif à mesure que les appels progressent.
- Améliorer l'organisation des données : étiquetez et trie les appels en fonction de [catégories personnalisées](#) (notamment les mots clés et les phrases, les sentiments, le temps de conversation et les interruptions).
- Résumez les aspects importants d'un appel à l'aide de l'IA générative : utilisez le [résumé génératif des appels](#) pour obtenir un résumé concis de la transcription, qui inclut les éléments clés tels que les problèmes, les éléments d'action et les résultats abordés lors de l'appel.

Transcriptions en temps réel :

- Atténuer les escalades en temps réel : configurez des [alertes en temps réel](#) pour les phrases clés, telles qu'un client disant « parler à un responsable », afin de signaler les appels qui commencent à dégénérer. Vous pouvez créer des alertes en temps réel en utilisant des correspondances de catégories en temps réel.
- Améliorer le traitement des données des clients : utilisez l'[identification des données d'identification personnelle \(PII\)](#) ou l'[expurgation des données d'identification personnelle \(PII\)](#) dans votre sortie de transcription pour protéger la confidentialité des clients.
- Identifier des mots clés et des phrases personnalisés : utilisez [des catégories personnalisées](#) pour signaler des mots clés spécifiques lors d'un appel.

- Identifier automatiquement les problèmes : utilisez la [détection automatique des problèmes](#) pour obtenir un bref résumé de tous les problèmes identifiés lors d'un appel.
- Mesurer l'efficacité du personnel dans la création d'une expérience client positive : utilisez [l'analyse des sentiments](#) pour déterminer si vos agents sont capables de transformer un sentiment négatif en un sentiment positif à mesure que les appels progressent.
- Configurer l'assistance aux agents : utilisez les informations de votre choix pour fournir à vos agents une assistance proactive dans la résolution des appels des clients. Pour plus d'informations, consultez la section [Live Call Analytics and agent assist for your contact center with Amazon language AI services](#).

Pour comparer les fonctionnalités disponibles avec Call Analytics à celles disponibles pour Amazon Transcribe et Amazon Transcribe Medical, reportez-vous au [tableau des fonctionnalités](#).

Pour démarrer, consultez les sections [Démarrage d'une transcription d'analyse après appel](#) et [Démarrage d'une transcription Call Analytics en temps réel](#). Le résultat de Call Analytics est similaire à celui d'une tâche de transcription standard, mais contient des données d'analyse supplémentaires. Pour consulter un exemple de sortie, reportez-vous aux sections [Résultats d'analyse après appel](#) et [Sortie Call Analytics en temps réel](#).

Considérations et informations supplémentaires

Avant d'utiliser Call Analytics, notez les informations suivantes :

- Call Analytics ne prend en charge que l'audio à deux canaux, où un agent est présent sur un canal et un client sur un second canal.
- [Mise en file d'attente des offres d'emploi](#) est toujours activé pour les tâches d'analyse après appel. Vous êtes donc limité à 100 tâches Call Analytics simultanées. Pour demander une augmentation du quota, consultez la section [Service Quotas AWS](#).
- Les fichiers d'entrée pour les tâches d'analyse après appel ne peuvent pas dépasser 500 Mo et doivent durer moins de 4 heures. Notez que la limite de taille de fichier peut être inférieure pour certains formats de fichiers audio compressés non WAV.
- Si vous utilisez des catégories, vous devez créer toutes les catégories souhaitées avant de commencer une transcription Call Analytics. Aucune nouvelle catégorie ne peut être appliquée aux transcriptions existantes. Pour savoir comment créer une nouvelle catégorie, consultez les sections [Création de catégories pour les transcriptions après appel](#) et [Création de catégories pour les transcriptions en temps réel](#).

- Certains quotas d'analyse des appels diffèrent de ceux Amazon Transcribe de Amazon Transcribe Medical ; reportez-vous à la [référence AWS générale](#) pour plus de détails.

i Approfondissez vos connaissances avec le blog AWS Machine Learning

Pour plus d'informations sur les options Call Analytics, consultez les sections suivantes :

- [Post Call Analytics for your contact center with Amazon language AI services](#)
- [Live Call Analytics and agent assist for your contact center with Amazon language AI services](#)

Pour voir des exemples de résultats et de fonctionnalités d'analyse des appels, consultez notre [GitHub démo](#). Nous proposons également une application de [document JSON vers Word](#) pour convertir votre transcription dans un easy-to-read format.

Disponibilité des régions et quotas

Call Analytics est pris en charge dans les domaines suivants Régions AWS :

Région	Type de transcription
ap-northeast-1 (Tokyo)	post-call, real-time
ap-northeast-2 (Séoul)	post-call, real-time
ap-south-1 (Bombay)	post-call
ap-southeast-1 (Singapour)	post-call
ap-southeast-2 (Sydney)	post-call, real-time
ca-central-1, (Canada, Centre)	post-call, real-time
eu-central-1 (Francfort)	post-call, real-time
eu-west-2 (Londres)	post-call, real-time
us-east-1 (Virginie du Nord)	post-call, real-time

Région	Type de transcription
us-west-2 (Oregon)	post-call, real-time

Notez que la prise en charge de la région est différente pour [Amazon Transcribe](#), [Amazon Transcribe Medical](#) et Call Analytics.

Pour connaître les points de terminaison de chaque région prise en charge, consultez la section [Points de terminaison de service](#) dans la Référence générale AWS .

Pour obtenir la liste des quotas relatifs à vos transcriptions, consultez la section [Service Quotas](#) dans la Référence générale AWS . Certains quotas peuvent être modifiés sur demande. Si la colonne Ajustable contient « Oui », vous pouvez demander une augmentation. Pour ce faire, sélectionnez le lien fourni.

Analyse après appel

Call Analytics fournit des analyses après appel qui sont utiles pour suivre les tendances du service client.

Les transcriptions effectuées après appel fournissent les informations suivantes :

- [Caractéristiques des appels](#), notamment le temps de conversation, le temps de non-conversation, le volume du locuteur, les interruptions, la vitesse de conversation, les problèmes, les résultats et les éléments d'action.
- [Résumé génératif des appels](#), qui crée un résumé concis de l'ensemble de l'appel
- [Catégorisation personnalisée](#) avec des règles que vous pouvez utiliser pour affiner des mots clés et des critères spécifiques
- [Expurgation des données d'identification personnelle](#) de votre transcription de texte et de votre fichier audio
- [Sentiment du locuteur](#) à l'égard de chaque appelant à différents moments d'un appel

Informations après appel

Cette section détaille les informations disponibles pour les transcriptions d'analyses après appel.

Caractéristiques des appels

La fonctionnalité des caractéristiques des appels mesure la qualité des interactions agent-client en fonction des critères suivants :

- **Interruption** : mesure si et quand un participant coupe la parole à l'autre participant au milieu d'une phrase. Les interruptions fréquentes peuvent être associées à la grossièreté ou à la colère, et peuvent être corrélées à un sentiment négatif chez l'un des participants ou les deux.
- **Volume sonore** : mesure le volume auquel chaque participant parle. Utilisez cette métrique pour voir si l'appelant ou l'agent parle fort ou crie, ce qui indique souvent qu'il est contrarié. Cette métrique est représentée sous la forme d'une valeur normalisée (niveau de parole par seconde dans un segment donné) sur une échelle de 0 à 100, une valeur plus élevée indiquant une voix plus forte.
- **Temps de non-conversation** : mesure les périodes pendant lesquelles il n'y a pas de parole. Utilisez cette métrique pour voir s'il existe de longues périodes de silence, par exemple si un agent maintient un client en attente pendant une durée excessive.
- **Vitesse de conversation** : mesure la vitesse à laquelle les deux participants parlent. La compréhension peut être affectée si l'un des participants parle trop rapidement. Cette métrique est mesurée en mots par minute.
- **Temps de conversation** : mesure le temps (en millisecondes) pendant lequel chaque participant a parlé pendant l'appel. Utilisez cette métrique pour déterminer si un participant domine l'appel ou si le dialogue est équilibré.
- **Problèmes, résultats et éléments d'action** : identifie les problèmes, les résultats et les éléments d'action à partir de la transcription de l'appel.

Voici un [exemple de sortie](#).

Résumé génératif des appels

La synthèse générative des appels crée un résumé concis de l'ensemble de l'appel, en capturant les éléments clés tels que le motif de l'appel, les mesures prises pour résoudre le problème et les étapes suivantes.

Grâce au résumé génératif des appels, vous pouvez :

- réduire le besoin de prendre des notes manuellement pendant et après les appels ;

- améliorer l'efficacité des agents, car ils peuvent passer plus de temps à parler aux appelants qui attendent dans la file d'attente plutôt qu'à s'occuper du travail après l'appel ;
- accélérer les évaluations des superviseurs, car les récapitulatifs des appels sont beaucoup plus rapides à passer en revue que les transcriptions complètes.

Pour utiliser le résumé génératif des appels avec une tâche d'analyse après appel, consultez [Activer le résumé génératif des appels](#). Pour un exemple de sortie, voir [Exemple de sortie de résumé génératif des appels](#). Le résumé génératif des appels est facturé séparément (veuillez vous reporter à la [page de tarification](#)).

Note

Le résumé génératif des appels est actuellement disponible dans us-east-1 et us-west-2. Cette fonctionnalité est prise en charge par les dialectes anglais suivants : australien (en-AU), britannique (en-GB), indien (en-IN), irlandais (en-IE), écossais (en-AB), américain (en-US) et gallois (en-WL).

Catégorisation personnalisée

Utilisez la catégorisation des appels pour marquer des mots clés, des phrases, des sentiments ou des actions au sein d'un appel. Nos options de catégorisation peuvent vous aider à trier les escalades, telles que les appels à sentiment négatif comportant de nombreuses interruptions, ou à organiser les appels dans des catégories spécifiques, telles que les services de l'entreprise.

Les critères que vous pouvez ajouter à une catégorie sont les suivants :

- Temps de non-conversation : périodes pendant lesquelles ni le client ni l'agent ne parlent.
- Interruptions : lorsque le client ou l'agent interrompt l'autre personne.
- Sentiment du client ou de l'agent : sentiment du client ou de l'agent pendant une période donnée. Si au moins 50 % de la conversation (back-and-forth entre deux interlocuteurs) pendant une période donnée correspond au sentiment spécifié, Amazon Transcribe considère que le sentiment correspond.
- Mots clés ou phrases : correspond à une partie de la transcription en fonction d'une phrase exacte. Par exemple, si vous définissez un filtre pour la phrase « Je souhaite parler au responsable », Amazon Transcribe filtre cette phrase exacte.

Vous pouvez également signaler l'inverse des critères précédents (temps de conversation, absence d'interruptions, absence d'un sentiment et absence d'une phrase spécifique).

Voici un [exemple de sortie](#).

Pour plus d'informations sur les catégories ou pour savoir comment créer une nouvelle catégorie, consultez la section [Création de catégories pour les transcriptions après appel](#).

Expurgation des données sensibles

L'expurgation des données sensibles remplace les données d'identification personnelle (PII) dans la transcription du texte et le fichier audio. Une transcription expurgée remplace le texte original par [PII] ; un fichier audio expurgé remplace les informations personnelles prononcées par le silence. Ce paramètre est utile pour protéger les informations des clients.

Note

La rédaction des informations personnelles après un appel est prise en charge en anglais américain (en-US) et en espagnol américain (.es-US).

Pour consulter la liste des PII supprimées à l'aide de cette fonctionnalité, ou pour en savoir plus sur l'expurgation avec Amazon Transcribe, consultez la section [La suppression ou l'identification personnelle des données d'identification personnelle](#).

Voici un [exemple de sortie](#).

Analyse de sentiment

L'analyse des sentiments permet d'estimer ce que le client et l'agent ressentent tout au long de l'appel. Cette métrique est représentée à la fois sous la forme d'une valeur quantitative (avec une plage comprise entre 5 et -5) et d'une valeur qualitative (*positive*, *neutral*, *mixed* ou *negative*). Les valeurs quantitatives sont fournies par trimestre et par appel ; les valeurs qualitatives sont fournies par tour.

Cette métrique peut aider à déterminer si votre agent est capable de satisfaire un client mécontent à la fin de l'appel.

L'analyse des sentiments fonctionne out-of-the-box et ne prend donc pas en charge la personnalisation, telle que la formation des modèles ou les catégories personnalisées.

Voici un [exemple de sortie](#).

Création de catégories pour les transcriptions après appel

L'analyse après appel prend en charge la création de catégories personnalisées, ce qui vous permet d'adapter vos analyses de transcriptions aux besoins spécifiques de votre entreprise.

Vous pouvez créer autant de catégories que vous le souhaitez pour couvrir une gamme de scénarios différents. Pour chaque catégorie que vous créez, vous devez créer entre 1 et 20 règles. Chaque règle est basée sur l'un des quatre critères suivants : interruptions, mots clés, temps de non-conversation ou sentiment. Pour plus d'informations sur l'utilisation de ces critères dans le cadre de l'opération [CreateCallAnalyticsCategory](#), consultez la section [Critères de règles pour les catégories d'analyse après appel](#).

Si le contenu de votre média répond à toutes les règles que vous avez spécifiées dans une catégorie donnée, Amazon Transcribe attribue cette catégorie à votre sortie. Pour un exemple de correspondance de catégorie dans la sortie JSON, consultez la [sortie de catégorisation des appels](#).

Voici quelques exemples de ce que vous pouvez faire avec des catégories personnalisées :

- isoler les appels présentant des caractéristiques spécifiques, tels que les appels qui se terminent par un sentiment négatif du client ;
- identifier les tendances relatives aux problèmes des clients en signalant et en suivant des ensembles de mots clés spécifiques ;
- surveiller la conformité, par exemple si un agent prononce (ou omet) une phrase spécifique pendant les premières secondes d'un appel ;
- obtenir des informations sur l'expérience client en signalant les appels présentant de nombreuses interruptions de l'agent et un sentiment négatif de la part des clients ;
- comparer plusieurs catégories pour mesurer les corrélations, par exemple en analysant si l'utilisation d'une phrase de bienvenue par un agent est en corrélation avec le sentiment positif de la part du client.

Catégories après appel et catégories en temps réel

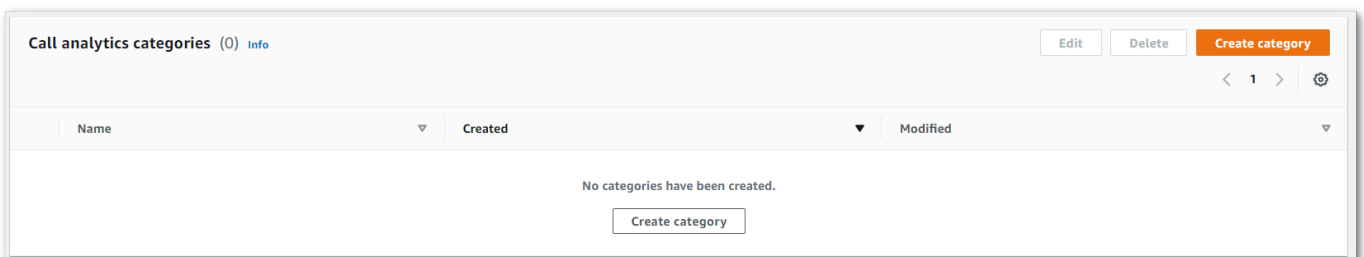
Lorsque vous créez une nouvelle catégorie, vous pouvez spécifier si vous souhaitez qu'elle soit créée en tant que catégorie d'analyse après appel (POST_CALL) ou en tant que catégorie Call Analytics en temps réel (REAL_TIME). Si vous ne spécifiez aucune option, votre catégorie est créée par défaut

en tant que catégorie après appel. Les correspondances de la catégories d'analyse après appel sont disponibles dans votre sortie une fois la transcription de votre analyse après appel terminée.

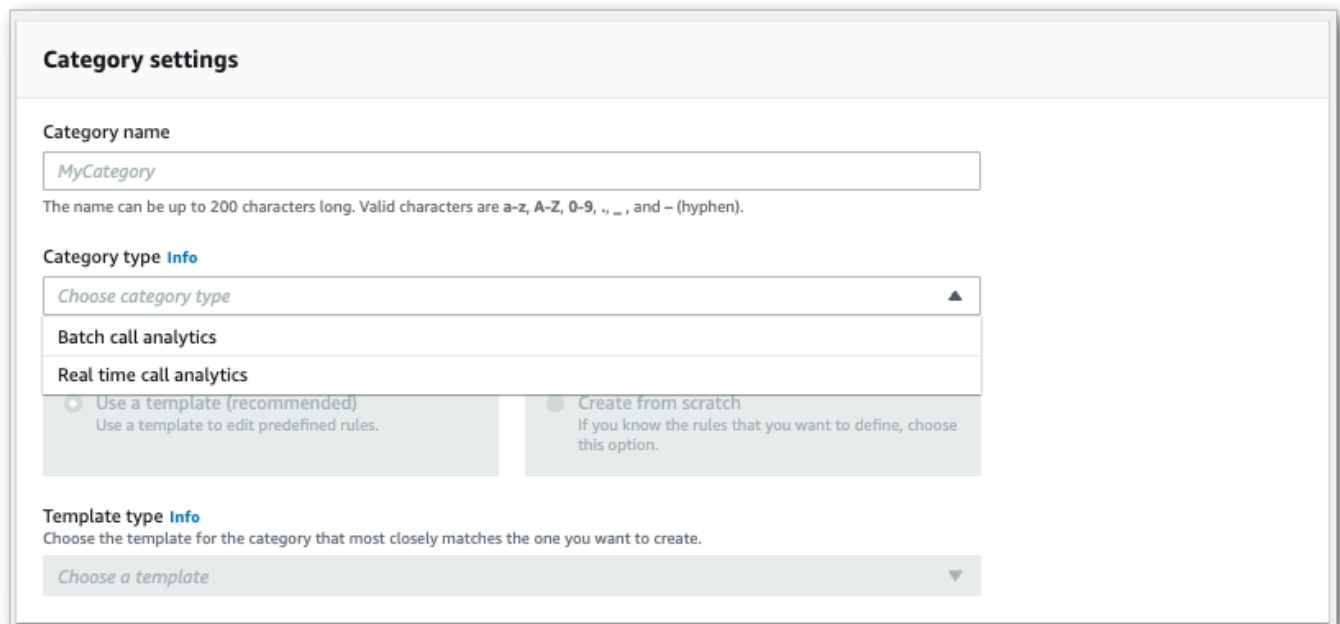
Pour créer une nouvelle catégorie pour les analyses après appel, vous pouvez utiliser la AWS Management Console, l'AWS CLI ou les kits SDK AWS . Consultez les exemples suivants :

AWS Management Console

1. Dans le volet de navigation, sous Amazon Transcribe, choisissez Amazon Transcribe Call Analytics.
2. Choisissez Catégories Call Analytics pour accéder à la page Catégories Call Analytics. Sélectionnez Créer une catégorie.



3. Vous êtes maintenant sur la page Créer une catégorie. Entrez un nom pour votre catégorie, puis choisissez « Analyse des appels par lots » dans le menu déroulant Type de catégorie.

The screenshot shows the 'Category settings' form. It has three main sections: 'Category name' with a text input field containing 'MyCategory' and a note that the name can be up to 200 characters long; 'Category type' with a dropdown menu showing 'Batch call analytics' selected; and 'Template type' with two radio button options: 'Use a template (recommended)' (which is selected) and 'Create from scratch'. There is also a 'Template type' dropdown menu at the bottom with 'Choose a template' selected.

4. Vous pouvez choisir un modèle pour créer votre catégorie ou en créer une de zéro.

Si vous utilisez un modèle : sélectionnez Utiliser un modèle (recommandé), choisissez le modèle souhaité, puis sélectionnez Créer une catégorie.

Category settings

Category name
MyCategory
The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, -, ., _ , and - (hyphen).

Category type [Info](#)
Batch call analytics ▼

Category creation method [Info](#)

Use a template (recommended)
Use a template to edit predefined rules.

Create from scratch
If you know the rules that you want to define, choose this option.

Template type [Info](#)
Choose the template for the category that most closely matches the one you want to create.

Choose a template ▲

- Non-talk time exceeds 5 minutes for the whole call
- Customer sentiment is negative for the last 5 minutes of the call
- Agent spoke over the customer more than 15 seconds for the entire call

ALL THE GIVE CONDITIONS MUST BE MET FOR A REPRESENTATIVE JOB TO BE ASSIGNED TO THE CATEGORY.

5. Si vous créez une catégorie personnalisée : sélectionnez Créer à partir de zéro.

Create category [Info](#)

Category settings

Category name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, ., _ , and - (hyphen).

Category creation method [Info](#)

Use a template (recommended)
Use a template to edit predefined rules.

Create from scratch
If you know the rules that you want to define, choose this option.

Rules

All the rule conditions must be met for a transcription job to be classified in this category.

▼ Rule 1 Delete rule

Rule type [Info](#)
Choose the rule that you want to define.

Add rule

You can add up to 19 more rules.

6. Ajoutez des règles à votre catégorie à l'aide du menu déroulant. Vous pouvez ajouter jusqu'à 20 règles par catégorie.

The screenshot shows the 'Rules' configuration page in Amazon Transcribe. At the top, it states: 'All the rule conditions must be met for a transcription job to be classified in this category.' Below this, there is a section for 'Rule 1' with a 'Delete rule' button. The rule's condition is: 'When no word has been spoken for more than 5 minute(s) during the entire call.' Underneath, the 'Rule type' is set to 'Non-talk time', with an 'Info' link. A dropdown menu is open, showing four options: 'Non-talk time' (selected), 'Interruption time', 'Transcript content match', and 'Transcript sentiment match'. Each option has a brief description of when the rule is triggered. At the bottom of the dropdown is an 'Add rule' button. A note at the very bottom says: 'You can add up to 19 more rules.'

7. Voici un exemple de catégorie comportant deux règles : un agent qui interrompt un client pendant plus de 15 secondes pendant l'appel et un sentiment négatif ressenti par le client ou l'agent au cours des deux dernières minutes de l'appel.

Rules

All the rule conditions must be met for a transcription job to be classified in this category.

▼ **Rule 1** Delete rule

When the duration of the interruption was more than 15 second(s) during the entire call when the speaker was agent.

Rule type [Info](#)
Choose the rule that you want to define.

Interruption time ▼

Logic [Info](#)
Define the conditions that must be met.

When the duration of the interruption was more than ▼

during the ▼

when the speaker was ▼

AND

▼ **Rule 2** Delete rule

When the sentiment is negative during the last 2 minute(s) when the speaker was either.

Rule type [Info](#)
Choose the rule that you want to define.

Transcript sentiment match ▼

Logic [Info](#)
Define the conditions that must be met.

When the sentiment is ▼

during the ▼ ▼

when the speaker was ▼

You can add up to 18 more rules.

8. Lorsque vous avez fini d'ajouter des règles à votre catégorie, choisissez Créer une catégorie.

AWS CLI

Cet exemple utilise la commande [create-call-analytics-category](#). Pour plus d'informations, consultez [CreateCallAnalyticsCategory](#), [CategoryProperties](#) et [Rule](#).

L'exemple suivant crée une catégorie avec les règles :

- Le client a été interrompu dans les 60 000 premières millisecondes. La durée de ces interruptions a été d'au moins 10 000 millisecondes.
- Il y a eu une période de silence d'au moins 20 000 millisecondes entre 10 % et 80 % de l'appel.
- L'agent a eu un sentiment négatif à un moment donné au cours de l'appel.
- Les mots « bienvenue » ou « bonjour » n'ont pas été utilisés pendant les 10 000 premières millisecondes de l'appel.

Cet exemple utilise la commande [create-call-analytics-category](#) et un corps de requête qui ajoute plusieurs règles à votre catégorie.

```
aws transcribe create-call-analytics-category \  
--cli-input-json file://filepath/my-first-analytics-category.json
```

Le fichier `my-first-analytics-category.json` contient le corps de requête suivant.

```
{  
  "CategoryName": "my-new-category",  
  "InputType": "POST_CALL",  
  "Rules": [  
    {  
      "InterruptionFilter": {  
        "AbsoluteTimeRange": {  
          "First": 60000  
        },  
        "Negate": false,  
        "ParticipantRole": "CUSTOMER",  
        "Threshold": 10000  
      }  
    },  
    {  
      "NonTalkTimeFilter": {  
        "Negate": false,  
        "RelativeTimeRange": {  
          "EndPercentage": 80,  
          "StartPercentage": 10  
        },  
        "Threshold": 20000  
      }  
    },  
    {
```

```

    "SentimentFilter": {
      "ParticipantRole": "AGENT",
      "Sentiments": [
        "NEGATIVE"
      ]
    },
    {
      "TranscriptFilter": {
        "Negate": true,
        "AbsoluteTimeRange": {
          "First": 10000
        },
        "Targets": [
          "welcome",
          "hello"
        ],
        "TranscriptFilterType": "EXACT"
      }
    }
  ]
}

```

AWS SDK for Python (Boto3)

Cet exemple utilise le AWS SDK for Python (Boto3) pour créer une catégorie à l'aide des Rules arguments `CategoryName` et de la méthode [create_call_analytics_category](#). Pour plus d'informations, consultez [CreateCallAnalyticsCategory](#), [CategoryProperties](#) et [Rule](#).

Pour d'autres exemples d'utilisation AWS des SDK, notamment des exemples spécifiques aux fonctionnalités, des scénarios et des exemples multiservices, reportez-vous au chapitre. [Exemples de code pour Amazon Transcribe à l'aide de kits de développement logiciel AWS](#)

L'exemple suivant crée une catégorie avec les règles :

- Le client a été interrompu dans les 60 000 premières millisecondes. La durée de ces interruptions a été d'au moins 10 000 millisecondes.
- Il y a eu une période de silence d'au moins 20 000 millisecondes entre 10 % et 80 % de l'appel.
- L'agent a eu un sentiment négatif à un moment donné au cours de l'appel.
- Les mots « bienvenue » ou « bonjour » n'ont pas été utilisés pendant les 10 000 premières millisecondes de l'appel.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
category_name = "my-new-category"
transcribe.create_call_analytics_category(
    CategoryName = category_name,
    InputType = POST_CALL,
    Rules = [
        {
            'InterruptionFilter': {
                'AbsoluteTimeRange': {
                    'First': 60000
                },
                'Negate': False,
                'ParticipantRole': 'CUSTOMER',
                'Threshold': 10000
            }
        },
        {
            'NonTalkTimeFilter': {
                'Negate': False,
                'RelativeTimeRange': {
                    'EndPercentage': 80,
                    'StartPercentage': 10
                },
                'Threshold': 20000
            }
        },
        {
            'SentimentFilter': {
                'ParticipantRole': 'AGENT',
                'Sentiments': [
                    'NEGATIVE'
                ]
            }
        },
        {
            'TranscriptFilter': {
                'Negate': True,
                'AbsoluteTimeRange': {
                    'First': 10000
                },
            },
        }
    ]
)
```



```

        'Targets': [
            'welcome',
            'hello'
        ],
        'TranscriptFilterType': 'EXACT'
    }
}
]
)

result = transcribe.get_call_analytics_category(CategoryName = category_name)
print(result)

```

Critères de règles pour les catégories d'analyse après appel

Cette section décrit les types de règles POST_CALL personnalisées que vous pouvez créer à l'aide de l'opération d'API [CreateCallAnalyticsCategory](#).

Correspondances d'interruptions

Les règles utilisant des interruptions (type de données [InterruptionFilter](#)) sont conçues pour correspondre à :

- des instances dans lesquelles un agent interrompt un client ;
- des instances dans lesquelles un client interrompt un agent ;
- tout participant interrompant l'autre ;
- l'absence d'interruptions.

Voici un exemple des paramètres disponibles avec [InterruptionFilter](#) :

```

"InterruptionFilter": {
  "AbsoluteTimeRange": {
    Specify the time frame, in milliseconds, when the match should occur
  },
  "RelativeTimeRange": {
    Specify the time frame, in percentage, when the match should occur
  },
  "Negate": Specify if you want to match the presence or absence of interruptions,
  "ParticipantRole": Specify if you want to match speech from the agent, the customer, or both,

```

```
"Threshold": Specify a threshold for the amount of time, in seconds, interruptions occurred during the call
},
```

Pour plus d'informations sur ces paramètres et les valeurs valides associées à chacun, consultez les rubriques [CreateCallAnalyticsCategory](#) et [InterruptionFilter](#).

Correspondances de mots clés

Les règles utilisant des mots clés (type de données [TranscriptFilter](#)) sont conçues pour correspondre à :

- des mots ou phrases personnalisés prononcés par l'agent, le client ou les deux ;
- des mots ou phrases personnalisés qui ne sont pas prononcés par l'agent, le client ou les deux ;
- des mots ou phrases personnalisés qui apparaissent dans un laps de temps spécifique.

Voici un exemple des paramètres disponibles avec [TranscriptFilter](#) :

```
"TranscriptFilter": {
  "AbsoluteTimeRange": {
    Specify the time frame, in milliseconds, when the match should occur
  },
  "RelativeTimeRange": {
    Specify the time frame, in percentage, when the match should occur
  },
  "Negate": Specify if you want to match the presence or absence of your custom keywords,
  "ParticipantRole": Specify if you want to match speech from the agent, the customer, or both,
  "Targets": [ The custom words and phrases you want to match ],
  "TranscriptFilterType": Use this parameter to specify an exact match for the specified targets
}
```

Pour plus d'informations sur ces paramètres et les valeurs valides associées à chacun, consultez les rubriques [CreateCallAnalyticsCategory](#) et [TranscriptFilter](#).

Correspondances de temps de non-conversation

Les règles utilisant des temps de non-conversation (type de données [NonTalkTimeFilter](#)) sont conçues pour correspondre à :

- la présence de silence à des moments précis de l'appel ;
- la présence de parole à des moments précis de l'appel.

Voici un exemple des paramètres disponibles avec [NonTalkTimeFilter](#) :

```
"NonTalkTimeFilter": {
  "AbsoluteTimeRange": {
    Specify the time frame, in milliseconds, when the match should occur
  },
  "RelativeTimeRange": {
    Specify the time frame, in percentage, when the match should occur
  },
  "Negate": Specify if you want to match the presence or absence of speech,
  "Threshold": Specify a threshold for the amount of time, in seconds, silence (or speech) occurred during the call
},
```

Pour plus d'informations sur ces paramètres et les valeurs valides associées à chacun, consultez les rubriques [CreateCallAnalyticsCategory](#) et [NonTalkTimeFilter](#).

Correspondances des sentiments

Les règles utilisant des sentiments (type de données [SentimentFilter](#)) sont conçues pour correspondre à :

- la présence ou l'absence d'un sentiment positif exprimé par le client, l'agent ou les deux à des moments précis de l'appel ;
- la présence ou l'absence d'un sentiment négatif exprimé par le client, l'agent ou les deux à des moments précis de l'appel ;
- la présence ou l'absence d'un sentiment neutre exprimé par le client, l'agent ou les deux à des moments précis de l'appel ;
- la présence ou l'absence d'un sentiment mitigé exprimé par le client, l'agent ou les deux à des moments précis de l'appel.

Voici un exemple des paramètres disponibles avec [SentimentFilter](#) :

```
"SentimentFilter": {
  "AbsoluteTimeRange": {
```

```
Specify the time frame, in milliseconds, when the match should occur
},
"RelativeTimeRange": {
Specify the time frame, in percentage, when the match should occur
},
"Negate": Specify if you want to match the presence or absence of your chosen
sentiment,
"ParticipantRole": Specify if you want to match speech from the agent, the
customer, or both,
"Sentiments": [ The sentiments you want to match ]
},
```

Pour plus d'informations sur ces paramètres et les valeurs valides associées à chacun, consultez les rubriques [CreateCallAnalyticsCategory](#) et [SentimentFilter](#).

Démarrage d'une transcription d'analyse après appel

Avant de commencer une transcription analytique après un appel, vous devez créer toutes les [catégories](#) que vous souhaitez associer Amazon Transcribe à votre audio.

Note

Les transcriptions Call Analytics ne peuvent pas être associées rétroactivement à de nouvelles catégories. Seules les catégories que vous créez avant de commencer une transcription Call Analytics peuvent être appliquées à cette sortie de transcription.

Si vous avez créé une ou plusieurs catégories et que votre fichier audio répond à toutes les règles d'au moins une de vos catégories, Amazon Transcribe indique à votre sortie la catégorie correspondante. Si vous choisissez de ne pas utiliser de catégories, ou si votre contenu audio ne correspond pas aux règles spécifiées dans vos catégories, votre transcription n'est pas signalée.

Pour démarrer une transcription d'analyse après appel, vous pouvez utiliser la AWS Management Console, l'AWS CLI ou les kits SDK AWS . Consultez les exemples suivants :

AWS Management Console

Utilisez la procédure suivante pour démarrer une tâche d'analyse après appel. Les appels qui correspondent à toutes les caractéristiques définies par une catégorie sont étiquetés avec cette catégorie.

1. Dans le volet de navigation, sous Analyse des Amazon Transcribe appels, sélectionnez Tâches d'analyse des appels.
2. Choisissez Créer une tâche.

Configure job - *optional* [Info](#)

Content removal

Content removal conceals information in the resulting transcript from your source audio file. Amazon Transcribe changes items in the transcript and does not modify the source audio.

PII redaction [Info](#)

Label the type of PII and also mask the content with the PII entity type in the transcription output. For example, (123) 456-7890 will be masked as [PHONE].

Vocabulary filtering [Info](#)

Vocabulary filtering can remove, mask or tag specified words in the final transcript.

Customization

Custom vocabulary [Info](#)

A custom vocabulary improves the accuracy of recognizing words and phrases specific to your use case.

Summarization

Generative call summarization [Info](#)

Generative call summarization provides a summary of the transcript, including important components of the conversation.


Categories

Create categories to classify calls. For example, you can create a category for all cancellation requests. When you run an analytics job, Amazon Transcribe applies that category to all calls that request cancellation.

Call analytics categories (1) [Info](#)

< 1 > 

	Name	Type	Created	Modified
<input type="radio"/>	CatchNegativeSentiment	POST_CALL	February 17 2023, 10:43 (UTC-08:00)	February 17 2023, 10:43 (UTC-08:00)

If the above categories aren't relevant to your use case, you can create a new category. [Create a new category.](#) 

3. Sur la page Spécifier les détails de la tâche, fournissez des informations sur votre tâche Call Analytics, notamment l'emplacement de vos données d'entrée.

Specify job details [Info](#)

Job settings

Name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, . (period), _ (underscore), and - (hyphen).

Model type [Info](#)

Choose the type of model to use for the transcription job.

General model
To use a model that is not specialized for a particular use case, choose this option. Configuration options vary between languages.

Custom language model
To use a model that you trained for your specific use case, choose this option. This model has fewer configuration options than the general model.

Language settings

You can transcribe your audio file in a language that you specify or have Amazon Transcribe identify and transcribe it in the predominant language.

Specific language [Info](#)
If you know the language spoken in your source audio, choose this option to get the most accurate results. The options available for additional processing vary between languages.

Automatic language identification [Info](#)
If you don't know the language spoken in your audio files, choose this option. You have access to fewer options for additional processing than if you choose **Specific language**.

Language

Choose the language of the input audio.

Input data [Info](#)

Input file location on S3

Choose an input audio or video file in Amazon S3.

Valid file formats: MP3, MP4, WAV, FLAC, AMR, OGG, and WebM.

Agent audio channel identification [Info](#)

Choose the channel that has the speech from the agent. The other channel is used for the customer's speech.

Spécifiez l' Amazon S3 emplacement souhaité pour vos données de sortie et IAM le rôle à utiliser.

Output data

Output data location type info [Info](#)

Service-managed S3 bucket
The output will be removed after 90 days when the job expires.

Customer specified S3 bucket
The output will not be removed from bucket even after the job expires.

Access permissions

IAM role [Info](#)

Use an existing IAM role

Create an IAM role
By choosing **Create job** you are authorizing creation of this role.

Permissions to access
Your role has access to these resources. The KMS key permission is used only if your input bucket is encrypted

Input S3 bucket and KMS decrypt permission to input bucket

Any S3 bucket and any KMS keys

Role name
Roles are prefixed with "AmazonTranscribeServiceRoleFullAccess-". Your newly created role has full access to the S3 bucket and KMS key for your account.

The name can be up to 64 characters long

▼ **Role permissions details**

Your new role has these permissions to give Amazon Transcribe access to the resources that you've specified.

Service	Access level	Resource
S3	List, Read, Write	All resources
Key Management Service	GenerateDataKey, Decrypt	All resources

4. Choisissez Suivant.

5. Pour Configurer la tâche, activez toutes les fonctionnalités facultatives que vous souhaitez inclure dans votre tâche Call Analytics. Si vous avez déjà créé des catégories, elles apparaissent dans le volet Catégories et sont automatiquement appliquées à votre tâche Call Analytics.

Configure job - *optional* [Info](#)

Content removal

Content removal conceals information in the resulting transcript from your source audio file. Amazon Transcribe changes items in the transcript and does not modify the source audio.

PII redaction [Info](#)

Label the type of PII and also mask the content with the PII entity type in the transcription output. For example, (123) 456-7890 will be masked as [PHONE].

Vocabulary filtering [Info](#)

Vocabulary filtering can remove, mask or tag specified words in the final transcript.

Customization

Custom vocabulary [Info](#)

A custom vocabulary improves the accuracy of recognizing words and phrases specific to your use case.

Summarization

Generative call summarization [Info](#)

Generative call summarization provides a summary of the transcript, including important components of the conversation.


Categories

Create categories to classify calls. For example, you can create a category for all cancellation requests. When you run an analytics job, Amazon Transcribe applies that category to all calls that request cancellation.

Call analytics categories (1) [Info](#)

< 1 > 

	Name	Type	Created	Modified
<input type="radio"/>	CatchNegativeSentiment	POST_CALL	February 17 2023, 10:43 (UTC-08:00)	February 17 2023, 10:43 (UTC-08:00)

If the above categories aren't relevant to your use case, you can create a new category. [Create a new category.](#) 

6. Choisissez Créer une tâche.

AWS CLI

Cet exemple utilise la commande [start-call-analytics-job](#) et le paramètre `channel-definitions`. Pour plus d'informations, consultez [StartCallAnalyticsJob](#) et [ChannelDefinition](#).

```
aws transcribe start-call-analytics-job \  
--region us-west-2 \  
--call-analytics-job-name my-first-call-analytics-job \  
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
--output-location s3://DOC-EXAMPLE-BUCKET/my-output-files/ \  
--data-access-role-arn arn:aws:iam::111122223333:role/ExampleRole \  
--channel-definitions ChannelId=0,ParticipantRole=AGENT \  
ChannelId=1,ParticipantRole=CUSTOMER
```

Voici un autre exemple utilisant la commande [start-call-analytics-job](#), et un corps de requête qui active Call Analytics pour cette tâche.

```
aws transcribe start-call-analytics-job \  
--region us-west-2 \  
--cli-input-json file://filepath/my-call-analytics-job.json
```

Le fichier `my-call-analytics-job.json` contient le corps de requête suivant.

```
{  
  "CallAnalyticsJobName": "my-first-call-analytics-job",  
  "DataAccessRoleArn": "arn:aws:iam::111122223333:role/ExampleRole",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
  },  
  "OutputLocation": "s3://DOC-EXAMPLE-BUCKET/my-output-files/",  
  "ChannelDefinitions": [  
    {  
      "ChannelId": 0,  
      "ParticipantRole": "AGENT"  
    },  
    {  
      "ChannelId": 1,  
      "ParticipantRole": "CUSTOMER"  
    }  
  ]  
}
```

```
}
```

AWS SDK for Python (Boto3)

Cet exemple utilise le AWS SDK for Python (Boto3) pour démarrer une tâche Call Analytics à l'aide de la méthode [start_call_analytics_job](#). Pour plus d'informations, consultez [StartCallAnalyticsJob](#) et [ChannelDefinition](#).

Pour d'autres exemples d'utilisation AWS des SDK, notamment des exemples spécifiques aux fonctionnalités, des scénarios et des exemples multiservices, reportez-vous au chapitre. [Exemples de code pour Amazon Transcribe à l'aide de kits de développement logiciel AWS](#)

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-call-analytics-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
output_location = "s3://DOC-EXAMPLE-BUCKET/my-output-files/"
data_access_role = "arn:aws:iam::111122223333:role/ExampleRole"
transcribe.start_call_analytics_job(
    CallAnalyticsJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    DataAccessRoleArn = data_access_role,
    OutputLocation = output_location,
    ChannelDefinitions = [
        {
            'ChannelId': 0,
            'ParticipantRole': 'AGENT'
        },
        {
            'ChannelId': 1,
            'ParticipantRole': 'CUSTOMER'
        }
    ]
)

while True:
    status = transcribe.get_call_analytics_job(CallAnalyticsJobName = job_name)
    if status['CallAnalyticsJob']['CallAnalyticsJobStatus'] in ['COMPLETED', 'FAILED']:
        break
```

```
print("Not ready yet...")
time.sleep(5)
print(status)
```

Résultats d'analyse après appel

Les transcriptions des analyses post-appel sont affichées dans un turn-by-turn format par segment. Elles incluent la catégorisation des appels, les caractéristiques des appels (scores de niveau sonore, interruptions, temps de non-conversation, vitesse de conversation), le récapitulatif des appels (problèmes, résultats et mesures à prendre), l'expurgation et les sentiments. Un récapitulatif des caractéristiques de la conversation est également fourni à la fin de la transcription.

Pour améliorer la précision et personnaliser davantage vos transcriptions en fonction de votre cas d'utilisation, par exemple en incluant des termes spécifiques au secteur, ajoutez des [vocabulaires personnalisés](#) ou des [modèles de langue personnalisés](#) à votre demande Call Analytics. Pour masquer, supprimer ou étiqueter des mots que vous ne voulez pas voir apparaître dans vos résultats de transcription, tels que des termes vulgaires, ajoutez un [filtrage du vocabulaire](#). Si vous n'êtes pas sûr du code de langue à transmettre au fichier multimédia, vous pouvez activer l'[identification de la langue par lots](#) pour identifier automatiquement la langue de le fichier multimédia.

Les sections suivantes présentent des exemples de sortie JSON au niveau des informations. Pour les résultats compilés, consultez la section [Compilation des résultats d'analyse après appel](#).

Catégorisation des appels

Voici à quoi ressemble une correspondance de catégorie dans votre sortie de transcription. Cet exemple montre que l'audio compris entre l'horodatage de 40 040 millisecondes et l'horodatage de 42 460 millisecondes correspond à la catégorie « résolution positive ». Dans ce cas, la catégorie personnalisée « résolution positive » exigeait un sentiment positif au cours des dernières secondes du discours.

```
"Categories": {
  "MatchedDetails": {
    "positive-resolution": {
      "PointsOfInterest": [
        {
          "BeginOffsetMillis": 40040,
          "EndOffsetMillis": 42460
        }
      ]
    }
  }
}
```

```
    }
  },
  "MatchedCategories": [
    " positive-resolution"
  ]
},
```

Caractéristiques des appels

Voici à quoi ressemblent les caractéristiques des appels dans votre sortie de transcription. Notez que les scores de niveau sonore sont fournis pour chaque tour de conversation, tandis que toutes les autres caractéristiques sont fournies à la fin de la transcription.

```
"LoudnessScores": [
  87.54,
  88.74,
  90.16,
  86.36,
  85.56,
  85.52,
  81.79,
  87.74,
  89.82
],
...

"ConversationCharacteristics": {
  "NonTalkTime": {
    "Instances": [],
    "TotalTimeMillis": 0
  },
  "Interruptions": {
    "TotalCount": 2,
    "TotalTimeMillis": 10700,
    "InterruptionsByInterrupter": {
      "AGENT": [
        {
          "BeginOffsetMillis": 26040,
          "DurationMillis": 5510,
          "EndOffsetMillis": 31550
        }
      ]
    }
  },
  ],
```

```

    "CUSTOMER": [
      {
        "BeginOffsetMillis": 770,
        "DurationMillis": 5190,
        "EndOffsetMillis": 5960
      }
    ]
  },
  "TotalConversationDurationMillis": 42460,
  ...,
  "TalkSpeed": {
    "DetailsByParticipant": {
      "AGENT": {
        "AverageWordsPerMinute": 150
      },
      "CUSTOMER": {
        "AverageWordsPerMinute": 167
      }
    }
  },
  "TalkTime": {
    "DetailsByParticipant": {
      "AGENT": {
        "TotalTimeMillis": 32750
      },
      "CUSTOMER": {
        "TotalTimeMillis": 18010
      }
    },
    "TotalTimeMillis": 50760
  }
},

```

Problèmes, éléments d'action et étapes suivantes

- Dans l'exemple suivant, les problèmes sont identifiés comme commençant au caractère 7 et se terminant au caractère 51, ce qui fait référence à la section suivante du texte : « J'aimerais annuler mon abonnement à une recette ».

```
"Content": "Well, I would like to cancel my recipe subscription.",
```



```
"IssuesDetected": [  
  {  
    "CharacterOffsets": {  
      "Begin": 7,  
      "End": 51  
    }  
  }  
],
```

- Dans l'exemple suivant, les résultats sont identifiés comme commençant au caractère 12 et se terminant au caractère 78, ce qui fait référence à la section suivante du texte : « J'ai apporté toutes les modifications à votre compte et cette réduction est maintenant appliquée ».

```
"Content": "Wonderful. I made all changes to your account and now this discount is applied, please check.",  
  
"OutcomesDetected": [  
  {  
    "CharacterOffsets": {  
      "Begin": 12,  
      "End": 78  
    }  
  }  
],
```

- Dans l'exemple suivant, les actions sont identifiées comme commençant au caractère 0 et se terminant au caractère 103, ce qui fait référence à la section suivante du texte : « Je vous enverrai un e-mail avec tous les détails aujourd'hui, et je vous rappellerai la semaine prochaine pour un suivi ».

```
"Content": "I will send an email with all the details to you today, and I will call you back next week to follow up. Have a wonderful evening.",  
  
"ActionItemsDetected": [  
  {  
    "CharacterOffsets": {  
      "Begin": 0,  
      "End": 103  
    }  
  }  
],
```

Résumé génératif des appels

Voici à quoi ressemble un résumé génératif des appels dans la sortie de transcription :

```
"ContactSummary": {
  "AutoGenerated": {
    "OverallSummary": {
      "Content": "A customer wanted to check to see if we had a bag allowance. We
told them that we didn't have it, but we could add the bag from Canada to Calgary and
then do the one coming back as well."
    }
  }
}
```

La tâche d'analyse s'achèvera sans génération de résumé dans les cas suivants :

- Contenu de conversation insuffisant : la conversation doit inclure au moins un tour de la part de l'agent et du client. Lorsque le contenu de la conversation est insuffisant, le service renvoie le code d'erreur `INSUSUFFICIENT_CONVERSATION_CONTENT`.
- Garde-corps de sécurité : La conversation doit respecter les garde-fous en place pour garantir la génération d'un résumé approprié. Lorsque ces barrières ne sont pas respectées, le service renvoie le code d'erreur `FAILED_SAFETY_GUIDELINES`.

Le code d'erreur se trouve `AnalyticsJobDetails` dans `Skipped` la section ci-dessous de la sortie. Vous pouvez également trouver la raison de l'erreur `CallAnalyticsJobDetails` dans la réponse de l'[GetCallAnalyticsJobAPI](#).

Exemple de sortie d'erreur

```
{
  "JobStatus": "COMPLETED",
  "AnalyticsJobDetails": {
    "Skipped": [
      {
        "Feature": "GENERATIVE_SUMMARIZATION",
        "ReasonCode": "INSUFFICIENT_CONVERSATION_CONTENT",
        "Message": "The conversation needs to have at least one turn from both
the participants to generate summary"
      }
    ]
  }
}
```

```

  },
  "LanguageCode": "en-US",
  "AccountId": "*****",
  "JobName": "Test2-copy",
  ...
}

```

Analyse de sentiment

Voici à quoi ressemble l'analyse des sentiments dans la sortie de transcription.

- Valeurs de turn-by-turn sentiment qualitatives :

```

"Content": "That's very sad to hear. Can I offer you a 50% discount to have you stay
with us?",

```

...

```

"BeginOffsetMillis": 12180,
"EndOffsetMillis": 16960,
"Sentiment": "NEGATIVE",
"ParticipantRole": "AGENT"

```

...

```

"Content": "That is a very generous offer. And I accept.",

```

...

```

"BeginOffsetMillis": 17140,
"EndOffsetMillis": 19860,
"Sentiment": "POSITIVE",
"ParticipantRole": "CUSTOMER"

```

- Valeurs quantitatives des sentiments pour l'ensemble de l'appel :

```

"Sentiment": {
  "OverallSentiment": {
    "AGENT": 2.5,
    "CUSTOMER": 2.1
  },

```

- Valeurs quantitatives des sentiments par participant et par trimestre d'appel :

```
"SentimentByPeriod": {
  "QUARTER": {
    "AGENT": [
      {
        "Score": 0.0,
        "BeginOffsetMillis": 0,
        "EndOffsetMillis": 9862
      },
      {
        "Score": -5.0,
        "BeginOffsetMillis": 9862,
        "EndOffsetMillis": 19725
      },
      {
        "Score": 5.0,
        "BeginOffsetMillis": 19725,
        "EndOffsetMillis": 29587
      },
      {
        "Score": 5.0,
        "BeginOffsetMillis": 29587,
        "EndOffsetMillis": 39450
      }
    ],
    "CUSTOMER": [
      {
        "Score": -2.5,
        "BeginOffsetMillis": 0,
        "EndOffsetMillis": 10615
      },
      {
        "Score": 5.0,
        "BeginOffsetMillis": 10615,
        "EndOffsetMillis": 21230
      },
      {
        "Score": 2.5,
        "BeginOffsetMillis": 21230,
        "EndOffsetMillis": 31845
      },
      {
        "Score": 5.0,
        "BeginOffsetMillis": 31845,
```

```
        "EndOffsetMillis": 42460
      }
    ]
  }
}
```

Expurgation des données d'identification personnelle (PII)

Voici à quoi ressemble l'expurgation des données d'identification personnelle (PII) dans la sortie de transcription.

```
"Content": "[PII], my name is [PII], how can I help?",
"Redaction": [{
  "Confidence": "0.9998",
  "Type": "NAME",
  "Category": "PII"
}]
```

Pour plus d'informations, reportez-vous à [Expurgation des données d'identification personnelle \(PII\) dans la tâche de lot](#).

Identification de la langue

Voici à quoi ressemble l'identification de la langue dans la sortie de transcription si la fonctionnalité est activée.

```
"LanguageIdentification": [{
  "Code": "en-US",
  "Score": "0.8299"
}, {
  "Code": "en-NZ",
  "Score": "0.0728"
}, {
  "Code": "zh-TW",
  "Score": "0.0695"
}, {
  "Code": "th-TH",
  "Score": "0.0156"
}, {
  "Code": "en-ZA",
  "Score": "0.0121"
}]
```

```
}]
```

Dans l'exemple de sortie ci-dessus, Identification de la langue remplit les codes de langue avec des scores de confiance. Le résultat avec le score le plus élevé est sélectionné comme code de langue pour la transcription. Pour plus de détails sur le mode, reportez-vous à [Identification des langues dominantes des médias](#).

Compilation des résultats d'analyse après appel

Par souci de concision, une partie du contenu est remplacée par des points de suspension dans la sortie de transcription suivante.

Cet exemple inclut une fonctionnalité optionnelle : le résumé génératif des appels.

```
{
  "JobStatus": "COMPLETED",
  "LanguageCode": "en-US",
  "Transcript": [
    {
      "LoudnessScores": [
        78.63,
        78.37,
        77.98,
        74.18
      ],
      "Content": "[PII], my name is [PII], how can I help?",
      ...

      "Content": "Well, I would like to cancel my recipe subscription.",
      "IssuesDetected": [
        {
          "CharacterOffsets": {
            "Begin": 7,
            "End": 51
          }
        }
      ],
      ...

      "Content": "That's very sad to hear. Can I offer you a 50% discount to have
you stay with us?",
```

```

    "Items": [
      ...
    ],
    "Id": "649afe93-1e59-4ae9-a3ba-a0a613868f5d",
    "BeginOffsetMillis": 12180,
    "EndOffsetMillis": 16960,
    "Sentiment": "NEGATIVE",
    "ParticipantRole": "AGENT"
  },
  {
    "LoudnessScores": [
      80.22,
      79.48,
      82.81
    ],
    "Content": "That is a very generous offer. And I accept.",
    "Items": [
      ...
    ],
    "Id": "f9266cba-34df-4ca8-9cea-4f62a52a7981",
    "BeginOffsetMillis": 17140,
    "EndOffsetMillis": 19860,
    "Sentiment": "POSITIVE",
    "ParticipantRole": "CUSTOMER"
  },
  {
    ...

    "Content": "Wonderful. I made all changes to your account and now this
discount is applied, please check.",
    "OutcomesDetected": [
      {
        "CharacterOffsets": {
          "Begin": 12,
          "End": 78
        }
      }
    ],
    ...

    "Content": "I will send an email with all the details to you today, and I
will call you back next week to follow up. Have a wonderful evening.",

```

```

    "Items": [
      ...
    ],
    "Id": "78cd0923-cafd-44a5-a66e-09515796572f",
    "BeginOffsetMillis": 31800,
    "EndOffsetMillis": 39450,
    "Sentiment": "POSITIVE",
    "ParticipantRole": "AGENT"
  },
  {
    "LoudnessScores": [
      78.54,
      68.76,
      67.76
    ],
    "Content": "Thank you very much, sir. Goodbye.",
    "Items": [
      ...
    ],
    "Id": "5c5e6be0-8349-4767-8447-986f995af7c3",
    "BeginOffsetMillis": 40040,
    "EndOffsetMillis": 42460,
    "Sentiment": "POSITIVE",
    "ParticipantRole": "CUSTOMER"
  }
],
...

"Categories": {
  "MatchedDetails": {
    "positive-resolution": {
      "PointsOfInterest": [
        {
          "BeginOffsetMillis": 40040,
          "EndOffsetMillis": 42460
        }
      ]
    }
  },
  "MatchedCategories": [
    "positive-resolution"
  ]
},

```


...

```
"ConversationCharacteristics": {
  "NonTalkTime": {
    "Instances": [],
    "TotalTimeMillis": 0
  },
  "Interruptions": {
    "TotalCount": 2,
    "TotalTimeMillis": 10700,
    "InterruptionsByInterrupter": {
      "AGENT": [
        {
          "BeginOffsetMillis": 26040,
          "DurationMillis": 5510,
          "EndOffsetMillis": 31550
        }
      ],
      "CUSTOMER": [
        {
          "BeginOffsetMillis": 770,
          "DurationMillis": 5190,
          "EndOffsetMillis": 5960
        }
      ]
    }
  },
  "TotalConversationDurationMillis": 42460,
  "Sentiment": {
    "OverallSentiment": {
      "AGENT": 2.5,
      "CUSTOMER": 2.1
    },
    "SentimentByPeriod": {
      "QUARTER": {
        "AGENT": [
          {
            "Score": 0.0,
            "BeginOffsetMillis": 0,
            "EndOffsetMillis": 9862
          },
          {
            "Score": -5.0,
```

```
        "BeginOffsetMillis": 9862,
        "EndOffsetMillis": 19725
    },
    {
        "Score": 5.0,
        "BeginOffsetMillis": 19725,
        "EndOffsetMillis": 29587
    },
    {
        "Score": 5.0,
        "BeginOffsetMillis": 29587,
        "EndOffsetMillis": 39450
    }
],
"CUSTOMER": [
    {
        "Score": -2.5,
        "BeginOffsetMillis": 0,
        "EndOffsetMillis": 10615
    },
    {
        "Score": 5.0,
        "BeginOffsetMillis": 10615,
        "EndOffsetMillis": 21230
    },
    {
        "Score": 2.5,
        "BeginOffsetMillis": 21230,
        "EndOffsetMillis": 31845
    },
    {
        "Score": 5.0,
        "BeginOffsetMillis": 31845,
        "EndOffsetMillis": 42460
    }
]
}
},
"TalkSpeed": {
    "DetailsByParticipant": {
        "AGENT": {
            "AverageWordsPerMinute": 150
        },
    },
}
```

```

        "CUSTOMER": {
            "AverageWordsPerMinute": 167
        }
    },
    "TalkTime": {
        "DetailsByParticipant": {
            "AGENT": {
                "TotalTimeMillis": 32750
            },
            "CUSTOMER": {
                "TotalTimeMillis": 18010
            }
        },
        "TotalTimeMillis": 50760
    },
    "ContactSummary": { // Optional feature - Generative call summarization
        "AutoGenerated": {
            "OverallSummary": {
                "Content": "The customer initially wanted to cancel but the agent
convinced them to stay by offering a 50% discount, which the customer accepted after
reconsidering cancelling given the significant savings. The agent ensured the discount
was applied and said they would follow up to ensure the customer remained happy with
the revised subscription."
            }
        }
    },
    "AnalyticsJobDetails": {
        "Skipped": []
    },
    ...
}

```

Activation du résumé génératif des appels

Note

Propulsé par Amazon Bedrock : AWS implémente la [détection automatique des abus](#). Le résumé après contact optimisé par l'IA générative étant basé sur Amazon Bedrock, les

utilisateurs peuvent tirer pleinement parti des contrôles mis en œuvre dans Amazon Bedrock pour renforcer la sûreté, la sécurité et l'utilisation responsable de l'intelligence artificielle (IA).

Pour utiliser le résumé génératif des appels avec une tâche d'analyse après appel, consultez les exemples suivants :

AWS Management Console

Dans le panneau Résumé, activez le résumé génératif des appels pour recevoir le résumé dans la sortie.

Configure job - *optional* [Info](#)

Content removal

Content removal conceals information in the resulting transcript from your source audio file. Amazon Transcribe changes items in the transcript and does not modify the source audio.

- PII redaction** [Info](#)
Label the type of PII and also mask the content with the PII entity type in the transcription output. For example, (123) 456-7890 will be masked as [PHONE].
- Vocabulary filtering** [Info](#)
Vocabulary filtering can remove, mask or tag specified words in the final transcript.

Customization

- Custom vocabulary** [Info](#)
A custom vocabulary improves the accuracy of recognizing words and phrases specific to your use case.

Summarization

- Generative call summarization** [Info](#)
Generative call summarization provides a summary of the transcript, including important components of the conversation.

Categories


Create categories to classify calls. For example, you can create a category for all cancellation requests. When you run an analytics job, Amazon Transcribe applies that category to all calls that request cancellation.

Call analytics categories (1) [Info](#)

< 1 > 

	Name	Type	Created	Modified
<input type="radio"/>	CatchNegativeSentiment	POST_CALL	February 17 2023, 10:43 (UTC-08:00)	February 17 2023, 10:43 (UTC-08:00)

Activation du résumé génératif des appels

If the above categories aren't relevant to your use case, you can create a new category. [Create a new category.](#) 

AWS CLI

Cet exemple utilise la commande [start-call-analytics-job](#) et le paramètre Settings avec les sous-paramètres Summarization. Pour plus d'informations, consultez [StartCallAnalyticsJob](#).

```
aws transcribe start-call-analytics-job \  
--region us-west-2 \  
--call-analytics-job-name my-first-call-analytics-job \  
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
--output-location s3://DOC-EXAMPLE-BUCKET/my-output-files/ \  
--data-access-role-arn arn:aws:iam::111122223333:role/ExampleRole \  
--channel-definitions ChannelId=0,ParticipantRole=AGENT  
ChannelId=1,ParticipantRole=CUSTOMER  
--settings '{"Summarization":{"GenerateAbstractiveSummary":true}}'
```

Voici un autre exemple utilisant la commande [start-call-analytics-job](#), et un corps de requête qui active le résumé pour cette tâche.

```
aws transcribe start-call-analytics-job \  
--region us-west-2 \  
--cli-input-json file://filepath/my-call-analytics-job.json
```

Le fichier `my-call-analytics-job.json` contient le corps de requête suivant.

```
{  
  "CallAnalyticsJobName": "my-first-call-analytics-job",  
  "DataAccessRoleArn": "arn:aws:iam::111122223333:role/ExampleRole",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
  },  
  "OutputLocation": "s3://DOC-EXAMPLE-BUCKET/my-output-files/",  
  "ChannelDefinitions": [  
    {  
      "ChannelId": 0,  
      "ParticipantRole": "AGENT"  
    },  
    {
```

```

    "ChannelId": 1,
    "ParticipantRole": "CUSTOMER"
  }
],
"Settings": {
  "Summarization":{
    "GenerateAbstractiveSummary": true
  }
}
}

```

AWS SDK for Python (Boto3)

Cet exemple utilise le AWS SDK for Python (Boto3) pour démarrer une analyse des appels avec la synthèse activée à l'aide de la méthode [start_call_analytics_job](#). Pour plus d'informations, consultez [StartCallAnalyticsJob](#).

Pour d'autres exemples d'utilisation AWS des SDK, notamment des exemples spécifiques aux fonctionnalités, des scénarios et des exemples multiservices, reportez-vous au chapitre. [Exemples de code pour Amazon Transcribe à l'aide de kits de développement logiciel AWS](#)

```

from __future__ import print_function
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-call-analytics-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
output_location = "s3://DOC-EXAMPLE-BUCKET/my-output-files/"
data_access_role = "arn:aws:iam::111122223333:role/ExampleRole"
transcribe.start_call_analytics_job(
    CallAnalyticsJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    DataAccessRoleArn = data_access_role,
    OutputLocation = output_location,
    ChannelDefinitions = [
        {
            'ChannelId': 0,
            'ParticipantRole': 'AGENT'
        },
    ],

```

```
{
  'ChannelId': 1,
  'ParticipantRole': 'CUSTOMER'
},
Settings = {
  "Summarization":
  {
    "GenerateAbstractiveSummary": true
  }
}
)

while True:
  status = transcribe.get_call_analytics_job(CallAnalyticsJobName = job_name)
  if status['CallAnalyticsJob']['CallAnalyticsJobStatus'] in ['COMPLETED', 'FAILED']:
    break
  print("Not ready yet...")
  time.sleep(5)
print(status)
```

Analyse Call Analytics en temps réel

L'analyse Call Analytics en temps réel fournit des informations en temps réel qui peuvent être utilisées pour résoudre des problèmes et atténuer les escalades à mesure qu'elles se produisent.

Les informations suivantes sont disponibles grâce à l'analyse Call Analytics en temps réel :

- Les [événements de catégorie](#) qui utilisent des règles pour signaler des mots clés et des phrases spécifiques. Les événements de catégorie peuvent être utilisés pour créer des [alertes en temps réel](#).
- La [détection des problèmes](#) qui identifient les problèmes évoqués dans chaque segment audio.
- L'[identification des PII \(données sensibles\)](#) dans votre transcription de texte.
- L'[expurgation des PII \(données sensibles\)](#) de votre transcription de texte.
- L'[analyse des sentiments](#) pour chaque segment de discours.

Outre l'analyse des appels en temps réel, Amazon Transcribe vous pouvez également effectuer des [analyses post-appel](#) sur votre flux multimédia. Vous pouvez inclure des analyses après appel dans votre demande Call Analytics en temps réel à l'aide du paramètre [PostCallAnalyticsSettings](#).

Informations en temps réel

Cette section détaille les informations disponibles pour les transcriptions Call Analytics en temps réel.

Événements de catégorie

Les événements de catégorie vous permettent de faire correspondre votre transcription à un mot clé ou une phrase exact. Par exemple, si vous définissez un filtre pour la phrase « Je souhaite parler au responsable », il Amazon Transcribe filtre pour cette phrase exacte.

Voici un [exemple de sortie](#).

Pour plus d'informations sur la création de catégories Call Analytics en temps réel, consultez la section [Création de catégories pour les transcriptions en temps réel](#).

Tip

Les événements de catégorie vous permettent de définir des alertes en temps réel. Pour plus d'informations, consultez la section [Création d'alertes en temps réel pour les correspondances de catégorie](#).

Détection des problèmes

La détection des problèmes fournit de brefs récapitulatifs des problèmes détectés dans chaque segment audio. La fonctionnalité de détection des problèmes vous permet de :

- réduire le besoin de prendre des notes manuellement pendant et après les appels ;
- améliorer l'efficacité des agents afin de leur permettre de répondre plus rapidement aux clients.

Note

La détection des problèmes est prise en charge avec les dialectes anglais suivants : australien (en-AU), britannique (en-GB) et américain (en-US).

La fonctionnalité de détection des problèmes fonctionne dans tous les secteurs d'activité et est basée sur le contexte. Il fonctionne out-of-the-box et ne prend donc pas en charge la personnalisation, telle que la formation des modèles ou les catégories personnalisées.

La détection des problèmes avec Call Analytics en temps réel est effectuée sur chaque segment audio complet.

Voici un [exemple de sortie](#).

Identification des PII (données sensibles)

L'identification des données sensibles indique les données d'identification personnelle (PII) dans la transcription de texte. Ce paramètre est utile pour protéger les informations des clients.

Note

L'identification des informations personnelles en temps réel est prise en charge avec les dialectes anglais suivants : australien (en-AU), britannique (en-GB), américain (en-US) et avec le dialecte espagnol (). es-US

L'identification des PII avec Call Analytics en temps réel est effectuée sur chaque segment audio complet.

Pour consulter la liste des informations personnelles identifiées à l'aide de cette fonctionnalité, ou pour en savoir plus sur l'identification par des informations personnelles Amazon Transcribe, consultez [La suppression ou l'identification personnelle des données d'identification personnelle](#).

Voici un [exemple de sortie](#).

Expurgation des PII (données sensibles)

L'expurgation des données sensibles remplace les informations d'identification personnelle (PII) dans votre transcription de texte par le type de PII (par exemple, [NAME]). Ce paramètre est utile pour protéger les informations des clients.

Note

La rédaction des informations personnelles en temps réel est prise en charge avec les dialectes anglais suivants : australien (en-AU), britannique (en-GB), américain (en-US) et avec le dialecte espagnol (). es-US

L'expurgation des PII avec Call Analytics en temps réel est effectuée sur chaque segment audio complet.

Pour consulter la liste des PII supprimées à l'aide de cette fonctionnalité, ou pour en savoir plus sur l'expurgation avec Amazon Transcribe, consultez la section [La suppression ou l'identification personnelle des données d'identification personnelle](#).

Voici un [exemple de sortie](#).

Analyse de sentiment

L'analyse des sentiments permet d'estimer ce que le client et l'agent ressentent tout au long de l'appel. Cette métrique est fournie pour chaque segment vocal et est représentée sous la forme d'une valeur qualitative (`positive`, `neutral`, `mixed` ou `negative`).

Ce paramètre vous permet d'évaluer qualitativement le sentiment général de chaque participant à l'appel et le sentiment de chaque participant pendant chaque segment de discours. Cette métrique peut aider à déterminer si votre agent est capable de satisfaire un client mécontent à la fin de l'appel.

L'analyse des sentiments avec Call Analytics en temps réel est effectuée sur chaque segment audio complet.

L'analyse des sentiments fonctionne out-of-the-box et ne prend donc pas en charge la personnalisation, telle que la formation des modèles ou les catégories personnalisées.

Voici un [exemple de sortie](#).

Création de catégories pour les transcriptions en temps réel

L'analyse Call Analytics en temps réel prend en charge la création de catégories personnalisées, que vous pouvez utiliser pour adapter vos analyses de transcriptions aux besoins spécifiques de votre entreprise.

Vous pouvez créer autant de catégories que vous le souhaitez pour couvrir une gamme de scénarios différents. Pour chaque catégorie que vous créez, vous devez créer entre 1 et 20 règles. Les transcriptions Call Analytics en temps réel ne prennent en charge que les règles qui utilisent [TranscriptFilter](#) (correspondances de mots clés). Pour plus d'informations sur l'utilisation de règles dans le cadre de l'opération [CreateCallAnalyticsCategory](#), consultez la section [Critères de règles pour les catégories Call Analytics en temps réel](#).

Si le contenu de votre média répond à toutes les règles que vous avez spécifiées dans une catégorie donnée, Amazon Transcribe attribue cette catégorie à votre sortie. Pour un exemple de correspondance de catégorie au format de sortie JSON, consultez [la sortie d'un événement de catégorie](#).

Voici quelques exemples de ce que vous pouvez faire avec des catégories personnalisées :

- identifier les problèmes qui nécessitent une attention immédiate en signalant et en suivant des ensembles de mots clés spécifiques ;
- surveiller la conformité, par exemple si un agent prononce (ou omet) une phrase spécifique ;
- marquer des mots et des phrases spécifiques en temps réel ; vous pouvez ensuite définir la correspondance de votre catégorie pour définir une alerte immédiate. Par exemple, si vous créez une catégorie Call Analytics en temps réel pour un client qui dit « parler à un responsable », vous pouvez définir une [alerte d'événement](#) pour cette correspondance de catégorie en temps réel qui notifie le responsable en service.

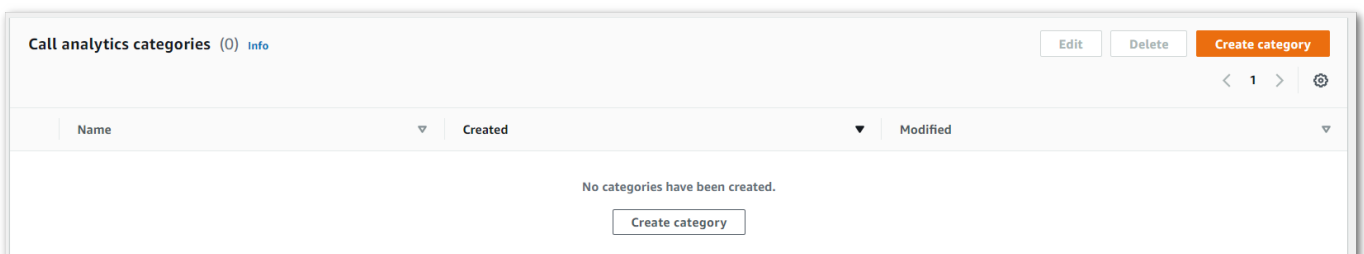
Catégories après appel et catégories en temps réel

Lorsque vous créez une nouvelle catégorie, vous pouvez spécifier si vous souhaitez qu'elle soit créée en tant que catégorie après appel (POST_CALL) ou en tant que catégorie en temps réel (REAL_TIME). Si vous ne spécifiez aucune option, votre catégorie est créée par défaut en tant que catégorie après appel. Les correspondances de catégories en temps réel peuvent être utilisées pour créer des alertes en temps réel. Pour plus d'informations, consultez [Création d'alertes en temps réel pour les correspondances de catégorie](#).

Pour créer une nouvelle catégorie pour Call Analytics en temps réel, vous pouvez utiliser la AWS Management Console, l'AWS CLI ou les kits SDK AWS . Consultez les exemples suivants :

AWS Management Console

1. Dans le volet de navigation, sous Amazon Transcribe, choisissez Amazon Transcribe Call Analytics.
2. Choisissez Catégories Call Analytics pour accéder à la page Catégories Call Analytics. Cliquez sur le bouton Créer une catégorie.



3. Vous êtes maintenant sur la page Créer une catégorie. Entrez un nom pour votre catégorie, puis choisissez « Analyse des appels en temps réel » dans le menu déroulant Type de catégorie.

Category settings

Category name
MyCategory
The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, ., _ , and - (hyphen).

Category type [Info](#)
Choose category type
Batch call analytics

Real time call analytics

Use a template (recommended)
Use a template to edit predefined rules.

Create from scratch
If you know the rules that you want to define, choose this option.

Template type [Info](#)
Choose the template for the category that most closely matches the one you want to create.
Choose a template

4. Vous pouvez choisir un modèle pour créer votre catégorie ou en créer une de zéro.

Si vous utilisez un modèle : sélectionnez Utiliser un modèle (recommandé), choisissez le modèle souhaité, puis sélectionnez Créer une catégorie.

Category settings

Category name
MyCategory
The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, ., _ , and - (hyphen).

Category type [Info](#)
Real time call analytics

Category creation method [Info](#)

Use a template (recommended)
Use a template to edit predefined rules.

Create from scratch
If you know the rules that you want to define, choose this option.

Template type [Info](#)
Choose the template for the category that most closely matches the one you want to create.
Choose a template

Customer content is negative and mentioned manager

5. Si vous créez une catégorie personnalisée : sélectionnez Créer à partir de zéro.

Create category [Info](#)

Category settings

Category name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, ., _ , and - (hyphen).

Category creation method [Info](#)

Use a template (recommended)
Use a template to edit predefined rules.

Create from scratch
If you know the rules that you want to define, choose this option.

Rules

All the rule conditions must be met for a transcription job to be classified in this category.

▼ Rule 1 Delete rule

Rule type [Info](#)
Choose the rule that you want to define.

Add rule

You can add up to 19 more rules.

6. Ajoutez des règles à votre catégorie à l'aide du menu déroulant. Vous pouvez ajouter jusqu'à 20 règles par catégorie. Avec les transcriptions Call Analytics en temps réel, vous ne pouvez inclure que des règles impliquant des correspondances de contenu de transcription. Toutes les correspondances sont signalés en temps réel.

Rules
All the rule conditions must be met for a transcription job to be classified in this category.

▼ Rule 1 Delete rule

Rule type [Info](#)
Choose the rule that you want to define.

Choose a rule type ▲

Transcript content match
Trigger the rule when the speaker says the words or phrases that you specify.

Add rule

You can add up to 19 more rules.

7. Voici un exemple de catégorie comportant une règle : un client qui dit « parler à un responsable » à tout moment au cours de l'appel.

Rules
All the rule conditions must be met for a transcription job to be classified in this category.

▼ Rule 1 Delete rule

When any of the words were mentioned during the entire call when the speaker was customer.

Rule type [Info](#)
Choose the rule that you want to define.

Transcript content match ▼

Logic [Info](#)
Define the conditions that must be met.

When any of the words were mentioned ▼

during the entire call ▼

when the speaker was customer ▼

Words or phrases [Info](#)
Enter the words or phrases that you want to look for in the transcript. You can enter up to 100 words or phrases.

speak to a manager Add a new word or phrase

The word or phrase can be up to 2,000 characters.

Add rule

You can add up to 19 more rules.

8. Lorsque vous avez fini d'ajouter des règles à votre catégorie, choisissez Créer une catégorie.

AWS CLI

Cet exemple utilise la commande [create-call-analytics-category](#). Pour plus d'informations, consultez [CreateCallAnalyticsCategory](#), [CategoryProperties](#) et [Rule](#).

L'exemple suivant crée une catégorie avec la règle :

- Le client a prononcé la phrase « parler au responsable » à n'importe quel moment de l'appel.

Cet exemple utilise la commande [create-call-analytics-category](#) et un corps de requête qui ajoute une règle à votre catégorie.

```
aws transcribe create-call-analytics-category \  
--cli-input-json file://filepath/my-first-analytics-category.json
```

Le fichier `my-first-analytics-category.json` contient le corps de requête suivant.

```
{  
  "CategoryName": "my-new-real-time-category",  
  "InputType": "REAL_TIME",  
  "Rules": [  
    {  
      "TranscriptFilter": {  
        "Negate": false,  
        "Targets": [  
          "speak to the manager"  
        ],  
        "TranscriptFilterType": "EXACT"  
      }  
    }  
  ]  
}
```

AWS SDK for Python (Boto3)

Cet exemple utilise le AWS SDK for Python (Boto3) pour créer une catégorie à l'aide des Rules arguments `CategoryName` et de la méthode [create_call_analytics_category](#). Pour plus d'informations, consultez [CreateCallAnalyticsCategory](#), [CategoryProperties](#) et [Rule](#).

Pour d'autres exemples d'utilisation AWS des SDK, notamment des exemples spécifiques aux fonctionnalités, des scénarios et des exemples multiservices, reportez-vous au chapitre. [Exemples de code pour Amazon Transcribe à l'aide de kits de développement logiciel AWS](#)

L'exemple suivant crée une catégorie avec la règle :

- Le client a prononcé la phrase « parler au responsable » à n'importe quel moment de l'appel.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
category_name = "my-new-real-time-category"
transcribe.create_call_analytics_category(
    CategoryName = category_name,
    InputType = "REAL_TIME",
    Rules = [
        {
            'TranscriptFilter': {
                'Negate': False,
                'Targets': [
                    'speak to the manager'
                ],
                'TranscriptFilterType': 'EXACT'
            }
        }
    ]
)

result = transcribe.get_call_analytics_category(CategoryName = category_name)
print(result)
```

Critères de règles pour les catégories Call Analytics en temps réel

Cette section décrit les types de règles REAL_TIME personnalisées que vous pouvez créer à l'aide de l'opération d'API [CreateCallAnalyticsCategory](#).

La détection des problèmes étant automatique, vous n'avez pas besoin de créer de règles ou de catégories pour signaler les problèmes.

Notez que seules les correspondances de mots clés sont prises en charge pour les transcriptions Call Analytics en temps réel. Si vous souhaitez créer des catégories incluant des interruptions, du silence

ou des sentiments, reportez-vous à la section [Critères de règles pour les catégories d'analyse après appel](#).

Correspondances de mots clés

Les règles utilisant des mots clés (type de données [TranscriptFilter](#)) sont conçues pour correspondre à :

- des mots ou phrases personnalisés prononcés par l'agent, le client ou les deux ;
- des mots ou phrases personnalisés qui ne sont pas prononcés par l'agent, le client ou les deux ;
- des mots ou phrases personnalisés qui apparaissent dans un laps de temps spécifique.

Voici un exemple des paramètres disponibles avec [TranscriptFilter](#) :

```
"TranscriptFilter": {
  "AbsoluteTimeRange": {
    Specify the time frame, in milliseconds, when the match should occur
  },
  "RelativeTimeRange": {
    Specify the time frame, in percentage, when the match should occur
  },
  "Negate": Specify if you want to match the presence or absence of your custom keywords,
  "ParticipantRole": Specify if you want to match speech from the agent, the customer, or both,
  "Targets": [ The custom words and phrases you want to match ],
  "TranscriptFilterType": Use this parameter to specify an exact match for the specified targets
}
```

Pour plus d'informations sur ces paramètres et les valeurs valides associées à chacun, consultez les rubriques [CreateCallAnalyticsCategory](#) et [TranscriptFilter](#).

Analyses après appel avec transcriptions en temps réel

L'analyse après appel est une fonctionnalité facultative disponible avec les transcriptions Call Analytics en temps réel. Outre les [informations d'analyse en temps réel](#) standard, les analyses après appel vous fournissent les informations suivantes :

- éléments d'action : répertorie tous les éléments d'action identifiés dans l'appel ;

- interruptions : mesure si et quand un participant coupe la parole à l'autre participant au milieu d'une phrase ;
- problèmes : fournit les problèmes identifiés dans l'appel ;
- niveau sonore : mesure le volume auquel chaque participant parle ;
- temps de non-conversation : mesure les périodes pendant lesquelles il n'y a pas de parole ;
- résultats : fournit le résultat, ou la résolution, identifié dans l'appel ;
- vitesse de conversation : mesure la vitesse à laquelle les deux participants parlent ;
- temps de conversation : mesure le temps (en millisecondes) pendant lequel chaque participant a parlé pendant l'appel.

Lorsqu'elle est activée, l'analyse post-appel à partir d'un flux audio produit une transcription similaire à une [analyse post-appel à partir d'un fichier audio](#) et la stocke dans le Amazon S3 bucket spécifié dans `OutputLocation`. En outre, l'analyse post-appel enregistre votre flux audio et l'enregistre sous forme de fichier audio (WAVformat) dans le même Amazon S3 compartiment. Si vous activez la rédaction, une transcription expurgée et un fichier audio expurgé sont également stockés dans le compartiment spécifié. Amazon S3 L'activation de l'analyse après appel avec votre flux audio produit entre deux et quatre fichiers, comme décrit ici :

- Si l'expurgation n'est pas activée, vos fichiers de sortie sont les suivants :
 1. une transcription non expurgée ;
 2. un fichier audio expurgé.
- Si l'expurgation est activée sans l'option non expurgée (`redacted`), vos fichiers de sortie sont les suivants :
 1. une transcription expurgée ;
 2. un fichier audio expurgé.
- Si l'expurgation est activée avec l'option non expurgée (`redacted_and_unredacted`), vos fichiers de sortie sont les suivants :
 1. une transcription expurgée ;
 2. un fichier audio expurgé.
 3. une transcription non expurgée ;
 4. un fichier audio expurgé.

Notez que si vous activez l'analyse après appel ([PostCallAnalyticsSettings](#)) avec votre demande et que vous utilisez un flux FLAC ou OPUS-0GG, `loudnessScore` n'apparaît pas dans votre transcription et aucun enregistrement audio de votre flux n'est créé. Transcribe peut également ne pas être en mesure de fournir des analyses post-appel pour les flux audio de longue durée d'une durée supérieure à 90 minutes.

Pour en savoir plus sur les informations disponibles avec l'analyse après appel des flux audio, consultez la section [post-call analytics insights](#).

Tip

Si vous activez l'analyse après appel avec votre demande Call Analytics en temps réel, toutes vos catégories `POST_CALL` et `REAL-TIME` sont appliquées à votre transcription d'analyse après appel.

Activation de l'analyse après appel

Pour activer l'analyse après appel, vous devez inclure le paramètre [PostCallAnalyticsSettings](#) dans votre demande Call Analytics en temps réel. Les paramètres suivants doivent être inclus lorsque `PostCallAnalyticsSettings` est activé :

- `OutputLocation`: le Amazon S3 compartiment dans lequel vous souhaitez stocker votre transcription après un appel.
- `DataAccessRoleArn` : ARN (Amazon Resource Name) du rôle Amazon S3 autorisé à accéder au compartiment Amazon S3 spécifié. Notez que vous devez également utiliser la [stratégie d'approbation des analyses en temps réel](#).

Si vous souhaitez obtenir une version expurgée de votre transcription, vous pouvez inclure `ContentRedactionOutput` ou `ContentRedactionType` dans votre demande. Pour plus d'informations sur ces paramètres, consultez [StartCallAnalyticsStreamTranscription](#) dans la Référence d'API.

Pour démarrer une transcription d'analyse des appels en temps réel avec l'analyse post-appel activée, vous pouvez utiliser le AWS Management Console(démo uniquement), HTTP/2 ou WebSockets Pour obtenir des exemples, consultez [Démarrage d'une transcription Call Analytics en temps réel](#).

⚠ Important

Actuellement, le AWS Management Console seul propose une démo d'analyse des appels en temps réel avec des exemples audio préchargés. Si vous souhaitez utiliser votre propre audio, vous devez utiliser l'API (HTTP/2 ou un SDK). WebSockets

Exemple de résultats d'analyse après appel

Les transcriptions post-appel sont affichées dans un turn-by-turn format par segment. Elles incluent les caractéristiques des appels, le sentiment, le récapitulatif des appels, la détection des problèmes et (éventuellement) l'expurgation des PII. Si l'une de vos catégories après appel correspond au contenu audio, elle est également présente dans votre sortie.

Pour améliorer la précision et personnaliser davantage vos transcriptions en fonction de votre cas d'utilisation, par exemple en incluant des termes spécifiques au secteur, ajoutez des [vocabulaires personnalisés](#) ou des [modèles de langue personnalisés](#) à votre demande Call Analytics. Pour masquer, supprimer ou étiqueter des mots que vous ne voulez pas voir apparaître dans vos résultats de transcription, tels que des termes vulgaires, ajoutez un [filtrage du vocabulaire](#).

Voici un exemple de sortie d'analyse après appel compilée :

```
{
  "JobStatus": "COMPLETED",
  "LanguageCode": "en-US",
  "AccountId": "1234567890",
  "Channel": "VOICE",
  "Participants": [{
    "ParticipantRole": "AGENT"
  },
  {
    "ParticipantRole": "CUSTOMER"
  }],
  "SessionId": "12a3b45c-de6f-78g9-0123-45h6ab78c901",
  "ContentMetadata": {
    "Output": "Raw"
  }
  "Transcript": [{
    "LoudnessScores": [
      78.63,
      78.37,
```

```

        77.98,
        74.18
    ],
    "Content": "[PII], my name is [PII], how can I help?",

    ...

    "Content": "Well, I would like to cancel my recipe subscription.",
    "IssuesDetected": [{
        "CharacterOffsets": {
            "Begin": 7,
            "End": 51
        }
    }],

    ...

    "Content": "That's very sad to hear. Can I offer you a 50% discount to have you
stay with us?",
    "Id": "649afe93-1e59-4ae9-a3ba-a0a613868f5d",
    "BeginOffsetMillis": 12180,
    "EndOffsetMillis": 16960,
    "Sentiment": "NEGATIVE",
    "ParticipantRole": "AGENT"
},
{
    "LoudnessScores": [
        80.22,
        79.48,
        82.81
    ],
    "Content": "That is a very generous offer. And I accept.",
    "Id": "f9266cba-34df-4ca8-9cea-4f62a52a7981",
    "BeginOffsetMillis": 17140,
    "EndOffsetMillis": 19860,
    "Sentiment": "POSITIVE",
    "ParticipantRole": "CUSTOMER"
},

    ...

    "Content": "Wonderful. I made all changes to your account and now this discount
is applied, please check.",
    "OutcomesDetected": [{
    "CharacterOffsets": {

```

```

        "Begin": 12,
        "End": 78
    }
}],

    ...

    "Content": "I will send an email with all the details to you today, and I will
call you back next week to follow up. Have a wonderful evening.",
    "Id": "78cd0923-cafd-44a5-a66e-09515796572f",
    "BeginOffsetMillis": 31800,
    "EndOffsetMillis": 39450,
    "Sentiment": "POSITIVE",
    "ParticipantRole": "AGENT"
},
{
    "LoudnessScores": [
        78.54,
        68.76,
        67.76
    ],
    "Content": "Thank you very much, sir. Goodbye.",
    "Id": "5c5e6be0-8349-4767-8447-986f995af7c3",
    "BeginOffsetMillis": 40040,
    "EndOffsetMillis": 42460,
    "Sentiment": "POSITIVE",
    "ParticipantRole": "CUSTOMER"
}
],

    ...

"Categories": {
    "MatchedDetails": {
        "positive-resolution": {
            "PointsOfInterest": [{
                "BeginOffsetMillis": 40040,
                "EndOffsetMillis": 42460
            }]
        }
    },
    "MatchedCategories": [
        "positive-resolution"
    ]
}

```

```
},
...
"ConversationCharacteristics": {
  "NonTalkTime": {
    "Instances": [],
    "TotalTimeMillis": 0
  },
  "Interruptions": {
    "TotalCount": 2,
    "TotalTimeMillis": 10700,
    "InterruptionsByInterrupter": {
      "AGENT": [{
        "BeginOffsetMillis": 26040,
        "DurationMillis": 5510,
        "EndOffsetMillis": 31550
      }],
      "CUSTOMER": [{
        "BeginOffsetMillis": 770,
        "DurationMillis": 5190,
        "EndOffsetMillis": 5960
      }]
    }
  },
  "TotalConversationDurationMillis": 42460,
  "Sentiment": {
    "OverallSentiment": {
      "AGENT": 2.5,
      "CUSTOMER": 2.1
    },
    "SentimentByPeriod": {
      "QUARTER": {
        "AGENT": [{
          "Score": 0.0,
          "BeginOffsetMillis": 0,
          "EndOffsetMillis": 9862
        }],
        {
          "Score": -5.0,
          "BeginOffsetMillis": 9862,
          "EndOffsetMillis": 19725
        }
      }
    }
  }
}
```



```
        "Score": 5.0,  
        "BeginOffsetMillis": 19725,  
        "EndOffsetMillis": 29587  
    },  
    {  
        "Score": 5.0,  
        "BeginOffsetMillis": 29587,  
        "EndOffsetMillis": 39450  
    }  
],  
"CUSTOMER": [{  
    "Score": -2.5,  
    "BeginOffsetMillis": 0,  
    "EndOffsetMillis": 10615  
},  
{  
    "Score": 5.0,  
    "BeginOffsetMillis": 10615,  
    "EndOffsetMillis": 21230  
},  
{  
    "Score": 2.5,  
    "BeginOffsetMillis": 21230,  
    "EndOffsetMillis": 31845  
},  
{  
    "Score": 5.0,  
    "BeginOffsetMillis": 31845,  
    "EndOffsetMillis": 42460  
}  
]  
}  
}  
},  
"TalkSpeed": {  
    "DetailsByParticipant": {  
        "AGENT": {  
            "AverageWordsPerMinute": 150  
        },  
        "CUSTOMER": {  
            "AverageWordsPerMinute": 167  
        }  
    }  
}  
},
```

```
"TalkTime": {
  "DetailsByParticipant": {
    "AGENT": {
      "TotalTimeMillis": 32750
    },
    "CUSTOMER": {
      "TotalTimeMillis": 18010
    }
  },
  "TotalTimeMillis": 50760
},
...
}
```

Démarrage d'une transcription Call Analytics en temps réel

Avant de commencer une transcription des analyses d'appels en temps réel, vous devez créer toutes les [catégories](#) que vous Amazon Transcribe souhaitez associer à votre appel.

Note


Les transcriptions Call Analytics ne peuvent pas être associées rétroactivement à de nouvelles catégories. Seules les catégories que vous créez avant de commencer une transcription Call Analytics peuvent être appliquées à cette sortie de transcription.

Si vous avez créé une ou plusieurs catégories et que votre fichier audio répond à toutes les règles d'au moins une de vos catégories, Amazon Transcribe indique à votre sortie les catégories correspondantes. Si vous choisissez de ne pas utiliser de catégories, ou si votre contenu audio ne correspond pas aux règles spécifiées dans vos catégories, votre transcription n'est pas signalée.

Pour inclure des analyses après appel dans votre transcription Call Analytics en temps réel, vous devez fournir un compartiment Amazon S3 dans votre demande à l'aide du paramètre `OutputLocation`. Vous devez également inclure un ARN `DataAccessRoleArn` disposant d'autorisations d'écriture sur le compartiment spécifié. Une transcription distincte est produite et stockée dans le compartiment spécifié à la fin de votre session de streaming Call Analytics en temps réel.

Grâce à l'analyse Call Analytics en temps réel, vous avez également la possibilité de créer des alertes de catégorie en temps réel. Pour obtenir des instructions, consultez la section [Création d'alertes en temps réel pour les correspondances de catégorie](#).


Pour démarrer une transcription Call Analytics en temps réel, vous pouvez utiliser le AWS Management Console HTTP/2 ou WebSockets; consultez les exemples suivants :

 Important

Actuellement, le AWS Management Console seul propose une démo d'analyse des appels en temps réel avec des exemples audio préchargés. Si vous souhaitez utiliser votre propre audio, vous devez utiliser l'API (HTTP/2 ou un SDK). WebSockets

AWS Management Console

Utilisez la procédure suivante pour démarrer une demande Call Analytics. Les appels qui correspondent à toutes les caractéristiques définies par une catégorie sont étiquetés avec cette catégorie.

 Note

Une démo est disponible dans la AWS Management Console uniquement. Pour démarrer une transcription d'analyse personnalisée en temps réel, vous devez utiliser l'[API](#).

1. Dans le volet de navigation, sous Analyse des Amazon Transcribe appels, choisissez Analyser un appel en temps réel.


Amazon Transcribe > Real-time Analytics

Real-time Analytics [Info](#)

Transcribe Real-time Call Analytics combines powerful speech-to-text and natural language processing (NLP) models that are trained specifically to understand customer service and sales calls. With Transcribe Call Analytics, developers can get a redacted and unredacted transcript, and insights such as customer and agent sentiment, detected issues, and supervisor alerts during the live call.

How it works

This demo experience has been configured to use preloaded audio examples of customer-agent interactions. Before starting the demo, you can optionally create categories in the Category Management page and update content redaction settings under the advance settings




Step 1: Specify input audio

Input audio file

Insurance complaints (en-US) ▼


▶ 00:00/00:00



Step 2: Review call categories - optional

Categorize your calls based on custom keywords or phrases.

[View categories](#)



Step 3: Configure output - optional

Apply content redaction settings to your calls.

[Configure advanced settings](#)

Post-call Analytics

Post-call analytics enabled with real-time analytics provides consolidated transcript and audio backup, with the associated analytics, along with further insights such as call summaries and conversation characteristics like non-talk time, interruptions, loudness, and talk speed, after the end of the call in the provided Amazon S3 bucket.

Post-call Analytics

[Start streaming](#)

- Pour l'étape 1 : Spécifier l'audio d'entrée, choisissez un fichier de test de démonstration dans le menu déroulant.



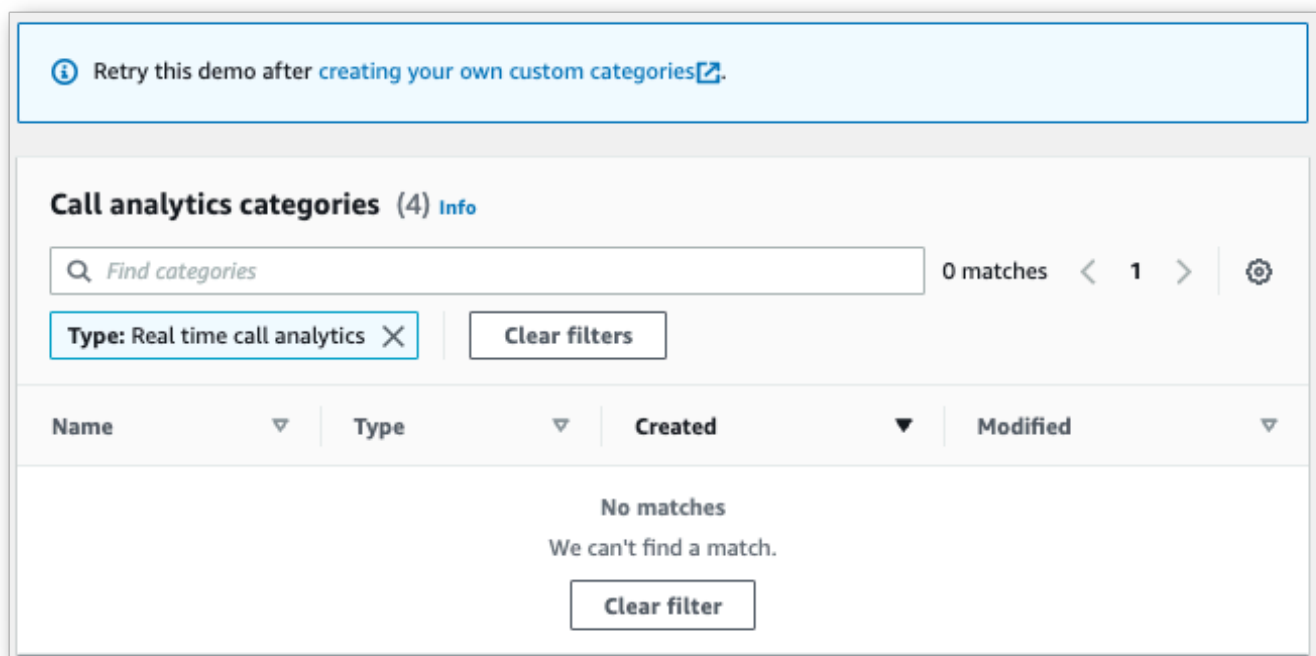
Step 1: Specify input audio

Input audio file

Insurance complaints (en-US)	▲
Insurance complaints (en-US)	✓
Hospitality complaints (en-US)	

- Pour l'étape 2 : Examiner les catégories d'appels, vous avez la possibilité de consulter les catégories Call Analytics en temps réel précédemment créées. Toutes les catégories Call Analytics en temps réel sont appliquées à votre transcription.

Si vous choisissez Afficher les catégories, un nouveau volet affiche vos catégories Call Analytics en temps réel existantes et fournit un lien pour en créer de nouvelles.



4. Pour l'étape 3 : Configurer l'entrée et la sortie, vous avez la possibilité d'appliquer des paramètres supplémentaires.

Si vous choisissez Configurer les paramètres avancés, un nouveau volet s'affiche, dans lequel vous pouvez définir les paramètres d'expurgation du contenu.

Use the following options to identify or redact content from your transcript. Other settings such as Custom Vocabulary, Custom Language Models, Partial results stabilization, Vocabulary Filtering are available through the API, SDK, CLI

▼ Content removal

PII Identification & redaction [Info](#)

Identify or redact one or more types of personally identifiable information (PII) in your transcript

Select PII detection type

Identification only

Label the type of PII identified but not redact it in the transcription output

Identification & redaction

Label the type of PII and also mask the content with the PII entity type in the transcription output. For example, (123)456-7890 will be masked as [PHONE]

Select PII entity types (11 of 11 selected)

Select All

Financial (6 of 6 selected)

BANK_ACCOUNT_NUMBER

BANK_ROUTING

CREDIT_DEBIT_NUMBER

CREDIT_DEBIT_CVV

CREDIT_DEBIT_EXPIRY

PIN

Personal (5 of 5 selected)


NAME

ADDRESS

PHONE

EMAIL

SSN

 The updates that you make here will only be applied when you start stream again.

Cancel

Save

Une fois que vous avez effectué toutes vos sélections, choisissez Enregistrer pour revenir à la page principale.

- Pour appliquer des analyses supplémentaires, vous pouvez activer les analyses après appel. Vous obtenez ainsi les mêmes analyses qu'une transcription d'analyse après appel, notamment en ce qui concerne les interruptions, le niveau sonore, le temps de non-conversation, la vitesse de conversation, le temps de conversation, les problèmes, les mesures à prendre et les résultats. Les résultats d'analyse après appel sont stockés dans un fichier distinct de votre transcription Call Analytics en temps réel.

Post-call Analytics

Post-call analytics enabled with real-time analytics provides consolidated transcript and audio backup, with the associated analytics, along with further insights such as call summaries and conversation characteristics like non-talk time, interruptions, loudness, and talk speed, after the end of the call in the provided Amazon S3 bucket.

Post-call Analytics

Si vous appliquez des analyses après un appel, vous devez spécifier une destination du fichier Amazon S3 de sortie et un IAM rôle. Vous pouvez éventuellement choisir de chiffrer votre sortie.

Post-call Analytics
Post-call analytics enabled with real-time analytics provides consolidated transcript and audio backup, with the associated analytics, along with further insights such as call summaries and conversation characteristics like non-talk time, interruptions, loudness, and talk speed, after the end of the call in the provided Amazon S3 bucket.

Post-call Analytics

Output file destination on S3 [Info](#)
Choose the location to store the output of the post-call analytics. If you input a location in an Amazon S3 bucket that doesn't yet exist, it will be created for you.

Resource URI

Format: s3://bucket, s3://bucket/prefix/, or s3://bucket/prefix/object.

Encryption [Info](#)

IAM role [Info](#)

[Create an IAM role](#) that grants access to the output bucket and KMS key (if specified) with the trust policy shown below
▶ Trust Policy

6. Choisissez Démarrer le streaming.

flux HTTP/2

Cet exemple crée une demande HTTP/2 avec Call Analytics activé. Pour plus d'informations sur l'utilisation du streaming HTTP/2 avec Amazon Transcribe, consultez [Configuration d'un flux HTTP/2](#). Pour plus de détails sur les paramètres et les en-têtes spécifiques à Amazon Transcribe, voir [StartCallAnalyticsStreamTranscription](#).

Cet exemple inclut une [analyse après appel](#). Si vous ne souhaitez pas d'analyse après appel, supprimez la section `PostCallAnalyticsSettings` de la demande.

Notez que l'événement de configuration présenté dans l'exemple suivant doit être transmis en tant que premier événement du flux.

```
POST /stream-transcription HTTP/2
host: transcribestreaming.us-west-2.amazonaws.com
X-Amz-Target: com.amazonaws.transcribe.Transcribe.StartCallAnalyticsStreamTranscription
Content-Type: application/vnd.amazon.eventstream
X-Amz-Content-Sha256: string
X-Amz-Date: 20220208T235959Z
Authorization: AWS4-HMAC-SHA256 Credential=access-key/20220208/us-west-2/transcribe/
aws4_request, SignedHeaders=content-type;host;x-amz-content-sha256;x-amz-date;x-amz-
target;x-amz-security-token, Signature=string
```

```
x-amzn-transcribe-language-code: en-US
x-amzn-transcribe-media-encoding: flac
x-amzn-transcribe-sample-rate: 16000
transfer-encoding: chunked

{
  "AudioStream": {
    "AudioEvent": {
      "AudioChunk": blob
    },
    "ConfigurationEvent": {
      "ChannelDefinitions": [
        {
          "ChannelId": 0,
          "ParticipantRole": "AGENT"
        },
        {
          "ChannelId": 1,
          "ParticipantRole": "CUSTOMER"
        }
      ],
      "PostCallAnalyticsSettings": {
        "OutputLocation": "s3://DOC-EXAMPLE-BUCKET/my-output-files/",
        "DataAccessRoleArn": "arn:aws:iam::111122223333:role/ExampleRole"
      }
    }
  }
}
```

Les définitions des paramètres se trouvent dans la [référence d'API](#) ; les paramètres communs à toutes les opérations d' AWS API sont répertoriés dans la section [Paramètres communs](#).

WebSocket flux

Cet exemple crée une URL présignée qui utilise Call Analytics dans un WebSocket flux. Les sauts de ligne ont été ajoutés pour faciliter la lecture. Pour plus d'informations sur l'utilisation WebSocket des flux avec Amazon Transcribe, consultez [Configuration d'un WebSocket stream](#). Pour plus de détails sur les paramètres, consultez la section [StartCallAnalyticsStreamTranscription](#).

Cet exemple inclut une [analyse après appel](#). Si vous ne souhaitez pas d'analyse après appel, supprimez la section `PostCallAnalyticsSettings` de la demande.

Notez que l'événement de configuration présenté dans l'exemple suivant doit être transmis en tant que premier événement du flux.

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/call-analytics-stream-
transcription-websocket?
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-
west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=string
&X-Amz-SignedHeaders=content-type%3Bhost%3Bx-amz-date
&language-code=en-US
&media-encoding=flac
&sample-rate=16000

{
  "AudioStream": {
    "AudioEvent": {
      "AudioChunk": blob
    },
    "ConfigurationEvent": {
      "ChannelDefinitions": [
        {
          "ChannelId": 0,
          "ParticipantRole": "AGENT"
        },
        {
          "ChannelId": 1,
          "ParticipantRole": "CUSTOMER"
        }
      ],
      "PostCallAnalyticsSettings": {
        "OutputLocation": "s3://DOC-EXAMPLE-BUCKET/my-output-files/",
        "DataAccessRoleArn": "arn:aws:iam::111122223333:role/ExampleRole"
      }
    }
  }
}
```

Les définitions des paramètres se trouvent dans la [référence d'API](#) ; les paramètres communs à toutes les opérations d' AWS API sont répertoriés dans la section [Paramètres communs](#).

i Tip

Le HTTP/2 et les WebSocket exemples ci-dessus incluent des analyses post-appel. Si vous ne souhaitez pas d'analyse après appel, supprimez la section `PostCallAnalyticsSettings` de la demande.

Si vous activez `PostCallAnalyticsSettings`, vous devez envoyer un événement de configuration en tant que premier événement. Votre événement de configuration inclut les paramètres pour `ChannelDefinitions` et `PostStreamAnalyticsSettings`, comme indiqué dans les exemples précédents.

Les données binaires sont transmises sous forme de message binaire avec `content-type application/octet-stream`, et l'événement de configuration est transmis sous forme de message texte avec `content-type application/json`.

Pour plus d'informations, consultez [Configuration d'une transcription en streaming](#).

Création d'alertes en temps réel pour les correspondances de catégorie

Pour configurer des alertes en temps réel, vous devez d'abord créer une catégorie [TranscriptFilterType](#) avec l'indicateur `REAL_TIME`. Cet indicateur permet d'appliquer votre catégorie aux transcriptions Call Analytics en temps réel.

Pour obtenir des instructions sur la création d'une catégorie, consultez la section [Création de catégories pour les transcriptions en temps réel](#).

Lorsque vous démarrez votre transcription Call Analytics en temps réel, toutes les catégories avec l'indicateur `REAL_TIME` sont automatiquement appliquées à votre sortie de transcription au niveau du segment. Si une correspondance `TranscriptFilterType` se produit, elle apparaît dans la section `CategoryEvent` de votre transcription. Vous pouvez ensuite utiliser ce paramètre et ses sous-paramètres `MatchedCategories` et `MatchedDetails` pour configurer des alertes en temps réel personnalisées.

Voici un exemple de sortie de transcription Call Analytics en temps réel pour une correspondance `CategoryEvent` :

```
"CategoryEvent": {
  "MatchedCategories": [ "shipping-complaint" ],
  "MatchedDetails": {
```

```
    "my package never arrived" : {  
      "TimestampRanges": [  
        {  
          "BeginOffsetMillis": 19010,  
          "EndOffsetMillis": 22690  
        }  
      ]  
    }  
  },  
},
```

L'exemple précédent représente une correspondance textuelle exacte avec le discours « mon colis n'est jamais arrivé », qui représente une règle relevant de la catégorie « plainte relative à l'expédition ».

Vous pouvez configurer votre alerte en temps réel pour inclure n'importe quelle combinaison de paramètres répertoriés. Par exemple, vous pouvez configurer votre alerte pour inclure uniquement la phrase correspondante (`MatchedDetails`) ou uniquement le nom de catégorie (`MatchedCategories`). Vous pouvez également configurer votre alerte pour inclure tous les paramètres.

La façon dont vous configurez vos alertes en temps réel dépend des interfaces de votre organisation et du type d'alerte que vous souhaitez. Par exemple, vous pouvez définir une correspondance `CategoryEvent` pour envoyer une notification contextuelle, un e-mail, un SMS ou toute autre alerte que votre système peut accepter.

Sortie Call Analytics en temps réel

Les transcriptions d'analyse des appels en temps réel sont affichées dans un turn-by-turn format par segment. Elles incluent des événements de catégorie, la détection des problèmes, les sentiments, ainsi que l'identification et l'expurgation des PII. Les événements de catégorie vous permettent de définir des alertes en temps réel. Pour plus d'informations, consultez la section [Création d'alertes en temps réel pour les correspondances de catégorie](#).

Pour améliorer la précision et personnaliser davantage vos transcriptions en fonction de votre cas d'utilisation, par exemple en incluant des termes spécifiques au secteur, ajoutez des [vocabulaires personnalisés](#) ou des [modèles de langue personnalisés](#) à votre demande Call Analytics. Pour masquer, supprimer ou étiqueter des mots que vous ne voulez pas voir apparaître dans vos résultats de transcription, tels que des termes vulgaires, ajoutez un [filtrage du vocabulaire](#).

Les sections suivantes présentent des exemples de sortie JSON pour les transcriptions Call Analytics en temps réel.

Événements de catégorie

Voici à quoi ressemble une correspondance de catégorie dans votre sortie de transcription. Cet exemple montre que l'audio entre l'horodatage de 19 010 millisecondes et l'horodatage de 22 690 millisecondes correspond à la catégorie « plainte réseau ». Dans ce cas, la catégorie personnalisée « plainte réseau » exigeait que le client indique « problèmes de réseau » (correspondance exacte des mots).

```
"CategoryEvent": {
  "MatchedCategories": [
    "network-complaint"
  ],
  "MatchedDetails": {
    "network issues" : {
      "TimestampRanges": [
        {
          "BeginOffsetMillis": 9299375,
          "EndOffsetMillis": 7899375
        }
      ]
    }
  }
},
```

Détection des problèmes

Voici à quoi ressemble une correspondance de détection de problèmes dans votre sortie de transcription. Cet exemple montre que le texte compris entre le caractère 26 et le caractère 62 décrit un problème.

```
"UtteranceEvent": {
  ...
  "Transcript": "Wang Xiulan I'm tired of the network issues my phone is having.",
  ...
  "IssuesDetected": [
    {
      "CharacterOffsets": {
        "BeginOffsetChar": 26,
        "EndOffsetChar": 62
      }
    }
  ]
}
```

```

    }
  }
],
},

```

Sentiment

Voici à quoi ressemble l'analyse des sentiments dans votre sortie de transcription.

```

"UtteranceEvent": {
  ...
  "Sentiment": "NEGATIVE",
  "Items": [{
    ...

```

Identification des PII

Voici à quoi ressemble l'identification des PII dans votre sortie de transcription.

```

"Entities": [
  {
    "Content": "Wang Xiulan",
    "Category": "PII",
    "Type": "NAME",
    "BeginOffsetMillis": 7999375,
    "EndOffsetMillis": 199375,
    "Confidence": 0.9989
  }
],

```

Expurgation des PII

Voici à quoi ressemble l'expurgation des PII dans votre sortie de transcription.

```

"Content": "[NAME]. Hi, [NAME]. I'm [NAME] Happy to be helping you today.",
"Redaction": {
  "RedactedTimestamps": [
    {
      "BeginOffsetMillis": 32670,
      "EndOffsetMillis": 33343
    },
    {

```

```

        "BeginOffsetMillis": 33518,
        "EndOffsetMillis": 33858
    },
    {
        "BeginOffsetMillis": 34068,
        "EndOffsetMillis": 34488
    }
]
},

```

Sortie Call Analytics en temps réel compilée

Par souci de concision, une partie du contenu est remplacée par des points de suspension dans la sortie de transcription suivante.

```

{
  "CallAnalyticsTranscriptResultStream": {
    "BadRequestException": {},
    "ConflictException": {},
    "InternalFailureException": {},
    "LimitExceededException": {},
    "ServiceUnavailableException": {},
    "UtteranceEvent": {
      "UtteranceId": "58c27f92-7277-11ec-90d6-0242ac120003",
      "ParticipantRole": "CUSTOMER",
      "IsPartial": false,
      "Transcript": "Wang Xiulan I'm tired of the network issues my phone is
having.",
      "BeginOffsetMillis": 19010,
      "EndOffsetMillis": 22690,
      "Sentiment": "NEGATIVE",
      "Items": [{
        "Content": "Wang",
        "BeginOffsetMillis": 379937,
        "EndOffsetMillis": 299375,
        "Type": "pronunciation",
        "Confidence": 0.9961,
        "VocabularyFilterMatch": false
      }],
      {
        "Content": "Xiulan",
        "EndOffsetMillis": 5899375,
        "BeginOffsetMillis": 3899375,

```

```
    "Type": "pronunciation",
    "Confidence": 0.9961,
    "VocabularyFilterMatch": false
  },
  ...
  {
    "Content": "network",
    "EndOffsetMillis": 199375,
    "BeginOffsetMillis": 9299375,
    "Type": "pronunciation",
    "Confidence": 0.9961,
    "VocabularyFilterMatch": false
  },
  {
    "Content": "issues",
    "EndOffsetMillis": 7899375,
    "BeginOffsetMillis": 5999375,
    "Type": "pronunciation",
    "Confidence": 0.9961,
    "VocabularyFilterMatch": false
  },
  {
    "Content": "my",
    "EndOffsetMillis": 9199375,
    "BeginOffsetMillis": 7999375,
    "Type": "pronunciation",
    "Confidence": 0.9961,
    "VocabularyFilterMatch": false
  },
  {
    "Content": "phone",
    "EndOffsetMillis": 199375,
    "BeginOffsetMillis": 9299375,
    "Type": "pronunciation",
    "Confidence": 0.9961,
    "VocabularyFilterMatch": false
  },
  ...
],
"Entities": [{
  "Content": "Wang Xiulan",
  "Category": "PII",
  "Type": "NAME",
  "BeginOffsetMillis": 7999375,
```

```
        "EndOffsetMillis": 199375,
        "Confidence": 0.9989
    }],
    "IssuesDetected": [{
        "CharacterOffsets": {
            "BeginOffsetChar": 26,
            "EndOffsetChar": 62
        }
    }
  ],
  "CategoryEvent": {
    "MatchedCategories": [
      "network-complaint"
    ],
    "MatchedDetails": {
      "network issues" : {
        "TimestampRanges": [
          {
            "BeginOffsetMillis": 9299375,
            "EndOffsetMillis": 7899375
          }
        ]
      }
    }
  }
}
```

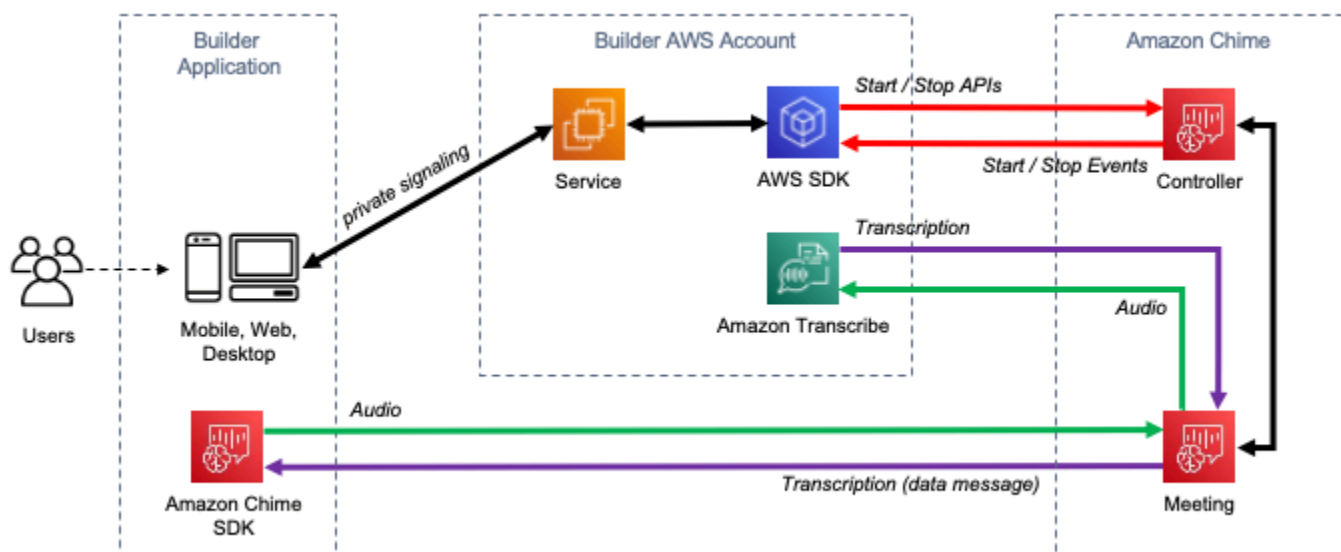

Transcription de vos Amazon Chime appels en temps réel

Amazon Transcribe est intégré au Amazon Chime SDK, ce qui facilite la transcription en temps réel de vos Amazon Chime appels.

Lorsque vous demandez une transcription à l'aide de l'API Amazon Chime SDK, le streaming audio Amazon Chime commence Amazon Transcribe et continue de le faire pendant toute la durée de l'appel.

Le Amazon Chime SDK utilise son algorithme « locuteur actif » pour sélectionner les deux principaux locuteurs actifs, puis envoie leur audio sur deux canaux distincts via un seul flux. Amazon Transcribe Les participants à la réunion reçoivent des transcriptions attribuées par l'utilisateur via des messages de données du Amazon Chime SDK. Vous pouvez consulter des exemples de livraison dans le [guide du développeur du Amazon Chime SDK](#).

Le flux de données d'une Amazon Chime transcription est illustré dans le schéma suivant :



Pour plus d'informations et des instructions détaillées sur la configuration des Amazon Chime transcriptions en temps réel, reportez-vous à la section [Utilisation de la transcription en direct du Amazon Chime SDK](#) dans le Guide du développeur du Amazon Chime SDK. Pour les opérations d'API, reportez-vous à la [référence de l'API Amazon Chime SDK](#).

 Approfondissez vos connaissances grâce au blog sur AWS Machine Learning

Pour en savoir plus sur l'amélioration de la précision grâce à des transcriptions en temps réel, consultez :

- [Amazon Chime Les réunions du SDK prennent désormais en charge la transcription en direct avec Amazon Transcribe et Amazon Transcribe Medical](#)
- [Amazon Chime SDK pour solution de télémédecine](#)

Exemples de code pour Amazon Transcribe à l'aide de kits de développement logiciel AWS

Les exemples de code suivants montrent comment utiliser Amazon Transcribe avec un kit de développement AWS logiciel (SDK).

Les actions sont des extraits de code de programmes plus larges et doivent être exécutées dans leur contexte. Alors que les actions vous indiquent comment appeler des fonctions de service individuelles, vous pouvez les voir en contexte dans leurs scénarios associés et dans des exemples interservices.

Les Scénarios sont des exemples de code qui vous montrent comment accomplir une tâche spécifique en appelant plusieurs fonctions au sein d'un même service.

Les Exemples de services croisés sont des exemples d'applications fonctionnant sur plusieurs Services AWS.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultez [Utilisation de ce service avec un AWS SDK](#). Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes du kit de développement logiciel (SDK).

Exemples de code

- [Actions pour Amazon Transcribe à l'aide des SDK AWS](#)
 - [Utilisation CreateVocabulary avec un AWS SDK ou une CLI](#)
 - [Utilisation DeleteMedicalTranscriptionJob avec un AWS SDK ou une CLI](#)
 - [Utilisation DeleteTranscriptionJob avec un AWS SDK ou une CLI](#)
 - [Utilisation DeleteVocabulary avec un AWS SDK ou une CLI](#)
 - [Utilisation GetTranscriptionJob avec un AWS SDK ou une CLI](#)
 - [Utilisation GetVocabulary avec un AWS SDK ou une CLI](#)
 - [Utilisation ListMedicalTranscriptionJobs avec un AWS SDK ou une CLI](#)
 - [Utilisation ListTranscriptionJobs avec un AWS SDK ou une CLI](#)
 - [Utilisation ListVocabularies avec un AWS SDK ou une CLI](#)
 - [Utilisation StartMedicalTranscriptionJob avec un AWS SDK ou une CLI](#)
 - [Utilisation StartStreamTranscriptionAsync avec un AWS SDK ou une CLI](#)

- [Utilisation StartTranscriptionJob avec un AWS SDK ou une CLI](#)
- [Utilisation UpdateVocabulary avec un AWS SDK ou une CLI](#)
- [Scénarios pour Amazon Transcribe à l'aide de kits de développement logiciel AWS](#)
 - [Créez et affinez un vocabulaire personnalisé Amazon Transcribe à l'aide d'un SDK AWS](#)
 - [Transcrivez du son et obtenez des données de travail avec Amazon Transcribe à l'aide d'un SDK AWS](#)
- [Exemples multiservices pour Amazon AWS Transcribe utilisant des SDK](#)
 - [Créer une application Amazon Transcribe](#)
 - [Créer une application de streaming Amazon Transcribe](#)
 - [Convertissez du texte en parole et de nouveau en texte à l'aide d'un AWS SDK](#)

Actions pour Amazon Transcribe à l'aide des SDK AWS

Les exemples de code suivants montrent comment effectuer des actions Amazon Transcribe individuelles à l'aide AWS des SDK. Ces extraits appellent l'API Amazon Transcribe et sont des extraits de code de programmes plus volumineux qui doivent être exécutés en contexte. Chaque exemple inclut un lien vers GitHub, où vous pouvez trouver des instructions pour configurer et exécuter le code.

Les exemples suivants incluent uniquement les actions les plus couramment utilisées. Pour obtenir la liste complète, veuillez consulter la [référence de l'API Amazon Transcribe](#).

Exemples

- [Utilisation CreateVocabulary avec un AWS SDK ou une CLI](#)
- [Utilisation DeleteMedicalTranscriptionJob avec un AWS SDK ou une CLI](#)
- [Utilisation DeleteTranscriptionJob avec un AWS SDK ou une CLI](#)
- [Utilisation DeleteVocabulary avec un AWS SDK ou une CLI](#)
- [Utilisation GetTranscriptionJob avec un AWS SDK ou une CLI](#)
- [Utilisation GetVocabulary avec un AWS SDK ou une CLI](#)
- [Utilisation ListMedicalTranscriptionJobs avec un AWS SDK ou une CLI](#)
- [Utilisation ListTranscriptionJobs avec un AWS SDK ou une CLI](#)
- [Utilisation ListVocabularies avec un AWS SDK ou une CLI](#)

- [Utilisation StartMedicalTranscriptionJob avec un AWS SDK ou une CLI](#)
- [Utilisation StartStreamTranscriptionAsync avec un AWS SDK ou une CLI](#)
- [Utilisation StartTranscriptionJob avec un AWS SDK ou une CLI](#)
- [Utilisation UpdateVocabulary avec un AWS SDK ou une CLI](#)

Utilisation **CreateVocabulary** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser `CreateVocabulary`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- [Création et affinage d'un vocabulaire personnalisé](#)

.NET

AWS SDK for .NET

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

```
/// <summary>
/// Create a custom vocabulary using a list of phrases. Custom vocabularies
/// improve transcription accuracy for one or more specific words.
/// </summary>
/// <param name="languageCode">The language code of the vocabulary.</param>
/// <param name="phrases">Phrases to use in the vocabulary.</param>
/// <param name="vocabularyName">Name for the vocabulary.</param>
/// <returns>The state of the custom vocabulary.</returns>
public async Task<VocabularyState> CreateCustomVocabulary(LanguageCode
languageCode,
    List<string> phrases, string vocabularyName)
{
    var response = await _amazonTranscribeService.CreateVocabularyAsync(
```

```
        new CreateVocabularyRequest
        {
            LanguageCode = languageCode,
            Phrases = phrases,
            VocabularyName = vocabularyName
        });
    return response.VocabularyState;
}
```

- Pour plus de détails sur l'API, reportez-vous [CreateVocabulary](#) à la section Référence des AWS SDK for .NET API.

CLI

AWS CLI

Pour créer un vocabulaire personnalisé

L'exemple `create-vocabulary` suivant crée un vocabulaire personnalisé. Pour créer un vocabulaire personnalisé, vous devez avoir créé un fichier texte contenant tous les termes que vous souhaitez transcrire de manière plus précise. Pour `vocabulary-file-uri`, spécifiez l'URI Amazon Simple Storage Service (Amazon S3) de ce fichier texte. Pour le code de langue, spécifiez un code de langue correspondant à la langue du vocabulaire personnalisé. Pour le nom de vocabulaire, spécifiez le nom que vous souhaitez donner au vocabulaire personnalisé.

```
aws transcribe create-vocabulary \
  --language-code language-code \
  --vocabulary-name cli-vocab-example \
  --vocabulary-file-uri s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/the-text-file-
for-the-custom-vocabulary.txt
```

Sortie :

```
{
  "VocabularyName": "cli-vocab-example",
  "LanguageCode": "language-code",
  "VocabularyState": "PENDING"
}
```

Pour plus d'informations, consultez [Vocabulaires personnalisés](#) dans le Guide du développeur Amazon Transcribe.

- Pour plus de détails sur l'API, reportez-vous [CreateVocabulary](#) à la section Référence des AWS CLI commandes.

Python

SDK pour Python (Boto3)

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

```
def create_vocabulary(
    vocabulary_name, language_code, transcribe_client, phrases=None,
    table_uri=None
):
    """
    Creates a custom vocabulary that can be used to improve the accuracy of
    transcription jobs. This function returns as soon as the vocabulary
    processing
    is started. Call get_vocabulary to get the current status of the vocabulary.
    The vocabulary is ready to use when its status is 'READY'.

    :param vocabulary_name: The name of the custom vocabulary.
    :param language_code: The language code of the vocabulary.
        For example, en-US or nl-NL.
    :param transcribe_client: The Boto3 Transcribe client.
    :param phrases: A list of comma-separated phrases to include in the
    vocabulary.
    :param table_uri: A table of phrases and pronunciation hints to include in
    the
        vocabulary.
    :return: Information about the newly created vocabulary.
    """
    try:
        vocab_args = {"VocabularyName": vocabulary_name, "LanguageCode":
        language_code}
        if phrases is not None:
```

```
        vocab_args["Phrases"] = phrases
    elif table_uri is not None:
        vocab_args["VocabularyFileUri"] = table_uri
        response = transcribe_client.create_vocabulary(**vocab_args)
        logger.info("Created custom vocabulary %s.", response["VocabularyName"])
    except ClientError:
        logger.exception("Couldn't create custom vocabulary %s.",
            vocabulary_name)
        raise
    else:
        return response
```

- Pour plus de détails sur l'API, consultez [CreateVocabulary](#) le AWS manuel de référence de l'API SDK for Python (Boto3).

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultez [Utilisation de ce service avec un AWS SDK](#). Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **DeleteMedicalTranscriptionJob** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser `DeleteMedicalTranscriptionJob`.

.NET

AWS SDK for .NET

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

```
/// <summary>
```



```
    /// Delete a medical transcription job. Also deletes the transcript
    /// associated with the job.
    /// </summary>
    /// <param name="jobName">Name of the medical transcription job to delete.</
param>
    /// <returns>True if successful.</returns>
    public async Task<bool> DeleteMedicalTranscriptionJob(string jobName)
    {
        var response = await
        _amazonTranscribeService.DeleteMedicalTranscriptionJobAsync(
            new DeleteMedicalTranscriptionJobRequest()
            {
                MedicalTranscriptionJobName = jobName
            });
        return response.HttpStatusCode == HttpStatusCode.OK;
    }
}
```

- Pour plus de détails sur l'API, reportez-vous [DeleteMedicalTranscriptionJob](#) à la section Référence des AWS SDK for .NET API.

CLI

AWS CLI

Pour supprimer une tâche de transcription médicale

L'exemple `delete-medical-transcription-job` suivant crée une tâche de transcription médicale.

```
aws transcribe delete-medical-transcription-job \
  --medical-transcription-job-name medical-transcription-job-name
```

Cette commande ne produit aucun résultat.

Pour plus d'informations, consultez [DeleteMedicalTranscriptionJob](#) le guide du développeur Amazon Transcribe.

- Pour plus de détails sur l'API, reportez-vous [DeleteMedicalTranscriptionJob](#) à la section Référence des AWS CLI commandes.

JavaScript

SDK pour JavaScript (v3)

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

Créez le client.

```
import { TranscribeClient } from "@aws-sdk/client-transcribe";
// Set the AWS Region.
const REGION = "REGION"; //e.g. "us-east-1"
// Create an Amazon Transcribe service client object.
const transcribeClient = new TranscribeClient({ region: REGION });
export { transcribeClient };
```

Supprimez une tâche de transcription médicale.

```
// Import the required AWS SDK clients and commands for Node.js
import { DeleteMedicalTranscriptionJobCommand } from "@aws-sdk/client-transcribe";
import { transcribeClient } from "../libs/transcribeClient.js";

// Set the parameters
export const params = {
  MedicalTranscriptionJobName: "MEDICAL_JOB_NAME", // For example,
  'medical_transcription_demo'
};

export const run = async () => {
  try {
    const data = await transcribeClient.send(
      new DeleteMedicalTranscriptionJobCommand(params)
    );
    console.log("Success - deleted");
    return data; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
```

```
}  
};  
run();
```

- Pour de plus amples informations, consultez le [Guide du développeur AWS SDK for JavaScript](#).
- Pour plus de détails sur l'API, reportez-vous [DeleteMedicalTranscriptionJob](#) à la section Référence des AWS SDK for JavaScript API.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultez [Utilisation de ce service avec un AWS SDK](#). Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **DeleteTranscriptionJob** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser `DeleteTranscriptionJob`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- [Création et affinage d'un vocabulaire personnalisé](#)

.NET

AWS SDK for .NET

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

```
/// <summary>  
/// Delete a transcription job. Also deletes the transcript associated with  
the job.
```

```
/// </summary>
/// <param name="jobName">Name of the transcription job to delete.</param>
/// <returns>True if successful.</returns>
public async Task<bool> DeleteTranscriptionJob(string jobName)
{
    var response = await
_amazonTranscribeService.DeleteTranscriptionJobAsync(
    new DeleteTranscriptionJobRequest()
    {
        TranscriptionJobName = jobName
    });
    return response.HttpStatusCode == HttpStatusCode.OK;
}
```

- Pour plus de détails sur l'API, reportez-vous [DeleteTranscriptionJob](#) à la section Référence des AWS SDK for .NET API.

CLI

AWS CLI

Pour supprimer l'une des tâches de transcription

L'exemple `delete-transcription-job` suivant supprime l'une des tâches de transcription.

```
aws transcribe delete-transcription-job \
  --transcription-job-name your-transcription-job
```

Cette commande ne produit aucun résultat.

Pour plus d'informations, consultez [DeleteTranscriptionJob](#) le guide du développeur Amazon Transcribe.

- Pour plus de détails sur l'API, reportez-vous [DeleteTranscriptionJob](#) à la section Référence des AWS CLI commandes.

JavaScript

SDK pour JavaScript (v3)

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

Supprimez une tâche de transcription.

```
// Import the required AWS SDK clients and commands for Node.js
import { DeleteTranscriptionJobCommand } from "@aws-sdk/client-transcribe";
import { transcribeClient } from "../libs/transcribeClient.js";

// Set the parameters
export const params = {
  TranscriptionJobName: "JOB_NAME", // Required. For example, 'transcription_demo'
};

export const run = async () => {
  try {
    const data = await transcribeClient.send(
      new DeleteTranscriptionJobCommand(params)
    );
    console.log("Success - deleted");
    return data; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Créez le client.

```
import { TranscribeClient } from "@aws-sdk/client-transcribe";
// Set the AWS Region.
const REGION = "REGION"; //e.g. "us-east-1"
// Create an Amazon Transcribe service client object.
```

```
const transcribeClient = new TranscribeClient({ region: REGION });
export { transcribeClient };
```

- Pour de plus amples informations, consultez le [Guide du développeur AWS SDK for JavaScript](#).
- Pour plus de détails sur l'API, reportez-vous [DeleteTranscriptionJob](#) à la section Référence des AWS SDK for JavaScript API.

Python

SDK pour Python (Boto3)

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

```
def delete_job(job_name, transcribe_client):
    """
    Deletes a transcription job. This also deletes the transcript associated with
    the job.

    :param job_name: The name of the job to delete.
    :param transcribe_client: The Boto3 Transcribe client.
    """
    try:
        transcribe_client.delete_transcription_job(TranscriptionJobName=job_name)
        logger.info("Deleted job %s.", job_name)
    except ClientError:
        logger.exception("Couldn't delete job %s.", job_name)
        raise
```

- Pour plus de détails sur l'API, consultez [DeleteTranscriptionJob](#) le AWS manuel de référence de l'API SDK for Python (Boto3).

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultez [Utilisation de ce service avec un AWS SDK](#). Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **DeleteVocabulary** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser `DeleteVocabulary`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- [Création et affinage d'un vocabulaire personnalisé](#)

.NET

AWS SDK for .NET

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

```
/// <summary>
/// Delete an existing custom vocabulary.
/// </summary>
/// <param name="vocabularyName">Name of the vocabulary to delete.</param>
/// <returns>True if successful.</returns>
public async Task<bool> DeleteCustomVocabulary(string vocabularyName)
{
    var response = await _amazonTranscribeService.DeleteVocabularyAsync(
        new DeleteVocabularyRequest
        {
            VocabularyName = vocabularyName
        });
    return response.HttpStatusCode == HttpStatusCode.OK;
}
```

- Pour plus de détails sur l'API, reportez-vous [DeleteVocabulary](#) à la section Référence des AWS SDK for .NET API.

CLI

AWS CLI

Pour supprimer un vocabulaire personnalisé

L'exemple `delete-vocabulary` suivant supprime un vocabulaire personnalisé.

```
aws transcribe delete-vocabulary \  
  --vocabulary-name vocabulary-name
```

Cette commande ne produit aucun résultat.

Pour plus d'informations, consultez [Vocabulaires personnalisés](#) dans le Guide du développeur Amazon Transcribe.

- Pour plus de détails sur l'API, reportez-vous [DeleteVocabulary](#) à la section Référence des AWS CLI commandes.

Python

SDK pour Python (Boto3)

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

```
def delete_vocabulary(vocabulary_name, transcribe_client):  
    """  
    Deletes a custom vocabulary.  
  
    :param vocabulary_name: The name of the vocabulary to delete.  
    :param transcribe_client: The Boto3 Transcribe client.  
    """
```



```
try:
    transcribe_client.delete_vocabulary(VocabularyName=vocabulary_name)
    logger.info("Deleted vocabulary %s.", vocabulary_name)
except ClientError:
    logger.exception("Couldn't delete vocabulary %s.", vocabulary_name)
    raise
```

- Pour plus de détails sur l'API, consultez [DeleteVocabulary](#) le AWS manuel de référence de l'API SDK for Python (Boto3).

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultez [Utilisation de ce service avec un AWS SDK](#). Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **GetTranscriptionJob** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser `GetTranscriptionJob`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans les exemples de code suivants :

- [Création et affinage d'un vocabulaire personnalisé](#)
- [Transcription d'un fichier audio et obtention de données sur la tâche](#)

.NET

AWS SDK for .NET

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

```
/// <summary>
/// Get details about a transcription job.
/// </summary>
/// <param name="jobName">A unique name for the transcription job.</param>
/// <returns>A TranscriptionJob instance with information on the requested
job.</returns>
public async Task<TranscriptionJob> GetTranscriptionJob(string jobName)
{
    var response = await _amazonTranscribeService.GetTranscriptionJobAsync(
        new GetTranscriptionJobRequest()
        {
            TranscriptionJobName = jobName
        });
    return response.TranscriptionJob;
}
```

- Pour plus de détails sur l'API, reportez-vous [GetTranscriptionJob](#) à la section Référence des AWS SDK for .NET API.

CLI

AWS CLI

Pour obtenir des informations sur une tâche de transcription spécifique

L'exemple `get-transcription-job` suivant permet d'obtenir des informations sur une tâche de transcription spécifique. Pour accéder aux résultats de la transcription, utilisez le `TranscriptFileUri` paramètre. Utilisez le `MediaFileUri` paramètre pour voir quel fichier audio vous avez transcrit avec cette tâche. Vous pouvez utiliser l'objet Paramètres pour voir les fonctionnalités facultatives que vous avez activées dans la tâche de transcription.

```
aws transcribe get-transcription-job \
  --transcription-job-name your-transcription-job
```

Sortie :

```
{
  "TranscriptionJob": {
    "TranscriptionJobName": "your-transcription-job",
```

```
"TranscriptionJobStatus": "COMPLETED",
"LanguageCode": "language-code",
"MediaSampleRateHertz": 48000,
"MediaFormat": "mp4",
"Media": {
  "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.file-
extension"
},
"Transcript": {
  "TranscriptFileUri": "https://Amazon-S3-file-location-of-
transcription-output"
},
"StartTime": "2020-09-18T22:27:23.970000+00:00",
"CreationTime": "2020-09-18T22:27:23.948000+00:00",
"CompletionTime": "2020-09-18T22:28:21.197000+00:00",
"Settings": {
  "ChannelIdentification": false,
  "ShowAlternatives": false
},
"IdentifyLanguage": true,
"IdentifiedLanguageScore": 0.8672199249267578
}
```

Pour plus d'informations, consultez [Getting Started \(interface de ligne de AWS commande\)](#) dans le manuel Amazon Transcribe Developer Guide.

- Pour plus de détails sur l'API, reportez-vous [GetTranscriptionJob](#) à la section Référence des AWS CLI commandes.

Python

SDK pour Python (Boto3)

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

```
def get_job(job_name, transcribe_client):
```

```
"""
Gets details about a transcription job.

:param job_name: The name of the job to retrieve.
:param transcribe_client: The Boto3 Transcribe client.
:return: The retrieved transcription job.
"""
try:
    response = transcribe_client.get_transcription_job(
        TranscriptionJobName=job_name
    )
    job = response["TranscriptionJob"]
    logger.info("Got job %s.", job["TranscriptionJobName"])
except ClientError:
    logger.exception("Couldn't get job %s.", job_name)
    raise
else:
    return job
```

- Pour plus de détails sur l'API, consultez [GetTranscriptionJob](#) le AWS manuel de référence de l'API SDK for Python (Boto3).

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultez [Utilisation de ce service avec un AWS SDK](#). Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **GetVocabulary** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser `GetVocabulary`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- [Création et affinage d'un vocabulaire personnalisé](#)

.NET

AWS SDK for .NET

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

```
/// <summary>
/// Get information about a custom vocabulary.
/// </summary>
/// <param name="vocabularyName">Name of the vocabulary.</param>
/// <returns>The state of the custom vocabulary.</returns>
public async Task<VocabularyState> GetCustomVocabulary(string vocabularyName)
{
    var response = await _amazonTranscribeService.GetVocabularyAsync(
        new GetVocabularyRequest()
        {
            VocabularyName = vocabularyName
        });
    return response.VocabularyState;
}
```

- Pour plus de détails sur l'API, reportez-vous [GetVocabulary](#) à la section Référence des AWS SDK for .NET API.

CLI

AWS CLI

Pour obtenir des informations sur un vocabulaire personnalisé

L'exemple `get-vocabulary` suivant permet d'obtenir des informations sur un vocabulaire personnalisé précédemment créé.

```
aws transcribe get-vocabulary \
```

```
--vocabulary-name cli-vocab-1
```

Sortie :

```
{
  "VocabularyName": "cli-vocab-1",
  "LanguageCode": "language-code",
  "VocabularyState": "READY",
  "LastModifiedTime": "2020-09-19T23:22:32.836000+00:00",
  "DownloadUri": "https://link-to-download-the-text-file-used-to-create-your-
custom-vocabulary"
}
```

Pour plus d'informations, consultez [Vocabulaires personnalisés](#) dans le Guide du développeur Amazon Transcribe.

- Pour plus de détails sur l'API, reportez-vous [GetVocabulary](#) à la section Référence des AWS CLI commandes.

Python

SDK pour Python (Boto3)

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

```
def get_vocabulary(vocabulary_name, transcribe_client):
    """
    Gets information about a custom vocabulary.

    :param vocabulary_name: The name of the vocabulary to retrieve.
    :param transcribe_client: The Boto3 Transcribe client.
    :return: Information about the vocabulary.
    """
    try:
        response =
transcribe_client.get_vocabulary(VocabularyName=vocabulary_name)
        logger.info("Got vocabulary %s.", response["VocabularyName"])
```

```
except ClientError:
    logger.exception("Couldn't get vocabulary %s.", vocabulary_name)
    raise
else:
    return response
```

- Pour plus de détails sur l'API, consultez [GetVocabulary](#) le AWS manuel de référence de l'API SDK for Python (Boto3).

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultez [Utilisation de ce service avec un AWS SDK](#). Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **ListMedicalTranscriptionJobs** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser `ListMedicalTranscriptionJobs`.

.NET

AWS SDK for .NET

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

```
/// <summary>
/// List medical transcription jobs, optionally with a name filter.
/// </summary>
/// <param name="jobNameContains">Optional name filter for the medical
transcription jobs.</param>
/// <returns>A list of summaries about medical transcription jobs.</returns>
public async Task<List<MedicalTranscriptionJobSummary>>
ListMedicalTranscriptionJobs(
```

```
        string? jobNameContains = null)
    {
        var response = await
        _amazonTranscribeService.ListMedicalTranscriptionJobsAsync(
            new ListMedicalTranscriptionJobsRequest()
            {
                JobNameContains = jobNameContains
            });
        return response.MedicalTranscriptionJobSummaries;
    }
```

- Pour plus de détails sur l'API, reportez-vous [ListMedicalTranscriptionJobs](#) à la section Référence des AWS SDK for .NET API.

CLI

AWS CLI

Pour répertorier les tâches de transcription médicale

L'`list-medical-transcription-job` exemple suivant répertorie les tâches de transcription médicale associées à votre AWS compte et à votre région. Pour obtenir plus d'informations sur une tâche de transcription particulière, copiez la valeur d'un `MedicalTranscriptionJobName` paramètre dans le résultat de transcription et spécifiez cette valeur pour l'`MedicalTranscriptionJobNameoption` de la `get-medical-transcription-job` commande. Pour voir d'autres tâches de transcription, copiez la valeur du `NextToken` paramètre, réexécutez la `list-medical-transcription-jobs` commande et spécifiez cette valeur dans l'`--next-tokenoption`.

```
aws transcribe list-medical-transcription-jobs
```

Sortie :

```
{
  "NextToken": "3/PblzkiGhzjER3KHuQt2fmbPLF7cDYafjFMEoGn440N/
gsuUSTIkGyanvRE6WMXfd/ZTEc2EZj+P9eii/
z102FDYli6RLI0WoRX4RwMisVrh9G0Kie0Y8ikBCdtqLZB10Wa9McC+eb01
+LaDtZPC4u6ttoHLRlEfzqstHXSgapXg3tEBtm9piIaPB6MOM5BB6t86+qtmocTR/
qrteHZBBudhTfbCwhsxaqujHiiUvFdm3BQbKKWIW06yV9b+4f38oD2lVIan
```



```

+vfUs3gBYA15VTDmXXzQPbQ0HPjtwmFI+IWX15nSUjWuN3TUy1HgPwzDaYT8qBtu0Z+3UG4V6b
+K2CC0XszXg5rBq9hYgNzy4XoFh/6s5DoSnzq49Q9xHgHdT2yBADFmvFK7myZBs j75+2vQZ0SVpWUPy3WT/32zFAc
+mFYfUjtTZ8n/jq7aQEjQ42A
+X/7K6Jg0cdVPtEg8P1Dr5kgYYG3q30mYXX37U3FZuJmnTI63VtIXsNn0U5eGoY0btpk00Nq9UkzgSJxqj84ZD5n
+S0EGy9ZUYBJRRcGeYUM3Q4DbSJfUwSAqcFdLIWZdp8qIREMQIBWY7BLwSdyqsQo2vRrd53hm5aWM7SVf6pPq6X/
IXR5+1eU00D8/coaTT4ES2DerbV6RkV4o0VT1d0SdVX/
MmtkNG8nYj8PqU07w7988quh1ZP6D80veJS1q73tUUR9MjnGernW2tAnvnLNhdefBcD
+sZVfYq3iBMFY7wTy1P1G6NqW9GrYDYox3tTPWLD7phpbVSyKrh/
PdYrps5UxnsGoA1b7L/FfAXDfUoGrGUB4N3JsPYXX9D++g+6gV1qBBs/
WfF934aKqfD6UTggm/zV3GA0WiBpfvAZRvEb924i6yGHyMC7y5401ZAwSBupmI
+FFd13CaP04kN1vJlth6aM5vUPXg4BpyUhtbRhwD/KxCvf9K0tLJGyL1A=="
  "MedicalTranscriptionJobSummaries": [
    {
      "MedicalTranscriptionJobName": "vocabulary-dictation-medical-
transcription-job",
      "CreationTime": "2020-09-21T21:17:27.016000+00:00",
      "StartTime": "2020-09-21T21:17:27.045000+00:00",
      "CompletionTime": "2020-09-21T21:17:59.561000+00:00",
      "LanguageCode": "en-US",
      "TranscriptionJobStatus": "COMPLETED",
      "OutputLocationType": "CUSTOMER_BUCKET",
      "Specialty": "PRIMARYCARE",
      "Type": "DICTATION"
    },
    {
      "MedicalTranscriptionJobName": "alternatives-dictation-medical-
transcription-job",
      "CreationTime": "2020-09-21T21:01:14.569000+00:00",
      "StartTime": "2020-09-21T21:01:14.592000+00:00",
      "CompletionTime": "2020-09-21T21:01:43.606000+00:00",
      "LanguageCode": "en-US",
      "TranscriptionJobStatus": "COMPLETED",
      "OutputLocationType": "CUSTOMER_BUCKET",
      "Specialty": "PRIMARYCARE",
      "Type": "DICTATION"
    },
    {
      "MedicalTranscriptionJobName": "alternatives-conversation-medical-
transcription-job",
      "CreationTime": "2020-09-21T19:09:18.171000+00:00",
      "StartTime": "2020-09-21T19:09:18.199000+00:00",
      "CompletionTime": "2020-09-21T19:10:22.516000+00:00",
      "LanguageCode": "en-US",
      "TranscriptionJobStatus": "COMPLETED",

```

```
        "OutputLocationType": "CUSTOMER_BUCKET",
        "Specialty": "PRIMARYCARE",
        "Type": "CONVERSATION"
    },
    {
        "MedicalTranscriptionJobName": "speaker-id-conversation-medical-
transcription-job",
        "CreationTime": "2020-09-21T18:43:37.157000+00:00",
        "StartTime": "2020-09-21T18:43:37.265000+00:00",
        "CompletionTime": "2020-09-21T18:44:21.192000+00:00",
        "LanguageCode": "en-US",
        "TranscriptionJobStatus": "COMPLETED",
        "OutputLocationType": "CUSTOMER_BUCKET",
        "Specialty": "PRIMARYCARE",
        "Type": "CONVERSATION"
    },
    {
        "MedicalTranscriptionJobName": "multichannel-conversation-medical-
transcription-job",
        "CreationTime": "2020-09-20T23:46:44.053000+00:00",
        "StartTime": "2020-09-20T23:46:44.081000+00:00",
        "CompletionTime": "2020-09-20T23:47:35.851000+00:00",
        "LanguageCode": "en-US",
        "TranscriptionJobStatus": "COMPLETED",
        "OutputLocationType": "CUSTOMER_BUCKET",
        "Specialty": "PRIMARYCARE",
        "Type": "CONVERSATION"
    }
}
]
```

Pour plus d'informations, consultez <https://docs.aws.amazon.com/transcribe/latest/dg/batch-med-transcription.html> dans le manuel Amazon Transcribe Developer Guide.

- Pour plus de détails sur l'API, reportez-vous [ListMedicalTranscriptionJobs](#) à la section Référence des AWS CLI commandes.

JavaScript

SDK pour JavaScript (v3)

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

Créez le client.

```
import { TranscribeClient } from "@aws-sdk/client-transcribe";
// Set the AWS Region.
const REGION = "REGION"; //e.g. "us-east-1"
// Create an Amazon Transcribe service client object.
const transcribeClient = new TranscribeClient({ region: REGION });
export { transcribeClient };
```

Répertoriez les tâches de transcription médicale.

```
// Import the required AWS SDK clients and commands for Node.js
import { StartMedicalTranscriptionJobCommand } from "@aws-sdk/client-transcribe";
import { transcribeClient } from "../libs/transcribeClient.js";

// Set the parameters
export const params = {
  MedicalTranscriptionJobName: "MEDICAL_JOB_NAME", // Required
  OutputBucketName: "OUTPUT_BUCKET_NAME", // Required
  Specialty: "PRIMARYCARE", // Required. Possible values are 'PRIMARYCARE'
  Type: "JOB_TYPE", // Required. Possible values are 'CONVERSATION' and
  'DICTATION'
  LanguageCode: "LANGUAGE_CODE", // For example, 'en-US'
  MediaFormat: "SOURCE_FILE_FORMAT", // For example, 'wav'
  Media: {
    MediaFileUri: "SOURCE_FILE_LOCATION",
    // The S3 object location of the input media file. The URI must be in the
    same region
    // as the API endpoint that you are calling. For example,
    // "https://transcribe-demo.s3-REGION.amazonaws.com/hello_world.wav"
  },
};
```

```
};

export const run = async () => {
  try {
    const data = await transcribeClient.send(
      new StartMedicalTranscriptionJobCommand(params)
    );
    console.log("Success - put", data);
    return data; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};

run();
```

- Pour de plus amples informations, consultez le [Guide du développeur AWS SDK for JavaScript](#).
- Pour plus de détails sur l'API, reportez-vous [ListMedicalTranscriptionJobs](#) à la section Référence des AWS SDK for JavaScript API.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultez [Utilisation de ce service avec un AWS SDK](#). Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **ListTranscriptionJobs** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser `ListTranscriptionJobs`.

.NET

AWS SDK for .NET

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

```
/// <summary>
/// List transcription jobs, optionally with a name filter.
/// </summary>
/// <param name="jobNameContains">Optional name filter for the transcription
jobs.</param>
/// <returns>A list of transcription job summaries.</returns>
public async Task<List<TranscriptionJobSummary>>
ListTranscriptionJobs(string? jobNameContains = null)
{
    var response = await _amazonTranscribeService.ListTranscriptionJobsAsync(
        new ListTranscriptionJobsRequest()
        {
            JobNameContains = jobNameContains
        });
    return response.TranscriptionJobSummaries;
}
```

- Pour plus de détails sur l'API, reportez-vous [ListTranscriptionJobs](#) à la section Référence des AWS SDK for .NET API.

CLI

AWS CLI

Pour répertorier les tâches de transcription

L'`list-transcription-job`exemple suivant répertorie les tâches de transcription associées à votre AWS compte et à votre région.

```
aws transcribe list-transcription-jobs
```

Sortie :

```
{
  "NextToken": "NextToken",
  "TranscriptionJobSummaries": [
    {
      "TranscriptionJobName": "speak-id-job-1",
      "CreationTime": "2020-08-17T21:06:15.391000+00:00",
      "StartTime": "2020-08-17T21:06:15.416000+00:00",
```

```
    "CompletionTime": "2020-08-17T21:07:05.098000+00:00",
    "LanguageCode": "language-code",
    "TranscriptionJobStatus": "COMPLETED",
    "OutputLocationType": "SERVICE_BUCKET"
  },
  {
    "TranscriptionJobName": "job-1",
    "CreationTime": "2020-08-17T20:50:24.207000+00:00",
    "StartTime": "2020-08-17T20:50:24.230000+00:00",
    "CompletionTime": "2020-08-17T20:52:18.737000+00:00",
    "LanguageCode": "language-code",
    "TranscriptionJobStatus": "COMPLETED",
    "OutputLocationType": "SERVICE_BUCKET"
  },
  {
    "TranscriptionJobName": "sdk-test-job-4",
    "CreationTime": "2020-08-17T20:32:27.917000+00:00",
    "StartTime": "2020-08-17T20:32:27.956000+00:00",
    "CompletionTime": "2020-08-17T20:33:15.126000+00:00",
    "LanguageCode": "language-code",
    "TranscriptionJobStatus": "COMPLETED",
    "OutputLocationType": "SERVICE_BUCKET"
  },
  {
    "TranscriptionJobName": "Diarization-speak-id",
    "CreationTime": "2020-08-10T22:10:09.066000+00:00",
    "StartTime": "2020-08-10T22:10:09.116000+00:00",
    "CompletionTime": "2020-08-10T22:26:48.172000+00:00",
    "LanguageCode": "language-code",
    "TranscriptionJobStatus": "COMPLETED",
    "OutputLocationType": "SERVICE_BUCKET"
  },
  {
    "TranscriptionJobName": "your-transcription-job-name",
    "CreationTime": "2020-07-29T17:45:09.791000+00:00",
    "StartTime": "2020-07-29T17:45:09.826000+00:00",
    "CompletionTime": "2020-07-29T17:46:20.831000+00:00",
    "LanguageCode": "language-code",
    "TranscriptionJobStatus": "COMPLETED",
    "OutputLocationType": "SERVICE_BUCKET"
  }
]
```

Pour plus d'informations, consultez [Getting Started \(interface de ligne de AWS commande\)](#) dans le manuel Amazon Transcribe Developer Guide.

- Pour plus de détails sur l'API, reportez-vous [ListTranscriptionJobs](#) à la section Référence des AWS CLI commandes.

Java

SDK pour Java 2.x

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

```
public class ListTranscriptionJobs {
    public static void main(String[] args) {
        TranscribeClient transcribeClient = TranscribeClient.builder()
            .region(Region.US_EAST_1)
            .build();

        listTranscriptionJobs(transcribeClient);
    }

    public static void listTranscriptionJobs(TranscribeClient
transcribeClient) {
        ListTranscriptionJobsRequest listJobsRequest =
ListTranscriptionJobsRequest.builder()
            .build();

        transcribeClient.listTranscriptionJobsPaginator(listJobsRequest).stream()
            .flatMap(response ->
response.transcriptionJobSummaries().stream())
            .forEach(jobSummary -> {
                System.out.println("Job Name: " +
jobSummary.transcriptionJobName());
                System.out.println("Job Status: " +
jobSummary.transcriptionJobStatus());
                System.out.println("Output Location: " +
jobSummary.outputLocationType());
            });
    }
}
```

```
        // Add more information as needed

        // Retrieve additional details for the job if necessary
        GetTranscriptionJobResponse jobDetails =
transcribeClient.getTranscriptionJob(
            GetTranscriptionJobRequest.builder()

                .transcriptionJobName(jobSummary.transcriptionJobName())
                .build());

        // Display additional details
        System.out.println("Language Code: " +
jobDetails.transcriptionJob().languageCode());
        System.out.println("Media Format: " +
jobDetails.transcriptionJob().mediaFormat());
        // Add more details as needed

        System.out.println("-----");
    });
}
}
```

- Pour plus de détails sur l'API, reportez-vous [ListTranscriptionJobs](#) à la section Référence des AWS SDK for Java 2.x API.

JavaScript

SDK pour JavaScript (v3)

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

Répertoriez les tâches de transcription.

```
// Import the required AWS SDK clients and commands for Node.js

import { ListTranscriptionJobsCommand } from "@aws-sdk/client-transcribe";
```



```
import { transcribeClient } from "../libs/transcribeClient.js";

// Set the parameters
export const params = {
  JobNameContains: "KEYWORD", // Not required. Returns only transcription
  // job names containing this string
};

export const run = async () => {
  try {
    const data = await transcribeClient.send(
      new ListTranscriptionJobsCommand(params)
    );
    console.log("Success", data.TranscriptionJobSummaries);
    return data; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Créez le client.

```
import { TranscribeClient } from "@aws-sdk/client-transcribe";
// Set the AWS Region.
const REGION = "REGION"; //e.g. "us-east-1"
// Create an Amazon Transcribe service client object.
const transcribeClient = new TranscribeClient({ region: REGION });
export { transcribeClient };
```

- Pour de plus amples informations, consultez le [Guide du développeur AWS SDK for JavaScript](#).
- Pour plus de détails sur l'API, reportez-vous [ListTranscriptionJobs](#) à la section Référence des AWS SDK for JavaScript API.

Python

SDK pour Python (Boto3)

Note

Il y en a plus à ce sujet [GitHub](#). Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

```
def list_jobs(job_filter, transcribe_client):
    """
    Lists summaries of the transcription jobs for the current AWS account.

    :param job_filter: The list of returned jobs must contain this string in
    their
                       names.
    :param transcribe_client: The Boto3 Transcribe client.
    :return: The list of retrieved transcription job summaries.
    """
    try:
        response =
transcribe_client.list_transcription_jobs(JobNameContains=job_filter)
        jobs = response["TranscriptionJobSummaries"]
        next_token = response.get("NextToken")
        while next_token is not None:
            response = transcribe_client.list_transcription_jobs(
                JobNameContains=job_filter, NextToken=next_token
            )
            jobs += response["TranscriptionJobSummaries"]
            next_token = response.get("NextToken")
        logger.info("Got %s jobs with filter %s.", len(jobs), job_filter)
    except ClientError:
        logger.exception("Couldn't get jobs with filter %s.", job_filter)
        raise
    else:
        return jobs
```

- Pour plus de détails sur l'API, consultez [ListTranscriptionJobs](#) le AWS manuel de référence de l'API SDK for Python (Boto3).

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultez [Utilisation de ce service avec un AWS SDK](#). Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **ListVocabularies** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser `ListVocabularies`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- [Création et affinage d'un vocabulaire personnalisé](#)

.NET

AWS SDK for .NET

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

```
/// <summary>
/// List custom vocabularies for the current account. Optionally specify a
name
/// filter and a specific state to filter the vocabularies list.
/// </summary>
/// <param name="nameContains">Optional string the vocabulary name must
contain.</param>
/// <param name="stateEquals">Optional state of the vocabulary.</param>
/// <returns>List of information about the vocabularies.</returns>
public async Task<List<VocabularyInfo>> ListCustomVocabularies(string?
nameContains = null,
```

```
VocabularyState? stateEquals = null)
{
    var response = await _amazonTranscribeService.ListVocabulariesAsync(
        new ListVocabulariesRequest()
        {
            NameContains = nameContains,
            StateEquals = stateEquals
        });
    return response.Vocabularies;
}
```

- Pour plus de détails sur l'API, reportez-vous [ListVocabularies](#) à la section Référence des AWS SDK for .NET API.

CLI

AWS CLI

Pour répertorier les vocabulaires personnalisés

L'`list-vocabularies`exemple suivant répertorie les vocabulaires personnalisés associés à votre AWS compte et à votre région.

```
aws transcribe list-vocabularies
```

Sortie :

```
{
  "NextToken": "NextToken",
  "Vocabularies": [
    {
      "VocabularyName": "ards-test-1",
      "LanguageCode": "language-code",
      "LastModifiedTime": "2020-04-27T22:00:27.330000+00:00",
      "VocabularyState": "READY"
    },
    {
      "VocabularyName": "sample-test",
      "LanguageCode": "language-code",

```

```
    "LastModifiedTime": "2020-04-24T23:04:11.044000+00:00",
    "VocabularyState": "READY"
  },
  {
    "VocabularyName": "CRLF-to-LF-test-3-1",
    "LanguageCode": "language-code",
    "LastModifiedTime": "2020-04-24T22:12:22.277000+00:00",
    "VocabularyState": "READY"
  },
  {
    "VocabularyName": "CRLF-to-LF-test-2",
    "LanguageCode": "language-code",
    "LastModifiedTime": "2020-04-24T21:53:50.455000+00:00",
    "VocabularyState": "READY"
  },
  {
    "VocabularyName": "CRLF-to-LF-1-1",
    "LanguageCode": "language-code",
    "LastModifiedTime": "2020-04-24T21:39:33.356000+00:00",
    "VocabularyState": "READY"
  }
]
}
```

Pour plus d'informations, consultez [Vocabulaires personnalisés](#) dans le Guide du développeur Amazon Transcribe.

- Pour plus de détails sur l'API, reportez-vous [ListVocabularies](#) à la section Référence des AWS CLI commandes.

Python

SDK pour Python (Boto3)

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

```
def list_vocabularies(vocabulary_filter, transcribe_client):
```

```
"""
Lists the custom vocabularies created for this AWS account.

:param vocabulary_filter: The returned vocabularies must contain this string
in
                           their names.
:param transcribe_client: The Boto3 Transcribe client.
:return: The list of retrieved vocabularies.
"""
try:
    response =
transcribe_client.list_vocabularies(NameContains=vocabulary_filter)
    vocabs = response["Vocabularies"]
    next_token = response.get("NextToken")
    while next_token is not None:
        response = transcribe_client.list_vocabularies(
            NameContains=vocabulary_filter, NextToken=next_token
        )
        vocabs += response["Vocabularies"]
        next_token = response.get("NextToken")
    logger.info(
        "Got %s vocabularies with filter %s.", len(vocabs), vocabulary_filter
    )
except ClientError:
    logger.exception(
        "Couldn't list vocabularies with filter %s.", vocabulary_filter
    )
    raise
else:
    return vocabs
```

- Pour plus de détails sur l'API, consultez [ListVocabularies](#) le AWS manuel de référence de l'API SDK for Python (Boto3).

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultez [Utilisation de ce service avec un AWS SDK](#). Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **StartMedicalTranscriptionJob** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser `StartMedicalTranscriptionJob`.

.NET

AWS SDK for .NET

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

```
/// <summary>
/// Start a medical transcription job for a media file. This method returns
/// as soon as the job is started.
/// </summary>
/// <param name="jobName">A unique name for the medical transcription job.</
param>
/// <param name="mediaFileUri">The URI of the media file, typically an Amazon
S3 location.</param>
/// <param name="mediaFormat">The format of the media file.</param>
/// <param name="outputBucketName">Location for the output, typically an
Amazon S3 location.</param>
/// <param name="transcriptionType">Conversation or dictation transcription
type.</param>
/// <returns>A MedicalTransactionJob instance with information on the new
job.</returns>
public async Task<MedicalTranscriptionJob> StartMedicalTranscriptionJob(
    string jobName, string mediaFileUri,
    MediaFormat mediaFormat, string outputBucketName,
    Amazon.TranscribeService.Type transcriptionType)
{
    var response = await
    _amazonTranscribeService.StartMedicalTranscriptionJobAsync(
        new StartMedicalTranscriptionJobRequest()
        {
            MedicalTranscriptionJobName = jobName,
            Media = new Media()
```

```
        {
            MediaFileUri = mediaFileUri
        },
        MediaFormat = mediaFormat,
        LanguageCode =
            LanguageCode
                .EnUS, // The value must be en-US for medical
transcriptions.
        OutputBucketName = outputBucketName,
        OutputKey =
            jobName, // The value is a key used to fetch the output of
the transcription.
        Specialty = Specialty.PRIMARYCARE, // The value PRIMARYCARE must
be set.
        Type = transcriptionType
    });
    return response.MedicalTranscriptionJob;
}
```

- Pour plus de détails sur l'API, reportez-vous [StartMedicalTranscriptionJob](#) à la section Référence des AWS SDK for .NET API.

CLI

AWS CLI

Exemple 1 : Transcrire une dictée médicale enregistrée sous forme de fichier audio

L'exemple `start-medical-transcription-job` suivant transcrit un fichier audio. Vous spécifiez l'emplacement de la sortie de transcription dans le paramètre `OutputBucketName`.

```
aws transcribe start-medical-transcription-job \  
  --cli-input-json file://myfile.json
```

Contenu de `myfile.json` :

```
{  
  "MedicalTranscriptionJobName": "simple-dictation-medical-transcription-job",  
  "LanguageCode": "language-code",  
  "Specialty": "PRIMARYCARE",
```



```

    "Type": "DICTATION",
    "OutputBucketName": "DOC-EXAMPLE-BUCKET",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
    }
  }
}

```

Sortie :

```

{
  "MedicalTranscriptionJob": {
    "MedicalTranscriptionJobName": "simple-dictation-medical-transcription-
job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
    },
    "StartTime": "2020-09-20T00:35:22.256000+00:00",
    "CreationTime": "2020-09-20T00:35:22.218000+00:00",
    "Specialty": "PRIMARYCARE",
    "Type": "DICTATION"
  }
}

```

Pour plus d'informations, consultez [Présentation de la transcription par lots](#) dans le Guide du développeur Amazon Transcribe.

Exemple 2 : Transcrire un dialogue entre un médecin et son patient enregistré sous forme de fichier audio

L'exemple `start-medical-transcription-job` suivant transcrit un fichier audio contenant un dialogue entre un médecin et son patient. Vous spécifiez l'emplacement de la sortie de transcription dans le `OutputBucketName` paramètre.

```

aws transcribe start-medical-transcription-job \
  --cli-input-json file://mysecondfile.json

```

Contenu de `mysecondfile.json` :

```

{
  "MedicalTranscriptionJobName": "simple-dictation-medical-transcription-job",

```

```
"LanguageCode": "language-code",
"Specialty": "PRIMARYCARE",
>Type": "CONVERSATION",
"OutputBucketName": "DOC-EXAMPLE-BUCKET",
"Media": {
  "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
}
}
```

Sortie :

```
{
  "MedicalTranscriptionJob": {
    "MedicalTranscriptionJobName": "simple-conversation-medical-
transcription-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
    },
    "StartTime": "2020-09-20T23:19:49.965000+00:00",
    "CreationTime": "2020-09-20T23:19:49.941000+00:00",
    "Specialty": "PRIMARYCARE",
    "Type": "CONVERSATION"
  }
}
```

Pour plus d'informations, consultez [Présentation de la transcription par lots](#) dans le Guide du développeur Amazon Transcribe.

Exemple 3 : Transcrire un fichier audio multicanal d'un dialogue entre un médecin et son patient

L'exemple `start-medical-transcription-job` suivant transcrit le son de chaque canal du fichier audio et fusionne les transcriptions distinctes de chaque canal en une seule sortie de transcription. Vous spécifiez l'emplacement de la sortie de transcription dans le paramètre `OutputBucketName`.

```
aws transcribe start-medical-transcription-job \
  --cli-input-json file://mythirdfile.json
```

Contenu de `mythirdfile.json` :

```
{
  "MedicalTranscriptionJobName": "multichannel-conversation-medical-
transcription-job",
  "LanguageCode": "language-code",
  "Specialty": "PRIMARYCARE",
  "Type": "CONVERSATION",
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
  },
  "Settings": {
    "ChannelIdentification": true
  }
}
```

Sortie :

```
{
  "MedicalTranscriptionJob": {
    "MedicalTranscriptionJobName": "multichannel-conversation-medical-
transcription-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
    },
    "StartTime": "2020-09-20T23:46:44.081000+00:00",
    "CreationTime": "2020-09-20T23:46:44.053000+00:00",
    "Settings": {
      "ChannelIdentification": true
    },
    "Specialty": "PRIMARYCARE",
    "Type": "CONVERSATION"
  }
}
```

Pour plus d'informations, consultez [Identification des canaux](#) dans le Guide du développeur Amazon Transcribe.

Exemple 4 : Transcrire un fichier audio d'un dialogue entre un médecin et son patient et identifier les locuteurs dans la sortie de transcription

L'exemple `start-medical-transcription-job` suivant transcrit le fichier audio et étiquette le discours de chaque locuteur dans la sortie de transcription. Vous spécifiez l'emplacement de la sortie de transcription dans le paramètre `OutputBucketName`.

```
aws transcribe start-medical-transcription-job \  
  --cli-input-json file://myfourthfile.json
```

Contenu de `myfourthfile.json` :

```
{  
  "MedicalTranscriptionJobName": "speaker-id-conversation-medical-  
transcription-job",  
  "LanguageCode": "language-code",  
  "Specialty": "PRIMARYCARE",  
  "Type": "CONVERSATION",  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"  
  },  
  "Settings": {  
    "ShowSpeakerLabels": true,  
    "MaxSpeakerLabels": 2  
  }  
}
```

Sortie :

```
{  
  "MedicalTranscriptionJob": {  
    "MedicalTranscriptionJobName": "speaker-id-conversation-medical-  
transcription-job",  
    "TranscriptionJobStatus": "IN_PROGRESS",  
    "LanguageCode": "language-code",  
    "Media": {  
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"  
    },  
    "StartTime": "2020-09-21T18:43:37.265000+00:00",  
    "CreationTime": "2020-09-21T18:43:37.157000+00:00",  
    "Settings": {  
      "ShowSpeakerLabels": true,  
      "MaxSpeakerLabels": 2  
    },  
  },  
}
```

```
    "Specialty": "PRIMARYCARE",  
    "Type": "CONVERSATION"  
  }  
}
```

Pour plus d'informations, consultez [Identification des locuteurs](#) dans le Guide du développeur Amazon Transcribe.

Exemple 5 : Transcrire une conversation médicale enregistrée sous forme de fichier audio avec jusqu'à deux alternatives de transcription

L'exemple `start-medical-transcription-job` suivant crée jusqu'à deux transcriptions alternatives à partir d'un seul fichier audio. Un niveau de confiance est associé à chaque transcription. Par défaut, Amazon Transcribe renvoie la transcription avec le niveau de confiance le plus élevé. Vous pouvez spécifier qu'Amazon Transcribe renvoie des transcriptions supplémentaires avec des niveaux de confiance inférieurs. Vous spécifiez l'emplacement de la sortie de transcription dans le paramètre `OutputBucketName`.

```
aws transcribe start-medical-transcription-job \  
  --cli-input-json file://myfifthfile.json
```

Contenu de `myfifthfile.json` :

```
{  
  "MedicalTranscriptionJobName": "alternatives-conversation-medical-  
transcription-job",  
  "LanguageCode": "language-code",  
  "Specialty": "PRIMARYCARE",  
  "Type": "CONVERSATION",  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"  
  },  
  "Settings": {  
    "ShowAlternatives": true,  
    "MaxAlternatives": 2  
  }  
}
```

Sortie :

```
{
  "MedicalTranscriptionJob": {
    "MedicalTranscriptionJobName": "alternatives-conversation-medical-
transcription-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
    },
    "StartTime": "2020-09-21T19:09:18.199000+00:00",
    "CreationTime": "2020-09-21T19:09:18.171000+00:00",
    "Settings": {
      "ShowAlternatives": true,
      "MaxAlternatives": 2
    },
    "Specialty": "PRIMARYCARE",
    "Type": "CONVERSATION"
  }
}
```

Pour plus d'informations, consultez [Transcriptions alternatives](#) dans le Guide du développeur Amazon Transcribe.

Exemple 6 : Transcrire le fichier audio d'une dictée médicale avec jusqu'à deux transcriptions alternatives

L'exemple `start-medical-transcription-job` suivant transcrit un fichier audio et utilise un filtre de vocabulaire pour masquer les mots indésirables. Vous spécifiez l'emplacement de la sortie de transcription dans le `OutputBucketName` paramètre.

```
aws transcribe start-medical-transcription-job \
  --cli-input-json file://mysixthfile.json
```

Contenu de `mysixthfile.json` :

```
{
  "MedicalTranscriptionJobName": "alternatives-conversation-medical-
transcription-job",
  "LanguageCode": "language-code",
  "Specialty": "PRIMARYCARE",
  "Type": "DICTATION",
```

```

"OutputBucketName": "DOC-EXAMPLE-BUCKET",
"Media": {
  "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
},
"Settings": {
  "ShowAlternatives": true,
  "MaxAlternatives": 2
}
}

```

Sortie :

```

{
  "MedicalTranscriptionJob": {
    "MedicalTranscriptionJobName": "alternatives-dictation-medical-
transcription-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
    },
    "StartTime": "2020-09-21T21:01:14.592000+00:00",
    "CreationTime": "2020-09-21T21:01:14.569000+00:00",
    "Settings": {
      "ShowAlternatives": true,
      "MaxAlternatives": 2
    },
    "Specialty": "PRIMARYCARE",
    "Type": "DICTATION"
  }
}

```

Pour plus d'informations, consultez [Transcriptions alternatives](#) dans le Guide du développeur Amazon Transcribe.

Exemple 7 : Transcrire un fichier audio d'une dictée médicale avec une précision accrue en utilisant un vocabulaire personnalisé

L'exemple `start-medical-transcription-job` suivant transcrit un fichier audio et utilise un vocabulaire médical personnalisé que vous avez créé précédemment pour augmenter la précision de la transcription. Vous spécifiez l'emplacement de la sortie de transcription dans le paramètre `OutputBucketName`.

```
aws transcribe start-transcription-job \  
  --cli-input-json file://myseventhfile.json
```

Contenu de mysixthfile.json :

```
{  
  "MedicalTranscriptionJobName": "vocabulary-dictation-medical-transcription-  
job",  
  "LanguageCode": "language-code",  
  "Specialty": "PRIMARYCARE",  
  "Type": "DICTATION",  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"  
  },  
  "Settings": {  
    "VocabularyName": "cli-medical-vocab-1"  
  }  
}
```

Sortie :

```
{  
  "MedicalTranscriptionJob": {  
    "MedicalTranscriptionJobName": "vocabulary-dictation-medical-  
transcription-job",  
    "TranscriptionJobStatus": "IN_PROGRESS",  
    "LanguageCode": "language-code",  
    "Media": {  
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"  
    },  
    "StartTime": "2020-09-21T21:17:27.045000+00:00",  
    "CreationTime": "2020-09-21T21:17:27.016000+00:00",  
    "Settings": {  
      "VocabularyName": "cli-medical-vocab-1"  
    },  
    "Specialty": "PRIMARYCARE",  
    "Type": "DICTATION"  
  }  
}
```


Pour plus d'informations, consultez [Vocabulaires personnalisés médicaux](#) dans le Guide du développeur Amazon Transcribe.

- Pour plus de détails sur l'API, reportez-vous [StartMedicalTranscriptionJob](#) à la section Référence des AWS CLI commandes.

JavaScript

SDK pour JavaScript (v3)

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

Créez le client.

```
import { TranscribeClient } from "@aws-sdk/client-transcribe";
// Set the AWS Region.
const REGION = "REGION"; //e.g. "us-east-1"
// Create an Amazon Transcribe service client object.
const transcribeClient = new TranscribeClient({ region: REGION });
export { transcribeClient };
```

Démarrez une tâche de transcription médicale.

```
// Import the required AWS SDK clients and commands for Node.js
import { StartMedicalTranscriptionJobCommand } from "@aws-sdk/client-transcribe";
import { transcribeClient } from "./libs/transcribeClient.js";

// Set the parameters
export const params = {
  MedicalTranscriptionJobName: "MEDICAL_JOB_NAME", // Required
  OutputBucketName: "OUTPUT_BUCKET_NAME", // Required
  Specialty: "PRIMARYCARE", // Required. Possible values are 'PRIMARYCARE'
  Type: "JOB_TYPE", // Required. Possible values are 'CONVERSATION' and
  'DICTATION'
  LanguageCode: "LANGUAGE_CODE", // For example, 'en-US'
  MediaFormat: "SOURCE_FILE_FORMAT", // For example, 'wav'
```

```
Media: {
  MediaFileUri: "SOURCE_FILE_LOCATION",
  // The S3 object location of the input media file. The URI must be in the
  // same region
  // as the API endpoint that you are calling. For example,
  // "https://transcribe-demo.s3-REGION.amazonaws.com/hello_world.wav"
},
};

export const run = async () => {
  try {
    const data = await transcribeClient.send(
      new StartMedicalTranscriptionJobCommand(params)
    );
    console.log("Success - put", data);
    return data; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

- Pour de plus amples informations, consultez le [Guide du développeur AWS SDK for JavaScript](#).
- Pour plus de détails sur l'API, reportez-vous [StartMedicalTranscriptionJob](#) à la section Référence des AWS SDK for JavaScript API.


Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultez [Utilisation de ce service avec un AWS SDK](#). Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **StartStreamTranscriptionAsync** avec un AWS SDK ou une CLI

L'exemple de code suivant montre comment utiliser `StartStreamTranscriptionAsync`.

C++

Kit de développement logiciel (SDK) for C++

 Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

```
int main() {
    Aws::SDKOptions options;

    Aws::InitAPI(options);
    {
        //TODO(User): Set to the region of your AWS account.
        const Aws::String region = Aws::Region::US_WEST_2;

        //Load a profile that has been granted AmazonTranscribeFullAccess AWS
        managed permission policy.
        Aws::Client::ClientConfiguration config;
#ifdef _WIN32
        // ATTENTION: On Windows with the AWS C++ SDK, this example only runs if
        the SDK is built
        // with the curl library.
        // For more information, see the accompanying ReadMe.
        // For more information, see "Building the SDK for Windows with curl".
        // https://docs.aws.amazon.com/sdk-for-cpp/v1/developer-guide/setup-
        windows.html
        //TODO(User): Update to the location of your .crt file.
        config.caFile = "C:/curl/bin/curl-ca-bundle.crt";
#endif
        config.region = region;

        TranscribeStreamingServiceClient client(config);
        StartStreamTranscriptionHandler handler;
        handler.SetOnErrorCallback(
            [](const Aws::Client::AWSError<TranscribeStreamingServiceErrors>
            &error) {
                std::cerr << "ERROR: " + error.GetMessage() << std::endl;
            });
    }
}
```

```

    //SetTranscriptEventCallback called for every 'chunk' of file
    transcribed.
    // Partial results are returned in real time.
    handler.SetTranscriptEventCallback([](const TranscriptEvent &ev) {
        for (auto &&r: ev.GetTranscript().GetResults()) {
            if (r.GetIsPartial()) {
                std::cout << "[partial] ";
            }
            else {
                std::cout << "[Final] ";
            }
            for (auto &&alt: r.GetAlternatives()) {
                std::cout << alt.GetTranscript() << std::endl;
            }
        }
    });

    StartStreamTranscriptionRequest request;
    request.SetMediaSampleRateHertz(SAMPLE_RATE);
    request.SetLanguageCode(LanguageCode::en_US);
    request.SetMediaEncoding(
        MediaEncoding::pcm); // wav and aiff files are PCM formats.
    request.SetEventStreamHandler(handler);

    auto OnStreamReady = [](AudioStream &stream) {
        Aws::FStream file(FILE_NAME, std::ios_base::in |
std::ios_base::binary);
        if (!file.is_open()) {
            std::cerr << "Failed to open " << FILE_NAME << '\n';
        }
        std::array<char, BUFFER_SIZE> buf;
        int i = 0;
        while (file) {
            file.read(&buf[0], buf.size());

            if (!file)
                std::cout << "File: only " << file.gcount() << " could be
read"
                    << std::endl;

            Aws::Vector<unsigned char> bits{buf.begin(), buf.end()};
            AudioEvent event(std::move(bits));
            if (!stream) {
                std::cerr << "Failed to create a stream" << std::endl;
            }
        }
    };

```

```

        break;
    }
    //The std::basic_istream::gcount() is used to count the
characters in the given string. It returns
    //the number of characters extracted by the last read()
operation.
    if (file.gcount() > 0) {
        if (!stream.WriteAudioEvent(event)) {
            std::cerr << "Failed to write an audio event" <<
std::endl;
                break;
            }
        }
    else {
        break;
    }
    std::this_thread::sleep_for(std::chrono::milliseconds(
        25)); // Slow down because we are streaming from a
file.
    }
    if (!stream.WriteAudioEvent(
        AudioEvent())) {
        // Per the spec, we have to send an empty event (an event
without a payload) at the end.
        std::cerr << "Failed to send an empty frame" << std::endl;
    }
    else {
        std::cout << "Successfully sent the empty frame" <<
std::endl;
    }
    stream.flush();
    stream.Close();
};

    Aws::Utils::Threading::Semaphore signaling(0 /*initialCount*/, 1 /
*maxCount*/);
    auto OnResponseCallback = [&signaling](
        const TranscribeStreamingServiceClient * /*unused*/,
        const Model::StartStreamTranscriptionRequest & /*unused*/,
        const Model::StartStreamTranscriptionOutcome &outcome,
        const std::shared_ptr<const Aws::Client::AsyncCallerContext> & /
*unused*/) {

        if (!outcome.IsSuccess()) {

```

```
        std::cerr << "Transcribe streaming error "
                << outcome.GetError().GetMessage() << std::endl;
    }

    signaling.Release();
};

std::cout << "Starting..." << std::endl;
client.StartStreamTranscriptionAsync(request, OnStreamReady,
OnResponseCallback,
                                   nullptr /*context*/);
signaling.WaitOne(); // Prevent the application from exiting until we're
done.
std::cout << "Done" << std::endl;
}

Aws::ShutdownAPI(options);

return 0;
}
```

- Pour plus de détails sur l'API, reportez-vous [StartStreamTranscriptionAsync](#) à la section Référence des AWS SDK for C++ API.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultez [Utilisation de ce service avec un AWS SDK](#). Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **StartTranscriptionJob** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser `StartTranscriptionJob`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans les exemples de code suivants :

- [Création et affinage d'un vocabulaire personnalisé](#)
- [Transcription d'un fichier audio et obtention de données sur la tâche](#)

.NET

AWS SDK for .NET

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

```
/// <summary>
/// Start a transcription job for a media file. This method returns
/// as soon as the job is started.
/// </summary>
/// <param name="jobName">A unique name for the transcription job.</param>
/// <param name="mediaFileUri">The URI of the media file, typically an Amazon
S3 location.</param>
/// <param name="mediaFormat">The format of the media file.</param>
/// <param name="languageCode">The language code of the media file, such as
en-US.</param>
/// <param name="vocabularyName">Optional name of a custom vocabulary.</
param>
/// <returns>A TranscriptionJob instance with information on the new job.</
returns>
public async Task<TranscriptionJob> StartTranscriptionJob(string jobName,
string mediaFileUri,
MediaFormat mediaFormat, LanguageCode languageCode, string?
vocabularyName)
{
    var response = await _amazonTranscribeService.StartTranscriptionJobAsync(
        new StartTranscriptionJobRequest()
        {
            TranscriptionJobName = jobName,
            Media = new Media()
            {
                MediaFileUri = mediaFileUri
            },
            MediaFormat = mediaFormat,
            LanguageCode = languageCode,
            Settings = vocabularyName != null ? new Settings()
            {
```

```
        VocabularyName = vocabularyName
    } : null
    });
    return response.TranscriptionJob;
}
```

- Pour plus de détails sur l'API, reportez-vous [StartTranscriptionJob](#) à la section Référence des AWS SDK for .NET API.

CLI

AWS CLI

Exemple 1 : Transcrire un fichier audio

L'exemple `start-transcription-job` suivant transcrit un fichier audio.

```
aws transcribe start-transcription-job \  
  --cli-input-json file://myfile.json
```

Contenu de `myfile.json` :

```
{  
  "TranscriptionJobName": "cli-simple-transcription-job",  
  "LanguageCode": "the-language-of-your-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-  
file-name.file-extension"  
  }  
}
```

Pour plus d'informations, consultez [Getting Started \(interface de ligne de AWS commande\)](#) dans le manuel Amazon Transcribe Developer Guide.

Exemple 2 : Transcrire un fichier audio multicanal

L'exemple `start-transcription-job` suivant transcrit le fichier audio multicanal.

```
aws transcribe start-transcription-job \  
  --cli-input-json file://myfile.json
```



```
--cli-input-json file://mysecondfile.json
```

Contenu de `mysecondfile.json` :

```
{
  "TranscriptionJobName": "cli-channelid-job",
  "LanguageCode": "the-language-of-your-transcription-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-
file-name.file-extension"
  },
  "Settings":{
    "ChannelIdentification":true
  }
}
```

Sortie :

```
{
  "TranscriptionJob": {
    "TranscriptionJobName": "cli-channelid-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "the-language-of-your-transcription-job",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-
file-name.file-extension"
    },
    "StartTime": "2020-09-17T16:07:56.817000+00:00",
    "CreationTime": "2020-09-17T16:07:56.784000+00:00",
    "Settings": {
      "ChannelIdentification": true
    }
  }
}
```

Pour plus d'informations, consultez [Transcription de l'audio multicanal](#) dans le Guide du développeur Amazon Transcribe.

Exemple 3 : Transcrire un fichier audio et identifier les différents locuteurs

L'exemple `start-transcription-job` suivant transcrit le fichier audio et identifie les locuteurs dans la sortie de transcription.

```
aws transcribe start-transcription-job \  
  --cli-input-json file://mythirdfile.json
```

Contenu de mythirdfile.json :

```
{  
  "TranscriptionJobName": "cli-speakerid-job",  
  "LanguageCode": "the-language-of-your-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-  
file-name.file-extension"  
  },  
  "Settings":{  
    "ShowSpeakerLabels": true,  
    "MaxSpeakerLabels": 2  
  }  
}
```

Sortie :

```
{  
  "TranscriptionJob": {  
    "TranscriptionJobName": "cli-speakerid-job",  
    "TranscriptionJobStatus": "IN_PROGRESS",  
    "LanguageCode": "the-language-of-your-transcription-job",  
    "Media": {  
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-  
file-name.file-extension"  
    },  
    "StartTime": "2020-09-17T16:22:59.696000+00:00",  
    "CreationTime": "2020-09-17T16:22:59.676000+00:00",  
    "Settings": {  
      "ShowSpeakerLabels": true,  
      "MaxSpeakerLabels": 2  
    }  
  }  
}
```

Pour plus d'informations, consultez [Identification des locuteurs](#) dans le Guide du développeur Amazon Transcribe.

Exemple 4 : Transcrire un fichier audio et masquer les mots indésirables dans la sortie de transcription

L'exemple `start-transcription-job` suivant transcrit un fichier audio et utilise un filtre de vocabulaire que vous avez créé précédemment pour masquer les mots indésirables.

```
aws transcribe start-transcription-job \  
  --cli-input-json file://myfourthfile.json
```

Contenu de `myfourthfile.json` :

```
{  
  "TranscriptionJobName": "cli-filter-mask-job",  
  "LanguageCode": "the-language-of-your-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-file-name.file-extension"  
  },  
  "Settings": {  
    "VocabularyFilterName": "your-vocabulary-filter",  
    "VocabularyFilterMethod": "mask"  
  }  
}
```

Sortie :

```
{  
  "TranscriptionJob": {  
    "TranscriptionJobName": "cli-filter-mask-job",  
    "TranscriptionJobStatus": "IN_PROGRESS",  
    "LanguageCode": "the-language-of-your-transcription-job",  
    "Media": {  
      "MediaFileUri": "s3://Amazon-S3-Prefix/your-media-file.file-extension"  
    },  
    "StartTime": "2020-09-18T16:36:18.568000+00:00",  
    "CreationTime": "2020-09-18T16:36:18.547000+00:00",  
    "Settings": {  
      "VocabularyFilterName": "your-vocabulary-filter",  
      "VocabularyFilterMethod": "mask"  
    }  
  }  
}
```

```
}
```

Pour plus d'informations, consultez [Filtrage des transcriptions](#) dans le Guide du développeur Amazon Transcribe.

Exemple 5 : Transcrire un fichier audio et supprimer les mots indésirables dans la sortie de transcription

L'exemple `start-transcription-job` suivant transcrit un fichier audio et utilise un filtre de vocabulaire que vous avez créé précédemment pour masquer les mots indésirables.

```
aws transcribe start-transcription-job \  
  --cli-input-json file://myfifthfile.json
```

Contenu de `myfifthfile.json` :

```
{  
  "TranscriptionJobName": "cli-filter-remove-job",  
  "LanguageCode": "the-language-of-your-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-  
file-name.file-extension"  
  },  
  "Settings": {  
    "VocabularyFilterName": "your-vocabulary-filter",  
    "VocabularyFilterMethod": "remove"  
  }  
}
```

Sortie :

```
{  
  "TranscriptionJob": {  
    "TranscriptionJobName": "cli-filter-remove-job",  
    "TranscriptionJobStatus": "IN_PROGRESS",  
    "LanguageCode": "the-language-of-your-transcription-job",  
    "Media": {  
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-  
file-name.file-extension"  
    },  
    "StartTime": "2020-09-18T16:36:18.568000+00:00",  
  }  
}
```

```

    "CreationTime": "2020-09-18T16:36:18.547000+00:00",
    "Settings": {
      "VocabularyFilterName": "your-vocabulary-filter",
      "VocabularyFilterMethod": "remove"
    }
  }
}

```

Pour plus d'informations, consultez [Filtrage des transcriptions](#) dans le Guide du développeur Amazon Transcribe.

Exemple 6 : Transcrire un fichier audio avec une précision accrue en utilisant un vocabulaire personnalisé

L'exemple `start-transcription-job` suivant transcrit un fichier audio et utilise un filtre de vocabulaire que vous avez créé précédemment pour masquer les mots indésirables.

```

aws transcribe start-transcription-job \
  --cli-input-json file://mysixthfile.json

```

Contenu de `mysixthfile.json` :

```

{
  "TranscriptionJobName": "cli-vocab-job",
  "LanguageCode": "the-language-of-your-transcription-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-file-name.file-extension"
  },
  "Settings": {
    "VocabularyName": "your-vocabulary"
  }
}

```

Sortie :

```

{
  "TranscriptionJob": {
    "TranscriptionJobName": "cli-vocab-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "the-language-of-your-transcription-job",
    "Media": {

```

```

        "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-
file-name.file-extension"
    },
    "StartTime": "2020-09-18T16:36:18.568000+00:00",
    "CreationTime": "2020-09-18T16:36:18.547000+00:00",
    "Settings": {
        "VocabularyName": "your-vocabulary"
    }
}
}

```

Pour plus d'informations, consultez [Filtrage des transcriptions](#) dans le Guide du développeur Amazon Transcribe.

Exemple 7 : Identifier la langue d'un fichier audio et le transcrire

L'exemple `start-transcription-job` suivant transcrit un fichier audio et utilise un filtre de vocabulaire que vous avez créé précédemment pour masquer les mots indésirables.

```

aws transcribe start-transcription-job \
  --cli-input-json file://myseventhfile.json

```

Contenu de `myseventhfile.json` :

```

{
  "TranscriptionJobName": "cli-identify-language-transcription-job",
  "IdentifyLanguage": true,
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-
file-name.file-extension"
  }
}

```

Sortie :

```

{
  "TranscriptionJob": {
    "TranscriptionJobName": "cli-identify-language-transcription-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-
file-name.file-extension"
    }
  }
}

```

```

    },
    "StartTime": "2020-09-18T22:27:23.970000+00:00",
    "CreationTime": "2020-09-18T22:27:23.948000+00:00",
    "IdentifyLanguage": true
  }
}

```

Pour plus d'informations, consultez [Identification de la langue](#) dans le Guide du développeur Amazon Transcribe.

Exemple 8 : Transcrire un fichier audio contenant des données d'identification personnelle expurgées

L'exemple `start-transcription-job` suivant transcrit le fichier audio et expurge les données d'identification personnelle dans la sortie de transcription.

```

aws transcribe start-transcription-job \
  --cli-input-json file://myeighthfile.json

```

Contenu de `myeighthfile.json` :

```

{
  "TranscriptionJobName": "cli-redaction-job",
  "LanguageCode": "language-code",
  "Media": {
    "MediaFileUri": "s3://Amazon-S3-Prefix/your-media-file.file-extension"
  },
  "ContentRedaction": {
    "RedactionOutput": "redacted",
    "RedactionType": "PII"
  }
}

```

Sortie :

```

{
  "TranscriptionJob": {
    "TranscriptionJobName": "cli-redaction-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {

```

```

        "MediaFileUri": "s3://Amazon-S3-Prefix/your-media-file.file-
extension"
      },
      "StartTime": "2020-09-25T23:49:13.195000+00:00",
      "CreationTime": "2020-09-25T23:49:13.176000+00:00",
      "ContentRedaction": {
        "RedactionType": "PII",
        "RedactionOutput": "redacted"
      }
    }
  }
}

```

Pour plus d'informations, consultez [Expurgation automatique du contenu](#) dans le Guide du développeur Amazon Transcribe.

Exemple 9 : Générer une transcription contenant des données d'identification personnelle (PII) expurgées et une transcription non expurgée

L'exemple `start-transcription-job` suivant génère deux transcriptions du fichier audio, l'une avec les données d'identification personnelle expurgées, et l'autre sans aucune expurgation.

```

aws transcribe start-transcription-job \
  --cli-input-json file://myninthfile.json

```

Contenu de `myninthfile.json` :

```

{
  "TranscriptionJobName": "cli-redaction-job-with-unredacted-transcript",
  "LanguageCode": "language-code",
  "Media": {
    "MediaFileUri": "s3://Amazon-S3-Prefix/your-media-file.file-extension"
  },
  "ContentRedaction": {
    "RedactionOutput": "redacted_and_unredacted",
    "RedactionType": "PII"
  }
}

```

Sortie :

```

{

```



```
"TranscriptionJob": {
  "TranscriptionJobName": "cli-redaction-job-with-unredacted-transcript",
  "TranscriptionJobStatus": "IN_PROGRESS",
  "LanguageCode": "language-code",
  "Media": {
    "MediaFileUri": "s3://Amazon-S3-Prefix/your-media-file.file-
extension"
  },
  "StartTime": "2020-09-25T23:59:47.677000+00:00",
  "CreationTime": "2020-09-25T23:59:47.653000+00:00",
  "ContentRedaction": {
    "RedactionType": "PII",
    "RedactionOutput": "redacted_and_unredacted"
  }
}
}
```

Pour plus d'informations, consultez [Expurgation automatique du contenu](#) dans le Guide du développeur Amazon Transcribe.

Exemple 10 : Utiliser un modèle de langage personnalisé que vous avez créé précédemment pour transcrire un fichier audio.

L'exemple `start-transcription-job` suivant transcrit le fichier audio avec un modèle de langage personnalisé que vous avez créé précédemment.

```
aws transcribe start-transcription-job \
  --cli-input-json file://mytenthfile.json
```

Contenu de `mytenthfile.json` :

```
{
  "TranscriptionJobName": "cli-clm-2-job-1",
  "LanguageCode": "language-code",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.file-extension"
  },
  "ModelSettings": {
    "LanguageModelName": "cli-clm-2"
  }
}
```

Sortie :

```
{
  "TranscriptionJob": {
    "TranscriptionJobName": "cli-clm-2-job-1",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.file-
extension"
    },
    "StartTime": "2020-09-28T17:56:01.835000+00:00",
    "CreationTime": "2020-09-28T17:56:01.801000+00:00",
    "ModelSettings": {
      "LanguageModelName": "cli-clm-2"
    }
  }
}
```

Pour plus d'informations, consultez [Amélioration de la précision de la transcription spécifique à un domaine grâce à des modèles de langue personnalisés](#) dans le Guide du développeur Amazon Transcribe.

- Pour plus de détails sur l'API, reportez-vous [StartTranscriptionJob](#) à la section Référence des AWS CLI commandes.

Java

SDK pour Java 2.x

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

```
public class TranscribeStreamingDemoApp {
    private static final Region REGION = Region.US_EAST_1;
    private static TranscribeStreamingAsyncClient client;

    public static void main(String args[])
```

```
throws URISyntaxException, ExecutionException, InterruptedException,
LineUnavailableException {

    client = TranscribeStreamingAsyncClient.builder()
        .credentialsProvider(getCredentials())
        .region(REGION)
        .build();

    CompletableFuture<Void> result =
client.startStreamTranscription(getRequest(16_000),
    new AudioStreamPublisher(getStreamFromMic()),
    getResponseHandler());

    result.get();
    client.close();
}

private static InputStream getStreamFromMic() throws LineUnavailableException
{

    // Signed PCM AudioFormat with 16kHz, 16 bit sample size, mono
    int sampleRate = 16000;
    AudioFormat format = new AudioFormat(sampleRate, 16, 1, true, false);
    DataLine.Info info = new DataLine.Info(TargetDataLine.class, format);

    if (!AudioSystem.isLineSupported(info)) {
        System.out.println("Line not supported");
        System.exit(0);
    }

    TargetDataLine line = (TargetDataLine) AudioSystem.getLine(info);
    line.open(format);
    line.start();

    InputStream audioStream = new AudioInputStream(line);
    return audioStream;
}

private static AwsCredentialsProvider getCredentials() {
    return DefaultCredentialsProvider.create();
}

private static StartStreamTranscriptionRequest getRequest(Integer
mediaSampleRateHertz) {
```

```
        return StartStreamTranscriptionRequest.builder()
            .languageCode(LanguageCode.EN_US.toString())
            .mediaEncoding(MediaEncoding.PCM)
            .mediaSampleRateHertz(mediaSampleRateHertz)
            .build();
    }

    private static StartStreamTranscriptionResponseHandler getResponseHandler() {
        return StartStreamTranscriptionResponseHandler.builder()
            .onResponse(r -> {
                System.out.println("Received Initial response");
            })
            .onError(e -> {
                System.out.println(e.getMessage());
                StringWriter sw = new StringWriter();
                e.printStackTrace(new PrintWriter(sw));
                System.out.println("Error Occurred: " + sw.toString());
            })
            .onComplete(() -> {
                System.out.println("=== All records stream successfully
===");
            })
            .subscriber(event -> {
                List<Result> results = ((TranscriptEvent)
event).transcript().results();
                if (results.size() > 0) {
                    if (!
results.get(0).alternatives().get(0).transcript().isEmpty()) {

                        System.out.println(results.get(0).alternatives().get(0).transcript());
                    }
                }
            })
            .build();
    }

    private InputStream getStreamFromFile(String audioFileName) {
        try {
            File inputFile = new
File(getClass().getClassLoader().getResource(audioFileName).getFile());
            InputStream audioStream = new FileInputStream(inputFile);
            return audioStream;
        } catch (FileNotFoundException e) {
            throw new RuntimeException(e);
        }
    }
}
```

```
    }  
}  
  
private static class AudioStreamPublisher implements Publisher<AudioStream> {  
    private final InputStream inputStream;  
    private static Subscription currentSubscription;  
  
    private AudioStreamPublisher(InputStream inputStream) {  
        this.inputStream = inputStream;  
    }  
  
    @Override  
    public void subscribe(Subscriber<? super AudioStream> s) {  
  
        if (this.currentSubscription == null) {  
            this.currentSubscription = new SubscriptionImpl(s, inputStream);  
        } else {  
            this.currentSubscription.cancel();  
            this.currentSubscription = new SubscriptionImpl(s, inputStream);  
        }  
        s.onSubscribe(currentSubscription);  
    }  
}  
  
public static class SubscriptionImpl implements Subscription {  
    private static final int CHUNK_SIZE_IN_BYTES = 1024 * 1;  
    private final Subscriber<? super AudioStream> subscriber;  
    private final InputStream inputStream;  
    private ExecutorService executor = Executors.newFixedThreadPool(1);  
    private AtomicLong demand = new AtomicLong(0);  
  
    SubscriptionImpl(Subscriber<? super AudioStream> s, InputStream  
inputStream) {  
        this.subscriber = s;  
        this.inputStream = inputStream;  
    }  
  
    @Override  
    public void request(long n) {  
        if (n <= 0) {  
            subscriber.onError(new IllegalArgumentException("Demand must be  
positive"));  
        }  
    }  
}
```

```
demand.getAndAdd(n);

executor.submit(() -> {
    try {
        do {
            ByteBuffer audioBuffer = getNextEvent();
            if (audioBuffer.remaining() > 0) {
                AudioEvent audioEvent =
audioEventFromBuffer(audioBuffer);
                subscriber.onNext(audioEvent);
            } else {
                subscriber.onComplete();
                break;
            }
        } while (demand.decrementAndGet() > 0);
    } catch (Exception e) {
        subscriber.onError(e);
    }
});
}

@Override
public void cancel() {
    executor.shutdown();
}

private ByteBuffer getNextEvent() {
    ByteBuffer audioBuffer = null;
    byte[] audioBytes = new byte[CHUNK_SIZE_IN_BYTES];

    int len = 0;
    try {
        len = inputStream.read(audioBytes);

        if (len <= 0) {
            audioBuffer = ByteBuffer.allocate(0);
        } else {
            audioBuffer = ByteBuffer.wrap(audioBytes, 0, len);
        }
    } catch (IOException e) {
        throw new UncheckedIOException(e);
    }

    return audioBuffer;
}
```

```
    }

    private AudioEvent audioEventFromBuffer(ByteBuffer bb) {
        return AudioEvent.builder()
            .audioChunk(SdkBytes.fromByteBuffer(bb))
            .build();
    }
}
}
```

- Pour plus de détails sur l'API, reportez-vous [StartTranscriptionJob](#) à la section Référence des AWS SDK for Java 2.x API.

JavaScript

SDK pour JavaScript (v3)

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

Démarrez une tâche de transcription.

```
// Import the required AWS SDK clients and commands for Node.js
import { StartTranscriptionJobCommand } from "@aws-sdk/client-transcribe";
import { transcribeClient } from "../libs/transcribeClient.js";

// Set the parameters
export const params = {
    TranscriptionJobName: "JOB_NAME",
    LanguageCode: "LANGUAGE_CODE", // For example, 'en-US'
    MediaFormat: "SOURCE_FILE_FORMAT", // For example, 'wav'
    Media: {
        MediaFileUri: "SOURCE_LOCATION",
        // For example, "https://transcribe-demo.s3-REGION.amazonaws.com/
hello_world.wav"
    },
    OutputBucketName: "OUTPUT_BUCKET_NAME"
```

```
};

export const run = async () => {
  try {
    const data = await transcribeClient.send(
      new StartTranscriptionJobCommand(params)
    );
    console.log("Success - put", data);
    return data; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};

run();
```

Créez le client.

```
import { TranscribeClient } from "@aws-sdk/client-transcribe";
// Set the AWS Region.
const REGION = "REGION"; //e.g. "us-east-1"
// Create an Amazon Transcribe service client object.
const transcribeClient = new TranscribeClient({ region: REGION });
export { transcribeClient };
```

- Pour de plus amples informations, consultez le [Guide du développeur AWS SDK for JavaScript](#).
- Pour plus de détails sur l'API, reportez-vous [StartTranscriptionJob](#) à la section Référence des AWS SDK for JavaScript API.

Python

SDK pour Python (Boto3)

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).


```
def start_job(
    job_name,
    media_uri,
    media_format,
    language_code,
    transcribe_client,
    vocabulary_name=None,
):
    """
    Starts a transcription job. This function returns as soon as the job is
    started.
    To get the current status of the job, call get_transcription_job. The job is
    successfully completed when the job status is 'COMPLETED'.

    :param job_name: The name of the transcription job. This must be unique for
                     your AWS account.
    :param media_uri: The URI where the audio file is stored. This is typically
                     in an Amazon S3 bucket.
    :param media_format: The format of the audio file. For example, mp3 or wav.
    :param language_code: The language code of the audio file.
                          For example, en-US or ja-JP
    :param transcribe_client: The Boto3 Transcribe client.
    :param vocabulary_name: The name of a custom vocabulary to use when
    transcribing
                           the audio file.
    :return: Data about the job.
    """
    try:
        job_args = {
            "TranscriptionJobName": job_name,
            "Media": {"MediaFileUri": media_uri},
            "MediaFormat": media_format,
            "LanguageCode": language_code,
        }
        if vocabulary_name is not None:
            job_args["Settings"] = {"VocabularyName": vocabulary_name}
        response = transcribe_client.start_transcription_job(**job_args)
        job = response["TranscriptionJob"]
        logger.info("Started transcription job %s.", job_name)
    except ClientError:
        logger.exception("Couldn't start transcription job %s.", job_name)
        raise
    else:
```

```
return job
```

- Pour plus de détails sur l'API, consultez [StartTranscriptionJob](#) le AWS manuel de référence de l'API SDK for Python (Boto3).

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultez [Utilisation de ce service avec un AWS SDK](#). Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **UpdateVocabulary** avec un AWS SDK ou une CLI

Les exemples de code suivants montrent comment utiliser `UpdateVocabulary`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- [Création et affinage d'un vocabulaire personnalisé](#)

.NET

AWS SDK for .NET

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

```
/// <summary>  
/// Update a custom vocabulary with new values. Update overwrites all  
existing information.  
/// </summary>  
/// <param name="languageCode">The language code of the vocabulary.</param>  
/// <param name="phrases">Phrases to use in the vocabulary.</param>  
/// <param name="vocabularyName">Name for the vocabulary.</param>
```

```
/// <returns>The state of the custom vocabulary.</returns>
public async Task<VocabularyState> UpdateCustomVocabulary(LanguageCode
languageCode,
    List<string> phrases, string vocabularyName)
{
    var response = await _amazonTranscribeService.UpdateVocabularyAsync(
        new UpdateVocabularyRequest()
        {
            LanguageCode = languageCode,
            Phrases = phrases,
            VocabularyName = vocabularyName
        });
    return response.VocabularyState;
}
```

- Pour plus de détails sur l'API, reportez-vous [UpdateVocabulary](#) à la section Référence des AWS SDK for .NET API.

CLI

AWS CLI

Pour mettre à jour un vocabulaire personnalisé avec de nouveaux termes.

L'exemple `update-vocabulary` suivant remplace les termes utilisés pour créer un vocabulaire personnalisé par les nouveaux termes que vous fournissez. Prérequis : pour remplacer les termes d'un vocabulaire personnalisé, vous avez besoin d'un fichier contenant les nouveaux termes.

```
aws transcribe update-vocabulary \
    --vocabulary-file-uri s3://DOC-EXAMPLE-BUCKET/Amazon-S3-Prefix/custom-
vocabulary.txt \
    --vocabulary-name custom-vocabulary \
    --language-code language-code
```

Sortie :

```
{
  "VocabularyName": "custom-vocabulary",
```

```
"LanguageCode": "language",
"VocabularyState": "PENDING"
}
```

Pour plus d'informations, consultez [Vocabulaires personnalisés](#) dans le Guide du développeur Amazon Transcribe.

- Pour plus de détails sur l'API, reportez-vous [UpdateVocabulary](#) à la section Référence des AWS CLI commandes.

Python

SDK pour Python (Boto3)

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

```
def update_vocabulary(
    vocabulary_name, language_code, transcribe_client, phrases=None,
    table_uri=None
):
    """
    Updates an existing custom vocabulary. The entire vocabulary is replaced with
    the contents of the update.

    :param vocabulary_name: The name of the vocabulary to update.
    :param language_code: The language code of the vocabulary.
    :param transcribe_client: The Boto3 Transcribe client.
    :param phrases: A list of comma-separated phrases to include in the
    vocabulary.
    :param table_uri: A table of phrases and pronunciation hints to include in
    the
        vocabulary.
    """
    try:
        vocab_args = {"VocabularyName": vocabulary_name, "LanguageCode":
language_code}
        if phrases is not None:
            vocab_args["Phrases"] = phrases
```

```
elif table_uri is not None:
    vocab_args["VocabularyFileUri"] = table_uri
    response = transcribe_client.update_vocabulary(**vocab_args)
    logger.info("Updated custom vocabulary %s.", response["VocabularyName"])
except ClientError:
    logger.exception("Couldn't update custom vocabulary %s.",
vocabulary_name)
    raise
```

- Pour plus de détails sur l'API, consultez [UpdateVocabulary](#) le AWS manuel de référence de l'API SDK for Python (Boto3).

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultez [Utilisation de ce service avec un AWS SDK](#). Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Scénarios pour Amazon Transcribe à l'aide de kits de développement logiciel AWS

Les exemples de code suivants vous montrent comment implémenter des scénarios courants dans Amazon Transcribe à l'aide AWS de kits SDK. Ces scénarios vous montrent comment accomplir des tâches spécifiques en appelant plusieurs fonctions dans Amazon Transcribe. Chaque scénario inclut un lien vers GitHub, où vous pouvez trouver des instructions sur la façon de configurer et d'exécuter le code.

Exemples

- [Créez et affinez un vocabulaire personnalisé Amazon Transcribe à l'aide d'un SDK AWS](#)
- [Transcrivez du son et obtenez des données de travail avec Amazon Transcribe à l'aide d'un SDK AWS](#)

Créez et affinez un vocabulaire personnalisé Amazon Transcribe à l'aide d'un SDK AWS

L'exemple de code suivant illustre comment :

- télécharger un fichier audio dans Amazon S3 ;
- exécuter une tâche Amazon Transcribe pour transcrire le fichier et obtenir des résultats ;
- créer et affiner un vocabulaire personnalisé pour améliorer la précision de la transcription ;
- exécuter des tâches avec des vocabulaires personnalisés et obtenir des résultats.

Python

SDK pour Python (Boto3)

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

Transcrivez un fichier audio contenant une lecture de Jabberwocky par Lewis Carroll. Commencez par créer des fonctions qui encapsulent les actions Amazon Transcribe.

```
def start_job(
    job_name,
    media_uri,
    media_format,
    language_code,
    transcribe_client,
    vocabulary_name=None,
):
    """
    Starts a transcription job. This function returns as soon as the job is
    started.
    To get the current status of the job, call get_transcription_job. The job is
    successfully completed when the job status is 'COMPLETED'.

    :param job_name: The name of the transcription job. This must be unique for
        your AWS account.
    :param media_uri: The URI where the audio file is stored. This is typically
        in an Amazon S3 bucket.
    :param media_format: The format of the audio file. For example, mp3 or wav.
    :param language_code: The language code of the audio file.
        For example, en-US or ja-JP
    :param transcribe_client: The Boto3 Transcribe client.
```

```

    :param vocabulary_name: The name of a custom vocabulary to use when
transcribing
                                the audio file.
:return: Data about the job.
"""
try:
    job_args = {
        "TranscriptionJobName": job_name,
        "Media": {"MediaFileUri": media_uri},
        "MediaFormat": media_format,
        "LanguageCode": language_code,
    }
    if vocabulary_name is not None:
        job_args["Settings"] = {"VocabularyName": vocabulary_name}
    response = transcribe_client.start_transcription_job(**job_args)
    job = response["TranscriptionJob"]
    logger.info("Started transcription job %s.", job_name)
except ClientError:
    logger.exception("Couldn't start transcription job %s.", job_name)
    raise
else:
    return job

def get_job(job_name, transcribe_client):
    """
    Gets details about a transcription job.

    :param job_name: The name of the job to retrieve.
    :param transcribe_client: The Boto3 Transcribe client.
    :return: The retrieved transcription job.
    """
    try:
        response = transcribe_client.get_transcription_job(
            TranscriptionJobName=job_name
        )
        job = response["TranscriptionJob"]
        logger.info("Got job %s.", job["TranscriptionJobName"])
    except ClientError:
        logger.exception("Couldn't get job %s.", job_name)
        raise
    else:
        return job
```

```
def delete_job(job_name, transcribe_client):
    """
    Deletes a transcription job. This also deletes the transcript associated with
    the job.

    :param job_name: The name of the job to delete.
    :param transcribe_client: The Boto3 Transcribe client.
    """
    try:
        transcribe_client.delete_transcription_job(TranscriptionJobName=job_name)
        logger.info("Deleted job %s.", job_name)
    except ClientError:
        logger.exception("Couldn't delete job %s.", job_name)
        raise

def create_vocabulary(
    vocabulary_name, language_code, transcribe_client, phrases=None,
    table_uri=None
):
    """
    Creates a custom vocabulary that can be used to improve the accuracy of
    transcription jobs. This function returns as soon as the vocabulary
    processing
    is started. Call get_vocabulary to get the current status of the vocabulary.
    The vocabulary is ready to use when its status is 'READY'.

    :param vocabulary_name: The name of the custom vocabulary.
    :param language_code: The language code of the vocabulary.
        For example, en-US or nl-NL.
    :param transcribe_client: The Boto3 Transcribe client.
    :param phrases: A list of comma-separated phrases to include in the
    vocabulary.
    :param table_uri: A table of phrases and pronunciation hints to include in
    the
        vocabulary.
    :return: Information about the newly created vocabulary.
    """
    try:
```



```
    vocab_args = {"VocabularyName": vocabulary_name, "LanguageCode":
language_code}
    if phrases is not None:
        vocab_args["Phrases"] = phrases
    elif table_uri is not None:
        vocab_args["VocabularyFileUri"] = table_uri
    response = transcribe_client.create_vocabulary(**vocab_args)
    logger.info("Created custom vocabulary %s.", response["VocabularyName"])
except ClientError:
    logger.exception("Couldn't create custom vocabulary %s.",
vocabulary_name)
    raise
else:
    return response

def get_vocabulary(vocabulary_name, transcribe_client):
    """
    Gets information about a custom vocabulary.

    :param vocabulary_name: The name of the vocabulary to retrieve.
    :param transcribe_client: The Boto3 Transcribe client.
    :return: Information about the vocabulary.
    """
    try:
        response =
transcribe_client.get_vocabulary(VocabularyName=vocabulary_name)
        logger.info("Got vocabulary %s.", response["VocabularyName"])
    except ClientError:
        logger.exception("Couldn't get vocabulary %s.", vocabulary_name)
        raise
    else:
        return response

def update_vocabulary(
    vocabulary_name, language_code, transcribe_client, phrases=None,
    table_uri=None
):
    """
    Updates an existing custom vocabulary. The entire vocabulary is replaced with
    the contents of the update.
```

```

:param vocabulary_name: The name of the vocabulary to update.
:param language_code: The language code of the vocabulary.
:param transcribe_client: The Boto3 Transcribe client.
:param phrases: A list of comma-separated phrases to include in the
vocabulary.
:param table_uri: A table of phrases and pronunciation hints to include in
the
                vocabulary.
"""
try:
    vocab_args = {"VocabularyName": vocabulary_name, "LanguageCode":
language_code}
    if phrases is not None:
        vocab_args["Phrases"] = phrases
    elif table_uri is not None:
        vocab_args["VocabularyFileUri"] = table_uri
    response = transcribe_client.update_vocabulary(**vocab_args)
    logger.info("Updated custom vocabulary %s.", response["VocabularyName"])
except ClientError:
    logger.exception("Couldn't update custom vocabulary %s.",
vocabulary_name)
    raise

def list_vocabularies(vocabulary_filter, transcribe_client):
    """
    Lists the custom vocabularies created for this AWS account.

    :param vocabulary_filter: The returned vocabularies must contain this string
in
                            their names.
    :param transcribe_client: The Boto3 Transcribe client.
    :return: The list of retrieved vocabularies.
    """
    try:
        response =
transcribe_client.list_vocabularies(NameContains=vocabulary_filter)
        vocabs = response["Vocabularies"]
        next_token = response.get("NextToken")
        while next_token is not None:
            response = transcribe_client.list_vocabularies(
                NameContains=vocabulary_filter, NextToken=next_token

```

```

        )
        vocabs += response["Vocabularies"]
        next_token = response.get("NextToken")
    logger.info(
        "Got %s vocabularies with filter %s.", len(vocabs), vocabulary_filter
    )
except ClientError:
    logger.exception(
        "Couldn't list vocabularies with filter %s.", vocabulary_filter
    )
    raise
else:
    return vocabs

def delete_vocabulary(vocabulary_name, transcribe_client):
    """
    Deletes a custom vocabulary.

    :param vocabulary_name: The name of the vocabulary to delete.
    :param transcribe_client: The Boto3 Transcribe client.
    """
    try:
        transcribe_client.delete_vocabulary(VocabularyName=vocabulary_name)
        logger.info("Deleted vocabulary %s.", vocabulary_name)
    except ClientError:
        logger.exception("Couldn't delete vocabulary %s.", vocabulary_name)
        raise

```

Appelez les fonctions de l'encapsuleur pour transcrire le fichier audio sans vocabulaire personnalisé, puis avec différentes versions d'un vocabulaire personnalisé pour obtenir de meilleurs résultats.

```

def usage_demo():
    """Shows how to use the Amazon Transcribe service."""
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    s3_resource = boto3.resource("s3")
    transcribe_client = boto3.client("transcribe")

```

```
print("-" * 88)
print("Welcome to the Amazon Transcribe demo!")
print("-" * 88)

bucket_name = f"jabber-bucket-{time.time_ns()}"
print(f"Creating bucket {bucket_name}.")
bucket = s3_resource.create_bucket(
    Bucket=bucket_name,
    CreateBucketConfiguration={
        "LocationConstraint": transcribe_client.meta.region_name
    },
)
media_file_name = ".media/Jabberwocky.mp3"
media_object_key = "Jabberwocky.mp3"
print(f"Uploading media file {media_file_name}.")
bucket.upload_file(media_file_name, media_object_key)
media_uri = f"s3://{bucket.name}/{media_object_key}"

job_name_simple = f"Jabber-{time.time_ns()}"
print(f"Starting transcription job {job_name_simple}.")
start_job(
    job_name_simple,
    f"s3://{bucket.name}/{media_object_key}",
    "mp3",
    "en-US",
    transcribe_client,
)
transcribe_waiter = TranscribeCompleteWaiter(transcribe_client)
transcribe_waiter.wait(job_name_simple)
job_simple = get_job(job_name_simple, transcribe_client)
transcript_simple = requests.get(
    job_simple["Transcript"]["TranscriptFileUri"]
).json()
print(f"Transcript for job {transcript_simple['jobName']}:")
print(transcript_simple["results"]["transcripts"][0]["transcript"])

print("-" * 88)
print(
    "Creating a custom vocabulary that lists the nonsense words to try to "
    "improve the transcription."
)
vocabulary_name = f"Jabber-vocabulary-{time.time_ns()}"
create_vocabulary(
```

```
    vocabulary_name,
    "en-US",
    transcribe_client,
    phrases=[
        "brillig",
        "slithy",
        "borogoves",
        "mome",
        "raths",
        "Jub-Jub",
        "frumious",
        "manxome",
        "Tumtum",
        "uffish",
        "whiffling",
        "tulgey",
        "thou",
        "frabjous",
        "callooh",
        "callay",
        "chortled",
    ],
)
vocabulary_ready_waiter = VocabularyReadyWaiter(transcribe_client)
vocabulary_ready_waiter.wait(vocabulary_name)

job_name_vocabulary_list = f"Jabber-vocabulary-list-{time.time_ns()}"
print(f"Starting transcription job {job_name_vocabulary_list}.")
start_job(
    job_name_vocabulary_list,
    media_uri,
    "mp3",
    "en-US",
    transcribe_client,
    vocabulary_name,
)
transcribe_waiter.wait(job_name_vocabulary_list)
job_vocabulary_list = get_job(job_name_vocabulary_list, transcribe_client)
transcript_vocabulary_list = requests.get(
    job_vocabulary_list["Transcript"]["TranscriptFileUri"]
).json()
print(f"Transcript for job {transcript_vocabulary_list['jobName']}:")
print(transcript_vocabulary_list["results"]["transcripts"][0]["transcript"])
```

```

print("-" * 88)
print(
    "Updating the custom vocabulary with table data that provides additional
"
    "pronunciation hints."
)
table_vocab_file = "jabber-vocabulary-table.txt"
bucket.upload_file(table_vocab_file, table_vocab_file)
update_vocabulary(
    vocabulary_name,
    "en-US",
    transcribe_client,
    table_uri=f"s3://{bucket.name}/{table_vocab_file}",
)
vocabulary_ready_waiter.wait(vocabulary_name)

job_name_vocab_table = f"Jabber-vocab-table-{time.time_ns()}"
print(f"Starting transcription job {job_name_vocab_table}.")
start_job(
    job_name_vocab_table,
    media_uri,
    "mp3",
    "en-US",
    transcribe_client,
    vocabulary_name=vocabulary_name,
)
transcribe_waiter.wait(job_name_vocab_table)
job_vocab_table = get_job(job_name_vocab_table, transcribe_client)
transcript_vocab_table = requests.get(
    job_vocab_table["Transcript"]["TranscriptFileUri"]
).json()
print(f"Transcript for job {transcript_vocab_table['jobName']}:")
print(transcript_vocab_table["results"]["transcripts"][0]["transcript"])

print("-" * 88)
print("Getting data for jobs and vocabularies.")
jabber_jobs = list_jobs("Jabber", transcribe_client)
print(f"Found {len(jabber_jobs)} jobs:")
for job_sum in jabber_jobs:
    job = get_job(job_sum["TranscriptionJobName"], transcribe_client)
    print(
        f"\t{job['TranscriptionJobName']}, {job['Media']['MediaFileUri']}, "
        f"{job['Settings'].get('VocabularyName')}"
    )

```

```
jabber_vocabs = list_vocabularies("Jabber", transcribe_client)
print(f"Found {len(jabber_vocabs)} vocabularies:")
for vocab_sum in jabber_vocabs:
    vocab = get_vocabulary(vocab_sum["VocabularyName"], transcribe_client)
    vocab_content = requests.get(vocab["DownloadUri"]).text
    print(f"\t{vocab['VocabularyName']} contents:")
    print(vocab_content)

print("-" * 88)
print("Deleting demo jobs.")
for job_name in [job_name_simple, job_name_vocabulary_list,
job_name_vocab_table]:
    delete_job(job_name, transcribe_client)
print("Deleting demo vocabulary.")
delete_vocabulary(vocabulary_name, transcribe_client)
print("Deleting demo bucket.")
bucket.objects.delete()
bucket.delete()
print("Thanks for watching!")
```

- Pour plus d'informations sur l'API, consultez les rubriques suivantes dans AWS SDK for Python (Boto3) API Reference.
 - [CreateVocabulary](#)
 - [DeleteTranscriptionJob](#)
 - [DeleteVocabulary](#)
 - [GetTranscriptionJob](#)
 - [GetVocabulary](#)
 - [ListVocabularies](#)
 - [StartTranscriptionJob](#)
 - [UpdateVocabulary](#)

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultez [Utilisation de ce service avec un AWS SDK](#). Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Transcrivez du son et obtenez des données de travail avec Amazon Transcribe à l'aide d'un SDK AWS

Les exemples de code suivants montrent comment :

- Démarrez une tâche de transcription avec Amazon Transcribe.
- Attendez que la tâche se termine.
- Obtenez l'URI dans lequel la transcription est stockée.

Pour plus d'informations, consultez la section [Getting started with Amazon Transcribe](#).

Java

SDK pour Java 2.x

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

Transcrit un fichier PCM.

```
/**
 * To run this AWS code example, ensure that you have set up your development
 * environment, including your AWS credentials.
 *
 * For information, see this documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-
 * started.html
 */

public class TranscribeStreamingDemoFile {
    private static final Region REGION = Region.US_EAST_1;
    private static TranscribeStreamingAsyncClient client;

    public static void main(String args[]) throws ExecutionException,
    InterruptedException {
```



```
final String USAGE = "\n" +
    "Usage:\n" +
    "  <file> \n\n" +
    "Where:\n" +
    "  file - the location of a PCM file to transcribe. In this
example, ensure the PCM file is 16 hertz (Hz). \n";

if (args.length != 1) {
    System.out.println(USAGE);
    System.exit(1);
}

String file = args[0];
client = TranscribeStreamingAsyncClient.builder()
    .region(REGION)
    .build();

CompletableFuture<Void> result =
client.startStreamTranscription(getRequest(16_000),
    new AudioStreamPublisher(getStreamFromFile(file)),
    getResponseHandler());

result.get();
client.close();
}

private static InputStream getStreamFromFile(String file) {
    try {
        File inputFile = new File(file);
        InputStream audioStream = new FileInputStream(inputFile);
        return audioStream;

    } catch (FileNotFoundException e) {
        throw new RuntimeException(e);
    }
}

private static StartStreamTranscriptionRequest getRequest(Integer
mediaSampleRateHertz) {
    return StartStreamTranscriptionRequest.builder()
        .languageCode(LanguageCode.EN_US)
        .mediaEncoding(MediaEncoding.PCM)
        .mediaSampleRateHertz(mediaSampleRateHertz)
        .build();
}
```

```

    }

    private static StartStreamTranscriptionResponseHandler getResponseHandler() {
        return StartStreamTranscriptionResponseHandler.builder()
            .onResponse(r -> {
                System.out.println("Received Initial response");
            })
            .onError(e -> {
                System.out.println(e.getMessage());
                StringWriter sw = new StringWriter();
                e.printStackTrace(new PrintWriter(sw));
                System.out.println("Error Occurred: " + sw.toString());
            })
            .onComplete(() -> {
                System.out.println("=== All records stream successfully
===");
            })
            .subscriber(event -> {
                List<Result> results = ((TranscriptEvent)
event).transcript().results();
                if (results.size() > 0) {
                    if (!
results.get(0).alternatives().get(0).transcript().isEmpty()) {

                        System.out.println(results.get(0).alternatives().get(0).transcript());
                    }
                }
            })
            .build();
    }

    private static class AudioStreamPublisher implements Publisher<AudioStream> {
        private final InputStream inputStream;
        private static Subscription currentSubscription;

        private AudioStreamPublisher(InputStream inputStream) {
            this.inputStream = inputStream;
        }

        @Override
        public void subscribe(Subscriber<? super AudioStream> s) {

            if (this.currentSubscription == null) {
                this.currentSubscription = new SubscriptionImpl(s, inputStream);
            }
        }
    }

```

```
        } else {
            this.currentSubscription.cancel();
            this.currentSubscription = new SubscriptionImpl(s, inputStream);
        }
        s.onSubscribe(currentSubscription);
    }
}

public static class SubscriptionImpl implements Subscription {
    private static final int CHUNK_SIZE_IN_BYTES = 1024 * 1;
    private final Subscriber<? super AudioStream> subscriber;
    private final InputStream inputStream;
    private ExecutorService executor = Executors.newFixedThreadPool(1);
    private AtomicLong demand = new AtomicLong(0);

    SubscriptionImpl(Subscriber<? super AudioStream> s, InputStream
inputStream) {
        this.subscriber = s;
        this.inputStream = inputStream;
    }

    @Override
    public void request(long n) {
        if (n <= 0) {
            subscriber.onError(new IllegalArgumentException("Demand must be
positive"));
        }

        demand.getAndAdd(n);

        executor.submit(() -> {
            try {
                do {
                    ByteBuffer audioBuffer = getNextEvent();
                    if (audioBuffer.remaining() > 0) {
                        AudioEvent audioEvent =
audioEventFromBuffer(audioBuffer);
                        subscriber.onNext(audioEvent);
                    } else {
                        subscriber.onComplete();
                        break;
                    }
                }
                while (demand.decrementAndGet() > 0);
            } catch (Exception e) {
```

```

        subscriber.onError(e);
    }
    });
}

@Override
public void cancel() {
    executor.shutdown();
}

private ByteBuffer getNextEvent() {
    ByteBuffer audioBuffer = null;
    byte[] audioBytes = new byte[CHUNK_SIZE_IN_BYTES];

    int len = 0;
    try {
        len = inputStream.read(audioBytes);

        if (len <= 0) {
            audioBuffer = ByteBuffer.allocate(0);
        } else {
            audioBuffer = ByteBuffer.wrap(audioBytes, 0, len);
        }
    } catch (IOException e) {
        throw new UncheckedIOException(e);
    }

    return audioBuffer;
}

private AudioEvent audioEventFromBuffer(ByteBuffer bb) {
    return AudioEvent.builder()
        .audioChunk(SdkBytes.fromByteBuffer(bb))
        .build();
}
}
}

```

Transcrit l'audio en streaming depuis le microphone de l'ordinateur.

```

public class TranscribeStreamingDemoApp {
    private static final Region REGION = Region.US_EAST_1;

```

```
private static TranscribeStreamingAsyncClient client;

public static void main(String args[])
    throws URISyntaxException, ExecutionException, InterruptedException,
LineUnavailableException {

    client = TranscribeStreamingAsyncClient.builder()
        .credentialsProvider(getCredentials())
        .region(REGION)
        .build();

    CompletableFuture<Void> result =
client.startStreamTranscription(getRequest(16_000),
    new AudioStreamPublisher(getStreamFromMic()),
    getResponseHandler());

    result.get();
    client.close();
}

private static InputStream getStreamFromMic() throws LineUnavailableException
{

    // Signed PCM AudioFormat with 16kHz, 16 bit sample size, mono
    int sampleRate = 16000;
    AudioFormat format = new AudioFormat(sampleRate, 16, 1, true, false);
    DataLine.Info info = new DataLine.Info(TargetDataLine.class, format);

    if (!AudioSystem.isLineSupported(info)) {
        System.out.println("Line not supported");
        System.exit(0);
    }

    TargetDataLine line = (TargetDataLine) AudioSystem.getLine(info);
    line.open(format);
    line.start();

    InputStream audioStream = new AudioInputStream(line);
    return audioStream;
}

private static AwsCredentialsProvider getCredentials() {
    return DefaultCredentialsProvider.create();
}
```

```
private static StartStreamTranscriptionRequest getRequest(Integer
mediaSampleRateHertz) {
    return StartStreamTranscriptionRequest.builder()
        .languageCode(LanguageCode.EN_US.toString())
        .mediaEncoding(MediaEncoding.PCM)
        .mediaSampleRateHertz(mediaSampleRateHertz)
        .build();
}

private static StartStreamTranscriptionResponseHandler getResponseHandler() {
    return StartStreamTranscriptionResponseHandler.builder()
        .onResponse(r -> {
            System.out.println("Received Initial response");
        })
        .onError(e -> {
            System.out.println(e.getMessage());
            StringWriter sw = new StringWriter();
            e.printStackTrace(new PrintWriter(sw));
            System.out.println("Error Occurred: " + sw.toString());
        })
        .onComplete(() -> {
            System.out.println("=== All records stream successfully
===");
        })
        .subscriber(event -> {
            List<Result> results = ((TranscriptEvent)
event).transcript().results();
            if (results.size() > 0) {
                if (!
results.get(0).alternatives().get(0).transcript().isEmpty()) {
                    System.out.println(results.get(0).alternatives().get(0).transcript());
                }
            }
        })
        .build();
}

private InputStream getStreamFromFile(String audioFileName) {
    try {
        File inputFile = new
File(getClass().getClassLoader().getResource(audioFileName).getFile());
        InputStream audioStream = new FileInputStream(inputFile);
    }
}
```

```
        return audioStream;
    } catch (FileNotFoundException e) {
        throw new RuntimeException(e);
    }
}

private static class AudioStreamPublisher implements Publisher<AudioStream> {
    private final InputStream inputStream;
    private static Subscription currentSubscription;

    private AudioStreamPublisher(InputStream inputStream) {
        this.inputStream = inputStream;
    }

    @Override
    public void subscribe(Subscriber<? super AudioStream> s) {

        if (this.currentSubscription == null) {
            this.currentSubscription = new SubscriptionImpl(s, inputStream);
        } else {
            this.currentSubscription.cancel();
            this.currentSubscription = new SubscriptionImpl(s, inputStream);
        }
        s.onSubscribe(currentSubscription);
    }
}

public static class SubscriptionImpl implements Subscription {
    private static final int CHUNK_SIZE_IN_BYTES = 1024 * 1;
    private final Subscriber<? super AudioStream> subscriber;
    private final InputStream inputStream;
    private ExecutorService executor = Executors.newFixedThreadPool(1);
    private AtomicLong demand = new AtomicLong(0);

    SubscriptionImpl(Subscriber<? super AudioStream> s, InputStream
inputStream) {
        this.subscriber = s;
        this.inputStream = inputStream;
    }

    @Override
    public void request(long n) {
        if (n <= 0) {
```

```
        subscriber.onError(new IllegalArgumentException("Demand must be
positive"));
    }

    demand.getAndAdd(n);

    executor.submit(() -> {
        try {
            do {
                ByteBuffer audioBuffer = getNextEvent();
                if (audioBuffer.remaining() > 0) {
                    AudioEvent audioEvent =
audioEventFromBuffer(audioBuffer);
                    subscriber.onNext(audioEvent);
                } else {
                    subscriber.onComplete();
                    break;
                }
            } while (demand.decrementAndGet() > 0);
        } catch (Exception e) {
            subscriber.onError(e);
        }
    });
}

@Override
public void cancel() {
    executor.shutdown();
}

private ByteBuffer getNextEvent() {
    ByteBuffer audioBuffer = null;
    byte[] audioBytes = new byte[CHUNK_SIZE_IN_BYTES];

    int len = 0;
    try {
        len = inputStream.read(audioBytes);

        if (len <= 0) {
            audioBuffer = ByteBuffer.allocate(0);
        } else {
            audioBuffer = ByteBuffer.wrap(audioBytes, 0, len);
        }
    } catch (IOException e) {
```



```
        throw new UncheckedIOException(e);
    }

    return audioBuffer;
}

private AudioEvent audioEventFromBuffer(ByteBuffer bb) {
    return AudioEvent.builder()
        .audioChunk(SdkBytes.fromByteBuffer(bb))
        .build();
}
}
```

- Pour plus d'informations sur l'API consultez les rubriques suivantes dans la référence de l'API AWS SDK for Java 2.x .
 - [GetTranscriptionJob](#)
 - [StartTranscriptionJob](#)

Python

SDK pour Python (Boto3)

Note

Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le [référentiel d'exemples de code AWS](#).

```
import time
import boto3

def transcribe_file(job_name, file_uri, transcribe_client):
    transcribe_client.start_transcription_job(
        TranscriptionJobName=job_name,
        Media={"MediaFileUri": file_uri},
        MediaFormat="wav",
        LanguageCode="en-US",
```

```
)

max_tries = 60
while max_tries > 0:
    max_tries -= 1
    job =
transcribe_client.get_transcription_job(TranscriptionJobName=job_name)
    job_status = job["TranscriptionJob"]["TranscriptionJobStatus"]
    if job_status in ["COMPLETED", "FAILED"]:
        print(f"Job {job_name} is {job_status}.")
        if job_status == "COMPLETED":
            print(
                f"Download the transcript from\n"
                f"\t{job['TranscriptionJob']['Transcript']}"
                f'["TranscriptFileUri"]}.')
            )
            break
    else:
        print(f"Waiting for {job_name}. Current status is {job_status}.")
        time.sleep(10)

def main():
    transcribe_client = boto3.client("transcribe")
    file_uri = "s3://test-transcribe/answer2.wav"
    transcribe_file("Example-job", file_uri, transcribe_client)

if __name__ == "__main__":
    main()
```

- Pour plus d'informations sur l'API, consultez les rubriques suivantes dans AWS SDK for Python (Boto3) API Reference.
 - [GetTranscriptionJob](#)
 - [StartTranscriptionJob](#)

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultez [Utilisation de ce service avec un AWS SDK](#). Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Exemples multiservices pour Amazon AWS Transcribe utilisant des SDK

Les exemples d'applications suivants utilisent des AWS SDK pour associer Amazon Transcribe à d'autres applications. Services AWS Chaque exemple inclut un lien vers GitHub, où vous pouvez trouver des instructions sur la façon de configurer et d'exécuter l'application.

Exemples

- [Créer une application Amazon Transcribe](#)
- [Créer une application de streaming Amazon Transcribe](#)
- [Convertissez du texte en parole et de nouveau en texte à l'aide d'un AWS SDK](#)

Créer une application Amazon Transcribe

L'exemple de code suivant montre comment utiliser Amazon Transcribe pour transcrire et afficher des enregistrements vocaux dans le navigateur.

JavaScript

SDK pour JavaScript (v3)

Créez une application qui utilise Amazon Transcribe pour transcrire et afficher des enregistrements vocaux dans le navigateur. L'application utilise deux compartiments Amazon Simple Storage Service (Amazon S3), l'un pour héberger le code de l'application, l'autre pour stocker les transcriptions. L'application utilise un groupe d'utilisateurs Amazon Cognito pour authentifier vos utilisateurs. Les utilisateurs authentifiés disposent des autorisations AWS Identity and Access Management (IAM) nécessaires pour accéder aux services requis. AWS

Pour obtenir le code source complet et les instructions de configuration et d'exécution, consultez l'exemple complet sur [GitHub](#).

Cet exemple est également disponible dans le [AWS SDK for JavaScript guide du développeur v3](#).

Les services utilisés dans cet exemple

- Amazon Cognito Identity
- Amazon S3

- Amazon Transcribe

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultez [Utilisation de ce service avec un AWS SDK](#). Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Créer une application de streaming Amazon Transcribe

L'exemple de code suivant montre comment créer une application qui enregistre, transcrit et traduit de l'audio en direct en temps réel, et envoie les résultats par e-mail.

JavaScript

SDK pour JavaScript (v3)

Montre comment utiliser Amazon Transcribe afin de créer une application qui enregistre, transcrit et traduit de l'audio en direct en temps réel, et envoie les résultats par e-mail à l'aide d'Amazon Simple Email Service (Amazon SES).

Pour obtenir le code source complet et les instructions de configuration et d'exécution, consultez l'exemple complet sur [GitHub](#).

Les services utilisés dans cet exemple

- Amazon Comprehend
- Amazon SES
- Amazon Transcribe
- Amazon Translate

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultez [Utilisation de ce service avec un AWS SDK](#). Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Convertissez du texte en parole et de nouveau en texte à l'aide d'un AWS SDK

L'exemple de code suivant illustre comment :

- Utilisez Amazon Polly pour synthétiser un fichier d'entrée en texte brut (UTF-8) en un fichier audio.

- Chargez le fichier audio sur un compartiment Amazon S3.
- Utilisez Amazon Transcribe pour convertir le fichier audio en texte.
- Affichez le texte.

Rust

SDK pour Rust

Utilisez Amazon Polly pour synthétiser un fichier d'entrée en texte brut (UTF-8) en un fichier audio, charger le fichier audio dans un compartiment Amazon S3, utiliser Amazon Transcribe pour convertir ce fichier audio en texte et afficher le texte.

Pour obtenir le code source complet et les instructions de configuration et d'exécution, consultez l'exemple complet sur [GitHub](#).

Les services utilisés dans cet exemple

- Amazon Polly
- Amazon S3
- Amazon Transcribe

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultez [Utilisation de ce service avec un AWS SDK](#). Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Sécurité dans Amazon Transcribe

Chez AWS, la sécurité dans le cloud est notre priorité numéro 1. En tant que client d'AWS, vous bénéficiez d'un centre de données et d'une architecture réseau conçus pour répondre aux exigences des organisations les plus pointilleuses en termes de sécurité.

La sécurité est une responsabilité partagée entre AWS et vous. Le [modèle de responsabilité partagée](#) décrit cela comme la sécurité du cloud et la sécurité dans le cloud :

- Sécurité du cloud : AWS est responsable de la protection de l'infrastructure qui exécute les AWS services dans le cloud AWS Cloud. AWS vous des services que vous utiliser en toute sécurité. Des auditeurs tiers testent et vérifient régulièrement l'efficacité de notre sécurité dans le cadre des [AWS programmes de conformité](#). Pour en savoir plus sur les programmes de conformité qui s'appliquent à Amazon Transcribe, consultez [Services AWS concernés par le programme de conformité](#).
- Sécurité dans le cloud : votre responsabilité est déterminée par le AWS service que vous utiliser. Vous êtes également responsable d'autres facteurs, y compris de la sensibilité de vos données, des exigences de votre entreprise, ainsi que de la législation et de la réglementation applicables.

Cette documentation vous aide à comprendre comment appliquer le modèle de responsabilité partagée lors de l'utilisation de Amazon Transcribe. Les rubriques suivantes expliquent comment configurer Amazon Transcribe pour répondre à vos objectifs de sécurité et de conformité. Vous pouvez utiliser d'autres AWS services pour surveiller et sécuriser vos Amazon Transcribe ressources.

Rubriques

- [Identity and Access Management pour Amazon Transcribe](#)
- [Protection des données dans Amazon Transcribe](#)
- [Surveillance Amazon Transcribe](#)
- [Validation de conformité pour Amazon Transcribe](#)
- [Résilience dans Amazon Transcribe](#)
- [Sécurité de l'infrastructure dans Amazon Transcribe](#)
- [Analyse et gestion des vulnérabilités dans Amazon Transcribe](#)
- [Bonnes pratiques de sécurité pour Amazon Transcribe](#)

Identity and Access Management pour Amazon Transcribe

AWS Identity and Access Management (IAM) est un outil Service AWS qui permet à un administrateur de contrôler en toute sécurité l'accès aux AWS ressources. Les administrateurs IAM contrôlent qui peut être authentifié (connecté) et autorisé (autorisé) à utiliser Amazon Transcribe les ressources. IAM est un Service AWS outil que vous pouvez utiliser sans frais supplémentaires.

Rubriques

- [Public ciblé](#)
- [Authentification par des identités](#)
- [Gestion des accès à l'aide de politiques](#)
- [Comment Amazon Transcribe fonctionne avec IAM](#)
- [Prévention du problème de l'adjoint confus entre services](#)
- [Exemples de politiques basées sur l'identité Amazon Transcribe](#)
- [Résolution des problèmes Amazon Transcribe d'identité et d'accès](#)

Public ciblé

La façon dont vous utilisez AWS Identity and Access Management (IAM) varie en fonction du travail que vous effectuez. Amazon Transcribe

Utilisateur du service : si vous utilisez le Amazon Transcribe service pour effectuer votre travail, votre administrateur vous fournit les informations d'identification et les autorisations dont vous avez besoin. Au fur et à mesure que vous utilisez de nouvelles Amazon Transcribe fonctionnalités pour effectuer votre travail, vous aurez peut-être besoin d'autorisations supplémentaires. En comprenant bien la gestion des accès, vous saurez demander les autorisations appropriées à votre administrateur. Si vous ne pouvez pas accéder à une fonctionnalité dans Amazon Transcribe, consultez [Résolution des problèmes Amazon Transcribe d'identité et d'accès](#).

Administrateur du service — Si vous êtes responsable des Amazon Transcribe ressources de votre entreprise, vous avez probablement un accès complet à Amazon Transcribe. C'est à vous de déterminer les Amazon Transcribe fonctionnalités et les ressources auxquelles les utilisateurs de votre service doivent accéder. Vous devez ensuite soumettre les demandes à votre administrateur IAM pour modifier les autorisations des utilisateurs de votre service. Consultez les informations sur cette page pour comprendre les concepts de base d'IAM. Pour en savoir plus sur

la manière dont votre entreprise peut utiliser IAM avec Amazon Transcribe, voir [Comment Amazon Transcribe fonctionne avec IAM](#).

Administrateur IAM – Si vous êtes un administrateur IAM, vous souhaitez peut-être en savoir plus sur la façon d'écrire des politiques pour gérer l'accès à Amazon Transcribe. Pour consulter des exemples de politiques Amazon Transcribe basées sur l'identité que vous pouvez utiliser dans IAM, consultez. [Exemples de politiques basées sur l'identité Amazon Transcribe](#)

Authentification par des identités

L'authentification est la façon dont vous vous connectez à AWS l'aide de vos informations d'identification. Vous devez être authentifié (connecté à AWS) en tant qu'utilisateur IAM ou en assumant un rôle IAM. Utilisateur racine d'un compte AWS

Vous pouvez vous connecter en AWS tant qu'identité fédérée en utilisant les informations d'identification fournies par le biais d'une source d'identité. AWS IAM Identity Center Les utilisateurs (IAM Identity Center), l'authentification unique de votre entreprise et vos informations d'identification Google ou Facebook sont des exemples d'identités fédérées. Lorsque vous vous connectez avec une identité fédérée, votre administrateur aura précédemment configuré une fédération d'identités avec des rôles IAM. Lorsque vous accédez à AWS l'aide de la fédération, vous assumez indirectement un rôle.

Selon le type d'utilisateur que vous êtes, vous pouvez vous connecter au portail AWS Management Console ou au portail AWS d'accès. Pour plus d'informations sur la connexion à AWS, consultez la section [Comment vous connecter à votre compte Compte AWS dans](#) le guide de Connexion à AWS l'utilisateur.

Si vous y accédez AWS par programmation, AWS fournit un kit de développement logiciel (SDK) et une interface de ligne de commande (CLI) pour signer cryptographiquement vos demandes à l'aide de vos informations d'identification. Si vous n'utilisez pas d' AWS outils, vous devez signer vous-même les demandes. Pour plus d'informations sur l'utilisation de la méthode recommandée pour signer vous-même les demandes, consultez la section [Signature des demandes AWS d'API](#) dans le guide de l'utilisateur IAM.

Quelle que soit la méthode d'authentification que vous utilisez, vous devrez peut-être fournir des informations de sécurité supplémentaires. Par exemple, il vous AWS recommande d'utiliser l'authentification multifactorielle (MFA) pour renforcer la sécurité de votre compte. Pour en savoir plus, consultez [Authentification multifactorielle](#) dans le Guide de l'utilisateur AWS IAM Identity Center

et [Utilisation de l'authentification multifactorielle \(MFA\) dans l'interface AWS](#) dans le Guide de l'utilisateur IAM.

Compte AWS utilisateur root

Lorsque vous créez un Compte AWS, vous commencez par une identité de connexion unique qui donne un accès complet à toutes Services AWS les ressources du compte. Cette identité est appelée utilisateur Compte AWS root et est accessible en vous connectant avec l'adresse e-mail et le mot de passe que vous avez utilisés pour créer le compte. Il est vivement recommandé de ne pas utiliser l'utilisateur racine pour vos tâches quotidiennes. Protégez vos informations d'identification d'utilisateur racine et utilisez-les pour effectuer les tâches que seul l'utilisateur racine peut effectuer. Pour obtenir la liste complète des tâches qui vous imposent de vous connecter en tant qu'utilisateur root, consultez [Tâches nécessitant des informations d'identification d'utilisateur root](#) dans le Guide de l'utilisateur IAM.

Identité fédérée

La meilleure pratique consiste à obliger les utilisateurs humains, y compris ceux qui ont besoin d'un accès administrateur, à utiliser la fédération avec un fournisseur d'identité pour accéder à l'aide Services AWS d'informations d'identification temporaires.

Une identité fédérée est un utilisateur de l'annuaire des utilisateurs de votre entreprise, d'un fournisseur d'identité Web AWS Directory Service, du répertoire Identity Center ou de tout utilisateur qui y accède à l'aide des informations d'identification fournies Services AWS par le biais d'une source d'identité. Lorsque des identités fédérées y accèdent Comptes AWS, elles assument des rôles, qui fournissent des informations d'identification temporaires.

Pour une gestion des accès centralisée, nous vous recommandons d'utiliser AWS IAM Identity Center. Vous pouvez créer des utilisateurs et des groupes dans IAM Identity Center, ou vous pouvez vous connecter et synchroniser avec un ensemble d'utilisateurs et de groupes dans votre propre source d'identité afin de les utiliser dans toutes vos applications Comptes AWS et applications. Pour obtenir des informations sur IAM Identity Center, consultez [Qu'est-ce que IAM Identity Center ?](#) dans le Guide de l'utilisateur AWS IAM Identity Center .

Utilisateurs et groupes IAM

Un [utilisateur IAM](#) est une identité au sein de votre Compte AWS qui possède des autorisations spécifiques pour une seule personne ou application. Dans la mesure du possible, nous vous recommandons de vous appuyer sur des informations d'identification temporaires plutôt que de créer des utilisateurs IAM ayant des informations d'identification à long terme tels que les clés d'accès.

Toutefois, si certains cas d'utilisation spécifiques nécessitent des informations d'identification à long terme avec les utilisateurs IAM, nous vous recommandons de faire pivoter les clés d'accès. Pour plus d'informations, consultez [Rotation régulière des clés d'accès pour les cas d'utilisation nécessitant des informations d'identification](#) dans le Guide de l'utilisateur IAM.

Un [groupe IAM](#) est une identité qui concerne un ensemble d'utilisateurs IAM. Vous ne pouvez pas vous connecter en tant que groupe. Vous pouvez utiliser les groupes pour spécifier des autorisations pour plusieurs utilisateurs à la fois. Les groupes permettent de gérer plus facilement les autorisations pour de grands ensembles d'utilisateurs. Par exemple, vous pouvez avoir un groupe nommé IAMAdmins et accorder à ce groupe les autorisations d'administrer des ressources IAM.

Les utilisateurs sont différents des rôles. Un utilisateur est associé de manière unique à une personne ou une application, alors qu'un rôle est conçu pour être endossé par tout utilisateur qui en a besoin. Les utilisateurs disposent d'informations d'identification permanentes, mais les rôles fournissent des informations d'identification temporaires. Pour en savoir plus, consultez [Quand créer un utilisateur IAM \(au lieu d'un rôle\)](#) dans le Guide de l'utilisateur IAM.

Rôles IAM

Un [rôle IAM](#) est une identité au sein de votre Compte AWS dotée d'autorisations spécifiques. Le concept ressemble à celui d'utilisateur IAM, mais le rôle IAM n'est pas associé à une personne en particulier. Vous pouvez assumer temporairement un rôle IAM dans le en AWS Management Console [changeant de rôle](#). Vous pouvez assumer un rôle en appelant une opération d' AWS API AWS CLI ou en utilisant une URL personnalisée. Pour plus d'informations sur les méthodes d'utilisation des rôles, consultez [Utilisation de rôles IAM](#) dans le Guide de l'utilisateur IAM.

Les rôles IAM avec des informations d'identification temporaires sont utiles dans les cas suivants :

- Accès utilisateur fédéré – Pour attribuer des autorisations à une identité fédérée, vous créez un rôle et définissez des autorisations pour le rôle. Quand une identité externe s'authentifie, l'identité est associée au rôle et reçoit les autorisations qui sont définies par celui-ci. Pour obtenir des informations sur les rôles pour la fédération, consultez [Création d'un rôle pour un fournisseur d'identité tiers \(fédération\)](#) dans le Guide de l'utilisateur IAM. Si vous utilisez IAM Identity Center, vous configurez un jeu d'autorisations. IAM Identity Center met en corrélation le jeu d'autorisations avec un rôle dans IAM afin de contrôler à quoi vos identités peuvent accéder après leur authentification. Pour plus d'informations sur les jeux d'autorisations, consultez la rubrique [Jeux d'autorisations](#) dans le Guide de l'utilisateur AWS IAM Identity Center .
- Autorisations d'utilisateur IAM temporaires : un rôle ou un utilisateur IAM peut endosser un rôle IAM pour profiter temporairement d'autorisations différentes pour une tâche spécifique.

- **Accès intercompte** : vous pouvez utiliser un rôle IAM pour permettre à un utilisateur (principal de confiance) d'un compte différent d'accéder aux ressources de votre compte. Les rôles constituent le principal moyen d'accorder l'accès intercompte. Toutefois, dans certains Services AWS cas, vous pouvez associer une politique directement à une ressource (au lieu d'utiliser un rôle comme proxy). Pour connaître la différence entre les rôles et les politiques basées sur les ressources pour l'accès entre comptes, consultez la section [Accès aux ressources entre comptes dans IAM dans le guide de l'utilisateur IAM](#).
- **Accès multiservices** — Certains Services AWS utilisent des fonctionnalités dans d'autres Services AWS. Par exemple, lorsque vous effectuez un appel dans un service, il est courant que ce service exécute des applications dans Amazon EC2 ou stocke des objets dans Amazon S3. Un service peut le faire en utilisant les autorisations d'appel du principal, un rôle de service ou un rôle lié au service.
- **Sessions d'accès direct (FAS)** : lorsque vous utilisez un utilisateur ou un rôle IAM pour effectuer des actions AWS, vous êtes considéré comme un mandant. Lorsque vous utilisez certains services, vous pouvez effectuer une action qui initie une autre action dans un autre service. FAS utilise les autorisations du principal appelant et Service AWS, associées Service AWS à la demande, pour adresser des demandes aux services en aval. Les demandes FAS ne sont effectuées que lorsqu'un service reçoit une demande qui nécessite des interactions avec d'autres personnes Services AWS ou des ressources pour être traitée. Dans ce cas, vous devez disposer d'autorisations nécessaires pour effectuer les deux actions. Pour plus de détails sur la politique relative à la transmission de demandes FAS, consultez [Sessions de transmission d'accès](#).
- **Rôle de service** : il s'agit d'un [rôle IAM](#) attribué à un service afin de réaliser des actions en votre nom. Un administrateur IAM peut créer, modifier et supprimer une fonction du service à partir d'IAM. Pour plus d'informations, consultez [Création d'un rôle pour la délégation d'autorisations à un Service AWS](#) dans le Guide de l'utilisateur IAM.
- **Rôle lié à un service** — Un rôle lié à un service est un type de rôle de service lié à un. Service AWS Le service peut endosser le rôle afin d'effectuer une action en votre nom. Les rôles liés à un service apparaissent dans votre Compte AWS répertoire et appartiennent au service. Un administrateur IAM peut consulter, mais ne peut pas modifier, les autorisations concernant les rôles liés à un service.
- **Applications exécutées sur Amazon EC2** : vous pouvez utiliser un rôle IAM pour gérer les informations d'identification temporaires pour les applications qui s'exécutent sur une instance EC2 et qui envoient des demandes d'API. AWS CLI AWS Cette solution est préférable au stockage des clés d'accès au sein de l'instance EC2. Pour attribuer un AWS rôle à une instance EC2 et le mettre à la disposition de toutes ses applications, vous devez créer un profil d'instance attaché

à l'instance. Un profil d'instance contient le rôle et permet aux programmes qui s'exécutent sur l'instance EC2 d'obtenir des informations d'identification temporaires. Pour plus d'informations, consultez [Utilisation d'un rôle IAM pour accorder des autorisations à des applications s'exécutant sur des instances Amazon EC2](#) dans le Guide de l'utilisateur IAM.

Pour savoir dans quel cas utiliser des rôles ou des utilisateurs IAM, consultez [Quand créer un rôle IAM \(au lieu d'un utilisateur\)](#) dans le Guide de l'utilisateur IAM.

Gestion des accès à l'aide de politiques

Vous contrôlez l'accès en AWS créant des politiques et en les associant à AWS des identités ou à des ressources. Une politique est un objet AWS qui, lorsqu'il est associé à une identité ou à une ressource, définit leurs autorisations. AWS évalue ces politiques lorsqu'un principal (utilisateur, utilisateur root ou session de rôle) fait une demande. Les autorisations dans les politiques déterminent si la demande est autorisée ou refusée. La plupart des politiques sont stockées AWS sous forme de documents JSON. Pour plus d'informations sur la structure et le contenu des documents de politique JSON, consultez [Vue d'ensemble des politiques JSON](#) dans le Guide de l'utilisateur IAM.

Les administrateurs peuvent utiliser les politiques AWS JSON pour spécifier qui a accès à quoi. C'est-à-dire, quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

Par défaut, les utilisateurs et les rôles ne disposent d'aucune autorisation. Pour octroyer aux utilisateurs des autorisations d'effectuer des actions sur les ressources dont ils ont besoin, un administrateur IAM peut créer des politiques IAM. L'administrateur peut ensuite ajouter les politiques IAM aux rôles et les utilisateurs peuvent assumer les rôles.

Les politiques IAM définissent les autorisations d'une action, quelle que soit la méthode que vous utilisez pour exécuter l'opération. Par exemple, supposons que vous disposiez d'une politique qui autorise l'action `iam:GetRole`. Un utilisateur appliquant cette politique peut obtenir des informations sur le rôle à partir de AWS Management Console AWS CLI, de ou de l' AWS API.

Politiques basées sur l'identité

Les politiques basées sur l'identité sont des documents de politique d'autorisations JSON que vous pouvez attacher à une identité telle qu'un utilisateur, un groupe d'utilisateurs ou un rôle IAM. Ces politiques contrôlent quel type d'actions des utilisateurs et des rôles peuvent exécuter, sur quelles

ressources et dans quelles conditions. Pour découvrir comment créer une politique basée sur l'identité, consultez [Création de politiques IAM](#) dans le Guide de l'utilisateur IAM.

Les politiques basées sur l'identité peuvent être classées comme des politiques en ligne ou des politiques gérées. Les politiques en ligne sont intégrées directement à un utilisateur, groupe ou rôle. Les politiques gérées sont des politiques autonomes que vous pouvez associer à plusieurs utilisateurs, groupes et rôles au sein de votre Compte AWS. Les politiques gérées incluent les politiques AWS gérées et les politiques gérées par le client. Pour découvrir comment choisir entre une politique gérée et une politique en ligne, consultez [Choix entre les politiques gérées et les politiques en ligne](#) dans le Guide de l'utilisateur IAM.

politiques basées sur les ressources

Les politiques basées sur les ressources sont des documents de politique JSON que vous attachez à une ressource. Des politiques basées sur les ressources sont, par exemple, les politiques de confiance de rôle IAM et des politiques de compartiment. Dans les services qui sont compatibles avec les politiques basées sur les ressources, les administrateurs de service peuvent les utiliser pour contrôler l'accès à une ressource spécifique. Pour la ressource dans laquelle se trouve la politique, cette dernière définit quel type d'actions un principal spécifié peut effectuer sur cette ressource et dans quelles conditions. Vous devez [spécifier un principal](#) dans une politique basée sur les ressources. Les principaux peuvent inclure des comptes, des utilisateurs, des rôles, des utilisateurs fédérés ou. Services AWS

Les politiques basées sur les ressources sont des politiques en ligne situées dans ce service. Vous ne pouvez pas utiliser les politiques AWS gérées par IAM dans une stratégie basée sur les ressources.

Listes de contrôle d'accès (ACL)

Les listes de contrôle d'accès (ACL) vérifie quels principaux (membres de compte, utilisateurs ou rôles) ont l'autorisation d'accéder à une ressource. Les listes de contrôle d'accès sont similaires aux politiques basées sur les ressources, bien qu'elles n'utilisent pas le format de document de politique JSON.

Amazon S3 et Amazon VPC sont des exemples de services qui prennent en charge les ACL. AWS WAF Pour en savoir plus sur les listes de contrôle d'accès, consultez [Vue d'ensemble des listes de contrôle d'accès \(ACL\)](#) dans le Guide du développeur Amazon Simple Storage Service.

Autres types de politique

AWS prend en charge d'autres types de politiques moins courants. Ces types de politiques peuvent définir le nombre maximum d'autorisations qui vous sont accordées par des types de politiques plus courants.

- **Limite d'autorisations** : une limite d'autorisations est une fonctionnalité avancée dans laquelle vous définissez le nombre maximal d'autorisations qu'une politique basée sur l'identité peut accorder à une entité IAM (utilisateur ou rôle IAM). Vous pouvez définir une limite d'autorisations pour une entité. Les autorisations en résultant représentent la combinaison des politiques basées sur l'identité d'une entité et de ses limites d'autorisation. Les politiques basées sur les ressources qui spécifient l'utilisateur ou le rôle dans le champ `Principal` ne sont pas limitées par les limites d'autorisations. Un refus explicite dans l'une de ces politiques remplace l'autorisation. Pour plus d'informations sur les limites d'autorisations, consultez [Limites d'autorisations pour des entités IAM](#) dans le Guide de l'utilisateur IAM.
- **Politiques de contrôle des services (SCP)** — Les SCP sont des politiques JSON qui spécifient les autorisations maximales pour une organisation ou une unité organisationnelle (UO) dans AWS Organizations. AWS Organizations est un service permettant de regrouper et de gérer de manière centralisée des comptes AWS les multiples propriétés de votre entreprise. Si vous activez toutes les fonctionnalités d'une organisation, vous pouvez appliquer les politiques de contrôle des services (SCP) à l'un ou à l'ensemble de vos comptes. Le SCP limite les autorisations pour les entités figurant dans les comptes des membres, y compris chacune Utilisateur racine d'un compte AWS d'entre elles. Pour plus d'informations sur les organisations et les SCP, consultez [Fonctionnement des SCP](#) dans le Guide de l'utilisateur AWS Organizations .
- **Politiques de séance** : les politiques de séance sont des politiques avancées que vous utilisez en tant que paramètre lorsque vous créez par programmation une séance temporaire pour un rôle ou un utilisateur fédéré. Les autorisations de séance en résultant sont une combinaison des politiques basées sur l'identité de l'utilisateur ou du rôle et des politiques de séance. Les autorisations peuvent également provenir d'une politique basée sur les ressources. Un refus explicite dans l'une de ces politiques annule l'autorisation. Pour plus d'informations, consultez [politiques de séance](#) dans le Guide de l'utilisateur IAM.

Plusieurs types de politique

Lorsque plusieurs types de politiques s'appliquent à la requête, les autorisations en résultant sont plus compliquées à comprendre. Pour savoir comment AWS déterminer s'il faut autoriser

une demande lorsque plusieurs types de politiques sont impliqués, consultez la section [Logique d'évaluation des politiques](#) dans le guide de l'utilisateur IAM.

Comment Amazon Transcribe fonctionne avec IAM

Avant d'utiliser IAM pour gérer l'accès à Amazon Transcribe, découvrez les fonctionnalités IAM disponibles. Amazon Transcribe

IAM fonctionnalités que vous pouvez utiliser avec Amazon Transcribe

IAM fonctionnalité	Amazon Transcribe soutien
Politiques basées sur l'identité	Oui
Politiques basées sur les ressources	Non
Actions de politique	Oui
Ressources de politique	Oui
Clés de condition de politique (spécifiques au service)	Oui
ACL	Non
ABAC (identifications dans les politiques)	Partielle
Informations d'identification temporaires	Oui
Autorisations de principal	Oui
Fonctions de service	Oui
Rôles liés à un service	Non

Pour obtenir une vue d'ensemble du fonctionnement de la plupart des IAM fonctionnalités Amazon Transcribe et des autres AWS services, reportez-vous à la section [AWS Services compatibles IAM](#) dans le Guide de IAM l'utilisateur.

Stratégies basées sur l'identité pour Amazon Transcribe

Prend en charge les politiques basées sur l'identité Oui

Les politiques basées sur l'identité sont des documents de politique d'autorisations JSON que vous pouvez attacher à une identité telle qu'un utilisateur, un groupe d'utilisateurs ou un rôle IAM. Ces politiques contrôlent quel type d'actions des utilisateurs et des rôles peuvent exécuter, sur quelles ressources et dans quelles conditions. Pour découvrir comment créer une politique basée sur l'identité, consultez [Création de politiques IAM](#) dans le Guide de l'utilisateur IAM.

Avec les politiques IAM basées sur l'identité, vous pouvez spécifier des actions et ressources autorisées ou refusées, ainsi que les conditions dans lesquelles les actions sont autorisées ou refusées. Vous ne pouvez pas spécifier le principal dans une politique basée sur une identité car celle-ci s'applique à l'utilisateur ou au rôle auquel elle est attachée. Pour découvrir tous les éléments que vous utilisez dans une politique JSON, consultez [Références des éléments de politique JSON IAM](#) dans le Guide de l'utilisateur IAM.

Exemples de politiques basées sur l'identité pour Amazon Transcribe

Pour consulter des exemples de politiques Amazon Transcribe basées sur l'identité, consultez [Exemples de politiques basées sur l'identité Amazon Transcribe](#)

Politiques basées sur les ressources dans Amazon Transcribe

Prend en charge les politiques basées sur les ressources Non

Les politiques basées sur les ressources sont des documents de politique JSON que vous attachez à une ressource. Des politiques basées sur les ressources sont, par exemple, les politiques de confiance de rôle IAM et des politiques de compartiment. Dans les services qui sont compatibles avec les politiques basées sur les ressources, les administrateurs de service peuvent les utiliser pour contrôler l'accès à une ressource spécifique. Pour la ressource dans laquelle se trouve la politique, cette dernière définit quel type d'actions un principal spécifié peut effectuer sur cette ressource et dans quelles conditions. Vous devez [spécifier un principal](#) dans une politique basée sur les

ressources. Les principaux peuvent inclure des comptes, des utilisateurs, des rôles, des utilisateurs fédérés ou. Services AWS

Pour permettre un accès intercompte, vous pouvez spécifier un compte entier ou des entités IAM dans un autre compte en tant que principal dans une politique basée sur les ressources. L'ajout d'un principal entre comptes à une politique basée sur les ressources ne représente qu'une partie de l'instauration de la relation d'approbation. Lorsque le principal et la ressource sont différents Comptes AWS, un administrateur IAM du compte sécurisé doit également accorder à l'entité principale (utilisateur ou rôle) l'autorisation d'accéder à la ressource. Pour ce faire, il attache une politique basée sur une identité à l'entité. Toutefois, si une politique basée sur des ressources accorde l'accès à un principal dans le même compte, aucune autre politique basée sur l'identité n'est requise. Pour plus d'informations, consultez [la section Accès aux ressources entre comptes dans IAM](#) dans le guide de l'utilisateur d'IAM.

Actions de politique pour Amazon Transcribe

Prend en charge les actions de politique	Oui
--	-----

Les administrateurs peuvent utiliser les politiques AWS JSON pour spécifier qui a accès à quoi. C'est-à-dire, quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

L'élément `Action` d'une politique JSON décrit les actions que vous pouvez utiliser pour autoriser ou refuser l'accès à une politique. Les actions de stratégie portent généralement le même nom que l'opération AWS d'API associée. Il existe quelques exceptions, telles que les actions avec autorisations uniquement qui n'ont pas d'opération API correspondante. Certaines opérations nécessitent également plusieurs actions dans une politique. Ces actions supplémentaires sont nommées actions dépendantes.

Intégration d'actions dans une stratégie afin d'accorder l'autorisation d'exécuter les opérations associées.

Pour consulter la liste des Amazon Transcribe actions, reportez-vous à la section [Actions définies par Amazon Transcribe](#) dans la référence d'autorisation de service.

Les actions de politique en Amazon Transcribe cours utilisent le `transcribe` préfixe avant l'action. Pour indiquer plusieurs actions dans une seule déclaration, séparez-les par des virgules.

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

Vous pouvez aussi spécifier plusieurs actions à l'aide de caractères génériques (*). Par exemple, pour spécifier toutes les actions qui commencent par le mot `List`, incluez l'action suivante :

```
"Action": "transcribe:List*"
```

Pour consulter des exemples de politiques Amazon Transcribe basées sur l'identité, consultez [Exemples de politiques basées sur l'identité Amazon Transcribe](#)

Ressources de politique pour Amazon Transcribe

Prend en charge les ressources de politique	Oui
---	-----

Les administrateurs peuvent utiliser les politiques AWS JSON pour spécifier qui a accès à quoi. C'est-à-dire, quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

L'élément de politique JSON `Resource` indique le ou les objets auxquels l'action s'applique. Les instructions doivent inclure un élément `Resource` ou `NotResource`. Il est recommandé de définir une ressource à l'aide de son [Amazon Resource Name \(ARN\)](#). Vous pouvez le faire pour des actions qui prennent en charge un type de ressource spécifique, connu sous la dénomination autorisations de niveau ressource.

Pour les actions qui ne sont pas compatibles avec les autorisations de niveau ressource, telles que les opérations de liste, utilisez un caractère générique (*) afin d'indiquer que l'instruction s'applique à toutes les ressources.

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

Pour consulter la liste des types de Amazon Transcribe ressources et de leurs ARN, voir [Ressources définies par Amazon Transcribe](#) dans la référence d'autorisation de service. Pour savoir grâce à

quelles actions vous pouvez spécifier l'ARN de chaque ressource, consultez [Actions définies par Amazon Transcribe](#).

Pour consulter des exemples de politiques Amazon Transcribe basées sur l'identité, consultez [Exemples de politiques basées sur l'identité Amazon Transcribe](#)

Clés de conditions de politique pour Amazon Transcribe

Prend en charge les clés de condition de politique spécifiques au service	Oui
---	-----

Les administrateurs peuvent utiliser les politiques AWS JSON pour spécifier qui a accès à quoi. C'est-à-dire, quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

L'élément `Condition` (ou le bloc `Condition`) vous permet de spécifier des conditions lorsqu'une instruction est appliquée. L'élément `Condition` est facultatif. Vous pouvez créer des expressions conditionnelles qui utilisent des [opérateurs de condition](#), tels que les signes égal ou inférieur à, pour faire correspondre la condition de la politique aux valeurs de la demande.

Si vous spécifiez plusieurs éléments `Condition` dans une instruction, ou plusieurs clés dans un seul élément `Condition`, AWS les évalue à l'aide d'une opération AND logique. Si vous spécifiez plusieurs valeurs pour une seule clé de condition, AWS évalue la condition à l'aide d'une OR opération logique. Toutes les conditions doivent être remplies avant que les autorisations associées à l'instruction ne soient accordées.

Vous pouvez aussi utiliser des variables d'espace réservé quand vous spécifiez des conditions. Par exemple, vous pouvez accorder à un utilisateur IAM l'autorisation d'accéder à une ressource uniquement si elle est balisée avec son nom d'utilisateur IAM. Pour plus d'informations, consultez [Éléments d'une politique IAM : variables et identifications](#) dans le Guide de l'utilisateur IAM.

AWS prend en charge les clés de condition globales et les clés de condition spécifiques au service. Pour voir toutes les clés de condition AWS globales, voir les clés de [contexte de condition AWS globales](#) dans le guide de l'utilisateur IAM.

Pour consulter la liste des clés de Amazon Transcribe condition, reportez-vous à la section [Clés de condition pour Amazon Transcribe](#) la référence d'autorisation de service. Pour savoir avec quelles

actions et ressources vous pouvez utiliser une clé de condition, consultez la section [Actions définies par Amazon Transcribe](#).

Pour consulter des exemples de politiques Amazon Transcribe basées sur l'identité, consultez. [Exemples de politiques basées sur l'identité Amazon Transcribe](#)

ACL dans Amazon Transcribe

Prend en charge les listes ACL	Non
--------------------------------	-----

Les listes de contrôle d'accès (ACL) vérifient quels principaux (membres de compte, utilisateurs ou rôles) ont l'autorisation d'accéder à une ressource. Les listes de contrôle d'accès sont similaires aux politiques basées sur les ressources, bien qu'elles n'utilisent pas le format de document de politique JSON.

ABAC avec Amazon Transcribe

Prise en charge d'ABAC (identifications dans les politiques)	Partielle
--	-----------

Le contrôle d'accès basé sur les attributs (ABAC) est une politique d'autorisation qui définit des autorisations en fonction des attributs. Dans AWS, ces attributs sont appelés balises. Vous pouvez associer des balises aux entités IAM (utilisateurs ou rôles) et à de nombreuses AWS ressources. L'étiquetage des entités et des ressources est la première étape d'ABAC. Vous concevez ensuite des politiques ABAC pour autoriser des opérations quand l'identification du principal correspond à celle de la ressource à laquelle il tente d'accéder.

L'ABAC est utile dans les environnements qui connaissent une croissance rapide et pour les cas où la gestion des politiques devient fastidieuse.

Pour contrôler l'accès basé sur des étiquettes, vous devez fournir les informations d'étiquette dans [l'élément de condition](#) d'une politique utilisant les clés de condition `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` ou `aws:TagKeys`.

Si un service prend en charge les trois clés de condition pour tous les types de ressources, alors la valeur pour ce service est Oui. Si un service prend en charge les trois clés de condition pour certains types de ressources uniquement, la valeur est Partielle.

Pour plus d'informations sur l'ABAC, consultez [Qu'est-ce que le contrôle d'accès basé sur les attributs \(ABAC\) ?](#) dans le Guide de l'utilisateur IAM. Pour accéder à un didacticiel décrivant les étapes de configuration de l'ABAC, consultez [Utilisation du contrôle d'accès par attributs \(ABAC\)](#) dans le Guide de l'utilisateur IAM.

Pour plus d'informations sur le balisage Amazon Transcribe des ressources, consultez [Étiquetage des ressources](#). Pour plus d'informations sur le contrôle d'accès basé sur des balises, consultez la section [Contrôle de l'accès aux ressources AWS à l'aide de balises](#).

Utilisation des informations d'identification temporaires avec Amazon Transcribe

Prend en charge les informations d'identification temporaires	Oui
---	-----

Certains Services AWS ne fonctionnent pas lorsque vous vous connectez à l'aide d'informations d'identification temporaires. Pour plus d'informations, y compris celles qui Services AWS fonctionnent avec des informations d'identification temporaires, consultez Services AWS la section relative à l'utilisation [d'IAM](#) dans le guide de l'utilisateur d'IAM.

Vous utilisez des informations d'identification temporaires si vous vous connectez à l' AWS Management Console aide d'une méthode autre qu'un nom d'utilisateur et un mot de passe. Par exemple, lorsque vous accédez à AWS l'aide du lien d'authentification unique (SSO) de votre entreprise, ce processus crée automatiquement des informations d'identification temporaires. Vous créez également automatiquement des informations d'identification temporaires lorsque vous vous connectez à la console en tant qu'utilisateur, puis changez de rôle. Pour plus d'informations sur le changement de rôle, consultez [Changement de rôle \(console\)](#) dans le Guide de l'utilisateur IAM.

Vous pouvez créer manuellement des informations d'identification temporaires à l'aide de l' AWS API AWS CLI or. Vous pouvez ensuite utiliser ces informations d'identification temporaires pour y accéder AWS. AWS recommande de générer dynamiquement des informations d'identification temporaires au lieu d'utiliser des clés d'accès à long terme. Pour plus d'informations, consultez [Informations d'identification de sécurité temporaires dans IAM](#).

Autorisations de principal interservices pour Amazon Transcribe

Prend en charge les sessions d'accès direct (FAS)	Oui
---	-----

Lorsque vous utilisez un utilisateur ou un rôle IAM pour effectuer des actions AWS, vous êtes considéré comme un mandant. Lorsque vous utilisez certains services, vous pouvez effectuer une action qui initie une autre action dans un autre service. FAS utilise les autorisations du principal appelant et Service AWS, associées Service AWS à la demande, pour adresser des demandes aux services en aval. Les demandes FAS ne sont effectuées que lorsqu'un service reçoit une demande qui nécessite des interactions avec d'autres personnes Services AWS ou des ressources pour être traitée. Dans ce cas, vous devez disposer d'autorisations nécessaires pour effectuer les deux actions. Pour plus de détails sur une politique lors de la formulation de demandes FAS, consultez [Transmission des sessions d'accès](#).

Rôles de service pour Amazon Transcribe

Prend en charge les fonctions du service	Oui
--	-----

Une fonction de service est un [rôle IAM](#) qu'un service endosse pour accomplir des actions en votre nom. Un administrateur IAM peut créer, modifier et supprimer une fonction du service à partir d'IAM. Pour plus d'informations, consultez [Création d'un rôle pour la délégation d'autorisations à un Service AWS](#) dans le Guide de l'utilisateur IAM.

Warning

La modification des autorisations associées à un rôle de service peut perturber Amazon Transcribe les fonctionnalités. Modifiez les rôles de service uniquement lorsque Amazon Transcribe vous recevez des instructions à cet effet.

Rôles liés à un service pour Amazon Transcribe

Prend en charge les rôles liés à un service	Non
---	-----

Un rôle lié à un service est un type de rôle de service lié à un. Service AWS Le service peut endosser le rôle afin d'effectuer une action en votre nom. Les rôles liés à un service apparaissent dans votre Compte AWS répertoire et appartiennent au service. Un administrateur IAM peut consulter, mais ne peut pas modifier, les autorisations concernant les rôles liés à un service.

Amazon Transcribe ne prend pas en charge les rôles liés à un service.

Pour en savoir plus sur la création ou la gestion de rôles liés à un service pour d'autres services, consultez la section [AWS Services compatibles avec](#). IAM Recherchez un service dans le tableau qui inclut un Yes dans la colonne Rôle lié à un service. Choisissez le lien Oui pour consulter la documentation du rôle lié à ce service.

Prévention du problème de l'adjoint confus entre services

Un adjoint confus est une entité (un service ou un compte) qui est contrainte par une autre entité à effectuer une action. Ce type d'usurpation d'identité peut se produire entre comptes et services.

Pour éviter toute confusion, AWS fournit des outils qui vous aident à protéger vos données pour tous les services en utilisant des principaux de service qui ont eu accès aux ressources de votre Compte AWS. Cette section se concentre sur la prévention de la confusion entre les services des adjoints, en particulier à Amazon Transcribe ; vous pouvez toutefois en savoir plus sur ce sujet dans la section du Guide de l'IAM utilisateur consacrée aux [problèmes liés à la confusion des adjoints](#).

Pour limiter les autorisations d'IAM accès Amazon Transcribe à vos ressources, nous vous recommandons d'utiliser les clés de contexte des conditions globales [aws:SourceArn](#) et [aws:SourceAccount](#) dans vos politiques de ressources.

Si vous utilisez ces deux clés de contexte de condition globale et que la `aws:SourceArn` valeur contient l'ID du Compte AWS, la `aws:SourceAccount` valeur et le Compte AWS in `aws:SourceArn` doivent utiliser le même Compte AWS ID lorsqu'ils sont utilisés dans la même déclaration de politique.

Utilisez `aws:SourceArn` si vous souhaitez qu'une seule ressource soit associée à l'accès entre services. Si vous souhaitez y associer une ressource à un Compte AWS un accès interservice, utilisez `aws:SourceAccount`.

Note

Le moyen le plus efficace de se protéger contre le problème de député confus consiste à utiliser la clé de contexte de condition `aws:SourceArn` globale avec l'ARN complet de la ressource. Si vous ne connaissez pas l'ARN complet ou si vous spécifiez plusieurs ressources, utilisez la clé de contexte de condition `aws:SourceArn` globale avec des caractères génériques (*) pour les parties inconnues de l'ARN. Par exemple, `arn:aws:transcribe::123456789012:*`.

Pour un exemple de politique d'attribution de rôles qui montre comment vous pouvez éviter un problème de confusion entre adjoints, voir [Politique du député du député confus](#).

Exemples de politiques basées sur l'identité Amazon Transcribe

Par défaut, les utilisateurs et les rôles ne sont pas autorisés à créer ou modifier les ressources Amazon Transcribe. Ils ne peuvent pas non plus exécuter des tâches à l'aide de la AWS Management Console, de l'AWS Command Line Interface (AWS CLI) ou de l'API AWS. Pour octroyer aux utilisateurs des autorisations d'effectuer des actions sur les ressources dont ils ont besoin, un administrateur IAM doit créer des politiques IAM. L'administrateur peut ensuite ajouter les politiques IAM aux rôles et les utilisateurs peuvent assumer les rôles.

Pour apprendre à créer une politique basée sur l'identité IAM à l'aide de ces exemples de documents de politique JSON, consultez [Création de politiques dans l'onglet JSON](#) dans le Guide de l'utilisateur IAM.

Pour plus de détails sur les actions et les types de ressources définis par Amazon Transcribe, notamment le format des ARN pour chacun des types de ressources, consultez [Actions, ressources et clés de condition pour Amazon Transcribe](#) dans la Référence de l'autorisation de service.

Rubriques

- [Bonnes pratiques en matière de politiques](#)
- [Utilisation du AWS Management Console](#)
- [Autorisations requises pour les IAM rôles](#)
- [Autorisations requises pour les clés Amazon S3 de chiffrement](#)
- [Autorisation accordée aux utilisateurs pour afficher leurs propres autorisations](#)
- [AWS KMS stratégie de chiffrement du contexte](#)
- [Politique du député du député confus](#)
- [Affichage des tâches de transcription basées sur les balises](#)

Bonnes pratiques en matière de politiques

Les stratégies basées sur l'identité déterminent si une personne peut créer, consulter ou supprimer des ressources Amazon Transcribe dans votre compte. Ces actions peuvent entraîner des frais pour votre Compte AWS. Lorsque vous créez ou modifiez des politiques basées sur l'identité, suivez ces instructions et recommandations :

- Démarrer avec AWS gérées et évoluez vers les autorisations de moindre privilège - Pour commencer à accorder des autorisations à vos utilisateurs et charges de travail, utilisez les politiques gérées AWS qui accordent des autorisations dans de nombreux cas d'utilisation courants. Ils sont disponibles dans votre Compte AWS. Nous vous recommandons de réduire encore les autorisations en définissant des Politiques gérées par le client AWS qui sont spécifiques à vos cas d'utilisation. Pour de plus amples informations, consultez [Politiques gérées AWS](#) ou [Politiques gérées AWS pour les activités professionnelles](#) dans le Guide de l'utilisateur IAM.
- Accorder les autorisations de moindre privilège - Lorsque vous définissez des autorisations avec des politiques IAM, accordez uniquement les autorisations nécessaires à l'exécution d'une seule tâche. Pour ce faire, vous définissez les actions qui peuvent être entreprises sur des ressources spécifiques dans des conditions spécifiques, également appelées autorisations de moindre privilège. Pour plus d'informations sur l'utilisation de IAM pour appliquer des autorisations, consultez [Politiques et autorisations dans IAM](#) dans le Guide de l'utilisateur IAM.
- Utiliser des conditions dans les politiques IAM pour restreindre davantage l'accès - Vous pouvez ajouter une condition à vos politiques afin de limiter l'accès aux actions et aux ressources. Par exemple, vous pouvez écrire une condition de politique pour spécifier que toutes les demandes doivent être envoyées via SSL. Vous pouvez également utiliser des conditions pour accorder l'accès aux actions de service si elles sont utilisées via un Service AWS spécifique, comme AWS CloudFormation. Pour plus d'informations, consultez [Conditions pour éléments de politique JSON IAM](#) dans le Guide de l'utilisateur IAM.
- Utilisez IAM Access Analyzer pour valider vos politiques IAM afin de garantir des autorisations sécurisées et fonctionnelles - IAM Access Analyzer valide les politiques nouvelles et existantes de manière à ce que les politiques IAM respectent le langage de politique IAM (JSON) et les bonnes pratiques IAM. IAM Access Analyzer fournit plus de 100 vérifications de politiques et des recommandations exploitables pour vous aider à créer des politiques sécurisées et fonctionnelles. Pour de plus amples informations, consultez [Validation de politique IAM Access Analyzer](#) dans le Guide de l'utilisateur IAM.
- Authentification multifactorielle (MFA) nécessaire : si vous avez un scénario qui nécessite des utilisateurs IAM ou un utilisateur root dans votre Compte AWS, activez l'authentification multifactorielle pour une sécurité renforcée. Pour exiger le MFA lorsque des opérations d'API sont appelées, ajoutez des conditions MFA à vos politiques. Pour de plus amples informations, consultez [Configuration de l'accès aux API protégé par MFA](#) dans le Guide de l'utilisateur IAM.

Pour plus d'informations sur les bonnes pratiques dans IAM, consultez [Bonnes pratiques de sécurité dans IAM](#) dans le Guide de l'utilisateur IAM.

Utilisation du AWS Management Console

Pour accéder à la console Amazon Transcribe, vous devez disposer d'un ensemble minimum d'autorisations. Ces autorisations doivent vous permettre de répertorier et de consulter les informations relatives aux Amazon Transcribe ressources de votre Compte AWS. Si vous créez une stratégie basée sur l'identité qui est plus restrictive que l'ensemble minimum d'autorisations requis, la console ne fonctionnera pas comme prévu pour les entités (utilisateurs ou rôles) tributaires de cette stratégie.

Vous n'avez pas besoin d'accorder les autorisations minimales de console pour les utilisateurs qui effectuent des appels uniquement à AWS CLI ou à l'API AWS. Autorisez plutôt l'accès à uniquement aux actions qui correspondent à l'opération d'API qu'ils tentent d'effectuer.

Pour vous assurer qu'une entité (utilisateurs et rôles) peut utiliser le [AWS Management Console](#), associez-lui l'une des politiques AWS gérées suivantes.

- `AmazonTranscribeFullAccess`: accorde un accès complet à la création, à la lecture, à la mise à jour, à la suppression et à l'exécution de toutes les Amazon Transcribe ressources. Il permet également d'accéder aux Amazon S3 compartiments dont le nom figure `transcribe` dans le nom du compartiment.
- `AmazonTranscribeReadOnlyAccess`: accorde un accès en lecture seule aux Amazon Transcribe ressources afin que vous puissiez obtenir et répertorier les tâches de transcription et les vocabulaires personnalisés.

Note

Vous pouvez consulter les politiques d'autorisations gérées par en vous connectant à l'AWS Management Console et en les recherchant par nom de stratégie. Une recherche sur « transcribe » renvoie les deux politiques répertoriées ci-dessus (`AmazonTranscribeReadOnly` et `AmazonTranscribeFullAccess`).

Vous pouvez également créer vos propres stratégies IAM personnalisées afin d'accorder des autorisations pour les actions d'API Amazon Transcribe. Vous pouvez rattacher ces politiques personnalisées aux entités qui nécessitent ces autorisations.

Autorisations requises pour les IAM rôles

Si vous créez un IAM rôle à appeler Amazon Transcribe, il doit être autorisé à accéder au Amazon S3 compartiment. Le cas échéant, KMS key il doit également être utilisé pour crypter le contenu du compartiment. Reportez-vous aux sections suivantes pour des exemples de politiques.

Politiques de confiance

L'entité que vous utilisez pour effectuer votre demande de transcription doit disposer d'une politique de confiance lui permettant d'assumer ce rôle. Utilisez la politique de confiance suivante. Notez que si vous envoyez une demande d'analyse des appels en temps réel avec l'analyse post-appel activée, vous devez utiliser la « Politique de confiance pour l'analyse des appels en temps réel ».

Politique de confiance pour Amazon Transcribe

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "transcribe.amazonaws.com"
        ]
      },
      "Action": [
        "sts:AssumeRole"
      ],
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "111122223333"
        },
        "StringLike": {
          "aws:SourceArn": "arn:aws:transcribe:us-west-2:111122223333:*"
        }
      }
    }
  ]
}
```

Politique de confiance pour l'analyse des appels en temps réel

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "transcribe.streaming.amazonaws.com"
        ]
      },
      "Action": [
        "sts:AssumeRole"
      ],
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "111122223333"
        },
        "StringLike": {
          "aws:SourceArn": "arn:aws:transcribe:us-west-2:111122223333:*"
        }
      }
    }
  ]
}
```

Amazon S3politique relative aux compartiments d'entrée

La politique suivante donne à un IAM rôle l'autorisation d'accéder aux fichiers depuis le compartiment d'entrée spécifié.

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": [
      "s3:GetObject",
      "s3:ListBucket"
    ],
    "Resource": [
      "arn:aws:s3:::DOC-EXAMPLE-INPUT-BUCKET",
      "arn:aws:s3:::DOC-EXAMPLE-INPUT-BUCKET/*"
    ]
  }
}
```

```
}
```

Amazon S3 politique du compartiment de sortie

La politique suivante donne à un IAM rôle l'autorisation d'écrire des fichiers dans le compartiment de sortie spécifié.

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": [
      "s3:PutObject"
    ],
    "Resource": [
      "arn:aws:s3:::DOC-EXAMPLE-OUTPUT-BUCKET/*"
    ]
  }
}
```

Autorisations requises pour les clés Amazon S3 de chiffrement

Si vous utilisez un KMS key pour crypter un Amazon S3 compartiment, incluez les éléments suivants dans la KMS key politique. Ceci permet à Amazon Transcribe d'accéder au contenu du compartiment. Pour plus d'informations sur l'autorisation d'accès à KMS keys, consultez la section [Autoriser un Comptes AWS utilisateur externe à accéder à un KMS key](#) dans le Guide du AWS KMS développeur.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/ExampleRole"
      },
      "Action": [
        "kms:Decrypt"
      ],
      "Resource": "arn:aws:kms:us-west-2:111122223333:key/KMS-Example-KeyId"
    }
  ]
}
```

```
]
}
```

Autorisation accordée aux utilisateurs pour afficher leurs propres autorisations

Cet exemple montre comment créer une politique qui permet aux utilisateurs IAM d'afficher les politiques en ligne et gérées attachées à leur identité d'utilisateur. Cette politique inclut les autorisations nécessaires pour réaliser cette action sur la console ou par programmation à l'aide de l'AWS CLI ou de l'API AWS.

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

AWS KMS stratégie de chiffrement du contexte

La politique suivante accorde au IAM rôle «`ExampleRole`» l'autorisation d'utiliser les opérations AWS KMS Decrypt et Encrypt pour ce cas particulier KMS key. Cette politique ne fonctionne que pour les demandes comportant au moins une paire de contextes de chiffrement, dans ce cas «`color:indigoBlue`». Pour plus d'informations sur le contexte de AWS KMS chiffrement, consultez [Contexte de chiffrement AWS KMS](#).

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/ExampleRole"
      },
      "Action": [
        "kms:Decrypt",
        "kms:DescribeKey",
        "kms:Encrypt",
        "kms:GenerateDataKey*",
        "kms:ReEncrypt*"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "kms:EncryptionContext:color": "indigoBlue"
        }
      }
    }
  ]
}
```

Politique du député du député confus

Voici un exemple de politique d'attribution de rôles qui montre comment vous pouvez utiliser `aws:SourceArn` et `aws:SourceAccount` avec Amazon Transcribe pour éviter un problème de confusion entre adjoints. Pour plus d'informations sur la prévention de la confusion chez les adjoints, voir [Prévention du problème de l'adjoint confus entre services](#).

```
{
```

```

"Version": "2012-10-17",
"Statement": [
  {
    "Effect": "Allow",
    "Principal": {
      "Service": "transcribe.amazonaws.com"
    },
    "Action": [
      "sts:AssumeRole",
    ],
    "Condition": {
      "StringEquals": {
        "aws:SourceAccount": "111122223333"
      },
      "StringLike": {
        "aws:SourceArn": "arn:aws:transcribe:us-west-2:111122223333:*"
      }
    }
  }
]
}

```

Affichage des tâches de transcription basées sur les balises

Vous pouvez utiliser des conditions dans votre politique basée sur l'identité pour contrôler l'accès aux ressources Amazon Transcribe en fonction des balises. Cet exemple montre comment créer une politique qui autorise à consulter une tâche de transcription. Toutefois, l'autorisation est accordée uniquement si la balise de tâche de transcription `Owner` a la valeur du nom d'utilisateur de cet utilisateur. Cette politique accorde également les autorisations nécessaires pour réaliser cette action à l'aide de l'AWS Management Console.

Vous pouvez rattacher cette politique aux IAM entités de votre compte. Si un rôle nommé `test-role` tente de visualiser une tâche de transcription, la tâche de transcription doit être balisée `Owner=test-role` ou `owner=test-role` (les noms des clés conditionnelles ne distinguent pas les majuscules), sinon l'accès leur sera refusé. Pour de plus amples informations, veuillez consulter [Éléments de stratégie IAM JSON : condition](#) dans le Guide de l'utilisateur IAM.

Pour plus d'informations sur le balisage Amazon Transcribe, consultez [Étiquetage des ressources](#).

```

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

```



```
"Statement": [
  {
    "Sid": "ListTranscriptionJobsInConsole",
    "Effect": "Allow",
    "Action": "transcribe:ListTranscriptionJobs",
    "Resource": "*"
  },
  {
    "Sid": "ViewTranscriptionJobsIfOwner",
    "Effect": "Allow",
    "Action": "transcribe:GetTranscriptionJobs",
    "Resource": "arn:aws:transcribe:*:*:transcription-job/*",
    "Condition": {
      "StringEquals": {"aws:ResourceTag/Owner": "${aws:username}"}
    }
  }
]
```

Résolution des problèmes Amazon Transcribe d'identité et d'accès

Utilisez les informations suivantes pour diagnostiquer et résoudre les problèmes courants que vous pouvez rencontrer lors de l'utilisation de Amazon Transcribe and AWS Identity and Access Management (IAM).

Rubriques

- [Je ne suis pas autorisé à effectuer une action dans Amazon Transcribe](#)
- [Je ne suis pas autorisé à effectuer iam : PassRole](#)
- [Je souhaite permettre à des personnes extérieures Compte AWS à moi d'accéder à mes Amazon Transcribe ressources](#)

Je ne suis pas autorisé à effectuer une action dans Amazon Transcribe

Si vous recevez une erreur qui indique que vous n'êtes pas autorisé à effectuer une action, vos politiques doivent être mises à jour afin de vous permettre d'effectuer l'action.

L'exemple d'erreur suivant se produit quand l'utilisateur IAM `mateojackson` tente d'utiliser la console pour afficher des informations détaillées sur une ressource `my-example-widget` fictive, mais ne dispose pas des autorisations `transcribe:GetWidget` fictives.

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

Dans ce cas, la politique qui s'applique à l'utilisateur mateojackson doit être mise à jour pour autoriser l'accès à la ressource *my-example-widget* à l'aide de l'action `transcribe:GetWidget`.

Si vous avez besoin d'aide, contactez votre AWS administrateur. Votre administrateur vous a fourni vos informations d'identification de connexion.

Je ne suis pas autorisé à effectuer iam : PassRole

Si vous recevez une erreur selon laquelle vous n'êtes pas autorisé à exécuter `iam:PassRole` l'action, vos stratégies doivent être mises à jour afin de vous permettre de transmettre un rôle à Amazon Transcribe.

Certains vos Services AWS permettent de transmettre un rôle existant à ce service au lieu de créer un nouveau rôle de service ou un rôle lié à un service. Pour ce faire, un utilisateur doit disposer des autorisations nécessaires pour transmettre le rôle au service.

L'exemple d'erreur suivant se produit lorsqu'un utilisateur IAM nommé marymajor essaie d'utiliser la console pour exécuter une action dans Amazon Transcribe. Toutefois, l'action nécessite que le service ait des autorisations accordées par une fonction de service. Mary ne dispose pas des autorisations nécessaires pour transférer le rôle au service.

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

Dans ce cas, les politiques de Mary doivent être mises à jour pour lui permettre d'exécuter l'action `iam:PassRole`.

Si vous avez besoin d'aide, contactez votre AWS administrateur. Votre administrateur vous a fourni vos informations d'identification de connexion.

Je souhaite permettre à des personnes extérieures Compte AWS à moi d'accéder à mes Amazon Transcribe ressources

Vous pouvez créer un rôle que les utilisateurs provenant d'autres comptes ou les personnes extérieures à votre organisation pourront utiliser pour accéder à vos ressources. Vous pouvez

spécifier qui est autorisé à assumer le rôle. Pour les services qui prennent en charge les politiques basées sur les ressources ou les listes de contrôle d'accès (ACL), vous pouvez utiliser ces politiques pour donner l'accès à vos ressources.

Pour en savoir plus, consultez les éléments suivants :

- Pour savoir si ces fonctionnalités sont prises Amazon Transcribe en charge, consultez [Comment Amazon Transcribe fonctionne avec IAM](#).
- Pour savoir comment fournir l'accès à vos ressources sur celles Comptes AWS que vous possédez, consultez la section [Fournir l'accès à un utilisateur IAM dans un autre utilisateur Compte AWS que vous possédez](#) dans le Guide de l'utilisateur IAM.
- Pour savoir comment fournir l'accès à vos ressources à des tiers Comptes AWS, consultez la section [Fournir un accès à des ressources Comptes AWS détenues par des tiers](#) dans le guide de l'utilisateur IAM.
- Pour savoir comment fournir un accès par le biais de la fédération d'identité, consultez [Fournir un accès à des utilisateurs authentifiés en externe \(fédération d'identité\)](#) dans le Guide de l'utilisateur IAM.
- Pour connaître la différence entre l'utilisation de rôles et de politiques basées sur les ressources pour l'accès entre comptes, consultez la section Accès aux [ressources entre comptes dans IAM dans le guide de l'utilisateur d'IAM](#).

Protection des données dans Amazon Transcribe

Le [modèle de responsabilité partagée](#) AWS s'applique à la protection des données dans Amazon Transcribe. Comme décrit dans ce modèle, AWS est responsable de la protection de l'infrastructure globale sur laquelle l'ensemble du AWS Cloud s'exécute. La gestion du contrôle de votre contenu hébergé sur cette infrastructure relève de votre responsabilité. Vous êtes également responsable des tâches de configuration et de gestion de la sécurité des Services AWS que vous utilisez. Pour en savoir plus sur la confidentialité des données, consultez [Questions fréquentes \(FAQ\) sur la confidentialité des données](#). Pour en savoir plus sur la protection des données en Europe, consultez le billet de blog Modèle de responsabilité partagée [AWS et RGPD \(Règlement général sur la protection des données\)](#) sur le AWSBlog de sécurité.

À des fins de protection des données, nous vous recommandons de protéger les informations d'identification Compte AWS et de configurer les comptes utilisateur individuels avec AWS IAM Identity Center ou AWS Identity and Access Management (IAM). Ainsi, chaque utilisateur se

voit attribuer uniquement les autorisations nécessaires pour exécuter ses tâches. Nous vous recommandons également de sécuriser vos données comme indiqué ci-dessous :

- Utilisez l'authentification multifactorielle (MFA) avec chaque compte.
- Utilisez les certificats SSL/TLS pour communiquer avec les ressources AWS. Nous exigeons TLS 1.2 et recommandons TLS 1.3.
- Configurez une API (Interface de programmation) et le journal de l'activité des utilisateurs avec AWS CloudTrail.
- Utilisez des solutions de chiffrement AWS, ainsi que tous les contrôles de sécurité par défaut au sein des Services AWS.
- Utilisez des services de sécurité gérés avancés tels qu'Amazon Macie, qui contribuent à la découverte et à la sécurisation des données sensibles stockées dans Amazon S3.
- Si vous avez besoin de modules cryptographiques validés FIPS (Federal Information Processing Standard) 140-2 lorsque vous accédez à AWS via une CLI (Interface de ligne de commande) ou une API (Interface de programmation), utilisez un point de terminaison FIPS (Federal Information Processing Standard). Pour en savoir plus sur les points de terminaison FIPS (Federal Information Processing Standard) disponibles, consultez [Federal Information Processing Standard \(FIPS\) 140-2](#) (Normes de traitement de l'information fédérale).

Nous vous recommandons fortement de ne jamais placer d'informations confidentielles ou sensibles, telles que les adresses e-mail de vos clients, dans des balises ou des champs de texte libre tels que le champ Name (Nom). Cela est également valable lorsque vous utilisez Amazon Transcribe ou d'autres Services AWS à l'aide de la console, de l'API, d'AWS CLI ou des kits SDK AWS. Toutes les données que vous saisissez dans des balises ou des champs de texte de forme libre utilisés pour les noms peuvent être utilisées à des fins de facturation ou dans les journaux de diagnostic. Si vous fournissez une adresse URL à un serveur externe, nous vous recommandons fortement de ne pas inclure d'informations d'identification dans l'adresse URL permettant de valider votre demande adressée à ce serveur.

Confidentialité du trafic inter-réseaux

Un point de terminaison Amazon Virtual Private Cloud (Amazon VPC) pour Amazon Transcribe est une entité logique au sein d'un VPC qui autorise uniquement la connectivité à Amazon Transcribe. Amazon VPC achemine les demandes vers Amazon Transcribe et les réponses vers le VPC. Pour plus d'informations, consultez [Concepts AWS PrivateLink](#). Pour plus d'informations sur l'utilisation

des points de terminaison Amazon VPC, avec Amazon Transcribe, consultez [Amazon Transcribe et interface des points de terminaison VPC \(AWS PrivateLink\)](#).

Chiffrement des données

Le chiffrement des données fait référence à la protection des données en transit et au repos. Vous pouvez protéger vos données à l'aide d'Amazon S3 de clés gérées ou de KMS keys au repos, en complément du protocole TLS (Transport Layer Security) standard pendant leur transfert.

Chiffrement au repos

Amazon Transcribe utilise la clé par défaut (SSE-S3) pour le chiffrement côté serveur des transcriptions placées dans votre Amazon S3 compartiment.

Lorsque vous utilisez cette [StartTranscriptionJob](#) opération, vous pouvez spécifier la vôtre KMS key pour crypter la sortie d'une tâche de transcription.

Amazon Transcribe utilise un volume chiffré Amazon EBS avec la clé par défaut.

Chiffrement en transit

Amazon Transcribe utilise TLS 1.2 avec des certificats AWS pour chiffrer les données en transit. Cela inclut les transcriptions en streaming.

Gestion des clés

Amazon Transcribe fonctionne avec KMS keys pour fournir un cryptage amélioré de vos données. Avec Amazon S3, vous pouvez crypter votre support d'entrée lors de la création d'une tâche de transcription. L'intégration avec AWS KMS permet le cryptage de la sortie d'une [StartTranscriptionJob](#) demande.

Si vous ne spécifiez pas de KMS key, la sortie de la tâche de transcription est cryptée avec la clé par défaut (SSE-S3).

Pour plus d'informations AWS KMS, consultez le [Guide du AWS Key Management Service développeur](#).

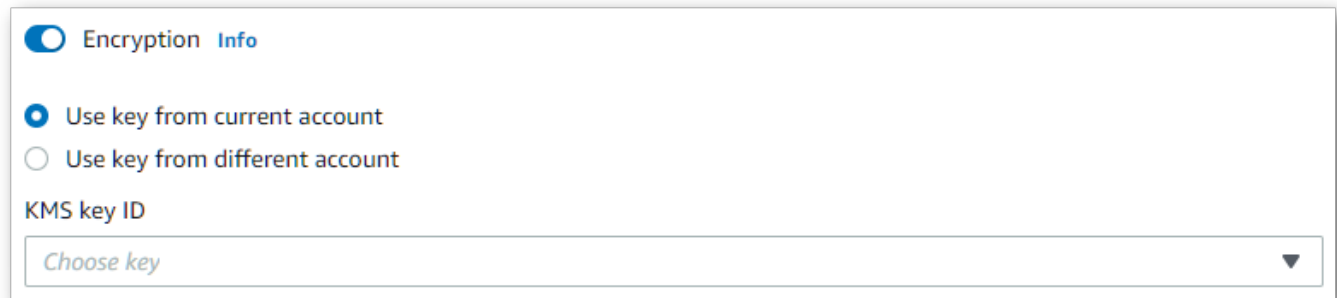
Gestion des clés à l'aide de l'AWS Management Console

Pour crypter la sortie de votre travail de transcription, vous pouvez choisir d'utiliser un KMS key pour le Compte AWS celui qui fait la demande ou un KMS key autre Compte AWS.

Si vous ne spécifiez pas de KMS key, la sortie de la tâche de transcription est cryptée avec la Amazon S3 clé par défaut (SSE-S3).

Pour activer le chiffrement de sortie :

1. Sous Output data (Données de sortie) choisissez Encryption (Chiffrement).



The screenshot shows the 'Encryption' settings in the Amazon Transcribe console. At the top, there is a blue radio button labeled 'Encryption Info'. Below it, there are two radio button options: 'Use key from current account' (which is selected) and 'Use key from different account'. Underneath these options is a text input field labeled 'KMS key ID' with a dropdown arrow on the right. The dropdown menu is open, showing the text 'Choose key'.

2. Choisissez si le KMS key provient de celui du Compte AWS que vous utilisez actuellement ou d'un autre Compte AWS. Si vous souhaitez utiliser une clé de la version actuelle du Compte AWS, choisissez la clé dans le KMS key ID. Si vous utilisez une clé provenant d'un autre Compte AWS, vous devez saisir l'ARN de la clé. Pour utiliser une clé provenant d'un autre Compte AWS, l'appelant doit disposer de permissions pour le KMS key. Reportez-vous à [la section Création d'une politique clé](#) pour plus d'informations.

Gestion des clés à l'aide de l'API

Pour utiliser le chiffrement de sortie avec l'API, vous devez le spécifier KMS key à l'aide du `OutputEncryptionKMSKeyId` paramètre de l'[StartTranscriptionJob](#) opération [StartCallAnalyticsJob](#) [StartMedicalTranscriptionJob](#), ou.

Si vous utilisez une clé située dans le courant du Compte AWS, vous pouvez spécifier votre KMS key clé de quatre manières :

1. Utilisez le KMS key identifiant lui-même. Par exemple, `1234abcd-12ab-34cd-56ef-1234567890ab`.
2. Utilisez un alias pour le KMS key ID. Par exemple, `alias/ExampleAlias`.
3. Utilisez l'Amazon Resource Name (ARN) pour le KMS key ID. Par exemple, `arn:aws:kms:region:account-ID:key/1234abcd-12ab-34cd-56ef-1234567890ab`.
4. Utilisez l'ARN pour le KMS key alias. Par exemple, `arn:aws:kms:region:account-ID:alias/ExampleAlias`.

Si vous utilisez une clé située dans une autre Compte AWS position que la clé actuelle Compte AWS, vous pouvez spécifier KMS key la vôtre de deux manières :

1. Utilisez l'ARN pour l'KMS keyID. Par exemple, `arn:aws:kms:region:account-ID:key/1234abcd-12ab-34cd-56ef-1234567890ab`.
2. Utilisez l'ARN pour l'KMS keyalias. Par exemple, `arn:aws:kms:region:account-ID:alias/ExampleAlias`.

Notez que l'entité qui fait la demande doit être autorisée à utiliser la spécifiée KMS key.

Contexte de chiffrement AWS KMS

AWS KMS Le contexte de chiffrement est une carte de paires clé:valeur non secrètes en texte brut. Cette carte représente des données authentifiées supplémentaires, appelées paires de contextes de chiffrement, qui fournissent un niveau de sécurité supplémentaire à vos données. Amazon Transcribe nécessite une clé de chiffrement symétrique pour crypter la sortie de transcription dans un Amazon S3 compartiment spécifié par le client. Pour en savoir plus, consultez [Clés asymétriques dans AWS KMS](#).

Lorsque vous créez vos paires de contextes de chiffrement, n'incluez pas d'informations sensibles. Le contexte de chiffrement n'est pas secret : il est visible en texte brut dans vos CloudTrail journaux (vous pouvez donc l'utiliser pour identifier et classer vos opérations cryptographiques).

Votre paire de contextes de chiffrement peut inclure des caractères spéciaux, tels que des traits de soulignement (), des tirets (-), des barres obliques (/,\) et des deux-points (:).

Tip

Il peut être utile de relier les valeurs de votre paire de contextes de chiffrement aux données cryptées. Bien que cela ne soit pas obligatoire, nous vous recommandons d'utiliser des métadonnées non sensibles liées à votre contenu crypté, telles que les noms de fichiers, les valeurs d'en-tête ou les champs de base de données non cryptés.

Pour utiliser le chiffrement de sortie avec l'API, définissez le `KMSEncryptionContext` paramètre dans l'[StartTranscriptionJob](#) opération. Afin de fournir un contexte de chiffrement pour l'opération de chiffrement en sortie, le `OutputEncryptionKMSKeyId` paramètre doit faire référence à un KMS key ID symétrique.

Vous pouvez utiliser [des clés AWS KMS conditionnelles IAM](#) associées à des politiques pour contrôler l'accès à un chiffrement symétrique en KMS key fonction du contexte de chiffrement utilisé dans la demande d'[opération cryptographique](#). Pour un exemple de politique contextuelle de chiffrement, reportez-vous à la section [AWS KMS stratégie de chiffrement du contexte](#).

L'utilisation d'un contexte de chiffrement est facultative, mais recommandée. Pour de plus amples informations, consultez [Contexte de chiffrement](#).

Refuser d'utiliser vos données pour améliorer le service

Par défaut, il Amazon Transcribe stocke et utilise les entrées vocales qu'il a traitées pour développer le service et améliorer continuellement votre expérience. Vous pouvez refuser que votre contenu soit utilisé à des fins de développement et d'amélioration Amazon Transcribe en utilisant une politique de [AWS Organizations désinscription](#). Pour plus d'informations sur la [désactivation des services IA](#).

Surveillance Amazon Transcribe

La surveillance joue un rôle important dans le maintien de la fiabilité, de la disponibilité Amazon Transcribe et des performances de vos autres AWS solutions. AWS fournit les outils de surveillance suivants pour surveiller Amazon Transcribe, signaler tout problème et prendre des mesures automatiques le cas échéant :

- Amazon CloudWatch surveille vos AWS ressources et les applications que vous exécutez AWS en temps réel. Vous pouvez collecter et suivre les métriques, créer des tableaux de bord personnalisés, et définir des alarmes qui vous informent ou prennent des mesures lorsqu'une métrique spécifique atteint un seuil que vous spécifiez. Par exemple, vous pouvez CloudWatch suivre l'utilisation du processeur ou d'autres indicateurs sur vos Amazon EC2 instances et lancer automatiquement de nouvelles instances en cas de besoin.
- Amazon CloudWatch Logs peut surveiller, stocker et accéder à vos fichiers journaux à partir d'Amazon EC2 instances et d'autres sources. CloudTrail CloudWatch Logs peut surveiller les informations contenues dans les fichiers journaux et vous avertir lorsque certains seuils sont atteints. Vous pouvez également archiver vos données de journaux dans une solution de stockage hautement durable.
- AWS CloudTrail capture les appels d'API et les événements associés effectués par vous ou en votre nom Compte AWS et envoie les fichiers journaux dans un Amazon S3 compartiment que vous spécifiez. Vous pouvez identifier les utilisateurs et les comptes appelés AWS, l'adresse IP source à partir de laquelle les appels ont été effectués et la date des appels.

Pour plus d'informations, consultez le Guide de l'utilisateur [Amazon CloudWatch](#).

Amazon EventBridge est un service sans serveur qui utilise des événements pour connecter les composants de l'application entre eux, ce qui vous permet de créer plus facilement des applications évolutives pilotées par des événements. EventBridge fournit un flux de données en temps réel à partir de vos propres applications, applications SaaS (Software as a Service) et AWS services et achemine ces données vers des cibles telles que Lambda. Vous pouvez surveiller les événements qui se produisent dans les services et créer des architectures basées sur les événements. Pour de plus amples informations, veuillez consulter le Guide de l'utilisateur [Amazon EventBridge](#).

Rubriques

- [Surveillance Amazon Transcribe avec Amazon CloudWatch](#)
- [Surveillance Amazon Transcribe avec AWS CloudTrail](#)
- [Utilisation Amazon EventBridge avec Amazon Transcribe](#)

Surveillance Amazon Transcribe avec Amazon CloudWatch

Vous pouvez surveiller Amazon Transcribe l'utilisation CloudWatch, qui collecte les données brutes et les transforme en indicateurs lisibles en temps quasi réel. Ces statistiques sont enregistrées pour une durée de 15 mois ; par conséquent, vous pouvez accéder aux informations historiques et acquérir un meilleur point de vue de la façon dont votre service ou application web s'exécute. Vous pouvez également définir des alarmes qui surveillent certains seuils et envoient des notifications ou prennent des mesures lorsque ces seuils sont atteints. Pour de plus amples informations, veuillez consulter le [Guide de l'utilisateur CloudWatch](#) .

Utilisation de Amazon CloudWatch métriques et de dimensions avec Amazon Transcribe

Amazon Transcribe prend en charge CloudWatch les métriques et les dimensions, qui sont des données qui peuvent vous aider à surveiller les performances. Les catégories de métriques prises en charge incluent le trafic, les erreurs, le transfert de données et la latence associés à vos tâches de transcription. Les métriques prises en charge se trouvent CloudWatch dans l'espace de noms AWS/Transcribe.

Note

CloudWatch les indicateurs de surveillance sont gratuits et ne sont pas pris en compte dans les quotas CloudWatch de service.

Pour plus d'informations sur les CloudWatch métriques, consultez la section [Utilisation Amazon CloudWatch des métriques](#).

Surveillance Amazon Transcribe avec AWS CloudTrail

Amazon Transcribe est intégré à AWS CloudTrail un service qui fournit un enregistrement des actions entreprises Amazon Transcribe par un utilisateur ou un rôle AWS Identity and Access Management (IAM), ou par un AWS service. CloudTrail capture tous les appels d'API pour Amazon Transcribe. Cela inclut les appels depuis les appels de code AWS Management Console et vers les Amazon Transcribe API, sous forme d'événements. En créant un suivi, vous pouvez permettre la diffusion continue d' CloudTrail événements, y compris des événements pour Amazon Transcribe, Amazon S3 dans un compartiment. Si vous ne créez pas de journal de suivi, vous pouvez toujours afficher les événements les plus récents dans la AWS Management Console CloudTrail sous Historique des événements. À l'aide des informations collectées par CloudTrail, vous pouvez voir chaque demande qui est faite Amazon Transcribe, l'adresse IP à partir de laquelle la demande a été faite, qui a fait la demande, quand elle a été faite et des détails supplémentaires.

Pour en savoir plus CloudTrail, consultez le [guide de l'AWS CloudTrail utilisateur](#).

Amazon Transcribe et CloudTrail

CloudTrail est activé sur votre compte Compte AWS lorsque vous créez le compte. Lorsqu'une activité se produit dans Amazon Transcribe, cette activité est enregistrée dans un CloudTrail événement avec d'autres Service AWS événements dans l'historique des CloudTrail événements. Vous pouvez consulter, rechercher et télécharger les événements récents dans votre Compte AWS. Pour plus d'informations, consultez [Affichage des événements avec l'historique des événements CloudTrail](#).

Pour obtenir un enregistrement continu des événements de votre compte Compte AWS, y compris des événements pour Amazon Transcribe, créez un parcours. Un journal est une configuration qui permet de CloudTrail transmettre des événements sous forme de fichiers journaux à un Amazon S3 compartiment spécifique. CloudTrail les fichiers journaux contiennent une ou plusieurs entrées de

journal. Un événement représente une demande individuelle d'une source quelconque. Il inclut des informations sur l'action demandée, la date et l'heure de l'action, les paramètres de la demande, etc. CloudTrail les fichiers journaux ne constituent pas une trace ordonnée des appels d'API publics, ils n'apparaissent donc pas dans un ordre spécifique.

Par défaut, lorsque vous créez un parcours dans le AWS Management Console, celui-ci s'applique à tous Régions AWS. Le journal enregistre les événements provenant de tous Régions AWS les éléments de la AWS partition et envoie les fichiers journaux dans le Amazon S3 compartiment que vous spécifiez. En outre, vous pouvez en configurer d'autres Services AWS pour analyser plus en détail les données d'événements collectées dans les CloudTrail journaux et agir en conséquence. section within

Pour plus d'informations, consultez :

- [Présentation de la création d'un journal d'activité](#)
- [CloudTrail Services et intégrations pris en charge](#)
- [Configuration Amazon SNS des notifications pour CloudTrail](#)
- [Réception de fichiers journaux CloudTrail de plusieurs régions](#) et [Réception de fichiers journaux CloudTrail de plusieurs comptes](#)

CloudTrail enregistre toutes les Amazon Transcribe actions, qui sont documentées dans la [référence de l'API](#). Par exemple, les [StartTranscriptionJob](#) opérations [CreateVocabularyGetTranscriptionJob](#), et génèrent des entrées dans les fichiers CloudTrail journaux.

Chaque événement ou entrée de journal contient des informations sur la personne ayant initié la demande. Cette information permet de déterminer les éléments suivants :

- Si la demande est faite avec les informations IAM d'identification root ou utilisateur
- Si la demande est effectuée avec des informations d'identification de sécurité temporaires pour un rôle ou un utilisateur fédéré IAM
- Si la demande est faite par un autre Service AWS

Pour plus d'informations, consultez la section [Élément userIdentity CloudTrail](#).

Vous pouvez également agréger des fichiers Amazon Transcribe journaux multiples Régions AWS ou multiples Comptes AWS dans un seul Amazon S3 compartiment. Pour plus d'informations, consultez les [sections Réception de fichiers CloudTrail journaux de plusieurs régions](#) et [Réception de fichiers CloudTrail journaux de plusieurs comptes](#).

Exemple : entrées de fichier Amazon Transcribe journal

Un suivi est une configuration qui permet de transmettre des événements sous forme de fichiers journaux à un Amazon S3 compartiment spécifié. CloudTrail les fichiers journaux contiennent une ou plusieurs entrées de journal. Un événement représente une demande individuelle d'une source quelconque. Il inclut des informations sur l'action demandée, telles que la date et l'heure de l'action, ainsi que les paramètres de la demande. CloudTrail les fichiers journaux ne constituent pas une trace ordonnée des appels d'API publics, ils n'apparaissent donc pas dans un ordre spécifique.

Les appels vers les opérations d'API [StartTranscriptionJob](#) et [GetTranscriptionJob](#) créent l'entrée suivante.

```
{
  "Records": [
    {
      "eventVersion": "1.05",
      "userIdentity": {
        "type": "IAMUser",
        "principalId": "111122223333",
        "arn": "arn:aws:iam:us-west-2:111122223333:user/my-user-name",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "my-user-name"
      },
      "eventTime": "2022-03-07T15:03:45Z",
      "eventSource": "transcribe.amazonaws.com",
      "eventName": "StartTranscriptionJob",
      "awsRegion": "us-west-2",
      "sourceIPAddress": "127.0.0.1",
      "userAgent": "[]",
      "requestParameters": {
        "mediaFormat": "flac",
        "languageCode": "en-US",
        "transcriptionJobName": "my-first-transcription-job",
        "media": {
          "mediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-media-file.flac"
        }
      },
      "responseElements": {
        "transcriptionJob": {
          "transcriptionJobStatus": "IN_PROGRESS",
          "mediaFormat": "flac",
          "creationTime": "2022-03-07T15:03:44.229000-08:00",

```

```
        "transcriptionJobName": "my-first-transcription-job",
        "languageCode": "en-US",
        "media": {
            "mediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-media-file.flac"
        }
    },
    "requestID": "47B8E8D397DCE7A6",
    "eventID": "cdc4b7ed-e171-4cef-975a-ad829d4123e8",
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
},
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "111122223333",
        "arn": "arn:aws:iam:us-west-2:111122223333:user/my-user-name",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "my-user-name"
    },
    "eventTime": "2022-03-07T15:07:11Z",
    "eventSource": "transcribe.amazonaws.com",
    "eventName": "GetTranscriptionJob",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "127.0.0.1",
    "userAgent": "[]",
    "requestParameters": {
        "transcriptionJobName": "my-first-transcription-job"
    },
    "responseElements": {
        "transcriptionJob": {
            "settings": {

            },
            "transcriptionJobStatus": "COMPLETED",
            "mediaFormat": "flac",
            "creationTime": "2022-03-07T15:03:44.229000-08:00",
            "transcriptionJobName": "my-first-transcription-job",
            "languageCode": "en-US",
            "media": {
                "mediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-media-file.flac"
            }
        },
    },
}
```

```
        "transcript": {
            "transcriptFileUri": "s3://DOC-EXAMPLE-BUCKET/my-first-
transcription-job.json"
        }
    },
    "requestID": "BD8798EACDD16751",
    "eventID": "607b9532-1423-41c7-b048-ec2641693c47",
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
]
```

Utilisation Amazon EventBridge avec Amazon Transcribe

Avec Amazon EventBridge, vous pouvez répondre aux changements d'état de vos Amazon Transcribe tâches en déclenchant des événements dans d'autres tâches Services AWS. Lorsqu'une tâche de transcription change d'état, envoie EventBridge automatiquement un événement vers un flux d'événements. Vous pouvez créer des règles qui définissent les événements que vous souhaitez surveiller dans le flux d'événements et l'action que EventBridge doit prendre lorsque ces événements se produisent. Par exemple, le routage de l'événement vers un autre service (ou une autre cible) qui peut alors effectuer une action. Vous pouvez, par exemple, configurer une règle pour acheminer un événement vers une AWS Lambda fonction lorsqu'une tâche de transcription est terminée avec succès. Pour définir des [règles EventBridge](#), reportez-vous aux sections suivantes.

Vous pouvez recevoir des notifications relatives à des événements via plusieurs canaux, notamment des e-mails, des notifications de chat [AWS Chatbot](#) ou des notifications push [AWS Console Mobile Application](#). Vous pouvez également consulter les notifications dans le [Centre de notifications de la console](#). Si vous souhaitez configurer des notifications, vous pouvez utiliser [Notifications des utilisateurs AWS](#). Notifications des utilisateurs AWS prend en charge l'agrégation, ce qui peut réduire le nombre de notifications que vous recevez lors d'événements spécifiques.

Définition des EventBridge règles

Pour définir EventBridge des règles, utilisez le [AWS Management Console](#). Lorsque vous définissez une règle, utilisez Amazon Transcribe comme nom de service. Pour un exemple de création d'une EventBridge règle, consultez la section [Amazon EventBridge règles](#).

Avant utilisation EventBridge, prenez note des définitions suivantes :

- Événement – Un événement indique un changement d'état de l'une de vos tâches de transcription. Par exemple, lorsque l'état `TranscriptionJobStatus` d'une tâche passe de `IN_PROGRESS` à `COMPLETED`.
- Cible – Une cible est un autre Service AWS qui traite un événement. Par exemple, AWS Lambda ou Amazon Simple Notification Service (Amazon SNS). Les cibles reçoivent les événements au format JSON.
- Règle : une règle fait correspondre les événements entrants que vous EventBridge souhaitez surveiller et les achemine vers une ou plusieurs cibles à des fins de traitement. Si une règle achemine un événement vers plusieurs cibles, toutes les cibles traitent l'événement en parallèle. Une règle peut personnaliser le fichier JSON envoyé à la cible.

Amazon EventBridge les événements sont émis dans la mesure du possible. Pour plus d'informations sur la création et la gestion d'événements dans EventBridge, consultez la section [Amazon EventBridge événements](#) du Guide de Amazon EventBridge l'utilisateur.

Voici un exemple de EventBridge règle initiée lorsque Amazon Transcribe le statut d'une tâche de transcription devient `COMPLETED` ou `FAILED`.

```
{
  "source": [
    "aws.transcribe"
  ],
  "detail-type": [
    "Transcribe Job State Change"
  ],
  "detail": {
    "TranscriptionJobStatus": [
      "COMPLETED",
      "FAILED"
    ]
  }
}
```

La règle contient les champs suivants :

- `source` – La source de l'événement. Car Amazon Transcribe c'est toujours le `aws.transcribe`.
- `detail-type` – Un identifiant pour les détails de l'événement. Pour Amazon Transcribe, c'est toujours `Transcribe Job State Change`.

- `detail` – Le nouvel état de la tâche de transcription. Dans cet exemple, la règle déclenche un événement lorsque l'état de la tâche devient `COMPLETED` ou `FAILED`.

Amazon Transcribe événements

Amazon EventBridge enregistre plusieurs Amazon Transcribe événements :

- [Événements de tâche de transcription](#)
- [Événements d'identification des langues](#)
- [Événements Call Analytics](#)
- [Événements après appel Call Analytics](#)
- [Événements de vocabulaire](#)

Ces événements contiennent tous les champs partagés suivants :

- `version` : la version des données de l'événement. Cette valeur est toujours `0`.
- `id`: identifiant unique généré par EventBridge l'événement.
- `detail-type` : un identifiant pour les détails de l'événement. Par exemple, `Transcribe Job State Change`.
- `source` : la source de l'événement. Car Amazon Transcribe c'est toujours le `casaws.transcribe`.
- `account`: Compte AWS ID du compte qui a généré l'appel d'API.
- `time` : la date et l'heure de diffusion de l'événement.
- `region`: L' Région AWS endroit dans lequel la demande est faite.
- `resources` : les ressources utilisées par l'appel d'API. Car Amazon Transcribe ce champ est toujours vide.
- `detail` : des détails supplémentaires sur l'événement.
 - `FailureReason` : ce champ est présent si l'état ou le statut passe à `FAILED` et décrit la raison de cet état ou de ce statut `FAILED`.
 - Chaque type d'événement comporte des champs uniques supplémentaires qui sont affichés sous `detail`. Ces champs uniques sont définis dans les sections suivantes après chaque exemple d'événement.

Événements de tâche de transcription

Lorsque l'état d'une tâche passe IN_PROGRESS de COMPLETED ou FAILED, cela Amazon Transcribe génère un événement. Pour identifier la tâche qui a changé d'état et déclencher l'événement dans votre cible, utilisez le champ `TranscriptionJobName` de l'événement. Un Amazon Transcribe événement contient les informations suivantes. Un champ `FailureReason` est ajouté sous `detail` si le statut de votre tâche de transcription est FAILED.

Notez que cet événement s'applique uniquement à l'opération de l'API [StartTranscriptionJob](#).

```
{
  "version": "0",
  "id": "event ID",
  "detail-type": "Transcribe Job State Change",
  "source": "aws.transcribe",
  "account": "111122223333",
  "time": "timestamp",
  "region": "us-west-2",
  "resources": [],
  "detail": {
    "TranscriptionJobName": "my-first-transcription-job",
    "TranscriptionJobStatus": "COMPLETED" (or "FAILED")
  }
}
```

- `TranscriptionJobName` : nom unique que vous avez choisi pour votre tâche de transcription.
- `TranscriptionJobStatus` : statut de la tâche de transcription. La valeur peut être COMPLETED ou FAILED.

Événements d'identification des langues

Quand vous activez l'[identification automatique des langues](#), Amazon Transcribe génère un événement lorsque l'état d'identification des langues est COMPLETED ou FAILED. Pour identifier la tâche qui a changé d'état et déclencher l'événement dans votre cible, utilisez le champ `JobName` de l'événement. Un événement Amazon Transcribe contient les informations suivantes : Un champ `FailureReason` est ajouté sous `detail` si votre statut d'identification des langues est FAILED.

Notez que cet événement s'applique uniquement à l'opération de l'API [StartTranscriptionJob](#) lorsque le paramètre [LanguageIdSettings](#) est inclus.

```
{
  "version": "0",
  "id": "event ID",
  "detail-type": "Language Identification State Change",
  "source": "aws.transcribe",
  "account": "111122223333",
  "time": "timestamp",
  "region": "us-west-2",
  "resources": [],
  "detail": {
    "JobType": "TranscriptionJob",
    "JobName": "my-first-lang-id-job",
    "LanguageIdentificationStatus": "COMPLETED" (or "FAILED")
  }
}
```

- JobType : pour les tâches de transcription, cette valeur doit être TranscriptionJob.
- JobName : nom unique de votre tâche de transcription.
- LanguageIdentificationStatus : statut de l'identification des langues dans une tâche de transcription. La valeur peut être COMPLETED ou FAILED.

Événements Call Analytics

Lorsque l'état d'une tâche [Call Analytics](#) passe de IN_PROGRESS à COMPLETED ou FAILED, Amazon Transcribe génère un événement. Pour identifier la tâche Call Analytics qui a changé d'état et déclencher l'événement dans votre cible, utilisez le champ JobName de l'événement. Un événement Amazon Transcribe contient les informations suivantes : Un champ FailureReason est ajouté sous detail si le statut de votre tâche Call Analytics est FAILED.

Notez que cet événement s'applique uniquement à l'opération de l'API [StartCallAnalyticsJob](#).

```
{
  "version": "0",
  "id": "event ID",
  "detail-type": "Call Analytics Job State Change",
  "source": "aws.transcribe",
  "account": "111122223333",
  "time": "timestamp",
  "region": "us-west-2",
  "resources": [],
```

```
"detail": {
  "JobName": "my-first-analytics-job",
  "JobStatus": "COMPLETED" (or "FAILED"),
  "AnalyticsJobDetails": { // only when you enable optional features such as
    Generative Call Summarization
      "Skipped": []
    }
  }
}
```

- **JobName** : nom unique de votre tâche de transcription Call Analytics.
- **JobStatus** : statut de votre tâche de transcription Call Analytics. Il peut correspondre à COMPLETED ou FAILED.
- **AnalyticsJobDetails**: les détails de votre tâche de transcription Call Analytics, y compris les informations sur les fonctionnalités d'analyse ignorées.

Événements après appel Call Analytics

Lorsque l'état d'une transcription d'[analyse après appel](#) passe de IN_PROGRESS à COMPLETED ou FAILED, Amazon Transcribe génère un événement. Pour identifier la tâche après appel Call Analytics qui a changé d'état et déclencher l'événement dans votre cible, utilisez le champ `StreamingSessionId` de l'événement.

Notez que cet événement s'applique uniquement à l'opération de l'API [StartCallAnalyticsStreamTranscription](#) lorsque le paramètre [PostCallAnalyticsSettings](#) est inclus.

Un événement COMPLETED contient les informations suivantes :

```
{
  "version": "0",
  "id": "event ID",
  "detail-type": "Call Analytics Post Call Job State Change",
  "source": "aws.transcribe",
  "account": "111122223333",
  "time": "timestamp",
  "region": "us-west-2",
  "resources": [],
  "detail": {
    "StreamingSessionId": "session-id",
```

```
    "PostCallStatus": "COMPLETED",
    "Transcript": {
      "RedactedTranscriptFileUri": "s3://DOC-EXAMPLE-BUCKET/my-output-files/my-
redacted-file.JSON",
      "TranscriptFileUri": "s3://DOC-EXAMPLE-BUCKET/my-output-files/my-file.JSON"
    },
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-output-files/my-redacted-
file.WAV",
      "RedactedMediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-output-files/my-
redacted-file.WAV"
    }
  }
}
```

Un événement FAILED contient les informations suivantes :

```
{
  "version": "0",
  "id": "event ID",
  "detail-type": "Call Analytics Post Call Job State Change",
  "source": "aws.transcribe",
  "account": "111122223333",
  "time": "timestamp",
  "region": "us-west-2",
  "resources": [],
  "detail": {
    "StreamingSessionId": "session-id",
    "PostCallStatus": "FAILED"
  }
}
```

- **StreamingSessionId** : numéro d'identification attribué à votre demande de transcription Call Analytics en temps réel.
- **PostCallStatus** : statut de votre transcription Call Analytics après appel. Il peut correspondre à COMPLETED ou FAILED.
- **Transcript** : URI de vos transcriptions expurgées et non expurgées.
- **Media** : URI de vos fichiers audio expurgés et non expurgés.

Événements de vocabulaire

Lorsque l'état d'un [vocabulaire personnalisé](#) passe PENDING de READY ou FAILED, cela Amazon Transcribe génère un événement. Pour identifier le vocabulaire personnalisé qui a changé d'état et déclencher l'événement dans votre cible, utilisez le champ `VocabularyName` de l'événement. Un Amazon Transcribe événement contient les informations suivantes. Un champ `FailureReason` est ajouté sous `detail` si l'état de votre vocabulaire personnalisé est FAILED.

Notez que cet événement s'applique uniquement à l'opération de l'API [CreateVocabulary](#).

```
{
  "version": "0",
  "id": "event ID",
  "detail-type": "Vocabulary State Change",
  "source": "aws.transcribe",
  "account": "111122223333",
  "time": "timestamp",
  "region": "us-west-2",
  "resources": [],
  "detail": {
    "VocabularyName": "unique-vocabulary-name",
    "VocabularyState": "READY" (or "FAILED")
  }
}
```

- `VocabularyName` : nom unique de votre vocabulaire personnalisé.
- `VocabularyState` : état de traitement de votre vocabulaire personnalisé. La valeur peut être READY ou FAILED.


Validation de conformité pour Amazon Transcribe

Pour savoir si un [programme Services AWS de conformité Service AWS s'inscrit dans le champ d'application de programmes de conformité](#) spécifiques, consultez Services AWS la section de conformité et sélectionnez le programme de conformité qui vous intéresse. Pour des informations générales, voir Programmes de [AWS conformité Programmes AWS](#) de .

Vous pouvez télécharger des rapports d'audit tiers à l'aide de AWS Artifact. Pour plus d'informations, voir [Téléchargement de rapports dans AWS Artifact](#) .

Votre responsabilité en matière de conformité lors de l'utilisation Services AWS est déterminée par la sensibilité de vos données, les objectifs de conformité de votre entreprise et les lois et réglementations applicables. AWS fournit les ressources suivantes pour faciliter la mise en conformité :

- [Guides de démarrage rapide sur la sécurité et la conformité](#) : ces guides de déploiement abordent les considérations architecturales et indiquent les étapes à suivre pour déployer des environnements de base axés sur AWS la sécurité et la conformité.
- [Architecture axée sur la sécurité et la conformité HIPAA sur Amazon Web Services](#) : ce livre blanc décrit comment les entreprises peuvent créer des applications AWS conformes à la loi HIPAA.

 Note

Tous ne Services AWS sont pas éligibles à la loi HIPAA. Pour plus d'informations, consultez le [HIPAA Eligible Services Reference](#).

- AWS Ressources de <https://aws.amazon.com/compliance/resources/> de conformité — Cette collection de classeurs et de guides peut s'appliquer à votre secteur d'activité et à votre région.
- [AWS Guides de conformité destinés aux clients](#) — Comprenez le modèle de responsabilité partagée sous l'angle de la conformité. Les guides résument les meilleures pratiques en matière de sécurisation Services AWS et décrivent les directives relatives aux contrôles de sécurité dans de nombreux cadres (notamment le National Institute of Standards and Technology (NIST), le Payment Card Industry Security Standards Council (PCI) et l'Organisation internationale de normalisation (ISO)).
- [Évaluation des ressources à l'aide des règles](#) du guide du AWS Config développeur : le AWS Config service évalue dans quelle mesure les configurations de vos ressources sont conformes aux pratiques internes, aux directives du secteur et aux réglementations.
- [AWS Security Hub](#)— Cela Service AWS fournit une vue complète de votre état de sécurité interne AWS. Security Hub utilise des contrôles de sécurité pour évaluer vos ressources AWS et vérifier votre conformité par rapport aux normes et aux bonnes pratiques du secteur de la sécurité. Pour obtenir la liste des services et des contrôles pris en charge, consultez [Référence des contrôles Security Hub](#).
- [Amazon GuardDuty](#) — Cela Service AWS détecte les menaces potentielles qui pèsent sur vos charges de travail Comptes AWS, vos conteneurs et vos données en surveillant votre environnement pour détecter toute activité suspecte et malveillante. GuardDuty peut vous aider

à répondre à diverses exigences de conformité, telles que la norme PCI DSS, en répondant aux exigences de détection des intrusions imposées par certains cadres de conformité.

- [AWS Audit Manager](#)— Cela vous Service AWS permet d'auditer en permanence votre AWS utilisation afin de simplifier la gestion des risques et la conformité aux réglementations et aux normes du secteur.

Résilience dans Amazon Transcribe

L'infrastructure mondiale d'AWS est construite autour de zones de disponibilité et de Régions AWS. Les Régions AWS fournissent plusieurs zones de disponibilité physiquement séparées et isolées, reliées par un réseau à latence faible, à débit élevé et à forte redondance. Avec les zones de disponibilité, vous pouvez concevoir et exploiter des applications et des bases de données qui basculent automatiquement d'une zone à l'autre sans interruption. Les zones de disponibilité sont davantage disponibles, tolérantes aux pannes et ont une plus grande capacité de mise à l'échelle que les infrastructures traditionnelles à un ou plusieurs centres de données.

Pour plus d'informations sur les Régions AWS et les zones de disponibilité, consultez [Infrastructure mondiale d'AWS](#).

Sécurité de l'infrastructure dans Amazon Transcribe

En tant que service géré, Amazon Transcribe il est protégé par la sécurité AWS globale du réseau. Pour plus d'informations sur les services de sécurité AWS et la manière dont AWS protège l'infrastructure, consultez la section [Sécurité du cloud AWS](#). Pour concevoir votre environnement AWS en utilisant les meilleures pratiques en matière de sécurité de l'infrastructure, consultez la section [Protection de l'infrastructure](#) dans le Security Pillar AWS Well-Architected Framework (Pilier de sécurité de l'infrastructure Well-Architected Framework).

Vous utilisez les appels d'API publiés AWS pour accéder à Amazon Transcribe via le réseau. Les clients doivent prendre en charge les éléments suivants :

- Protocole TLS (Transport Layer Security). Nous exigeons TLS 1.2 et nous recommandons TLS 1.3.
- Ses suites de chiffrement PFS (Perfect Forward Secrecy) comme DHE (Ephemeral Diffie-Hellman) ou ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). La plupart des systèmes modernes tels que Java 7 et les versions ultérieures prennent en charge ces modes.

En outre, les demandes doivent être signées à l'aide d'un ID de clé d'accès et d'une clé d'accès secrète associée à un principal IAM. Vous pouvez également utiliser [AWS Security Token Service](#) (AWS STS) pour générer des informations d'identification de sécurité temporaires et signer les demandes.

Analyse et gestion des vulnérabilités dans Amazon Transcribe

La configuration et les contrôles informatiques sont une responsabilité partagée entre AWS et vous, notre client. Pour de plus amples informations, veuillez consulter [Modèle de responsabilité partagée AWS](#).

Amazon Transcribe et interface des points de terminaison VPC (AWS PrivateLink)

Vous pouvez établir une connexion privée entre votre VPC et Amazon Transcribe en créant un point de terminaison de VPC d'interface. Les points de terminaison de l'interface sont alimentés par [AWS PrivateLink](#), une technologie que vous pouvez utiliser pour accéder en privé à Amazon Transcribe. Des API sans passerelle Internet, appareil NAT, connexion VPN ou AWS Direct Connect connexion. Les instances de votre VPC ne requièrent pas d'adresses IP publiques pour communiquer avec les API Amazon Transcribe. Le trafic entre votre VPC et Amazon Transcribe ne quitte pas le réseau Amazon.

Chaque point de terminaison d'interface est représenté par une ou plusieurs [interfaces réseau Elastic](#) dans vos sous-réseaux.

Pour de plus amples informations, consultez [Points de terminaison VPC \(AWS PrivateLink\)](#) dans le Guide de l'utilisateur Amazon VPC.

Considérations relatives aux points de terminaison de VPC Amazon Transcribe

Avant de configurer un point de terminaison VPC d'interface pour Amazon Transcribe, assurez-vous de vérifier [Propriétés et limites des points de terminaison de l'interface](#) dans le Amazon VPC Guide de l'utilisateur.

Amazon Transcribe prend en charge l'exécution d'appels en direction de toutes ses actions d'API depuis votre VPC.

Création d'un point de terminaison de VPC d'interface pour Amazon Transcribe

Vous pouvez créer un point de terminaison VPC pour Amazon Transcribe service utilisant le Amazon VPC AWS Management Console ou AWS CLI. Pour de plus amples informations, veuillez consulter [Création d'un point de terminaison d'interface](#) dans le Guide de l'utilisateur Amazon VPC.

Pour les transcriptions par lots dans Amazon Transcribe, créez un point de terminaison VPC en utilisant le nom de service suivant :

- `com.amazonaws.Etats-Unis-Ouest-2.transcribe`

Pour diffuser des transcriptions dans Amazon Transcribe, créez un point de terminaison VPC en utilisant le nom de service suivant :

- `com.amazonaws.Etats-Unis-Ouest-2.transcribe le streaming`

Si vous activez le DNS privé pour le point de terminaison, vous pouvez envoyer des demandes d'API à Amazon Transcribe en utilisant son nom DNS par défaut pour Région AWS, par exemple `transcribestreaming.us-east-2.amazonaws.com`.

Pour plus d'informations, voir [Accès à un service via un point de terminaison d'interface](#) dans le Amazon VPC Guide de l'utilisateur.

Création d'une stratégie de point de terminaison d'un VPC pour Amazon Transcribe

Vous pouvez associer une politique de point de terminaison à votre point de terminaison VPC qui contrôle l'accès au service de streaming ou au service de transcription par lots de Amazon Transcribe. La politique spécifie les informations suivantes :

- Le principal qui peut exécuter des actions.
- Les actions qui peuvent être effectuées.
- Les ressources sur lesquelles les actions peuvent être exécutées.

Pour de plus amples informations, veuillez consulter [Contrôle de l'accès aux services avec des points de terminaison de VPC](#) dans le Guide de l'utilisateur Amazon VPC.

Exemple : politique de point de terminaison VPC pour Amazon Transcribe actions de transcription par lots

Voici un exemple de politique de point de terminaison pour une transcription par lots dans Amazon Transcribe. Lorsqu'elle est attachée à un point de terminaison, cette politique accorde l'accès aux actions Amazon Transcribe répertoriées pour tous les principaux sur toutes les ressources.

```
{
  "Statement": [
    {
      "Principal": "*",
      "Effect": "Allow",
      "Action": [
        "transcribe:StartTranscriptionJob",
        "transcribe:ListTranscriptionJobs"
      ],
      "Resource": "*"
    }
  ]
}
```

Exemple : politique de point de terminaison VPC pour Amazon Transcribe actions de transcription en streaming

Voici un exemple de politique de point de terminaison pour une transcription en streaming dans Amazon Transcribe. Lorsqu'elle est attachée à un point de terminaison, cette politique accorde l'accès aux actions Amazon Transcribe répertoriées pour tous les principaux sur toutes les ressources.

```
{
  "Statement": [
    {
      "Principal": "*",
      "Effect": "Allow",
      "Action": [
        "transcribe:StartStreamTranscription",
        "transcribe:StartStreamTranscriptionWebsocket"
      ],
      "Resource": "*"
    }
  ]
}
```

Sous-réseaux partagés

Vous ne pouvez pas créer, décrire, modifier ou supprimer des points de terminaison VPC dans des sous-réseaux qui sont partagés avec vous. Toutefois, vous pouvez utiliser les points de terminaison d'un VPC dans les sous-réseaux qui sont partagés avec vous. Pour plus d'informations sur le partage de VPC, voir [Partagez votre VPC avec d'autres comptes](#) dans le Amazon Virtual Private Cloudguide.

Bonnes pratiques de sécurité pour Amazon Transcribe

Les bonnes pratiques suivantes doivent être considérées comme des instructions générales et ne représentent pas une solution de sécurité complète. Étant donné que ces bonnes pratiques peuvent ne pas être appropriées ou suffisantes pour votre environnement, considérez-les comme des recommandations.

- Utiliser le chiffrement des données, tel que le contexteAWS KMS de chiffrement

AWS KMSLe contexte de chiffrement est une carte de paires clé:valeur non secrètes en texte brut. Cette carte représente des données authentifiées supplémentaires, appelées paires de contextes de chiffrement, qui fournissent un niveau de sécurité supplémentaire à vos données.

Pour plus d'informations, consultez [Contexte de chiffrement AWS KMS](#).

- Utilisez des informations d'identification temporaires dans la mesure du possible

Dans la mesure du possible, utilisez des informations d'identification temporaires plutôt que comme des informations d'identification à long terme tels que les clés d'accès. pour les scénarios dans lesquels vous avez besoin d'IAMutilisateurs disposant d'un accès par programmation et d'informations d'identification à long terme, nous vous recommandons de faire pivoter les clés d'accès. La rotation régulière des titres de compétences à long terme vous aide à vous familiariser avec le processus. Cela est utile dans le cas où vous retrouvez dans une situation où vous devez alterner les informations d'identification, par exemple lorsqu'un employé quitte votre entreprise. Nous vous recommandons d'utiliserIAM les dernières informations utilisées pour faire pivoter et supprimer les clés d'accès en toute sécurité.

Pour plus d'informations, consultez la section [Rotation des clés d'accès](#) et [Bonnes pratiques de sécurité dansIAM](#).

- UtiliserIAM des rôles pour les applications et lesAWS services qui nécessitent unAmazon Transcribe accès

Utilisez un IAM rôle pour gérer des informations d'identification temporaires pour les applications ou services devant accéder à Amazon Transcribe. Lorsque vous utilisez un rôle, vous n'avez pas à distribuer des informations d'identification à long terme, tels que des mot de passe ou des clé d'accès, à une Amazon EC2 instance ou un AWS service. IAM les rôles peuvent fournir des autorisations temporaires que les applications peuvent utiliser lors de demandes de AWS ressources.

Pour plus d'informations, reportez-vous à la section [IAM Rôles](#) et [Scénarios courants pour les rôles : utilisateurs, applications et services](#).

- Utiliser un contrôle d'accès basé sur les balises

Vous pouvez utiliser des balises pour contrôler l'accès au sein de votre Comptes AWS. Des balises Amazon Transcribe In. peuvent être ajoutées aux tâches de transcription, aux vocabulaires personnalisés, aux filtres de vocabulaire personnalisés et aux modèles linguistiques personnalisés.

Pour plus d'informations, consultez [Contrôle d'accès basé sur les étiquettes](#).

- Utiliser des outils AWS de surveillance

La surveillance est un enjeu important pour assurer la fiabilité, la sécurité, la disponibilité et les performances d'Amazon Transcribe et de vos solutions AWS. Vous pouvez surveiller Amazon Transcribe en utilisant CloudTrail.

Pour plus d'informations, consultez [Surveillance Amazon Transcribe avec AWS CloudTrail](#).

- Activer les AWS Config

AWS Config peut évaluer, contrôler et évaluer les configurations de vos AWS ressources. À l'aide de AWS Config, vous pouvez consulter les modifications apportées aux configurations et aux relations entre les AWS ressources. Vous pouvez également consulter des historiques de configuration des ressources détaillés et déterminer votre conformité globale aux configurations spécifiées dans vos directives internes. Cela peut vous aider à simplifier le contrôle de la conformité, l'analyse de la sécurité, la gestion des modifications et le diagnostic de défaillances opérationnelles.

Pour plus d'informations, reportez-vous à la section [Qu'est-ce que c'est AWS Config ?](#)

Amazon Transcribe Médical

Amazon TranscribeMedical est un service de reconnaissance vocale automatique (ASR) conçu pour les professionnels de la santé qui souhaitent transcrire des discours liés à la médecine, tels que des notes dictées par un médecin, une surveillance de la sécurité des médicaments, des rendez-vous de télémédecine ou des conversations entre médecins et patients. Amazon Transcribe Les services médicaux sont disponibles soit par diffusion en temps réel (via microphone), soit par transcription d'un fichier téléchargé (par lots).

Important

Amazon TranscribeLa médecine ne remplace pas les conseils, le diagnostic ou le traitement d'un professionnel de la santé. Déterminez le seuil de confiance approprié pour votre cas et utilisez des seuils de confiance élevés dans les situations qui exigent une grande précision. Pour certains cas d'utilisation, les résultats doivent être examinés et vérifiés par des évaluateurs humains correctement formés. Amazon Transcribe Les transcriptions médicales ne doivent être utilisées dans les scénarios de soins aux patients qu'après vérification de l'exactitude et du bon jugement médical par des professionnels de la santé qualifiés.

Amazon TranscribeMedical fonctionne selon un modèle de responsabilité partagée, dans lequelAWS il est responsable de la protection de l'infrastructure qui gèreAmazon Transcribe Medical et vous êtes responsable de la gestion de vos données. Pour plus d'informations, consultez [Modèle de responsabilité partagée](#).

Amazon TranscribeMedical est disponible en anglais américain (en-US).

Pour de meilleurs résultats, utilisez un format audio sans perte, tel que FLAC ou WAV, avec un codage PCM 16 bits. Amazon Transcribe Medicalprend en charge des fréquences d'échantillonnage de 16 000 Hz ou plus.

Pour l'analyse de vos relevés de notes, vous pouvez en utiliser d'autresServices AWS, tels que [Amazon Comprehend Medical](#).

Spécialités prises en charge

Spécialité	Sous-spécialité	Entrée audio
Cardiologie	Aucun(e)	streaming uniquement
Neurologie	Aucun(e)	streaming uniquement
Oncologie	Aucun(e)	streaming uniquement
Soins de traitement par lots	Médecine familiale	traitement par lots, traitement par lots
Soins de traitement par lots	Médecine interne	traitement par lots, traitement par lots
Soins de traitement par lots	Obstétrique et gynécologie (OB-GYN)	traitement par lots, traitement par lots
Soins de traitement par lots	Pédiatrie	traitement par lots, traitement par lots
Radiologie	Aucun(e)	streaming uniquement
Urologie	Aucun(e)	streaming uniquement

Disponibilité et quotas par région

L'analyse des appels est prise en charge dans les domaines suivants Régions AWS :

Région	Type de transcription
af-south-1 (Le Cap)	lot
ap-east-1, Hong Kong)	lot
ap-northeast-1 (Tokyo)	traitement par lots, traitement par lots
ap-northeast-2 (Séoul)	traitement par lots, traitement par lots

Région	Type de transcription
ap-south-1 (Bombay)	lot
ap-southeast-1 (Singapour)	lot
ap-southeast-2 (Sydney)	traitement par lots, traitement par lots
ca-central-1, Canada, Centre)	traitement par lots, traitement par lots
eu-central-1 (Francfort)	traitement par lots, traitement par lots
eu-north-1 (Stockholm)	lot
eu-west-1 (Irlande)	traitement par lots, traitement par lots
eu-west-2 (Londres)	traitement par lots, traitement par lots
eu-west-3 (Paris)	lot
me-south-1 (Bahreïn)	lot
sa-east-1 (São Paulo)	traitement par lots, traitement par lots
us-east-1 (Virginie du Nord)	traitement par lots, traitement par lots
us-east-2 (Ohio)	traitement par lots, traitement par lots
us-gov-east-1 (GovCloud, US-Est)	traitement par lots, traitement par lots
us-gov-west-1 (GovCloud, US-Ouest)	traitement par lots, traitement par lots
us-west-1, San Francisco)	lot
us-west-2 (Oregon)	traitement par lots, traitement par lots

Notez que la prise en charge par région n'est pas la même pour [Amazon Transcribe](#) Amazon Transcribe Medical, et [Call Analytics](#).

Pour obtenir les points de terminaison pour chaque région prise en charge, consultez la section [Points de terminaison du service](#) dans la RéférenceAWS générale.

Pour obtenir la liste des quotas relatifs à vos transcriptions, reportez-vous aux [quotas de service](#) dans la RéférenceAWS générale. Certains quotas peuvent être modifiés sur demande. Si la colonne Réglable contient « Oui », vous pouvez demander une augmentation. Pour ce faire, sélectionnez le lien fourni.

Spécialités et termes de base

Lors de la création d'une tâche de transcription médicale, vous spécifiez la langue, la spécialité médicale et le type audio du fichier source. Vous entrez l'anglais américain (en-US) comme langue et PRIMARYCARE comme spécialité médicale. La saisie des soins primaires comme valeur vous permet de générer des transcriptions à partir de l'audio source dans les spécialités médicales suivantes :

- Médecine familiale
- Médecine interne
- Obstétrique et gynécologie (OB-GYN)
- Pédiatrie

Vous avez le choix entre la dictée et la conversation pour votre type d'audio. Choisissez la dictée pour les fichiers audio où le médecin fait un rapport sur une visite ou une intervention du patient. Choisissez la conversation pour les fichiers audio qui impliquent une conversation entre un médecin et un patient ou une conversation entre médecins.

Pour stocker la sortie de votre travail de transcription, sélectionnez un compartiment Amazon S3 que vous avez déjà créé. Pour plus d'informations sur les Amazon S3 compartiments, consultez la section [Mise en route avec Amazon Simple Storage Service](#).

Le nombre minimal de paramètres de requête à saisir dans l'exemple de JSON est indiqué ci-dessous.

```
{
  "MedicalTranscriptionJobName": "my-first-transcription-job",
  "LanguageCode": "en-US",
  "Media": {
    "MediaFileUri": "s3://path to your audio file"
  },
  "OutputBucketName": "your output bucket name",
  "Specialty": "PRIMARYCARE",
  "Type": "CONVERSATION"
```


}

Amazon TranscribeMedical vous permet de générer des transcriptions alternatives. Pour plus d'informations, veuillez consulter [Génération de transcriptions alternatives](#).

Vous pouvez également activer le partitionnement des haut-parleurs ou identifier les canaux de votre audio. Pour plus d'informations, consultez [Activation du partitionnement des haut-parleurs](#) et [Transcription audio multicanal](#).

Transcription de termes et de mesures médicaux

Amazon TranscribeMedical peut transcrire des termes et des mesures médicaux. Amazon TranscribeAbréviations des termes parlés dans les publications médicales. Par exemple, la « pression artérielle » est transcrite en tant que PA. Vous trouverez une liste des conventions utilisées parAmazon Transcribe Medical pour les termes et les mesures médicaux dans le tableau de cette page. La colonne Terme parlé fait référence au terme parlé dans l'audio source. La colonne Sortie fait référence à l'abréviation que vous voyez dans vos résultats de transcription.

Vous pouvez voir comment les termes prononcés dans l'audio source correspondent à la sortie de transcription ici.

Terme prononcé dans la source audio	Abréviation utilisée dans la sortie	Exemple de sortie
Centigrade	C	La température du patient est de 37,4 °C.
Celsius	C	La température du patient est de 37,4 °C.
Fahrenheit	F	La température du patient est de 101 °F.
Grammes	g	Une masse de 100 g a été extraite du patient.
Mètres	m	Le patient mesure 1,80 m.
Pieds	ft	Le patient mesure 6 pieds.

Terme prononcé dans la source audio	Abréviation utilisée dans la sortie	Exemple de sortie
Kilos	kg	Le patient pèse 80 kg.
Kilogrammes	kg	Le patient pèse 80 kg.
c c	cc	Le patient a reçu 100 cc de solution saline.
Centimètre cube	cc	Le patient a reçu 100 cc de solution saline.
Millilitre	mL	Le patient a excrété 100 mL d'urine.
Pression artérielle	PA	La PA du patient était élevée.
b p	PA	La PA du patient était élevée.
X sur Y	X/Y	La PA du patient était de 120/80.
Battements par min	BPM	Le patient avait une fibrillation auriculaire avec une fréquence cardiaque de 160 BPM.
Battements par minute	BPM	Le patient avait une fibrillation auriculaire avec une fréquence cardiaque de 160 BPM.
O 2	O2	La saturation en O2 du patient était de 98 %.
CO2	CO2	Le patient a besoin d'un soutien respiratoire pour une augmentation du CO2.
Après l'opération	POSTOP	Le patient est venu pour l'évaluation POSTOP.

Terme prononcé dans la source audio	Abréviation utilisée dans la sortie	Exemple de sortie
Post op	POSTOP	Le patient est venu pour l'évaluation POSTOP.
Tomodensitogramme	TDM	L'indication du patient d'une hémorragie cérébrale nécessitait l'utilisation d'un TDM.
Pouls 80	P 80	Les signes vitaux des patients étaient P 80, R 17, ...
Respiration 17	R 17	Les signes vitaux des patients étaient P 80, R 17, ...
entrée et sortie	I/O	Le patient avait un rythme sinusal E/S
L cinq	L5	Une ponction lombaire a été effectuée entre L4 et L5

Transcrire des numéros

Amazon TranscribeMedical transcrit les chiffres sous forme de chiffres plutôt que de mots. Par exemple, le nombre « mille deux cent quarante-deux » est transcrit sous la forme 1242.

Les numéros sont transcrits selon les règles suivantes.

Règle	Description
Convertir les nombres cardinaux supérieurs à 10 en chiffres.	<ul style="list-style-type: none"> « Cinquante cinq » > 55 « Cent » > 100 « Mille-trente-un » > 1031

Règle	Description
	<ul style="list-style-type: none"> « Cent vingt-trois millions quatre cent cinquante six mille sept cent huit neuf » > 123 456 789
Convertissez les nombres cardinaux suivis de « million » ou de « milliard » en chiffres suivis d'un mot quand « million » ou « milliard » n'est pas suivi d'un nombre.	<ul style="list-style-type: none"> « cent millions » > 100 millions « un milliard » > 1 milliard « deux virgule trois millions » > 2,3 millions
Convertir les nombres ordinaux supérieurs à 10 en chiffres.	<ul style="list-style-type: none"> « Quarante troisième » > 43e « Vingt-sixième avenue » > 26e avenue
Fractions converties en leur format numérique.	<ul style="list-style-type: none"> « Un quart » > 1/4 « Trois seizièmes » > 3/16 « Un demi » > 1/2 « un centième » > 1/100
Nombres de moins de 10 chiffres convertis s'il y en a plus d'un dans une ligne.	<ul style="list-style-type: none"> « Trois quatre cinq » > 345 « Mon numéro de téléphone est quatre deux cinq cinq cinq un deux un deux » > 4255551212
Les décimales sont indiqués par « virgule ».	<ul style="list-style-type: none"> « trois cent trois point virgule cinq » > 303,5 « trois virgule vingt trois » > 3,23 « zéro virgule quatre » > 0,4 « Virgule trois » > 0,3
Mot « pour cent » après un nombre converti en signe de pourcentage.	<ul style="list-style-type: none"> « Vingt-trois pour cent » > 23 % « Vingt-trois virgule quarante-cinq pour cent » > 23,45 %
Mots « dollar », dollar américain », « dollar australien », « AUD » ou « USD » après un nombre convertis en un symbole de dollar avant le nombre.	<ul style="list-style-type: none"> « un dollar et quinze cents » > \$ 1,15 « vingt trois USD » > \$23 « vingt trois dollars australiens » > \$23

Règle	Description
Convertir les mots « livres » ou « milligrammes » en « lbs » ou « mg »,	<ul style="list-style-type: none"> « vingt trois livres » > 23 livres « quarante-cinq milligrammes » > 45 mg
Mots « roupies », « roupies indiennes » ou « INR » après un nombre convertis en signe roupie (#) avant le nombre.	<ul style="list-style-type: none"> « vingt trois roupies » > #23 « cinquante roupies trente paise » > #50,30
Heures converties en nombres.	<ul style="list-style-type: none"> « sept a m heure de l'est » > 7 a.m. heure de l'est « douze trente p m » > 12 h 30 p.m.
<p>Combinez les années exprimés en deux chiffres en quatre.</p> <p>Valide uniquement pour les 20e, 21e et 22e siècles.</p>	<ul style="list-style-type: none"> « dix-neuf soixante-deux » > 1962 « l'année est vingt douze » > l'année 2012 « vingt dix-neuf » > 2019 « vingt un trente » > 2130
Dates converties en nombres.	<ul style="list-style-type: none"> « Mai cinquième vingt douze » > 5 mai 2012 « Mai cinq vingt douze » > 5 mai 2012 « cinq mai vingt douze » > 5 mai 2012
Plages de nombres séparées par le mot « à ».	<ul style="list-style-type: none"> « vingt trois à trente sept » > 23 à 37

Transcription d'une conversation médicale

Vous pouvez utiliser Amazon Transcribe Medical pour transcrire une conversation médicale entre un clinicien et un patient à l'aide d'une tâche de transcription par lots ou d'un flux en temps réel. Les tâches de transcription Batch vous permettent de transcrire des fichiers audio. Pour garantir que Amazon Transcribe Medical produit des résultats de transcription avec la plus grande précision possible, vous devez spécifier la spécialité médicale du clinicien dans votre travail ou votre flux de transcription.

Vous pouvez transcrire une visite clinicien-patient dans les spécialités médicales suivantes :

- Cardiologie — disponible en transcription en streaming uniquement

- Neurologie — disponible en transcription en streaming uniquement
- Oncologie — disponible en transcription en streaming uniquement
- Soins primaires — incluent les types de pratique médicale suivants :
 - Médecine familiale
 - Médecine interne
 - Obstétrique et gynécologie (OB-GYN)
 - Pédiatrie
- Urologie — disponible en transcription en streaming uniquement

Vous pouvez améliorer la précision de la transcription en utilisant des vocabulaires médicaux personnalisés. Pour plus d'informations sur le fonctionnement des vocabulaires médicaux personnalisés, voir [Améliorer la précision de la transcription grâce à des vocabulaires médicaux personnalisés](#).

Par défaut, Amazon Transcribe Medical renvoie la transcription avec le niveau de confiance le plus élevé. Si vous souhaitez le configurer pour renvoyer des transcriptions alternatives, consultez [Génération de transcriptions alternatives](#).

Pour plus d'informations sur la façon dont les chiffres et les mesures médicales apparaissent dans la sortie de transcription, voir [Transcrire des numéros](#) et [Transcription de termes et de mesures médicaux](#).

Rubriques

- [Transcription d'un fichier audio d'une conversation médicale](#)
- [Transcription d'une conversation médicale dans un flux en temps réel](#)
- [Activation du partitionnement des haut-parleurs](#)
- [Transcription audio multicanal](#)

Transcription d'un fichier audio d'une conversation médicale

Utilisez une tâche de transcription par lots pour transcrire les fichiers audio des conversations médicales. Vous pouvez l'utiliser pour transcrire un dialogue clinicien-patient. Vous pouvez démarrer une tâche de transcription par lots dans l'[StartMedicalTranscriptionJob](#) API ou dans le AWS Management Console.

Lorsque vous démarrez une tâche de transcription médicale avec l'[StartMedicalTranscriptionJob](#) API, vous spécifiez PRIMARYCARE la valeur du Specialty paramètre.

AWS Management Console

Pour transcrire un dialogue clinicien-patient (AWS Management Console)

Pour utiliser le AWS Management Console pour transcrire un dialogue clinicien-patient, créez une tâche de transcription et choisissez Conversation pour le type d'entrée audio.

1. Connectez-vous à [AWS Management Console](#).
2. Dans le panneau de navigation, sous Amazon Transcribe Médecine, sélectionnez Tâches de transcription.
3. Choisissez Create job (Créer une tâche).
4. Sur la page Spécifier les détails de la Job, sous Paramètres de la tâche, spécifiez les informations suivantes.
 - a. Nom : nom de la tâche de transcription.
 - b. Type d'entrée audio — Conversation
5. Pour les autres champs, spécifiez l'Amazon S3 emplacement de votre fichier audio et l'endroit où vous souhaitez stocker la sortie de votre travail de transcription.
6. Choisissez Suivant.
7. Sélectionnez Create (Créer).

API

Pour transcrire une conversation médicale à l'aide d'une tâche de transcription par lots (API)

- Pour l'[StartMedicalTranscriptionJob](#) API, spécifiez ce qui suit.
 - a. Pour `MedicalTranscriptionJobName`, spécifiez un nom unique dans votre Compte AWS.
 - b. Pour `LanguageCode`, spécifiez le code de langue qui correspond à la langue parlée dans votre fichier audio et à la langue de votre filtre de vocabulaire.
 - c. Pour le `MediaFileUri` paramètre de l'`Media` objet, spécifiez le nom du fichier audio que vous souhaitez transcrire.

- d. Pour `Specialty`, spécifiez la spécialité médicale du clinicien qui parle dans le fichier audio sous la forme `PRIMARYCARE`.
- e. Pour `Type`, spécifiez `CONVERSATION`.
- f. Pour `OutputBucketName`, spécifiez le Amazon S3 compartiment dans lequel stocker les résultats de transcription.

Voici un exemple de demande qui utilise le AWS SDK for Python (Boto3) pour transcrire une conversation médicale entre un clinicien de la `PRIMARYCARE` spécialité et un patient.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-med-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-audio-file.flac"
transcribe.start_medical_transcription_job(
    MedicalTranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'output-files/',
    LanguageCode = 'en-US',
    Specialty = 'PRIMARYCARE',
    Type = 'CONVERSATION'
)

while True:
    status = transcribe.get_medical_transcription_job(MedicalTranscriptionJobName =
job_name)
    if status['MedicalTranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED',
'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

L'exemple de code suivant montre les résultats de transcription d'une conversation clinicien-patient.


```
{
  "jobName": "conversation-medical-transcription-job",
  "accountId": "111122223333",
  "results": {
    "transcripts": [
      {
        "transcript": "... come for a follow up visit today..."
      }
    ],
    "items": [
      {
        ...
        "start_time": "4.85",
        "end_time": "5.12",
        "alternatives": [
          {
            "confidence": "1.0",
            "content": "come"
          }
        ],
        "type": "pronunciation"
      },
      {
        "start_time": "5.12",
        "end_time": "5.29",
        "alternatives": [
          {
            "confidence": "1.0",
            "content": "for"
          }
        ],
        "type": "pronunciation"
      },
      {
        "start_time": "5.29",
        "end_time": "5.33",
        "alternatives": [
          {
            "confidence": "0.9955",
            "content": "a"
          }
        ],
      },
    ]
  }
}
```

```
        "type": "pronunciation"
    },
    {
        "start_time": "5.33",
        "end_time": "5.66",
        "alternatives": [
            {
                "confidence": "0.9754",
                "content": "follow"
            }
        ],
        "type": "pronunciation"
    },
    {
        "start_time": "5.66",
        "end_time": "5.75",
        "alternatives": [
            {
                "confidence": "0.9754",
                "content": "up"
            }
        ],
        "type": "pronunciation"
    },
    {
        "start_time": "5.75",
        "end_time": "6.02",
        "alternatives": [
            {
                "confidence": "1.0",
                "content": "visit"
            }
        ]
    }
    ...
},
"status": "COMPLETED"
}
```

AWS CLI

Pour transcrire une conversation médicale à l'aide d'une tâche de transcription par lots (AWS CLI)

- Exécutez le code suivant.

```
aws transcribe start-medical-transcription-job \  
--region us-west-2 \  
--cli-input-json file://example-start-command.json
```

Le code suivant affiche le contenu de `example-start-command.json`.

```
{  
  "MedicalTranscriptionJobName": "my-first-med-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-audio-file.flac"  
  },  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "OutputKey": "my-output-files/",  
  "LanguageCode": "en-US",  
  "Specialty": "PRIMARYCARE",  
  "Type": "CONVERSATION"  
}
```

Transcription d'une conversation médicale dans un flux en temps réel

Vous pouvez transcrire le flux audio d'une conversation médicale à l'aide du protocole HTTP/2 ou [WebSocket](#) des protocoles. Pour plus d'informations sur la façon de démarrer un flux à l'aide WebSocket du protocole, consultez [Configuration d'un WebSocket stream](#). Pour démarrer un flux HTTP/2, utilisez l'[StartMedicalStreamTranscriptionAPI](#).

Vous pouvez transcrire de l'audio en streaming dans les spécialités médicales suivantes :

- Cardiologie
- Neurologie
- Oncologie

- Soins primaires
- Urologie

Chaque spécialité médicale comprend de nombreux types de procédures et de rendez-vous. Les cliniciens dictent donc de nombreux types de notes différents. Utilisez les exemples suivants pour vous aider à spécifier la valeur du paramètre `specialty` URI de la WebSocket demande ou du `specialty` paramètre de l'[StartMedicalStreamTranscriptionAPI](#) :

- Pour les consultations d'électrophysiologie ou d'échocardiographie, choisissez `CARDIOLOGY`.
- Pour les consultations en oncologie médicale, en oncologie chirurgicale ou en radio-oncologie, choisissez `ONCOLOGY`.
- Pour un médecin qui consulte un patient ayant subi un accident vasculaire cérébral, soit un accident ischémique transitoire, soit un accident vasculaire cérébral, choisissez `NEUROLOGY`.
- Pour une consultation sur l'incontinence urinaire, choisissez `UROLOGY`.
- Pour un examen annuel ou des visites de soins d'urgence, choisissez `PRIMARYCARE`.
- Pour les visites en milieu hospitalier, choisissez `PRIMARYCARE`.
- Pour les consultations concernant la fertilité, la ligature des trompes, l'insertion d'un DIU ou l'avortement, choisissez `PRIMARYCARE`.

AWS Management Console

Pour transcrire une conversation médicale en streaming (AWS Management Console)

AWS Management Console Pour transcrire un dialogue clinicien-patient en temps réel, choisissez l'option permettant de transcrire une conversation médicale, de démarrer la diffusion et de commencer à parler dans le microphone.

1. Connectez-vous à [AWS Management Console](#).
2. Dans le panneau de navigation, sous Amazon Transcribe Médecine, sélectionnez Transcription en temps réel.
3. Choisissez Conversation.
4. Pour Spécialité médicale, choisissez la spécialité du clinicien.
5. Choisissez Start streaming (Démarrer la diffusion).
6. Parlez dans le micro.

Transcription d'une conversation médicale dans un flux HTTP/2

Voici la syntaxe des paramètres d'une requête HTTP/2.

Pour transcrire le flux HTTP/2 d'une conversation médicale, utilisez l'[StartMedicalStreamTranscription](#) API et spécifiez les informations suivantes :

- **LanguageCode**— Le code de langue. La valeur valide est en-US
- **MediaEncoding**— Le codage utilisé pour le signal audio d'entrée. Les valeurs valides sont pcm, ogg-opus et flac.
- **Specialty**— La spécialité du professionnel de santé.
- **Type** – CONVERSATION

Pour améliorer la précision de la transcription de termes spécifiques dans un flux en temps réel, utilisez un vocabulaire personnalisé. Pour activer un vocabulaire personnalisé, définissez la valeur du `VocabularyName` paramètre sur le nom du vocabulaire personnalisé que vous souhaitez utiliser. Pour plus d'informations, veuillez consulter [Améliorer la précision de la transcription grâce à des vocabulaires médicaux personnalisés](#).

Pour étiqueter le discours de différents haut-parleurs, réglez le `ShowSpeakerLabel` paramètre sur `true`. Pour plus d'informations, veuillez consulter [Activation du partitionnement des haut-parleurs](#).

Pour plus d'informations sur la configuration d'un flux HTTP/2 pour transcrire une conversation médicale, consultez [Configuration d'un flux HTTP/2](#).

Transcrire une conversation médicale dans un WebSocket stream

Vous pouvez utiliser une WebSocket demande pour transcrire une conversation médicale. Lorsque vous faites une WebSocket demande, vous créez un URI présigné. Cet URI contient les informations nécessaires pour configurer le flux audio entre votre application et Amazon Transcribe Medical. Pour plus d'informations sur la création de WebSocket demandes, consultez [Configuration d'un WebSocket stream](#).

Utilisez le modèle suivant pour créer votre URI pré-signé.

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/medical-stream-  
transcription-websocket  
?language-code=LanguageCode  
&X-Amz-Algorithm=AWS4-HMAC-SHA256
```

```
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-  
west-2%2Ftranscribe%2Faws4_request  
&X-Amz-Date=20220208T235959Z  
&X-Amz-Expires=300  
&X-Amz-Security-Token=security-token  
&X-Amz-Signature=Signature Version 4 signature  
&X-Amz-SignedHeaders=host  
&media-encoding=flac  
&sample-rate=16000  
&session-id=sessionId  
&specialty=medicalSpecialty  
&type=CONVERSATION  
&vocabulary-name=vocabularyName  
&show-speaker-label=boolean
```

Pour améliorer la précision de la transcription de termes spécifiques dans un flux en temps réel, utilisez un vocabulaire personnalisé. Pour activer un vocabulaire personnalisé, définissez la valeur `vocabulary-name` de sur le nom du vocabulaire personnalisé que vous souhaitez utiliser. Pour plus d'informations, veuillez consulter [Améliorer la précision de la transcription grâce à des vocabulaires médicaux personnalisés](#).

Pour étiqueter le discours de différents haut-parleurs, définissez le `show-speaker-label` paramètre sur `true`. Pour plus d'informations, veuillez consulter [Activation du partitionnement des haut-parleurs](#).

Pour plus d'informations sur la création d'URI pré-signés, consultez [Configuration d'un WebSocket stream](#).

Activation du partitionnement des haut-parleurs

Pour activer le partitionnement des haut-parleurs dans Amazon Transcribe Medical, utilisez la diarisation des haut-parleurs. Cela vous permet de voir ce que le patient a dit et ce que le clinicien a dit dans la sortie de transcription.

Lorsque vous activez la diarisation des Amazon Transcribe locuteurs, Medical attribue à chaque locuteur un identifiant unique pour chaque locuteur. Un énoncé est une unité de parole qui est généralement séparée des autres énoncés par le silence. Dans la transcription par lots, un énoncé du clinicien peut recevoir une étiquette `pk_0` et un énoncé dont le patient peut recevoir une étiquette `pk_1`.

Si un énoncé d'un locuteur se chevauche avec un énoncé d'un autre locuteur, leAmazon Transcribe médecin les ordonne dans la transcription en fonction de leur heure de début. Les énoncés qui se chevauchent dans le flux audio d'entrée ne se chevauchent pas dans la transcription de sortie.

Vous pouvez activer la diarisation des haut-parleurs lorsque vous transcrivez un fichier audio à l'aide d'une tâche de transcription par lots ou dans le cadre d'un flux en temps réel.

Rubriques

- [Activation du partitionnement des haut-parleurs dans les transcriptions par lots](#)
- [Permettre le partitionnement des haut-parleurs dans les flux en temps réel](#)

Activation du partitionnement des haut-parleurs dans les transcriptions par lots

Vous pouvez activer le partitionnement des haut-parleurs dans une tâche de transcription par lots à l'aide de l'[StartMedicalTranscriptionJob](#)API ou duAWS Management Console. Cela vous permet de partitionner le texte par locuteur dans une conversation clinicien-patient et de déterminer qui a dit quoi dans la sortie de transcription.

AWS Management Console

AWS Management ConsolePour activer la diarisation des haut-parleurs dans votre tâche de transcription, vous devez activer l'identification audio, puis le partitionnement des haut-parleurs.

1. Connectez-vous à [AWS Management Console](#).
2. Dans le panneau de navigation, sousAmazon Transcribe Médecine, sélectionnez Tâches de transcription.
3. Choisissez Create job (Créer une tâche).
4. Sur la page Spécifier les détails de la tâche, fournissez des informations sur votre tâche de transcription.
5. Choisissez Suivant.
6. Activez l'identification audio.
7. Dans le champ Type d'identification audio, sélectionnez Partitionnement des haut-parleurs.
8. Dans Nombre maximum de haut-parleurs, entrez le nombre maximum de haut-parleurs que vous pensez être en train de parler dans votre fichier audio.
9. Sélectionnez Create (Créer).

API

Pour activer le partitionnement des haut-parleurs à l'aide d'une tâche de transcription par lots (API)

- Pour l'[StartMedicalTranscriptionJob](#) API, spécifiez ce qui suit.
 - a. Pour `MedicalTranscriptionJobName`, spécifiez un nom unique dans votre Compte AWS.
 - b. Pour `LanguageCode`, spécifiez le code de langue correspondant à la langue parlée dans le fichier audio.
 - c. Pour le `MediaFileUri` paramètre de l'objet `Media`, spécifiez le nom du fichier audio que vous souhaitez transcrire.
 - d. Pour `Specialty`, spécifiez la spécialité médicale du clinicien qui parle dans le fichier audio.
 - e. Pour `Type`, spécifiez `CONVERSATION`.
 - f. Pour `OutputBucketName`, spécifiez le Amazon S3 compartiment dans lequel stocker les résultats de transcription.
 - g. Pour l'objet `Settings`, spécifiez ce qui suit.
 - i. `ShowSpeakerLabels = true`.
 - ii. `MaxSpeakerLabels`— Un entier compris entre 2 et 10 pour indiquer le nombre de haut-parleurs qui, selon vous, parlent dans votre audio.

La demande suivante utilise le AWS SDK for Python (Boto3) pour démarrer une tâche de transcription par lots d'un dialogue entre un clinicien de soins primaires et un patient avec le partitionnement des haut-parleurs activé.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
transcribe.start_medical_transcription_job(
    MedicalTranscriptionJobName = job_name,
    Media={
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
```



```
OutputKey = 'my-output-files/',
LanguageCode = 'en-US',
Specialty = 'PRIMARYCARE',
Type = 'CONVERSATION',
OutputBucketName = 'DOC-EXAMPLE-BUCKET',
Settings = {'ShowSpeakerLabels': True,
           'MaxSpeakerLabels': 2
          }
)
while True:
    status = transcribe.get_medical_transcription_job(MedicalTranscriptionJobName =
job_name)
    if status['MedicalTranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED',
'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

L'exemple de code suivant montre les résultats de transcription d'une tâche de transcription lorsque le partitionnement des haut-parleurs est activé.

```
{
  "jobName": "job ID",
  "accountId": "111122223333",
  "results": {
    "transcripts": [
      {
        "transcript": "Professional answer."
      }
    ],
    "speaker_labels": {
      "speakers": 1,
      "segments": [
        {
          "start_time": "0.000000",
          "speaker_label": "spk_0",
          "end_time": "1.430",
          "items": [
            {
```

```
        "start_time": "0.100",
        "speaker_label": "spk_0",
        "end_time": "0.690"
    },
    {
        "start_time": "0.690",
        "speaker_label": "spk_0",
        "end_time": "1.210"
    }
]
}
],
},
"items": [
    {
        "start_time": "0.100",
        "end_time": "0.690",
        "alternatives": [
            {
                "confidence": "0.8162",
                "content": "Professional"
            }
        ],
        "type": "pronunciation"
    },
    {
        "start_time": "0.690",
        "end_time": "1.210",
        "alternatives": [
            {
                "confidence": "0.9939",
                "content": "answer"
            }
        ],
        "type": "pronunciation"
    },
    {
        "alternatives": [
            {
                "content": "."
            }
        ],
        "type": "punctuation"
    }
]
```

```
    ]
  },
  "status": "COMPLETED"
}
```

AWS CLI

Pour transcrire un fichier audio d'une conversation entre un clinicien pratiquant des soins primaires et un patient (AWS CLI)

- Exécutez le code suivant.

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--cli-input-json file://example-start-command.json
```

Le code suivant affiche le contenu de `example-start-command.json`.

```
{
  "MedicalTranscriptionJobName": "my-first-med-transcription-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-audio-file.flac"
  },
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "OutputKey": "my-output-files/",
  "LanguageCode": "en-US",
  "Specialty": "PRIMARYCARE",
  "Type": "CONVERSATION",
  "Settings":{
    "ShowSpeakerLabels": true,
    "MaxSpeakerLabels": 2
  }
}
```

Permettre le partitionnement des haut-parleurs dans les flux en temps réel

Pour partitionner les haut-parleurs et étiqueter leur discours dans un flux en temps réel, utilisez la [AWS Management Console](#) ou une requête de streaming. Le partitionnement des haut-parleurs fonctionne mieux pour deux à cinq haut-parleurs dans un flux. Bien que [Amazon Transcribe Medical](#) puisse partitionner plus de cinq haut-parleurs dans un flux, la précision des partitions diminue si vous dépassez ce nombre.

Pour démarrer une requête HTTP/2, utilisez l'[StartMedicalStreamTranscription](#) API.

Pour lancer une WebSocket demande, utilisez un URI pré-signé. L'URI contient les informations nécessaires pour configurer une communication bidirectionnelle entre votre application et [Amazon Transcribe Medical](#).

Activation du partitionnement des haut-parleurs dans les sons émis par votre microphone (AWS Management Console)

Vous pouvez utiliser le [AWS Management Console](#) pour démarrer une diffusion en temps réel d'une conversation entre le clinicien et le patient, ou une dictée prononcée dans votre microphone en temps réel.

1. Connectez-vous à [AWS Management Console](#).
2. Dans le volet de navigation, pour [Amazon Transcribe Médical](#), sélectionnez Transcription en temps réel.
3. Dans Type d'entrée audio, choisissez le type de discours médical que vous souhaitez transcrire.
4. Pour des paramètres supplémentaires, choisissez Partitionnement des haut-parleurs.
5. Choisissez Démarrer la diffusion pour commencer à transcrire votre audio en temps réel.
6. Parlez dans le micro.

Activation du partitionnement des haut-parleurs dans un flux HTTP/2

Pour activer le partitionnement des haut-parleurs dans le flux HTTP/2 d'une conversation médicale, utilisez l'[StartMedicalStreamTranscription](#) API et spécifiez les informations suivantes :

- Pour `LanguageCode`, spécifiez le code de langue correspondant à la langue du flux. La valeur valide est en-US.
- Pour `MediaSampleHertz`, spécifiez la fréquence d'échantillonnage de l'audio.
- Pour `Specialty`, précisez la spécialité médicale du prestataire.

- `ShowSpeakerLabel – true`

Pour plus d'informations sur la configuration d'un flux HTTP/2 pour transcrire une conversation médicale, consultez [Configuration d'un flux HTTP/2](#).

Activation du partitionnement des haut-parleurs dans une WebSocket demande

Pour partitionner les haut-parleurs en WebSocket flux à l'aide de l'API, utilisez le format suivant pour créer un URI pré-signé afin de démarrer une WebSocket demande et de `show-speaker-label` définir sur `true`.

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/medical-stream-
transcription-websocket
?language-code=languageCode
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-
west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=Signature Version 4 signature
&X-Amz-SignedHeaders=host
&media-encoding=flac
&sample-rate=16000
&session-id=sessionId
&specialty=medicalSpecialty
&type=CONVERSATION
&vocabulary-name=vocabularyName
&show-speaker-label=boolean
```

Le code suivant montre l'exemple de réponse tronqué d'une demande de streaming.

```
{
  "Transcript": {
    "Results": [
      {
        "Alternatives": [
          {
            "Items": [
```

```
{
  "Confidence": 0.97,
  "Content": "From",
  "EndTime": 18.98,
  "Speaker": "0",
  "StartTime": 18.74,
  "Type": "pronunciation",
  "VocabularyFilterMatch": false
},
{
  "Confidence": 1,
  "Content": "the",
  "EndTime": 19.31,
  "Speaker": "0",
  "StartTime": 19,
  "Type": "pronunciation",
  "VocabularyFilterMatch": false
},
{
  "Confidence": 1,
  "Content": "last",
  "EndTime": 19.86,
  "Speaker": "0",
  "StartTime": 19.32,
  "Type": "pronunciation",
  "VocabularyFilterMatch": false
},
...
{
  "Confidence": 1,
  "Content": "chronic",
  "EndTime": 22.55,
  "Speaker": "0",
  "StartTime": 21.97,
  "Type": "pronunciation",
  "VocabularyFilterMatch": false
},
...
  "Confidence": 1,
  "Content": "fatigue",
  "EndTime": 24.42,
  "Speaker": "0",
  "StartTime": 23.95,
  "Type": "pronunciation",
```

```

        "VocabularyFilterMatch": false
    },
    {
        "EndTime": 25.22,
        "StartTime": 25.22,
        "Type": "speaker-change",
        "VocabularyFilterMatch": false
    },
    {
        "Confidence": 0.99,
        "Content": "True",
        "EndTime": 25.63,
        "Speaker": "1",
        "StartTime": 25.22,
        "Type": "pronunciation",
        "VocabularyFilterMatch": false
    },
    {
        "Content": ".",
        "EndTime": 25.63,
        "StartTime": 25.63,
        "Type": "punctuation",
        "VocabularyFilterMatch": false
    }
],
    "Transcript": "From the last note she still has mild sleep deprivation and
chronic fatigue True."
}
],
"EndTime": 25.63,
"IsPartial": false,
"ResultId": "XXXXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX",
"StartTime": 18.74
}
]
}
}

```

Amazon TranscribeMedical interrompt votre flux audio entrant en fonction de segments vocaux naturels, tels qu'un changement de haut-parleur ou une pause dans le son. La transcription est renvoyée progressivement à votre application, chaque réponse contenant plus de paroles transcrites que la précédente, jusqu'à la transcription de la totalité du segment. Le code précédent

est un exemple tronqué de segment vocal entièrement transcrit. Les étiquettes des haut-parleurs apparaissent uniquement pour les segments entièrement transcrits.

La liste suivante montre l'organisation des objets et des paramètres dans une sortie de transcription en continu.

Transcript

Chaque segment vocal possède son propre `Transcript` objet.

Results

Chaque `Transcript` objet possède son propre `Results` objet. Cet objet contient le `isPartial` champ. Lorsque sa valeur est égale à `false`, les résultats renvoyés concernent un segment vocal complet.

Alternatives

Chaque `Results` objet possède un `Alternatives` objet.

Items

Chaque `Alternatives` objet possède son propre `Items` objet qui contient des informations sur chaque mot et chaque signe de ponctuation dans la sortie de transcription. Lorsque vous activez le partitionnement des haut-parleurs, chaque mot possède une `Speaker` étiquette pour les segments vocaux entièrement transcrits. Amazon Transcribe Medical utilise cette étiquette pour attribuer un entier unique à chaque haut-parleur du flux. Le `Type` paramètre ayant une valeur de `speaker-change` indique qu'une personne a cessé de parler et qu'une autre personne est sur le point de commencer.

Transcript

Chaque objet `Items` contient un segment vocal transcrit comme valeur du `Transcript` champ.

Pour plus d'informations sur WebSocket les demandes, consultez [Configuration d'un WebSocket stream](#).

Transcription audio multicanal

Si vous disposez d'un fichier ou d'un flux audio comportant plusieurs canaux, vous pouvez utiliser l'identification des canaux pour transcrire le discours provenant de chacun de ces canaux. Amazon Transcribe Medical transcrit le discours de chaque canal séparément. Il combine les transcriptions distinctes de chaque canal en une seule sortie de transcription.

Utilisez l'identification des canaux pour identifier les différents canaux de votre audio et transcrire le discours provenant de chacun de ces canaux. Activez cette option dans des situations telles que le scénario d'un appelant et d'un agent. Utilisez-le pour distinguer un appelant d'un agent dans les enregistrements ou les flux provenant de centres d'appels chargés de la surveillance de la sécurité des médicaments.

Vous pouvez activer l'identification des canaux pour le traitement par lots et le streaming en temps réel. La liste suivante explique comment l'activer pour chaque méthode.

- Transcription Batch —AWS Management Console et [StartMedicalTranscriptionJobAPI](#)
- Transcription en WebSocket continu : streaming et [StartMedicalStreamTranscriptionAPI](#)

Transcription de fichiers audio multicanaux

Lorsque vous transcrivez un fichier audio, Amazon Transcribe Medical renvoie une liste d'éléments pour chaque canal. Un élément est un mot transcrit ou un signe de ponctuation. Chaque mot a une heure de début et une heure de fin. Si une personne sur une chaîne parle à une autre personne sur une chaîne distincte, les heures de début et de fin des éléments de chaque chaîne se chevauchent pendant que les personnes parlent l'une sur l'autre.

Par défaut, vous pouvez transcrire des fichiers audio sur deux canaux. Vous pouvez demander une augmentation de quota si vous devez transcrire des fichiers comportant plus de deux canaux. Pour plus d'informations sur la demande d'augmentation de quota, consultez la section [Service AWSQuotas](#).

Pour transcrire du son multicanal dans une tâche de transcription par lots, utilisez l'[StartMedicalTranscriptionJobAPI](#) AWS Management Console ou.

AWS Management Console

Pour utiliser le AWS Management Console pour activer l'identification des canaux dans votre tâche de transcription par lots, vous devez activer l'identification audio, puis l'identification des canaux. L'identification des canaux est un sous-ensemble de l'identification audio dans le AWS Management Console.

1. Connectez-vous à [AWS Management Console](#).
2. Dans le panneau de navigation, sous Amazon Transcribe Médecine, sélectionnez Tâches de transcription.
3. Choisissez Create job (Créer une tâche).

4. Sur la page Spécifier les détails de la tâche, fournissez des informations sur votre tâche de transcription.
5. Choisissez Suivant.
6. Activez l'identification audio.
7. Pour Type d'identification audio, choisissez Identification du canal.
8. Sélectionnez Create (Créer).

API

Pour transcrire un fichier audio multicanal (API)

- Pour l'[StartMedicalTranscriptionJob](#) API, spécifiez ce qui suit.
 - a. Pour `TranscriptionJobName`, spécifiez un nom propre à votre Compte AWS.
 - b. Pour `LanguageCode`, spécifiez le code de langue correspondant à la langue parlée dans le fichier audio. La valeur valide est `en-US`.
 - c. Pour le `MediaFileUri` paramètre de l'objet `Media`, spécifiez le nom du fichier multimédia que vous souhaitez transcrire.
 - d. Pour l'objet `Settings`, définissez `ChannelIdentification` sur `true`.

Voici un exemple de demande utilisant le AWS SDK for Python (Boto3).

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-transcription-job"
job_name = "my-first-med-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
transcribe.start_medical_transcription_job(
    MedicalTranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'output-files/',
    LanguageCode = 'en-US',
```

```

    Specialty = 'PRIMARYCARE',
    Type = 'CONVERSATION',
    Settings = {
        'ChannelIdentification': True
    }
)
while True:
    status = transcribe.get_transcription_job(MedicalTranscriptionJobName = job_name)
    if status['MedicalTranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED',
'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)

```

AWS CLI

Pour transcrire un fichier audio multicanal à l'aide d'une tâche de transcription par lots (AWS CLI)

- Exécutez le code suivant.

```

aws transcribe start-medical-transcription-job \
--region us-west-2 \
--cli-input-json file://example-start-command.json

```

Ce qui suit est le code de *example-start-command.json*.

```

{
    "MedicalTranscriptionJobName": "my-first-med-transcription-job",
    "Media": {
        "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-audio-
file.flac"
    },
    "OutputBucketName": "DOC-EXAMPLE-BUCKET",
    "OutputKey": "my-output-files/",
    "LanguageCode": "en-US",
    "Specialty": "PRIMARYCARE",
    "Type": "CONVERSATION",
}

```

```
    "Settings":{
      "ChannelIdentification": true
    }
  }
```

Le code suivant montre la sortie de transcription d'un fichier audio contenant une conversation sur deux canaux.

```
{
  "jobName": "job id",
  "accountId": "111122223333",
  "results": {
    "transcripts": [
      {
        "transcript": "When you try ... It seems to ..."
      }
    ],
    "channel_labels": {
      "channels": [
        {
          "channel_label": "ch_0",
          "items": [
            {
              "start_time": "12.282",
              "end_time": "12.592",
              "alternatives": [
                {
                  "confidence": "1.0000",
                  "content": "When"
                }
              ],
              "type": "pronunciation"
            },
            {
              "start_time": "12.592",
              "end_time": "12.692",
              "alternatives": [
                {
                  "confidence": "0.8787",
                  "content": "you"
                }
              ]
            }
          ]
        }
      ]
    }
  }
}
```

```
    ],
    "type": "pronunciation"
  },
  {
    "start_time": "12.702",
    "end_time": "13.252",
    "alternatives": [
      {
        "confidence": "0.8318",
        "content": "try"
      }
    ],
    "type": "pronunciation"
  },
  ...
]
},
{
  "channel_label": "ch_1",
  "items": [
    {
      "start_time": "12.379",
      "end_time": "12.589",
      "alternatives": [
        {
          "confidence": "0.5645",
          "content": "It"
        }
      ],
      "type": "pronunciation"
    },
    {
      "start_time": "12.599",
      "end_time": "12.659",
      "alternatives": [
        {
          "confidence": "0.2907",
          "content": "seems"
        }
      ],
      "type": "pronunciation"
    },
    {
      "start_time": "12.669",
```

```
        "end_time": "13.029",
        "alternatives": [
          {
            "confidence": "0.2497",
            "content": "to"
          }
        ],
        "type": "pronunciation"
      },
      ...
    ]
  }
}
```

Transcription de flux audio multicanaux

Vous pouvez transcrire du son provenant de canaux distincts en HTTP/2 ou en WebSocket flux à l'aide de l'[StartMedicalStreamTranscriptionAPI](#).

Par défaut, vous pouvez transcrire des flux avec deux canaux. Vous pouvez demander une augmentation de quota si vous devez transcrire des flux comportant plus de deux canaux. Pour plus d'informations sur la demande d'augmentation de quota, consultez la section [QuotasAWS de service](#).

Transcription d'un son multicanal dans un flux HTTP/2

Pour transcrire du son multicanal dans un flux HTTP/2, utilisez l'[StartMedicalStreamTranscriptionAPI](#) et spécifiez les informations suivantes :

- **LanguageCode**— Code de langue de l'élément audio. La valeur valide est en-US.
- **MediaEncoding**— L'encodage de l'audio. Les valeurs valides sont ogg-opus, flac et pcm.
- **EnableChannelIdentification** – true
- **NumberOfChannels**— le nombre de chaînes dans votre flux audio en streaming.

Pour plus d'informations sur la configuration d'un flux HTTP/2 pour transcrire une conversation médicale, consultez [Configuration d'un flux HTTP/2](#).

Transcription de données audio multicanales dans un WebSocket flux

Pour partitionner les haut-parleurs en WebSocket flux, utilisez le format suivant pour créer un URI pré-signé et lancer une WebSocket demande. Spécifiez `enable-channel-identification` le

nombre `true` et le nombre de chaînes de votre stream dans `number-of-channels`. Un URI pré-signé contient les informations nécessaires pour établir une communication bidirectionnelle entre votre application et Amazon Transcribe Medical.

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/medical-stream-
transcription-websocket
?language-code=languageCode
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-
west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=Signature Version 4 signature
&X-Amz-SignedHeaders=host
&media-encoding=flac
&sample-rate=16000
&session-id=sessionId
&enable-channel-identification=true
&number-of-channels=2
```

Les définitions de paramètres se trouvent dans la [référence de l'API](#) ; les paramètres communs à toutes les opérations d'AWSAPI sont répertoriés dans la section [Paramètres communs](#).

Pour plus d'informations sur WebSocket les demandes, consultez [Configuration d'un WebSocket stream](#).

Sortie de streaming multicanal

La sortie d'une transcription en continu est la même pour HTTP/2 et les WebSocket requêtes. Voici un exemple de sortie.

```
{
  "resultId": "XXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXX",
  "startTime": 0.11,
  "endTime": 0.66,
  "isPartial": false,
  "alternatives": [
    {
      "transcript": "Left.",
```

```
    "items": [
      {
        "startTime": 0.11,
        "endTime": 0.45,
        "type": "pronunciation",
        "content": "Left",
        "vocabularyFilterMatch": false
      },
      {
        "startTime": 0.45,
        "endTime": 0.45,
        "type": "punctuation",
        "content": ".",
        "vocabularyFilterMatch": false
      }
    ]
  },
  "channelId": "ch_0"
}
```

Pour chaque segment de discours, un `channelId` drapeau indique à quel canal appartient le discours.

Transcrire une dictée médicale

Vous pouvez utiliser Amazon Transcribe Medical pour transcrire les notes médicales dictées par le clinicien à l'aide d'une tâche de transcription par lots ou d'un flux en temps réel. Les tâches de transcription Batch vous permettent de transcrire des fichiers audio. Vous spécifiez la spécialité médicale du clinicien dans votre travail ou votre flux de transcription afin de garantir que Amazon Transcribe Medical produit des résultats de transcription avec la plus grande précision possible.

Vous pouvez transcrire une dictée médicale dans les spécialités suivantes :

- Cardiologie — disponible en transcription en streaming uniquement
- Neurologie — disponible en transcription en streaming uniquement
- Oncologie — disponible en transcription en streaming uniquement
- Soins primaires — incluent les types de pratique médicale suivants :
 - Médecine familiale
 - Médecine interne

- Obstétrique et gynécologie (OB-GYN)
- Pédiatrie
- Radiologie — disponible en transcription en streaming uniquement
- Urologie — disponible en transcription en streaming uniquement

Vous pouvez améliorer la précision de la transcription en utilisant des vocabulaires personnalisés. Pour plus d'informations sur le fonctionnement des vocabulaires médicaux personnalisés, voir [Améliorer la précision de la transcription grâce à des vocabulaires médicaux personnalisés](#).

Par défaut, Amazon Transcribe Medical renvoie la transcription avec le niveau de confiance le plus élevé. Si vous souhaitez le configurer pour renvoyer des transcriptions alternatives, consultez [Génération de transcriptions alternatives](#).

Pour plus d'informations sur la façon dont les chiffres et les mesures médicales apparaissent dans la sortie de transcription, reportez-vous [Transcrire des numéros](#) aux sections et [Transcription de termes et de mesures médicaux](#).

Rubriques

- [Transcription d'un fichier audio d'une dictée médicale](#)
- [Transcription d'une dictée médicale dans un flux en temps réel](#)

Transcription d'un fichier audio d'une dictée médicale

Utilisez une tâche de transcription par lots pour transcrire les fichiers audio des conversations médicales. Vous pouvez l'utiliser pour transcrire un dialogue clinicien-patient. Vous pouvez démarrer une tâche de transcription par lots dans l'[StartMedicalTranscriptionJob](#) API ou dans le AWS Management Console.

Lorsque vous démarrez une tâche de transcription médicale avec l'[StartMedicalTranscriptionJob](#) API, vous spécifiez PRIMARYCARE la valeur du Specialty paramètre.

AWS Management Console

Pour transcrire un dialogue clinicien-patient (AWS Management Console)

Pour utiliser le AWS Management Console pour transcrire un dialogue clinicien-patient, créez une tâche de transcription et choisissez Conversation pour le type d'entrée audio.

1. Connectez-vous à [AWS Management Console](#).
2. Dans le panneau de navigation, sous Amazon Transcribe Medical, sélectionnez Tâches de transcription.
3. Choisissez Create job (Créer une tâche).
4. Sur la page Spécifier les détails de la Job, sous Paramètres de la tâche, spécifiez les informations suivantes.
 - a. Nom : nom de la tâche de transcription.
 - b. Type d'entrée audio : dictée
5. Pour les autres champs, spécifiez l'Amazon S3 emplacement de votre fichier audio et l'endroit où vous souhaitez stocker la sortie de votre travail de transcription.
6. Choisissez Suivant.
7. Sélectionnez Create (Créer).

API

Pour transcrire une conversation médicale à l'aide d'une tâche de transcription par lots (API)

- Pour l'[StartMedicalTranscriptionJob](#) API, spécifiez ce qui suit.
 - a. Pour `MedicalTranscriptionJobName`, spécifiez un nom unique dans votre Compte AWS.
 - b. Pour `LanguageCode`, spécifiez le code de langue qui correspond à la langue parlée dans votre fichier audio et à la langue de votre filtre de vocabulaire.
 - c. Dans le `MediaFileUri` paramètre de l'`Media` objet, spécifiez le nom du fichier audio que vous souhaitez transcrire.
 - d. Pour `Specialty`, spécifiez la spécialité médicale du clinicien qui parle dans le fichier audio.
 - e. Pour `Type`, spécifiez `DICTION`.
 - f. Pour `OutputBucketName`, spécifiez le Amazon S3 compartiment dans lequel stocker les résultats de transcription.

Voici un exemple de demande qui utilise le AWS SDK for Python (Boto3) pour transcrire la dictée médicale d'un clinicien de la PRIMARYCARE spécialité.

```
from __future__ import print_function
```

```
import time
import boto3
transcribe = boto3.client('transcribe')
job_name = "my-first-med-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-audio-file.flac"
transcribe.start_medical_transcription_job(
    MedicalTranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'my-output-files/',
    LanguageCode = 'en-US',
    Specialty = 'PRIMARYCARE',
    Type = 'DICTATION'
)
while True:
    status = transcribe.get_medical_transcription_job(MedicalTranscriptionJobName =
job_name)
    if status['MedicalTranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED',
'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

L'exemple de code suivant montre les résultats de transcription d'une dictée médicale.

```
{
  "jobName": "dictation-medical-transcription-job",
  "accountId": "111122223333",
  "results": {
    "transcripts": [
      {
        "transcript": "... came for a follow up visit today..."
      }
    ],
    "items": [
      {
        ...
      }
    ]
  }
}
```

```
    "start_time": "4.85",
    "end_time": "5.12",
    "alternatives": [
      {
        "confidence": "1.0",
        "content": "came"
      }
    ],
    "type": "pronunciation"
  },
  {
    "start_time": "5.12",
    "end_time": "5.29",
    "alternatives": [
      {
        "confidence": "1.0",
        "content": "for"
      }
    ],
    "type": "pronunciation"
  },
  {
    "start_time": "5.29",
    "end_time": "5.33",
    "alternatives": [
      {
        "confidence": "0.9955",
        "content": "a"
      }
    ],
    "type": "pronunciation"
  },
  {
    "start_time": "5.33",
    "end_time": "5.66",
    "alternatives": [
      {
        "confidence": "0.9754",
        "content": "follow"
      }
    ],
    "type": "pronunciation"
  },
  {
```

```

        "start_time": "5.66",
        "end_time": "5.75",
        "alternatives": [
            {
                "confidence": "0.9754",
                "content": "up"
            }
        ],
        "type": "pronunciation"
    },
    {
        "start_time": "5.75",
        "end_time": "6.02",
        "alternatives": [
            {
                "confidence": "1.0",
                "content": "visit"
            }
        ]
        ...
    },
    "status": "COMPLETED"
}

```

AWS CLI

Pour activer le partitionnement des haut-parleurs dans une tâche de transcription par lots (AWS CLI)

- Exécutez le code suivant.

```

aws transcribe start-medical-transcription-job \
--region us-west-2 \
--cli-input-json file://example-start-command.json

```

Le code suivant affiche le contenu de *example-start-command.json*.

```

{
    "MedicalTranscriptionJobName": "my-first-med-transcription-job",

```

```
"Media": {
  "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-audio-file.flac"
},
"OutputBucketName": "DOC-EXAMPLE-BUCKET",
"OutputKey": "my-output-files/",
"LanguageCode": "en-US",
"Specialty": "PRIMARYCARE",
"Type": "DICTATION"
}
```

Transcription d'une dictée médicale dans un flux en temps réel

Utilisez un WebSocket flux pour transcrire une dictée médicale sous forme de flux audio. Vous pouvez également utiliser leAWS Management Console pour transcrire le discours que vous ou d'autres personnes parlez directement dans un microphone.

Pour un HTTP/2 ou un WebSocket stream, vous pouvez transcrire du son dans les spécialités médicales suivantes :

- Cardiologie
- Oncologie
- Neurologie
- Principale
- Radiologie
- Urologie

Chaque spécialité médicale comprend de nombreux types de procédures et de rendez-vous. Les clinale dictent donc de nombreux types de notes. Utilisez les exemples suivants pour vous aider à spécifier la valeur du paramètre `specialty` URI de la WebSocket demande ou du `specialty` paramètre de l'[StartMedicalStreamTranscriptionAPI](#) :

- Pour une dictée après une électrophysiologie ou une échocardiogramme, choisissez `CARDIOLOGY`.
- Pour une dictée après une intervention d'oncologie chirurgicale ou de radio-oncologie, choisissez `ONCOLOGY`.
- Pour un médecin dictant des notes indiquant un diagnostic d'encéphalite, choisissez `NEUROLOGY`.
- Pour une dictée de notes d'intervention visant à éliminer un calcul vésical, choisissez `UROLOGY`.

- Pour une dictée des notes du clinicien après une consultation en médecine interne, choisissez `PRIMARYCARE`.
- Pour une dictée d'un médecin communiquant les résultats d'une tomodensitométrie, d'une tomodensitométrie, d'une IRM ou d'une radiographie, choisissez `RADIOLOGY`.
- Pour une dictée des notes du médecin après une consultation de gynécologie, choisissez `PRIMARYCARE`.

Pour améliorer la précision de la transcription de termes spécifiques dans un flux en temps réel, utilisez un vocabulaire personnalisé. Pour activer un vocabulaire personnalisé, définissez la valeur `vocabulary-name` de sur le nom du vocabulaire personnalisé que vous souhaitez utiliser.

Transcrire une dictée prononcée dans votre microphone à l'aide de l'[AWS Management Console](#)

[AWS Management Console](#) Pour transcrire le son en streaming d'une dictée médicale, choisissez l'option permettant de transcrire une dictée médicale, lancez la diffusion et commencez à parler dans le microphone.

Pour transcrire le son en streaming d'une dictée médicale (AWS Management Console)

1. Connectez-vous à [AWS Management Console](#).
2. Dans le panneau de navigation, sous [Amazon Transcribe Medical](#), sélectionnez [Transcription en temps réel](#).
3. Choisissez [Dictation](#).
4. Pour [Spécialité médicale](#), choisissez la spécialité médicale du clinicien qui parle dans le stream.
5. Choisissez [Start streaming](#) (Démarrer la diffusion).
6. Parlez dans le micro.

Transcription d'une dictée médicale dans un flux HTTP/2

Pour transcrire le flux HTTP/2 d'une dictée médicale, utilisez l'[StartMedicalStreamTranscription](#) API et spécifiez les informations suivantes :

- `LanguageCode`— Le code de langue. La valeur valide est `en-US`.
- `MediaEncoding`— Le codage utilisé pour le signal audio d'entrée. Les valeurs valides sont `pcm`, `ogg-opus` et `flac`.
- `Specialty`— La spécialité du professionnel de santé.

- Type – DICTATION

Pour plus d'informations sur la configuration d'un flux HTTP/2 pour transcrire une dictée médicale, consultez [Configuration d'un flux HTTP/2](#).

Utiliser une demande de WebSocket diffusion pour transcrire une dictée médicale

Pour transcrire une dictée médicale dans un flux en temps réel à l'aide d'une WebSocket demande, vous devez créer un URI présigné. Cet URI contient les informations nécessaires pour configurer le flux audio entre votre application et Amazon Transcribe Medical. Pour plus d'informations sur la création de WebSocket demandes, consultez [Configuration d'un WebSocket stream](#).

Utilisez le modèle suivant pour créer votre URI pré-signé.

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/medical-stream-
transcription-websocket
?language-code=languageCode
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-
west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=Signature Version 4 signature
&X-Amz-SignedHeaders=host
&media-encoding=flac
&sample-rate=16000
&session-id=sessionId
&specialty=medicalSpecialty
&type=DICTATION
&vocabulary-name=vocabularyName
&show-speaker-label=boolean
```

Pour plus d'informations sur la création d'URI pré-signés, consultez [Configuration d'un WebSocket stream](#).

Améliorer la précision de la transcription grâce à des vocabulaires médicaux personnalisés

Pour améliorer la précision de la transcription dans Amazon Transcribe Medical, créez et utilisez un ou plusieurs vocabulaires médicaux personnalisés. Un vocabulaire personnalisé est un ensemble de mots ou d'expressions spécifiques à un domaine. Cette collection contribue à améliorer les performances de Amazon Transcribe Medical en matière de transcription de ces mots ou phrases.

Vous êtes responsable de l'intégrité de vos données lorsque vous utilisez Amazon Transcribe Medical. Ne saisissez pas d'informations confidentielles, d'informations personnelles (PII) ou d'informations de santé protégées (PHI) dans un vocabulaire personnalisé.

Pour de meilleurs résultats, créez de petits vocabulaires personnalisés distincts qui aident chacun à transcrire un enregistrement audio spécifique. Vous bénéficiez d'une meilleure précision de transcription que si vous créez un vaste vocabulaire personnalisé à utiliser avec tous vos enregistrements.

Par défaut, vous pouvez avoir un vocabulaire personnalisé dans votre vocabulaire Compte AWS. La taille d'un vocabulaire personnalisé ne peut pas dépasser 50 Ko. Pour plus d'informations sur la demande d'augmentation du nombre de vocabulaires personnalisés que vous pouvez inclure dans votre fichier Compte AWS, consultez la section [Quotas AWS de service](#).

Les vocabulaires personnalisés sont disponibles en anglais américain (en-US).

Rubriques

- [Création d'un fichier texte pour votre vocabulaire médical personnalisé](#)
- [Utilisation d'un fichier texte pour créer un vocabulaire médical personnalisé](#)
- [Transcription d'un fichier audio à l'aide d'un vocabulaire médical personnalisé](#)
- [Transcription d'un flux en temps réel à l'aide d'un vocabulaire médical personnalisé](#)
- [Set de personnages pour Amazon Transcribe Medical](#)

Création d'un fichier texte pour votre vocabulaire médical personnalisé

Pour créer un vocabulaire personnalisé, vous devez créer un fichier texte au format UTF-8. Dans ce fichier, vous créez un tableau à quatre colonnes, chaque colonne indiquant un champ. Chaque champ indique à Amazon Transcribe Medical comment les termes spécifiques au domaine sont

prononcés ou comment afficher ces termes dans vos transcriptions. Vous stockez le fichier texte contenant ces champs dans un Amazon S3 compartiment.

Comprendre comment formater votre fichier texte

Pour créer un vocabulaire médical personnalisé, vous devez saisir les noms des colonnes sous forme de ligne d'en-tête. Vous entrez les valeurs pour chaque colonne située sous la ligne d'en-tête.

Les noms des quatre colonnes du tableau sont les suivants :

- **Phrase**— colonne requise, valeurs requises
- **IPA**— colonne requise, les valeurs peuvent être facultatives
- **SoundsLike**— colonne requise, les valeurs peuvent être facultatives
- **DisplayAs**— colonne requise, les valeurs peuvent être facultatives

Lorsque vous créez un vocabulaire personnalisé, assurez-vous que vous :

- Séparez chaque colonne par un seul caractère de tabulation. Amazon Transcribe affiche un message d'erreur si vous essayez de séparer les colonnes par des espaces ou plusieurs caractères de caractères de caractères de caractères de caractères de caractères de caractères de caractères de caractères.
- Assurez-vous qu'il n'y a pas d'espace final ou d'espace blanc après chaque valeur d'une colonne.

Assurez-vous que les valeurs que vous entrez pour chaque colonne :

- Comporte moins de 256 caractères, tirets compris
- Utilisez uniquement les caractères du jeu de caractères autorisé, voir [Set de personnages pour Amazon Transcribe Medical](#).

Saisie de valeurs pour les colonnes du tableau

Les informations suivantes expliquent comment spécifier des valeurs pour les quatre colonnes du tableau de caractères :

- **Phrase**— Le mot ou l'expression qui doit être reconnu. Vous devez entrer des valeurs dans cette colonne.

Si l'entrée est une expression, séparez les termes par un trait d'union (-). Par exemple, entrez **cerebral autosomal dominant arteriopathy with subcortical infarcts and leukoencephalopathy** en tant que **cerebral-autosomal-dominant-arteriopathy-with-subcortical-infarcts-and-leukoencephalopathy**.

Saisissez les acronymes ou les autres termes dont les lettres doivent être prononcées individuellement en séparant les lettres par un point, par exemple **D.N.A.** ou **S.T.E.M.I.**. Pour saisir le pluriel d'un acronyme, par exemple « **STEMIs** », séparez le « **s** » de l'acronyme par un tiret : **S.T.E.M.I-s**. Vous pouvez utiliser des lettres majuscules ou minuscules pour les acronymes.

La colonne `Phrase` est obligatoire. Vous pouvez utiliser n'importe quel caractère autorisé pour la langue saisie. Pour connaître les caractères autorisés, veuillez consulter [Set de personnages pour Amazon Transcribe Medical](#). Si vous ne spécifiez pas la colonne `DisplayAs`, Amazon Transcribe Medical utilise le contenu de la colonne `Phrase` dans le fichier de sortie.

- **IPA**(colonne obligatoire, les valeurs peuvent être facultatives) — Pour spécifier la prononciation d'un mot ou d'une phrase, vous pouvez inclure des caractères du [Alphabet phonétique international \(IPA\)](#) dans cette colonne. La colonne IPA ne peut pas inclure des espaces de début ou de fin, et vous devez utiliser un seul espace pour séparer chaque phonème dans la saisie. Par exemple, en anglais, pour l'expression **acute-respiratory-distress-syndrome**, vous saisissez **ə k j u t # # s p # # ə t # # i d # s t # # s s # n d # o # m**. Pour l'expression **A.L.L.**, vous saisissez **e# # l # l**.

Même si vous ne spécifiez pas le contenu de la colonne IPA, vous devez inclure une colonne IPA vide. Si vous incluez des valeurs dans la colonne IPA, vous ne pouvez pas fournir de valeurs pour la colonne `SoundsLike`.

Pour obtenir la liste des caractères phonétiques de l'alphabet phonétique international autorisés pour une langue spécifique, veuillez consulter [Set de personnages pour Amazon Transcribe Medical](#). L'anglais américain est la seule langue disponible en Amazon Transcribe médecine.

- **SoundsLike**(colonne requise, les valeurs peuvent être facultatives) — Vous pouvez diviser un mot ou une phrase en segments plus petits et fournir une prononciation pour chaque segment en utilisant l'orthographe standard de la langue afin d'imiter le son du mot. Par exemple, en anglais, vous pouvez fournir des conseils pour la prononciation de l'expression **cerebral-autosomal-dominant-arteriopathy-with-subcortical-infarcts-and-leukoencephalopathy** : **sir-e-brul-aut-o-som-ul-dah-mi-nant-ar-ter-ri-o-pa-ty-with-sub-cor-ti-cul-in-farcts-and-lewk-o-en-ce-phul-ah-pu-ty**. Le conseil pour l'expression

atrioventricular-nodal-reentrant-tachycardia serait : **ay-tree-o-ven-trick-u-lar-node-al-re-entr-ant-tack-ih-card-ia**. Les parties de l'indication doivent être séparées par un trait d'union (-).

Même si vous ne spécifiez pas de valeurs pour la colonne SoundsLike, vous devez inclure une colonne SoundsLike vide. Si vous incluez des valeurs dans la colonne SoundsLike, vous ne pouvez pas fournir de valeurs pour la colonne IPA.

Vous pouvez utiliser n'importe quel caractère autorisé pour la langue saisie. Pour obtenir la liste des caractères autorisés, veuillez consulter [Set de personnages pour Amazon Transcribe Medical](#).

- **DisplayAs**(colonne requise, les valeurs peuvent être facultatives) — Définit l'apparence du mot ou de la phrase lorsqu'il est sorti. Par exemple, si le terme ou l'expression est **cerebral-autosomal-dominant-arteriopathy-with-subcortical-infarcts-and-leukoencephalopathy**, vous pouvez décider d'afficher cerebral autosomal dominant arteriopathy with subcortical infarcts and leukoencephalopathy afin que les tirets n'apparaissent pas. Vous pouvez également spécifier DisplayAs comme CADASIL si vous souhaitez afficher l'acronyme au lieu du terme complet dans la sortie.

Si vous ne spécifiez pas la DisplayAs colonne, Amazon Transcribe Medical utilise la Phrase colonne du fichier d'entrée dans la sortie.

Vous pouvez utiliser n'importe quel caractère UTF-8 dans la colonne DisplayAs.

Vous pouvez inclure des espaces uniquement pour les valeurs des DisplayAs colonnes IPA et.

Pour créer le fichier texte de votre vocabulaire personnalisé, placez chaque mot ou phrase de votre fichier texte sur une ligne distincte. Séparez les colonnes par des caractères de tabulation. Incluez des espaces uniquement pour les valeurs des colonnes IPA et DisplayAs. Enregistrez le fichier avec l'extension .txt dans un Amazon S3 compartiment à l'emplacement Région AWS où vous utilisez Amazon Transcribe Medical pour créer votre vocabulaire personnalisé.

Si vous modifiez votre fichier texte dans Windows, assurez-vous que votre fichier est au format LF et non au format CRLF. Sinon, vous ne pourrez pas créer votre glossaire personnalisé. Certains éditeurs de texte vous permettent de modifier la mise en forme à l'aide des commandes Rechercher et Remplacer.

Les exemples suivants montrent du texte que vous pouvez utiliser pour créer des glossaires personnalisés. Pour créer un glossaire personnalisé à partir de ces exemples, copiez un exemple

dans un éditeur de texte, remplacez [TAB] par un caractère de tabulation et téléchargez le fichier texte enregistré dans Amazon S3.

```
Phrase[TAB]IPA[TAB]SoundsLike[TAB]DisplayAs
acute-respiratory-distress-syndrome[TAB][TAB][TAB]acute respiratory distress syndrome
A.L.L.[TAB]e# # 1 # 1[TAB][TAB]ALL
atrioventricular-nodal-reentrant-tachycardia[TAB][TAB]ay-tree-o-ven-trick-u-lar-node-
al-re-entr-ant-tack-ih-card-ia[TAB]
```

Vous pouvez entrer des colonnes dans n'importe quel ordre. Les exemples suivants montrent d'autres structures valides pour le fichier d'entrée du glossaire personnalisé.

```
Phrase[TAB]SoundsLike[TAB]IPA[TAB]DisplayAs
acute-respiratory-distress-syndrome[TAB][TAB][TAB]acute respiratory distress syndrome
A.L.L.[TAB][TAB]e# # 1 # 1[TAB]ALL
atrioventricular-nodal-reentrant-tachycardia[TAB]ay-tree-o-ven-trick-u-lar-node-al-re-
entr-ant-tack-ih-card-ia[TAB][TAB]
```

```
DisplayAs[TAB]SoundsLike[TAB]IPA[TAB]Phrase
acute respiratory distress syndrome[TAB][TAB][TAB]acute-respiratory-distress-syndrome
ALL[TAB][TAB]e# # 1 # 1[TAB]A.L.L.
[TAB]ay-tree-o-ven-trick-u-lar-node-al-re-entr-ant-tack-ih-card-ia[TAB]
[TAB]atrioventricular-nodal-reentrant-tachycardia
```

Pour faciliter la lecture, les tableaux suivants montrent plus clairement les exemples précédents au format html. Ils ne sont destinés qu'à illustrer les exemples.

Expression	IPA	SoundsLike	DisplayAs
acute-respiratory-distress-syndrome			acute respiratory distress syndrome
A.L.L.	eɪ ɛ ɛ		ALL
atrioventricular-nodal-reentrant-tachycardia		ay-tree-o-ven-trick-u-lar-node-al-re-entr-ant-tack-ih-card-ia	

Expression	SoundsLike	IPA	DisplayAs
acute-respiratory-distress-syndrome			acute respiratory distress syndrome
atrioventricular-nodal-reentrant-tachycardia	ay-tree-o-ven-trick-ular-node-al-re-entr-ant-tack-ih-card-ia		
A.L.L.		eɪ ɛ ɛ	ALL

DisplayAs	SoundsLike	IPA	Expression
acute respiratory distress syndrome			acute-respiratory-distress-syndrome
ALL		eɪ ɛ ɛ	A.L.L.
	ay-tree-o-ven-trick-ular-node-al-re-entr-ant-tack-ih-card-ia		atrioventricular-nodal-reentrant-tachycardia

Utilisation d'un fichier texte pour créer un vocabulaire médical personnalisé

Pour créer un vocabulaire personnalisé, vous devez avoir préparé un fichier texte contenant une collection de mots ou de phrases. Amazon Transcribe Medical utilise ce fichier texte pour créer un vocabulaire personnalisé que vous pouvez utiliser pour améliorer la précision de transcription de ces termes ou expressions. Vous pouvez créer un vocabulaire personnalisé à l'aide de l'[CreateMedicalVocabulary](#) API ou de la console Amazon Transcribe médicale.

AWS Management Console

Pour utiliser le AWS Management Console afin de créer un vocabulaire personnalisé, vous devez fournir l'Amazon S3URI du fichier texte contenant vos mots ou phrases.

1. Connectez-vous à [AWS Management Console](#).

2. Dans le panneau de navigation, sous Amazon Transcribe Médical, sélectionnez vocabulaire personnalisé de vocabulaire personnalisé.
3. Dans le champ Nom, sous Paramètres du vocabulaire, choisissez un nom pour votre vocabulaire personnalisé.
4. Spécifiez l'emplacement de votre fichier audio ou vidéo dans Amazon S3 :
 - Pour l'emplacement du fichier d'entrée de vocabulaire sur S3, sous Paramètres de vocabulaire, spécifiez l'Amazon S3URI qui identifie le fichier texte que vous utiliserez pour créer votre vocabulaire personnalisé.
 - Pour l'emplacement du fichier d'entrée de vocabulaire dans S3, choisissez Browse S3 pour rechercher le fichier texte et le sélectionner.
5. Choisissez Créer du vocabulaire.

Vous pouvez consulter l'état de traitement de votre vocabulaire personnalisé dans le AWS Management Console.

API

Pour créer un vocabulaire médical personnalisé (API)

- Pour l'[StartTranscriptionJob](#) API, spécifiez ce qui suit.
 - a. Pour LanguageCode, spécifiez en-US.
 - b. Pour VocabularyFileUri, spécifiez l'Amazon S3 emplacement du fichier texte que vous utilisez pour définir votre vocabulaire personnalisé.
 - c. Pour VocabularyName, spécifiez un nom pour votre vocabulaire personnalisé. Le nom que vous spécifiez doit être unique au sein de votre nomCompte AWS.

Pour voir l'état de traitement de votre vocabulaire personnalisé, utilisez l'[GetMedicalVocabulary](#) API.

Voici un exemple de demande utilisant le vocabulaire personnalisé AWS SDK for Python (Boto3) de vocabulaire personnalisé.

```
from __future__ import print_function
import time
import boto3
```

```
transcribe = boto3.client('transcribe', 'us-west-2')
vocab_name = "my-first-vocabulary"
response = transcribe.create_medical_vocabulary(
    VocabularyName = job_name,
    VocabularyFileUri = 's3://DOC-EXAMPLE-BUCKET/my-vocabularies/my-vocabulary-
table.txt'
    LanguageCode = 'en-US',
)

while True:
    status = transcribe.get_medical_vocabulary(VocabularyName = vocab_name)
    if status['VocabularyState'] in ['READY', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

AWS CLI

Pour activer le partitionnement des haut-parleurs dans une tâche de transcription par lots (AWS CLI)

- Exécutez le code suivant.

```
aws transcribe create-medical-vocabulary \
--vocabulary-name my-first-vocabulary \
--vocabulary-file-uri s3://DOC-EXAMPLE-BUCKET/my-vocabularies/my-vocabulary-
file.txt \
--language-code en-US
```

Transcription d'un fichier audio à l'aide d'un vocabulaire médical personnalisé

Utilisez le [StartMedicalTranscriptionJob](#) ou le AWS Management Console pour démarrer une tâche de transcription utilisant un vocabulaire personnalisé afin d'améliorer la précision de la transcription.

AWS Management Console

1. Connectez-vous à [AWS Management Console](#).

2. Dans le panneau de navigation, sous Amazon Transcribe Médical, sélectionnez Tâches de transcription.
3. Choisissez Create job (Créer une tâche).
4. Sur la page Spécifier les détails de la tâche, fournissez des informations sur votre tâche de transcription.
5. Choisissez Suivant.
6. Sous Personnalisation, activez Vocabulaire personnalisé.
7. Sous Sélection du vocabulaire, choisissez un vocabulaire personnalisé.
8. Sélectionnez Create (Créer).

API

Pour activer le partitionnement des haut-parleurs dans un fichier audio à l'aide d'une tâche de transcription par lots (API)

- Pour l'[StartMedicalTranscriptionJob](#) API, spécifiez ce qui suit.
 - a. Pour `MedicalTranscriptionJobName`, spécifiez un nom unique dans votre Compte AWS.
 - b. Pour `LanguageCode`, spécifiez le code de langue qui correspond à la langue parlée dans votre fichier audio et à la langue de votre filtre de vocabulaire.
 - c. Pour le `MediaFileUri` paramètre de l'`Media` objet, spécifiez le nom du fichier audio que vous souhaitez transcrire.
 - d. Pour `Specialty`, spécifiez la spécialité médicale du clinicien qui parle dans le fichier audio.
 - e. Pour `Type`, spécifiez si le fichier audio est une conversation ou une dictée.
 - f. Pour `OutputBucketName`, spécifiez le Amazon S3 compartiment dans lequel stocker les résultats de transcription.
 - g. Pour l'`Settings` objet, spécifiez ce qui suit.
 - `VocabularyName`— le nom de votre vocabulaire personnalisé.

La demande suivante utilise le AWS SDK for Python (Boto3) pour démarrer une tâche de transcription par lots avec un vocabulaire personnalisé.

```
from __future__ import print_function
```

```
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-med-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
transcribe.start_medical_transcription_job(
    MedicalTranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'my-output-files/',
    LanguageCode = 'en-US',
    Specialty = 'PRIMARYCARE',
    Type = 'CONVERSATION',
    Settings = {
        'VocabularyName': 'example-med-custom-vocab'
    }
)

while True:
    status = transcribe.get_medical_transcription_job(MedicalTranscriptionJobName =
job_name)
    if status['MedicalTranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED',
'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Transcription d'un flux en temps réel à l'aide d'un vocabulaire médical personnalisé

Pour améliorer la précision de la transcription dans un flux en temps réel, vous pouvez utiliser un vocabulaire personnalisé utilisant HTTP/2 ou WebSocket des flux. Pour démarrer une requête HTTP/2, utilisez l'[StartMedicalStreamTranscriptionAPI](#). Vous pouvez utiliser un vocabulaire personnalisé en temps réel àAWS Management Console l'aide de l'[StartMedicalStreamTranscriptionAPI](#) ou du WebSocket protocole.

Transcrire une dictée prononcée dans votre microphone (AWS Management Console)

AWS Management Console Pour transcrire le son en streaming d'une dictée médicale, choisissez l'option permettant de transcrire une dictée médicale, lancez la diffusion et commencez à parler dans le microphone.

Pour transcrire le son en streaming d'une dictée médicale (AWS Management Console)

1. Connectez-vous à [AWS Management Console](#).
2. Dans le panneau de navigation, sous Amazon Transcribe Médical, sélectionnez Transcription en temps réel de transcription.
3. Pour Spécialité médicale, choisissez la spécialité médicale du clinicien qui parle dans le stream.
4. Pour le type d'entrée audio, choisissez Conversation ou Dictée.
5. Pour Paramètres supplémentaires, choisissez Vocabulaire personnalisé.
 - Pour la sélection du vocabulaire, choisissez le vocabulaire personnalisé.
6. Choisissez Start streaming (Démarrer la diffusion).
7. Parlez dans le micro.

Activation du partitionnement des haut-parleurs dans un flux HTTP/2

Voici la syntaxe des paramètres d'une requête HTTP/2.

```
POST /medical-stream-transcription HTTP/2
host: transcribestreaming.us-west-2.amazonaws.com
authorization: Generated value
x-amz-target: com.amazonaws.transcribe.Transcribe.StartMedicalStreamTranscription
x-amz-content-sha256: STREAMING-MED-AWS4-HMAC-SHA256-EVENTS
x-amz-date: 20220208T235959Z
x-amzn-transcribe-session-id: my-first-http2-med-stream
x-amzn-transcribe-language-code: en-US
x-amzn-transcribe-media-encoding: flac
x-amzn-transcribe-sample-rate: 16000
x-amzn-transcribe-vocabulary-name: my-first-med-vocab
x-amzn-transcribe-specialty: PRIMARYCARE
x-amzn-transcribe-type: CONVERSATION
x-amzn-transcribe-show-speaker-label: true
Content-type: application/vnd.amazon.eventstream
transfer-encoding: chunked
```

Descriptions des paramètres :

- **host** : mettez à jour leRégion AWS ('us-west-2' dans l'exemple précédent) avec celuiRégion AWS que vous appelez. Pour obtenir la liste des points de terminaison validesRégions AWS, voir [Régions AWS Set Endpoints](#).
- **autorisation** : Ce champ est généré. Pour en savoir plus sur la création d'une signature, consultez [la section Signature deAWS demandes avec Signature version 4](#).
- **x-amz-target**: ne modifiez pas ce champ ; utilisez le contenu présenté dans l'exemple précédent.
- **x-amz-content-sha256** : Il s'agit d'un champ généré. Pour en savoir plus sur le calcul d'une signature, voir Signature de [AWSdemandes avec Signature version 4](#).
- **x-amz-date**: date et heure de création de signature. Le format est YYYYMMDDTHHMMSSZ, où YY=année, MM=mois, DD=jour, HH=heure, MM=minute, SS=secondes, et « T » et « Z » sont des caractères fixes. Pour plus d'informations, consultez la page [Gestion des dates dans Signature Version 4](#) de caractères caractères, sous forme de caractères, de caractères
- **x-amzn-transcribe-session-id** : nom de votre session de streaming.
- **x-amzn-transcribe-language-code** : L'encodage utilisé pour votre entrée audio. Reportez-vous à [StartMedicalStreamTranscriptionouLangues prises en charge et fonctionnalités spécifiques aux langues](#) pour obtenir une liste de valeurs valides.
- **x-amzn-transcribe-media-encoding** : L'encodage utilisé pour votre entrée audio. Les valeurs valides sont pcm, ogg-opus et flac.
- **x-amzn-transcribe-sample-rate** : fréquence d'échantillonnage du signal audio d'entrée (en Hertz). Amazon Transcribeprend en charge une plage de 8 000 Hz à 48 000 Hz. Un son de faible qualité, tel que le son d'un téléphone, est généralement d'environ 8 000 Hz. Un son de haute qualité varie généralement entre 16 000 Hz et 48 000 Hz. Notez que la fréquence d'échantillonnage que vous spécifiez doit correspondre à celle de votre audio.
- **x-amzn-transcribe-vocabulary-name** : Le nom du vocabulaire que vous souhaitez utiliser avec votre transcription.
- **x-amzn-transcribe-specialty**: La spécialité médicale à transcrire.
- **x-amzn-transcribe-type**: choisissez s'il s'agit d'une dictée ou d'une conversation.
- **x-amzn-transcribe-show-speaker-label** : pour activer la diarisation, cette valeur doit être true.
- **type de contenu** : ne modifiez pas ce champ ; utilisez le contenu présenté dans l'exemple précédent.

Activation du partitionnement des haut-parleurs dans une WebSocket demande

Pour partitionner les haut-parleurs en WebSocket flux à l'aide de l'API, utilisez le format suivant pour créer un URI pré-signé afin de lancer une WebSocket requête et de définir `vocabulary-name` le nom du vocabulaire personnalisé.

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/medical-stream-
transcription-websocket
?language-code=en-US
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-
west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=Signature Version 4 signature
&X-Amz-SignedHeaders=host
&media-encoding=flac
&sample-rate=16000
&session-id=sessionId
&specialty=medicalSpecialty
&type=CONVERSATION
&vocabulary-name=vocabularyName
&show-speaker-label=boolean
```

Set de personnages pour Amazon Transcribe Medical

Pour utiliser des vocabulaires personnalisés dans Amazon Transcribe Medical, utilisez le jeu de caractères suivant.

Jeu de caractères caractères caractères caractères caractères

Pour les glossaires personnalisés en anglais, vous pouvez utiliser les caractères suivants dans les colonnes `Phrase` et `SoundsLike` :

- a - z
- A - Z
- ' (apostrophe)
- - (trait d'union)
- . (point)

Vous pouvez utiliser les caractères de l'alphabet phonétique international dans la colonne IPA du fichier d'entrée du glossaire.

Caractère	Code	Caractère	Code
au	0061 028A	w	0077
aɪ	0061 026A	z	007A
b	0062	æ	00E6
d	0064	ð	00F0
eɪ	0065 026A	ŋ	014B
f	0066	ɑ	0251
g	0067	ɔ	0254
h	0068	ɔɪ	0254 026A
i	0069	ə	0259
j	006A	ɛ	025B
k	006B	ʒ	025D
l	006C	g	0261
ɹ	006C 0329	ɪ	026A
m	006D	ɹ	0279
n	006E	ʃ	0283
ŋ	006E 0329	ʊ	028A
ou	006F 028A	ʌ	028C
p	0070	ɹ	028D
s	0073	ʒ	0292

Caractère	Code	Caractère	Code
t	0074	dʒ	02A4
u	0075	tʃ	02A7
v	0076	θ	03B8

Identification des informations personnelles sur la santé (PHI) dans une transcription

Utilisez l'identification des informations médicales personnelles pour étiqueter les informations personnelles sur la Health (PHI) dans vos résultats de transcription. En examinant les étiquettes, vous pouvez trouver des informations personnelles qui pourraient être utilisées pour identifier un patient.

Vous pouvez identifier les données PHI à l'aide d'un flux en temps réel ou d'une tâche de transcription par lots.

Vous pouvez utiliser votre propre post-traitement pour rédiger les PHI identifiés dans la sortie de transcription.

Utilisez l'identification des informations personnelles sur la Health pour identifier les types de renseignements personnels sur la santé suivants :

- PHI personnels :
 - Noms : nom complet ou nom de famille et initiale
 - Sexe
 - Âge
 - Numéros de téléphone
 - Dates (sans compter l'année) directement liées au patient
 - Adresses e-mail
- PHI géographiques :
 - Adresse physique
 - Code postal
 - Nom du centre médical ou du cabinet

- Compte PHI :
 - Numéros de télécopie
 - Numéros de sécurité sociale (SSN)
 - Numéros de bénéficiaire de l'Health maladie
 - Numéros de comptes bancaires
 - Numéros de certificat ou de licence
- PHI du véhicule :
 - Numéro d'identification de véhicule (VIN)
 - numéro de plaque d'immatriculation
- Autres PHI supplémentaires :
 - Emplacement uniforme des ressources Web (URL)
 - Numéros d'adresse IP (Internet Protocol)

Amazon TranscribeHealth Insurance Portability and Accountability Act (HIPAA) Health Insurance Portability and Accountability Act (HIPAA) de 1996 (HIPAA) Pour plus d'informations, veuillez consulter [Amazon Transcribe Médical](#). Pour plus d'informations sur l'identification des données PHI dans un fichier audio, consultez [Identifier les données PHI dans un fichier audio](#). Pour plus d'informations sur l'identification des PHI dans un flux, consultez [Identifier les PHI dans un flux en temps réel](#).

Rubriques

- [Identifier les données PHI dans un fichier audio](#)
- [Identifier les PHI dans un flux en temps réel](#)

Identifier les données PHI dans un fichier audio

Utilisez un travail de transcription par lots pour transcrire des fichiers audio et identifier les informations médicales personnelles (PHI) qu'ils contiennent. Lorsque vous activez l'identification des informations personnelles sur la Health (PHI), Amazon Transcribe Medical étiquette les informations de santé identifiées dans les résultats de la transcription. Pour plus d'informations sur les PHI que Amazon Transcribe Medical peut identifier, voir [Identification des informations personnelles sur la santé \(PHI\) dans une transcription](#).

Vous pouvez démarrer une tâche de transcription par lots à l'aide de l'[StartMedicalTranscriptionJob](#) API ou de l'AWS Management Console.

AWS Management Console

Pour utiliser l'AWS Management Console pour transcrire un dialogue clinicien-patient, créez une tâche de transcription et choisissez Conversation pour le type d'entrée audio.

Pour transcrire un fichier audio et identifier son PHI (AWS Management Console)

1. Connectez-vous à [AWS Management Console](#).
2. Dans le panneau de navigation, sous Amazon Transcribe Medical, sélectionnez Tâches de transcription.
3. Choisissez Create job (Créer une tâche).
4. Sur la page Spécifier les détails de la Job, sous Paramètres de la tâche, spécifiez les informations suivantes.
 - a. Nom : nom de la tâche de transcription qui vous est propre Compte AWS.
 - b. Type d'entrée audio : conversation ou dictée.
5. Pour les autres champs, spécifiez l'Amazon S3 emplacement de votre fichier audio et l'endroit où vous souhaitez stocker la sortie de votre travail de transcription.
6. Choisissez Suivant.
7. Sous Paramètres audio, choisissez Identification PHI.
8. Sélectionnez Create (Créer).

API

Pour transcrire un fichier audio et identifier ses PHI à l'aide d'une tâche de transcription par lots (API)

- Pour l'[StartMedicalTranscriptionJob](#) API, spécifiez ce qui suit.
 - a. Pour `MedicalTranscriptionJobName`, spécifiez un nom qui soit unique à votre Compte AWS.
 - b. Pour `LanguageCode`, spécifiez le code de langue correspondant à la langue parlée dans votre fichier audio.
 - c. Pour le `MediaFileUri` paramètre de l'`Media` objet, spécifiez le nom du fichier audio que vous souhaitez transcrire.

- d. Pour `Specialty`, spécifiez la spécialité médicale du clinicien qui parle dans le fichier audio sous la forme `PRIMARYCARE`.
- e. Pour `Type`, spécifiez `CONVERSATION` ou `DICTIONATION`.
- f. Pour `OutputBucketName`, spécifiez le Amazon S3 compartiment dans lequel stocker les résultats de transcription.

Voici un exemple de demande qui utilise le AWS SDK for Python (Boto3) pour transcrire un fichier audio et identifier le PHI d'un patient.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe')
job_name = "my-first-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-audio-file.flac"
transcribe.start_medical_transcription_job(
    MedicalTranscriptionJobName = job_name,
    Media = {'MediaFileUri': job_uri},
    LanguageCode = 'en-US',
    ContentIdentificationType = 'PHI',
    Specialty = 'PRIMARYCARE',
    Type = 'type', # Specify 'CONVERSATION' for a medical conversation. Specify
    'DICTIONATION' for a medical dictation.
    OutputBucketName = 'DOC-EXAMPLE-BUCKET'
)
while True:
    status = transcribe.get_medical_transcription_job(MedicalTranscriptionJobName =
    job_name)
    if status['MedicalTranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED',
    'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

L'exemple de code suivant montre les résultats de la transcription avec le PHI du patient identifié.

```
{
  "jobName": "my-medical-transcription-job-name",
  "accountId": "111122223333",
  "results": {
    "transcripts": [{
      "transcript": "The patient's name is Bertrand."
    }],
    "items": [{
      "start_time": "0.0",
      "end_time": "0.37",
      "alternatives": [{
        "confidence": "0.9993",
        "content": "The"
      }],
      "type": "pronunciation"
    }, {
      "start_time": "0.37",
      "end_time": "0.44",
      "alternatives": [{
        "confidence": "0.9981",
        "content": "patient's"
      }],
      "type": "pronunciation"
    }, {
      "start_time": "0.44",
      "end_time": "0.52",
      "alternatives": [{
        "confidence": "1.0",
        "content": "name"
      }],
      "type": "pronunciation"
    }, {
      "start_time": "0.52",
      "end_time": "0.92",
      "alternatives": [{
        "confidence": "1.0",
        "content": "is"
      }],
      "type": "pronunciation"
    }, {
      "start_time": "0.92",
      "end_time": "0.9989",
```

```
    "alternatives": [{
      "confidence": "1.0",
      "content": "Bertrand"
    }],
    "type": "pronunciation"
  }, {
    "alternatives": [{
      "confidence": "0.0",
      "content": "."
    }],
    "type": "punctuation"
  }],
  "entities": [{
    "content": "Bertrand",
    "category": "PHI*-Personal*",
    "startTime": 0.92,
    "endTime": 1.2,
    "confidence": 0.9989
  }],
},
"status": "COMPLETED"
}
```

AWS CLI

Pour transcrire un fichier audio et identifier les données PHI à l'aide d'une tâche de transcription par lots (AWS CLI)

- Exécutez le code suivant.

```
aws transcribe start-medical-transcription-job \  
--medical-transcription-job-name my-medical-transcription-job-name \  
--language-code en-US \  
--media MediaFileUri="s3://DOC-EXAMPLE-BUCKET/my-input-files/my-audio-file.flac" \  
--output-bucket-name DOC-EXAMPLE-BUCKET \  
--specialty PRIMARYCARE \  
--type type \ # Choose CONVERSATION to transcribe a medical conversation.  
Choose DICTATION to transcribe a medical dictation.  
--content-identification-type PHI
```

Identifier les PHI dans un flux en temps réel

Vous pouvez identifier les informations personnelles sur la Health (PHI) dans HTTP/2 ou dans WebSocket des flux. Lorsque vous activez l'identification PHI, Amazon Transcribe Medical étiquette les PHI qu'il identifie dans les résultats de transcription. Pour plus d'informations sur les PHI que Amazon Transcribe Medical peut identifier, voir [Identification des informations personnelles sur la santé \(PHI\) dans une transcription](#).

Identifier les PHI lors d'une dictée prononcée dans votre microphone

Pour utiliser le AWS Management Console afin de transcrire le discours capté par votre microphone et d'identifier tout PHI, choisissez Dictation comme type d'entrée audio, lancez le flux et commencez à parler dans le microphone de votre ordinateur.

Pour identifier les données PHI dans une dictée à l'aide du AWS Management Console

1. Connectez-vous à [AWS Management Console](#).
2. Dans le panneau de navigation, choisissez Transcription en temps réel.
3. Pour Type d'entrée audio, choisissez Dictée.
4. Pour Paramètres supplémentaires, choisissez Identification PHI.
5. Choisissez Démarrer la diffusion et parlez dans le microphone.
6. Choisissez Arrêter la diffusion pour mettre fin à la dictée.

Identification des PHI dans un flux HTTP/2

Pour démarrer un flux HTTP/2 avec l'identification PHI activée, utilisez l'[StartMedicalStreamTranscription](#) API et spécifiez les informations suivantes :

- Pour `LanguageCode`, spécifiez le code de langue correspondant à la langue parlée dans le flux. Pour l'anglais américain, précisez `en-US`.
- Pour `MediaSampleHertz`, spécifiez la fréquence d'échantillonnage de l'audio.
- Pour `content-identification-type`, spécifiez PHI.

Identifier les données PHI dans un WebSocket flux

Pour démarrer un WebSocket flux avec l'identification PHI activée, utilisez le format suivant pour créer une URL présignée.

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/medical-stream-
transcription-websocket?
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-
west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=Signature Version 4 signature
&X-Amz-SignedHeaders=host
&language-code=en-US
&media-encoding=flac
&sample-rate=16000
&specialty=medical-specialty
&content-identification-type=PHI
```

Les définitions de paramètres se trouvent dans la [référence de l'API](#) ; les paramètres communs à toutes les opérations de l'AWSAPI sont répertoriés dans la section [Paramètres communs](#).

Génération de transcriptions alternatives

Lorsque vous utilisez Amazon Transcribe Medical, vous obtenez la transcription qui présente le plus haut niveau de confiance. Toutefois, vous pouvez configurer Amazon Transcribe Medical pour renvoyer des transcriptions supplémentaires avec des niveaux de confiance inférieurs.

Utilisez des transcriptions alternatives pour voir différentes interprétations de la transcription audio. Par exemple, dans une application qui permet à une personne de vérifier la transcription, vous pouvez présenter les transcriptions alternatives parmi lesquelles elle peut choisir.

Vous pouvez générer des transcriptions alternatives à l'aide de AWS Management Console ou de l'[StartMedicalTranscriptionJob](#) API.

AWS Management Console

AWS Management Console Pour générer des transcriptions alternatives, vous devez activer des résultats alternatifs lorsque vous configurez votre tâche.

1. Connectez-vous à [AWS Management Console](#).
2. Dans le panneau de navigation Amazon Transcribe, sous NET de transcription.
3. Choisissez Create job (Créer une tâche).

4. Sur la page Spécifier les détails de la tâche, fournissez des informations sur votre tâche de transcription.
5. Choisissez Next (Suivant).
6. Activez les résultats alternatifs.
7. Dans le champ Nombre maximal d'alternatives, entrez un entier compris entre 2 et 10, pour le nombre maximum de transcriptions alternatives que vous souhaitez voir apparaître dans la sortie.
8. Sélectionnez Create (Créer).

API

Pour séparer le texte par haut-parleur dans un fichier audio à l'aide d'une tâche de transcription par lots (API)

- Pour l'[StartMedicalTranscriptionJob](#) API, spécifiez ce qui suit.
 - a. Pour `MedicalTranscriptionJobName`, spécifiez un nom unique dans votre Compte AWS.
 - b. Pour `LanguageCode`, spécifiez le code de langue qui correspond à la langue parlée dans votre fichier audio et à la langue de votre filtre de vocabulaire.
 - c. Dans le `MediaFileUri` paramètre de l'`Media` objet, spécifiez l'emplacement du fichier audio que vous souhaitez transcrire.
 - d. Pour `Specialty`, spécifiez la spécialité médicale du clinicien qui parle dans le fichier audio.
 - e. `ParType` exemple, précisez si vous transcrivez une conversation médicale ou une dictée.
 - f. Pour `OutputBucketName`, spécifiez le Amazon S3 compartiment dans lequel stocker les résultats de transcription.
 - g. Pour l'`Settings` objet, spécifiez ce qui suit.
 - i. `ShowAlternatives` – `true`.
 - ii. `MaxAlternatives`- Un entier compris entre 2 et 10 pour indiquer le nombre de transcriptions alternatives que vous souhaitez voir apparaître dans la sortie de transcription.

La demande suivante utilise le AWS SDK for Python (Boto3) pour démarrer une tâche de transcription qui génère jusqu'à deux transcriptions alternatives.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-transcription-job"
job_uri = s3://DOC-EXAMPLE-BUCKET/my-input-files/my-audio-file.flac
transcribe.start_medical_transcription_job(
    MedicalTranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'my-output-files/',
    LanguageCode = 'en-US',
    Specialty = 'PRIMARYCARE',
    Type = 'CONVERSATION',
    Settings = {
        'ShowAlternatives': True,
        'MaxAlternatives': 2
    }
)

while True:
    status = transcribe.get_medical_transcription_job(MedicalTranscriptionJobName =
job_name)
    if status['MedicalTranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED',
'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

AWS CLI

Pour transcrire un fichier audio d'une conversation entre un clinicien de soins primaires et un patient dans un fichier audio (AWS CLI)

- Exécutez le code suivant.


```
aws transcribe start-transcription-job \  
--cli-input-json file://filepath/example-start-command.json
```

L'exemple suivant le `example-start-command.json`.

```
{  
  "MedicalTranscriptionJobName": "my-first-transcription-job",  
  "LanguageCode": "en-US",  
  "Specialty": "PRIMARYCARE",  
  "Type": "CONVERSATION",  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-audio-  
file.flac"  
  },  
  "Settings": {  
    "ShowAlternatives": true,  
    "MaxAlternatives": 2  
  }  
}
```

Amazon Transcribe Points de terminaison VPC médicaux et d'interface (AWS PrivateLink)

Vous pouvez établir une connexion privée entre votre VPC et Amazon Transcribe Médical en créant un point de terminaison VPC de l'interface. Les points de terminaison de l'interface sont alimentés par [AWS PrivateLink](#), une technologie qui vous permet d'accéder en privé Amazon Transcribe Des API médicales sans passerelle Internet, appareil NAT, connexion VPN ou AWS Direct Connect connexion. Les instances de votre VPC n'ont pas besoin d'adresses IP publiques pour communiquer avec Amazon Transcribe API médicales. Trafic entre votre VPC et Amazon Transcribe Medical ne quitte pas le réseau Amazon.

Chaque point de terminaison d'interface est représenté par une ou plusieurs [interfaces réseau Elastic](#) dans vos sous-réseaux.

Pour plus d'informations, voir [Points de terminaison d'interface VPC \(AWS PrivateLink\)](#) dans le [Amazon VPC Guide de l'utilisateur](#).

Considérations pour Amazon Transcribe Points de terminaison VPC médicaux

Avant de configurer un point de terminaison VPC d'interface pour Amazon Transcribe Médical, assurez-vous de vérifier [Propriétés et limites des points de terminaison de l'interface](#) dans le [Amazon VPC Guide de l'utilisateur](#).

Amazon Transcribe Medical prend en charge l'envoi d'appels vers toutes ses actions d'API depuis votre VPC.

Création d'un point de terminaison VPC d'interface pour Amazon Transcribe Médical

Vous pouvez créer un point de terminaison VPC pour Amazon Transcribe Service médical utilisant l'un des [AWS Management Console](#) ou le [AWS CLI](#). Pour de plus amples informations, veuillez consulter [Création d'un point de terminaison d'interface](#) dans le [Guide de l'utilisateur Amazon VPC](#).

Pour la transcription par lots dans Amazon Transcribe Medical, créez un point de terminaison VPC en utilisant le nom de service suivant :

- `com.amazonaws.usa-ouest-2.transcrire`

Pour diffuser la transcription dans Amazon Transcribe Medical, créez un point de terminaison VPC en utilisant le nom de service suivant :

- `com.amazonaws.usa-ouest-2.transcribe le streaming`

Si vous activez le DNS privé pour le point de terminaison, vous pouvez envoyer des demandes d'API à Amazon Transcribe Medical en utilisant son nom DNS par défaut pour Région AWS, par exemple, `transcribestreaming.us-east-2.amazonaws.com`.

Pour plus d'informations, voir [Accès à un service via un point de terminaison d'interface](#) dans le [Amazon VPC Guide de l'utilisateur](#).

Création d'une politique de point de terminaison VPC pour Amazon Transcribe Streaming médical

Vous pouvez associer une politique de point de terminaison à votre point de terminaison VPC qui contrôle l'accès à Amazon Transcribe Médical. La politique spécifie les informations suivantes :

- Le principal qui peut exécuter des actions.
- Les actions qui peuvent être effectuées.
- Les ressources sur lesquelles les actions peuvent être exécutées.

Pour plus d'informations, consultez [Contrôle de l'accès aux services avec points de terminaison d'un VPC](#) dans le Guide de l'utilisateur Amazon VPC.

Exemple : politique de point de terminaison VPC pour Amazon Transcribe Actions de transcription du streaming médical

Voici un exemple de politique de terminal pour la diffusion de la transcription dans Amazon Transcribe Médical. Lorsqu'elle est associée à un point de terminaison, cette politique autorise l'accès à la liste Amazon Transcribe Des actions médicales pour tous les directeurs et sur toutes les ressources.

```
{
  "Statement": [
    {
      "Principal": "*",
      "Effect": "Allow",
      "Action": [
        "transcribe:StartMedicalStreamTranscription",
      ],
      "Resource": "*"
    }
  ]
}
```

Exemple : politique de point de terminaison VPC pour Amazon Transcribe Actions de transcription médicale par lots

Voici un exemple de politique de terminal pour la transcription par lots dans Amazon Transcribe Médical. Lorsqu'elle est associée à un point de terminaison, cette politique autorise

l'accès à la liste Amazon TranscribeDes actions médicales pour tous les directeurs et sur toutes les ressources.

```
{
  "Statement": [
    {
      "Principal": "*",
      "Effect": "Allow",
      "Action": [
        "transcribe:StartMedicalTranscriptionJob"
      ],
      "Resource": "*"
    }
  ]
}
```

Sous-réseaux partagés

Vous ne pouvez pas créer, décrire, modifier ou supprimer des points de terminaison VPC dans des sous-réseaux qui sont partagés avec vous. Toutefois, vous pouvez utiliser les points de terminaison d'un VPC dans les sous-réseaux qui sont partagés avec vous. Pour plus d'informations sur le partage de VPC, voir [Partagez votre VPC avec d'autres comptes](#) dans le Amazon Virtual Private Cloudguide.

AWS HealthScribe

AWS HealthScribe est une nouvelle fonctionnalité de machine learning (ML) conforme à la loi HIPAA qui combine la reconnaissance vocale et l'IA générative pour transcrire les conversations patient-médecin et générer des notes cliniques faciles à consulter. AWS HealthScribe aide les fournisseurs de logiciels de santé à créer des applications cliniques qui réduisent la charge de documentation et améliorent l'expérience de consultation. Le service fournit automatiquement des transcriptions détaillées des conversations, identifie les rôles des intervenants, classe les dialogues, extrait les termes médicaux et génère des notes cliniques préliminaires. AWS HealthScribe combine ces fonctionnalités pour éliminer le besoin d'intégrer et d'optimiser des services d'IA distincts, ce qui vous permet d'accélérer la mise en œuvre.

Cas d'utilisation courants :

- Réduction du temps de documentation – Permettre aux médecins de compléter rapidement la documentation clinique grâce à des notes cliniques générées par l'IA, faciles à réviser, à ajuster et à finaliser dans votre application.
- Amélioration de l'efficacité des transcriptions médicales – Équiper les transpositeurs médicaux de transcriptions et de notes cliniques générées par l'IA, ainsi que de la bande audio de la consultation, afin d'accélérer les délais de traitement de la documentation.
- Récapitulatif efficace des visites des patients – Créer une expérience qui permet aux utilisateurs de se souvenir rapidement des points saillants de leur conversation dans votre application.

Important

Les résultats produits par AWS HealthScribe sont probabilistes et ne sont pas toujours exacts en raison de divers facteurs, notamment la qualité audio, le bruit de fond, la clarté du locuteur, la complexité de la terminologie médicale, les nuances linguistiques spécifiques au contexte et [la nature du machine learning et de l'IA générative](#). AWS HealthScribe est conçu pour être utilisé dans un rôle d'assistance pour les médecins et les transpositeurs médicaux. AWS Les résultats de HealthScribe ne doivent être utilisés que dans des scénarios de soins aux patients, y compris, mais sans s'y limiter, dans le cadre des dossiers médicaux électroniques, après vérification de leur exactitude et imposition d'un jugement médical solide par des professionnels de santé qualifiés. AWS Les résultats de HealthScribe ne remplacent

pas les conseils, diagnostics ou traitements médicaux professionnels et ne sont pas destinés à guérir, traiter, atténuer, prévenir ou diagnostiquer une maladie ou un problème de santé.

AWS HealthScribe fonctionne selon un modèle de responsabilité partagée, dans lequel AWS est responsable de la protection de l'infrastructure qui gère AWS HealthScribe et dans lequel vous êtes responsable de la gestion de vos données. Pour plus d'informations, consultez [Modèle de responsabilité partagée](#).

AWS HealthScribe est disponible dans la région USA Est (Virginie du Nord).

Le service est disponible en anglais américain (en-US). Pour un résultat optimal, utilisez un format audio sans perte tel que FLAC ou WAV, avec un encodage PCM 16 bits. AWS HealthScribe prend en charge des fréquences d'échantillonnage de 16 000 Hz ou plus.

AWS HealthScribe prend actuellement en charge les spécialités de médecine générale et d'orthopédie.

Une tâche AWS HealthScribe analyse les consultations médicales pour produire deux fichiers de sortie JSON : un fichier de [transcription](#) et un fichier de [documentation clinique](#).

Dans le fichier de transcription, outre la sortie de transcription standard étape par étape avec horodatage au niveau des mots, AWS HealthScribe fournit les éléments suivants :

- Détection du rôle des participants : pour distinguer les patients des médecins dans la transcription de la conversation.
- Découpage des transcriptions : pour classer les dialogues de transcription en fonction de leur pertinence clinique (discussion informelle, subjectif, objectif, etc.). Peut être utilisé pour afficher des parties spécifiques de la transcription.
- Entités cliniques : qui incluent des informations structurées telles que les médicaments, les affections médicales et les traitements mentionnés dans la conversation.

Dans le fichier de documentation clinique, AWS HealthScribe fournit les éléments suivants :

- Résumés contenant des notes résumées pour les sections clés de la documentation clinique telles que la plainte principale, l'historique de la maladie actuelle, l'examen des systèmes, les antécédents médicaux, l'évaluation et le plan.

- Liens des preuves qui relie chaque phrase utilisée dans les résumés des notes générés par l'IA à la transcription originale de la consultation, ce qui permet aux utilisateurs de valider plus facilement l'exactitude du résumé dans l'application.

Opérations d'API spécifiques à AWS HealthScribe :

- StartMedicalScribeJob
- ListMedicalScribeJobs
- GetMedicalScribeJob
- DeleteMedicalScribeJob

Pour afficher des exemples de demandes AWS HealthScribe, consultez [Démarrage d'une tâche AWS HealthScribe](#).

Fichier de transcription

Le fichier de transcription fournit le contenu de la conversation dans un format détaillé.

En outre, les informations suivantes sont fournies pour chaque tour de conversation :

- Rôle du participant – Chaque participant est étiqueté comme un médecin ou un patient. Si une conversation compte plusieurs participants dans chaque catégorie, un numéro est attribué à chaque participant. Par exemple, CLINICIAN_1, CLINICIAN_2 et PATIENT_1, PATIENT_2.
- Section : chaque tour de dialogue est attribué à l'une des quatre sections possibles en fonction du contenu identifié.
 - Subjectif : informations fournies par le patient au sujet de ses problèmes de santé.
 - Objectif : informations observées par le médecin par le biais d'examens physiques, de tests de laboratoire, d'imagerie ou de tests diagnostiques.
 - Évaluation et plan : informations relatives à l'évaluation du médecin et au plan de traitement.
 - Gestion du flux de visites : informations relatives aux conversations informelles ou aux transitions.
- Informations : extrait les entités cliniquement pertinentes (ClinicalEntity) présentes dans la conversation. AWS HealthScribe détecte toutes les entités cliniques prises en charge par [Amazon Comprehend Medical](#).

Pour des informations de sortie plus détaillées, consultez [Exemple de sortie de transcription](#).

Fichier de documentation clinique

Le fichier d'informations sur la documentation contient des résumés pour les sections clés suivantes de la documentation clinique.

Section	Description
PLAINTE PRINCIPALE	Brève description de la raison pour laquelle le patient consulte un médecin.
ANTÉCÉDENTS DE LA MALADIE ACTUELLE	Notes fournissant des informations sur la maladie du patient, notamment sur la gravité, l'apparition et la chronologie des symptômes, les traitements actuels et les zones touchées.
EXAMEN DES SYSTÈMES	Évaluation des symptômes des différents systèmes corporels rapportée par le patient.
ANTÉCÉDENTS MÉDICAUX	Détaille les problèmes médicaux, les interventions chirurgicales et les traitements antérieurs d'un patient.
ÉVALUATION	Notes fournissant des informations sur l'évaluation de l'état de santé du patient par le médecin.
PLAN	Notes faisant référence à tout traitement médical, à tout ajustement du mode de vie et à tout autre rendez-vous.

Chaque phrase présente dans le Summary inclut des références à la transcription originale de la consultation, ce qui permet aux utilisateurs de valider plus facilement l'exactitude du récapitulatif dans votre application. Assurer la traçabilité et la transparence des informations générées par l'IA est conforme aux principes de l'IA responsable comme l'explicabilité. Fournir ces références ainsi que les notes de synthèse aux médecins ou aux transcripateurs médicaux contribue à renforcer la confiance et à encourager une utilisation sûre de l'IA dans les milieux cliniques.

Chaque phrase du Summary contient EvidenceLinks qui fournit SegmentId pour les dialogues pertinents de la transcription qui ont été résumés.

Pour des informations de sortie plus détaillées, consultez [Exemple de sortie de documentation clinique](#).

Démarrage d'une tâche dans AWS HealthScribe

Vous pouvez démarrer une tâche AWS HealthScribe à l'aide de la CLI AWS ou de kits SDK AWS ; consultez les exemples suivants.

AWS CLI

Cet exemple utilise la commande [start-medical-scribe-job](#). Pour plus d'informations, consultez la section [StartMedicalScribeJob](#).

```
aws transcribe start-medical-scribe-job \  
--region us-west-2 \  
--medical-scribe-job-name my-first-medical-scribe-job \  
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
--output-bucket-name DOC-EXAMPLE-BUCKET \  
--DataAccessRoleArn=arn:aws:iam::111122223333:role/ExampleRole \  
--settings ShowSpeakerLabels=false,ChannelIdentification=true \  
--channel-definitions ChannelId=0,ParticipantRole=CLINICIAN \  
ChannelId=1,ParticipantRole=PATIENT
```

Voici un autre exemple d'utilisation de la commande [start-medical-scribe-job](#) et d'un corps de requête avec des paramètres supplémentaires.

```
aws transcribe start-medical-scribe-job \  
--region us-west-2 \  
--cli-input-json file://filepath/my-first-medical-scribe-job.json
```

Le fichier `my-first-medical-scribe-job.json` contient le corps de requête suivant.

```
{
```

```

"MedicalScribeJobName": "my-first-medical-scribe-job",
"Media": {
  "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
},
"OutputBucketName": "DOC-EXAMPLE-BUCKET",
"DataAccessRoleArn": "arn:aws:iam::111122223333:role/ExampleRole",
"Settings": {
  "ShowSpeakerLabels": false,
  "ChannelIdentification": true
},
"ChannelDefinitions": [
  {
    "ChannelId": 0,
    "ParticipantRole": "CLINICIAN"
  }, {
    "ChannelId": 1,
    "ParticipantRole": "PATIENT"
  }
]
}

```

AWS SDK for Python (Boto3)

L'exemple suivant utilise le AWS SDK for Python (Boto3) pour effectuer une demande [start_medical_scribe_job](#). Pour plus d'informations, consultez la section [StartMedicalScribeJob](#).

```

from __future__ import print_function
import time
import boto3

transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-medical-scribe-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
transcribe.start_medical_scribe_job(
    MedicalScribeJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    DataAccessRoleArn = 'arn:aws:iam::111122223333:role/ExampleRole',
    Settings = {
        'ShowSpeakerLabels': false,
        'ChannelIdentification': true
    },

```

```
ChannelDefinitions = [  
  {  
    'ChannelId': 0,  
    'ParticipantRole': 'CLINICIAN'  
  }, {  
    'ChannelId': 1,  
    'ParticipantRole': 'PATIENT'  
  }  
]  
)  
while True:  
    status = transcribe.get_medical_scribe_job(MedicalScribeJobName = job_name)  
    if status['MedicalScribeJob']['MedicalScribeJobStatus'] in ['COMPLETED', 'FAILED']:  
        break  
    print("Not ready yet...")  
    time.sleep(5)  
print(status)
```

Note

La console de gestion AWS ne prend actuellement pas en charge les tâches AWS HealthScribe.

Exemple de sortie

Outre une transcription, les demandes `StartMedicalScribeJob` génèrent un fichier de documentation clinique distinct. Les deux fichiers sont au format JSON et sont stockés dans l'emplacement de sortie que vous spécifiez dans votre demande. Voici des exemples de chaque type de sortie :

Exemple de sortie de transcription

Un fichier de transcription AWS HealthScribe (provenant d'une demande `StartMedicalScribeJob`) a le format suivant :

```
{  
  "Conversation": {  
    "ConversationId": "sampleConversationUUID",  
    "JobName": "sampleJobName",
```

```
"JobType": "ASYNC",
"LanguageCode": "en-US",
"ClinicalInsights": [
  {
    "Attributes": [],
    "Category": "MEDICAL_CONDITION",
    "InsightId": "insightUUID1",
    "InsightType": "ClinicalEntity",
    "Spans": [
      {
        "BeginCharacterOffset": 12,
        "Content": "pain",
        "EndCharacterOffset": 15,
        "SegmentId": "uuid1"
      }
    ],
    "Type": "DX_NAME"
  },
  {
    "Attributes": [],
    "Category": "TEST_TREATMENT_PROCEDURE",
    "InsightId": "insightUUID2",
    "InsightType": "ClinicalEntity",
    "Spans": [
      {
        "BeginCharacterOffset": 4,
        "Content": "mammogram",
        "EndCharacterOffset": 12,
        "SegmentId": "uuid2"
      }
    ],
    "Type": "TEST_NAME"
  },
  {
    "Attributes": [],
    "Category": "TEST_TREATMENT_PROCEDURE",
    "InsightId": "insightUUID3",
    "InsightType": "ClinicalEntity",
    "Spans": [
      {
        "BeginCharacterOffset": 15,
        "Content": "pap smear",
        "EndCharacterOffset": 23,
        "SegmentId": "uuid3"
      }
    ]
  }
]
```

```
    }
  ],
  "Type": "TEST_NAME"
},
{
  "Attributes": [],
  "Category": "MEDICATION",
  "InsightId": "insightUUID4",
  "InsightType": "ClinicalEntity",
  "Spans": [
    {
      "BeginCharacterOffset": 28,
      "Content": "phentermine",
      "EndCharacterOffset": 38,
      "SegmentId": "uuid4"
    }
  ],
  "Type": "GENERIC_NAME"
},
{
  "Attributes": [
    {
      "AttributeId": "attributeUUID1",
      "Spans": [
        {
          "BeginCharacterOffset": 38,
          "Content": "high",
          "EndCharacterOffset": 41,
          "SegmentId": "uuid5"
        }
      ]
    }
  ],
  "Type": "TEST_VALUE"
}
],
"Category": "TEST_TREATMENT_PROCEDURE",
"InsightId": "insightUUID5",
"InsightType": "ClinicalEntity",
"Spans": [
  {
    "BeginCharacterOffset": 14,
    "Content": "weight",
    "EndCharacterOffset": 19,
    "SegmentId": "uuid6"
  }
]
```

```
    ],
    "Type": "TEST_NAME"
  },
  {
    "Attributes": [],
    "Category": "ANATOMY",
    "InsightId": "insightUUID6",
    "InsightType": "ClinicalEntity",
    "Spans": [
      {
        "BeginCharacterOffset": 60,
        "Content": "heart",
        "EndCharacterOffset": 64,
        "SegmentId": "uuid7"
      }
    ],
    "Type": "SYSTEM_ORGAN_SITE"
  }
],
"TranscriptItems": [
  {
    "Alternatives": [
      {
        "Confidence": 0.7925,
        "Content": "Okay"
      }
    ],
    "BeginAudioTime": 0.16,
    "EndAudioTime": 0.6,
    "Type": "PRONUNCIATION"
  },
  {
    "Alternatives": [
      {
        "Confidence": 0,
        "Content": "."
      }
    ],
    "BeginAudioTime": 0,
    "EndAudioTime": 0,
    "Type": "PUNCTUATION"
  },
  {
    "Alternatives": [
```

```
    {
      "Confidence": 1,
      "Content": "Good"
    }
  ],
  "BeginAudioTime": 0.61,
  "EndAudioTime": 0.92,
  "Type": "PRONUNCIATION"
},
{
  "Alternatives": [
    {
      "Confidence": 1,
      "Content": "afternoon"
    }
  ],
  "BeginAudioTime": 0.92,
  "EndAudioTime": 1.54,
  "Type": "PRONUNCIATION"
},
{
  "Alternatives": [
    {
      "Confidence": 0,
      "Content": "."
    }
  ],
  "BeginAudioTime": 0,
  "EndAudioTime": 0,
  "Type": "PUNCTUATION"
},
{
  "Alternatives": [
    {
      "Confidence": 0.9924,
      "Content": "You"
    }
  ],
  "BeginAudioTime": 1.55,
  "EndAudioTime": 1.88,
  "Type": "PRONUNCIATION"
},
{
  "Alternatives": [
```

```
    {
      "Confidence": 1,
      "Content": "lost"
    }
  ],
  "BeginAudioTime": 1.88,
  "EndAudioTime": 2.19,
  "Type": "PRONUNCIATION"
},
{
  "Alternatives": [
    {
      "Confidence": 1,
      "Content": "one"
    }
  ],
  "BeginAudioTime": 2.19,
  "EndAudioTime": 2.4,
  "Type": "PRONUNCIATION"
},
{
  "Alternatives": [
    {
      "Confidence": 1,
      "Content": "lb"
    }
  ],
  "BeginAudioTime": 2.4,
  "EndAudioTime": 2.97,
  "Type": "PRONUNCIATION"
}
],
"TranscriptSegments": [
  {
    "BeginAudioTime": 0.16,
    "Content": "Okay.",
    "EndAudioTime": 0.6,
    "ParticipantDetails": {
      "ParticipantRole": "CLINICIAN_0"
    },
    "SectionDetails": {
      "SectionName": "SUBJECTIVE"
    },
    "SegmentId": "uuid1"
  }
]
```



```
  },
  {
    "BeginAudioTime": 0.61,
    "Content": "Good afternoon.",
    "EndAudioTime": 1.54,
    "ParticipantDetails": {
      "ParticipantRole": "CLINICIAN_0"
    },
    "SectionDetails": {
      "SectionName": "OTHER"
    },
    "SegmentId": "uuid2"
  },
  {
    "BeginAudioTime": 1.55,
    "Content": "You lost one lb.",
    "EndAudioTime": 2.97,
    "ParticipantDetails": {
      "ParticipantRole": "CLINICIAN_0"
    },
    "SectionDetails": {
      "SectionName": "SUBJECTIVE"
    },
    "SegmentId": "uuid3"
  },
  {
    "BeginAudioTime": 2.98,
    "Content": "Yeah, I think it, uh, do you feel more energy?",
    "EndAudioTime": 6.95,
    "ParticipantDetails": {
      "ParticipantRole": "CLINICIAN_0"
    },
    "SectionDetails": {
      "SectionName": "SUBJECTIVE"
    },
    "SegmentId": "uuid5"
  },
  {
    "BeginAudioTime": 6.96,
    "Content": "Yes.",
    "EndAudioTime": 7.88,
    "ParticipantDetails": {
      "ParticipantRole": "CLINICIAN_0"
    },
  },
```

```

    "SectionDetails": {
      "SectionName": "SUBJECTIVE"
    },
    "SegmentId": "uuid6"
  },
  {
    "BeginAudioTime": 7.89,
    "Content": "Uh, how about craving for the carbohydrate or sugar or fat or
anything?",
    "EndAudioTime": 17.93,
    "ParticipantDetails": {
      "ParticipantRole": "CLINICIAN_0"
    },
    "SectionDetails": {
      "SectionName": "SUBJECTIVE"
    },
    "SegmentId": "uuid7"
  }
]
}
}

```

Voici un autre exemple d'utilisation de la commande [start-medical-scribe-job](#) et d'un corps de requête avec des paramètres supplémentaires.

```

aws transcribe start-medical-scribe-job \
--region us-west-2 \
--cli-input-json file://filepath/my-first-medical-scribe-job.json

```

Le fichier `my-first-medical-scribe-job.json` contient le corps de requête suivant.

```

{
  "MedicalScribeJobName": "my-first-medical-scribe-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
  },
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "DataAccessRoleArn": "arn:aws:iam::111122223333:role/ExampleRole",
  "Settings": {

```

```
"ShowSpeakerLabels": false,
"ChannelIdentification": true
},
"ChannelDefinitions": [
  {
    "ChannelId": 0,
    "ParticipantRole": "CLINICIAN"
  }, {
    "ChannelId": 1,
    "ParticipantRole": "PATIENT"
  }
]
}
```

Exemple de sortie de documentation clinique

Un fichier d'informations sur la documentation (issu d'une demande `StartMedicalScribeJob`) a le format suivant :

```
{
  "ClinicalDocumentation": {
    "Sections": [
      {
        "SectionName": "CHIEF_COMPLAINT",
        "Summary": [
          {
            "EvidenceLinks": [
              {
                "SegmentId": "uuid1"
              },
              {
                "SegmentId": "uuid2"
              },
              {
                "SegmentId": "uuid3"
              },
              {
                "SegmentId": "uuid4"
              },
              {
                "SegmentId": "uuid5"
              }
            ]
          }
        ]
      }
    ]
  }
}
```

```

    },
    {
      "SegmentId": "uuid6"
    }
  ],
  "SummarizedSegment": "Weight loss."
}
]
},
{
  "SectionName": "HISTORY_OF_PRESENT_ILLNESS",
  "Summary": [
    {
      "EvidenceLinks": [
        {
          "SegmentId": "uuid7"
        },
        {
          "SegmentId": "uuid8"
        },
        {
          "SegmentId": "uuid9"
        },
        {
          "SegmentId": "uuid10"
        }
      ],
      "SummarizedSegment": "The patient is seen today for a follow-up of weight
loss."
    },
    {
      "EvidenceLinks": [
        {
          "SegmentId": "uuid11"
        },
        {
          "SegmentId": "uuid12"
        },
        {
          "SegmentId": "uuid13"
        }
      ],
      "SummarizedSegment": "They report feeling more energy and craving
carbohydrates, sugar, and fat."
    }
  ]
}

```

```
    },
    {
      "EvidenceLinks": [
        {
          "SegmentId": "uuid14"
        },
        {
          "SegmentId": "uuid15"
        },
        {
          "SegmentId": "uuid16"
        }
      ],
      "SummarizedSegment": "The patient is up to date on their mammogram and pap smear."
    },
    {
      "EvidenceLinks": [
        {
          "SegmentId": "uuid17"
        },
        {
          "SegmentId": "uuid18"
        },
        {
          "SegmentId": "uuid19"
        },
        {
          "SegmentId": "uuid20"
        }
      ],
      "SummarizedSegment": "The patient is taking phentermine and would like to continue."
    }
  ]
},
{
  "SectionName": "REVIEW_OF_SYSTEMS",
  "Summary": [
    {
      "EvidenceLinks": [
        {
          "SegmentId": "uuid21"
        }
      ],

```

```
    {
      "SegmentId": "uuid22"
    }
  ],
  "SummarizedSegment": "Patient reports intermittent headaches, occasional
chest pains but denies any recent fevers or chills."
},
{
  "EvidenceLinks": [
    {
      "SegmentId": "uuid23"
    },
    {
      "SegmentId": "uuid24"
    }
  ],
  "SummarizedSegment": "No recent changes in vision, hearing, or any
respiratory complaints."
}
],
},
{
  "SectionName": "PAST_MEDICAL_HISTORY",
  "Summary": [
    {
      "EvidenceLinks": [
        {
          "SegmentId": "uuid25"
        },
        {
          "SegmentId": "uuid26"
        }
      ],
      "SummarizedSegment": "Patient has a history of hypertension and was
diagnosed with Type II diabetes 5 years ago."
    },
    {
      "EvidenceLinks": [
        {
          "SegmentId": "uuid27"
        },
        {
          "SegmentId": "uuid28"
        }
      ]
    }
  ]
}
```

```
    ],
    "SummarizedSegment": "Underwent an appendectomy in the early '90s and had a
fracture in the left arm during childhood."
  }
]
},
{
  "SectionName": "ASSESSMENT",
  "Summary": [
    {
      "EvidenceLinks": [
        {
          "SegmentId": "uuid29"
        },
        {
          "SegmentId": "uuid30"
        }
      ],
      "SummarizedSegment": "Weight loss"
    }
  ]
},
{
  "SectionName": "PLAN",
  "Summary": [
    {
      "EvidenceLinks": [
        {
          "SegmentId": "uuid31"
        },
        {
          "SegmentId": "uuid32"
        },
        {
          "SegmentId": "uuid33"
        },
        {
          "SegmentId": "uuid34"
        }
      ],
      "SummarizedSegment": "For the condition of Weight loss: The patient was
given a 30-day supply of phentermine and was advised to follow up in 30 days."
    }
  ]
}
```

```
}  
  ]  
  }  
}
```

Chiffrement des données au repos pour AWS HealthScribe

AWS HealthScribe fournit un chiffrement par défaut pour protéger les données au repos sensibles des clients à l'aide de clés gérées par Amazon S3.

- Clés gérées par Amazon S3 (SSE-S3) : AWS HealthScribe utilise des clés gérées par Amazon S3 par défaut pour chiffrer automatiquement les fichiers intermédiaires. Vous ne pouvez pas afficher, gérer ou utiliser les clés gérées par Amazon S3, ou vérifier leur utilisation. Toutefois, vous n'avez pas besoin de prendre de mesure ou de modifier les programmes pour protéger les clés qui chiffrent les données. Pour plus d'informations, consultez [SSE-S3](#).

Le chiffrement des données au repos par défaut permet de réduire les frais opérationnels et la complexité liés à la protection des données sensibles. Dans le même temps, il vous permet de créer des applications sécurisées qui répondent aux exigences réglementaires et de conformité strictes en matière de chiffrement.

Bien que vous ne puissiez pas désactiver cette couche de chiffrement ou sélectionner un autre type de chiffrement, vous pouvez ajouter une deuxième couche de chiffrement aux clés gérées par Amazon S3 en choisissant une CMK lorsque vous créez une tâche avec AWS HealthScribe.

- CMK : AWS HealthScribe prend en charge l'utilisation d'une CMK symétrique que vous créez, détenez et gérez afin d'ajouter une deuxième couche de chiffrement au chiffrement existant détenu par AWS. Étant donné que vous avez le contrôle total de cette couche de chiffrement, vous pouvez effectuer les tâches suivantes :
 - Établissement et gestion des stratégies de clé
 - Établissement et gestion des politiques IAM et des octrois
 - Activation et désactivation des stratégies de clé
 - Rotation des matériaux de chiffrement de clé
 - Ajout de balises
 - Création d'alias de clé

- Planification des clés pour la suppression

Pour plus d'informations, consultez [CMK](#) dans le Guide du développeur AWS Key Management Service.

Note

AWS HealthScribe active automatiquement le chiffrement au repos à l'aide de clés détenues par AWS afin de protéger gratuitement les données d'identification personnelle. Toutefois, AWS KMS facture des frais liés à l'utilisation d'une CMK. Pour de plus amples informations sur la tarification, veuillez consulter [Tarification AWS Key Management Service](#).

Pour plus d'informations sur AWS KMS, consultez [Présentation de AWS Key Management Service](#).

Création d'une clé gérée par le client

Vous pouvez créer une CMK symétrique à l'aide de la AWS Management Console ou des API AWS KMS. Pour créer une CMK symétrique, suivez les étapes de [Création d'une CMK symétrique](#) dans le Guide du développeur AWS Key Management Service.

Les politiques de clés contrôlent l'accès à votre clé gérée par le client. Chaque clé gérée par le client doit avoir exactement une politique de clé, qui contient des instructions qui déterminent les personnes pouvant utiliser la clé et comment elles peuvent l'utiliser. Lorsque vous créez votre clé gérée par le client, vous pouvez spécifier une politique de clé. Pour plus d'informations, consultez [Gestion de l'accès aux CMK](#) dans le Guide du développeur AWS Key Management Service.

Si vous utilisez une clé dans le même compte que le rôle IAM que vous spécifiez comme [DataAccessRoleArn](#) dans la demande [StartMedicalScribeJob](#), vous n'avez pas besoin de mettre à jour la stratégie de clé. Pour utiliser la CMK dans un autre compte en tant que `DataAccessRole`, vous devez faire confiance au `DataAccessRoleArn` dans la stratégie de clé pour les actions suivantes :

- [kms:Encrypt](#) : autorise le chiffrement à l'aide de la CMK.
- [kms:Decrypt](#) : autorise le déchiffrement à l'aide de la CMK.
- [kms:DescribeKey](#) : fournit les détails des CMK pour permettre à AWS HealthScribe de valider la clé.

Voici un exemple de déclaration de politique que vous pouvez ajouter pour autoriser votre rôle IAM à utiliser la CMK sur plusieurs comptes :

```
"Statement" : [
  {
    "Sid": "Allow access to the DataAccessRole for StartMedicalScribeJob",
    "Effect": "Allow",
    "Principal": {
      "AWS": "arn:aws:iam::111122223333:role/DataAccessRole"
    },
    "Action": [
      "kms:DescribeKey",
      "kms:Encrypt",
      "kms:Decrypt",
      "kms:GenerateDataKey"
    ],
    "Resource" : "*"
  }
]
```

Que la clé gérée par le client et le DataAccessRole se trouvent dans le même compte ou dans des comptes différents, le DataAccessRole aura besoin d'autorisations pour effectuer les actions mentionnées ci-dessus à l'aide de la CMK. Voici un exemple de déclaration de politique que vous pouvez ajouter au DataAccessRole :

```
"Statement" : [
  {
    "Sid": "Allow role to perform AWS KMS actions for customer managed key",
    "Effect": "Allow",
    "Action": [
      "kms:DescribeKey",
      "kms:Encrypt",
      "kms:Decrypt"
    ],
    "Resource": "*"
  }
]
```

Pour plus d'informations sur la [spécification d'autorisations dans une politique](#), consultez le Guide du développeur AWS Key Management Service. Pour plus d'informations sur la [résolution des problèmes liés aux clés d'accès](#), consultez le Guide du développeur AWS Key Management Service.

Spécification d'une CMK pour AWS HealthScribe

Vous pouvez spécifier une CMK en tant que chiffrement de deuxième couche pour les demandes `StartMedicalScribeJob`. Lorsque vous créez une demande [StartMedicalScribeJob](#), vous pouvez spécifier une CMK en incluant le champ [OutputEncryptionKmsKeyId](#) dans la demande.

Contexte de chiffrement AWS KMS

Le contexte de chiffrement AWS KMS est une carte de paires clé-valeur non secrètes en texte brut. Cette carte représente des données authentifiées supplémentaires, appelées paires de contextes de chiffrement, qui fournissent une couche de sécurité supplémentaire aux données. AWS HealthScribe nécessite une clé de chiffrement symétrique pour chiffrer la sortie AWS HealthScribe dans un compartiment Amazon S3 spécifié par le client. Pour en savoir plus, consultez la section [Clés asymétriques dans AWS KMS](#).

Lorsque vous créez vos paires de contextes de chiffrement, n'incluez pas d'informations sensibles. Le contexte de chiffrement n'est pas secret : il est visible en texte brut dans les journaux CloudTrail (vous pouvez donc l'utiliser pour identifier et classer les opérations cryptographiques). La paire de contextes de chiffrement peut inclure des caractères spéciaux, notamment des traits de soulignement (), des tirets (-), des barres obliques (/, \) et deux points (:).

Tip

Il peut être utile de relier les valeurs de votre paire de contextes de chiffrement aux données chiffrées. Bien que cela ne soit pas obligatoire, nous vous recommandons d'utiliser des métadonnées non sensibles relatives à votre contenu chiffré, telles que les noms de fichiers, les valeurs d'en-tête ou les champs de base de données non chiffrés.

Pour utiliser le chiffrement de sortie avec l'API, définissez le paramètre

[KMSEncryptionContext](#) dans l'opération [StartMedicalScribeJob](#). Pour fournir un contexte de chiffrement pour l'opération de chiffrement de sortie, le paramètre

[OutputEncryptionKMSKeyId](#) doit faire référence à un ID de clé AWS KMS symétrique.

Vous pouvez utiliser des [clés de condition AWS KMS](#) avec des politiques IAM pour contrôler l'accès à une clé AWS KMS de chiffrement symétrique en fonction du contexte de chiffrement utilisé dans la demande d'[opération de chiffrement](#). Pour un exemple de politique de contexte de chiffrement, consultez [Politique de contexte de chiffrement AWS KMS](#).

L'utilisation du contexte de chiffrement est facultative mais recommandée. Pour plus d'informations, consultez la section [Contexte de chiffrement](#).

Historique du document pour Amazon Transcribe

- Dernière mise à jour de la documentation : 13 novembre 2023

Le tableau suivant décrit les modifications importantes apportées à chaque version de Amazon Transcribe. Pour recevoir les notifications de mise à jour de cette documentation, abonnez-vous à un flux RSS.

Modification	Description	Date
Mise à jour de la fonctionnalité	Mettez à jour le nombre maximum de haut-parleurs dans la diarisation à 30 au lieu de 10.	10 mai 2024
Mise à jour de la section	Mises à jour de la synthèse générative des appels et ajout de détails sur les sorties d'erreur.	30 avril 2024
Mise à jour de la section	Mises à jour sur les colonnes de vocabulaire personnalisées - IPA et SoundsLike.	30 avril 2024
Mise à jour de la fonctionnalité	Amazon Transcribe Call Analytics prend désormais en charge le résumé génératif des appels.	29 novembre 2023
Mise à jour de la section	Mettez à jour le nouveau format de sortie d'expurgation des données d'identification personnelle (PII) et de l'identification de la langue.	13 novembre 2023

Mise à jour de la fonctionnalité	La diarisation peut désormais être combinée à l'identification des canaux.	06 mars 2023
Mise à jour de la fonctionnalité	L'identification des canaux peut désormais être combinée à la diarisation.	06 mars 2023
Mise à jour de la section	IAM les meilleures pratiques ont été mises à jour.	13 février 2023
Nouvelles langues	Amazon Transcribe supporte désormais le vietnamien et le suédois.	6 décembre 2022
Nouvelle fonction	Amazon Transcribe prend désormais en charge l'analyse des appels en temps réel.	28 novembre 2022
Mise à jour de la fonctionnalité	L'expurgation et l'identification en streaming sont désormais disponibles en hindi et en thaï.	11 novembre 2022
Mise à jour de la section	De nouvelles catégories d'informations d'identification personnelles sont disponibles pour l'expurgation et l'identification en streaming.	14 septembre 2022
Mise à jour de la section	La section du modèle de langue personnalisé a été révisée.	18 juin 2022
Mise à jour de la section	L'identification de la langue par lots permet désormais d'identifier plusieurs langues par fichier audio.	31 mai 2022

Mise à jour du guide	La référence Amazon Transcribe d'API est désormais un guide autonome.	1er avril 2022
Nouveau chapitre	Un nouveau tableau comparatif pour Amazon Transcribe les Amazon Transcribe domaines médical et d'analyse des Amazon Transcribe appels est inclus.	21 mars 2022
Nouveau chapitre	Un nouveau chapitre d'exemples de code de kit SDK est inclus.	21 mars 2022
Mise à jour de la fonctionnalité	Call Analytics fournit désormais un récapitulatif des appels.	21 mars 2022
Mise à jour du chapitre	Le chapitre d'introduction présente désormais des cas Amazon Transcribe d'utilisation.	21 mars 2022
Mise à jour du chapitre	Le chapitre Mise en route a été mis à jour pour être spécifique à la méthode.	21 mars 2022
Mise à jour du chapitre	Le chapitre Streaming a été mis à jour et restructuré.	21 mars 2022
Mise à jour de la fonctionnalité	L'identification de la langue prend désormais en charge les vocabulaires personnalisés et les filtres de vocabulaire personnalisé avec les transcriptions en streaming.	11 mars 2022

Nouvel événement	Il existe un nouveau type d'événement : les événements de vocabulaire.	7 février 2022
Mise à jour de la section	Des mises à jour ont été apportées à la section sur les vocabulaires personnalisés.	20 janvier 2022
Nouvelle fonction	L'identification de la langue peut désormais être utilisée avec les transcriptions en streaming.	23 novembre 2021
Nouvelle fonction	L'identification de la langue peut désormais être utilisée avec des modèles de langue personnalisés, des vocabulaires personnalisés, le filtrage du vocabulaire et l'expurgation de contenu.	29 octobre 2021
Nouvelle fonction	Amazon Transcribe prend désormais en charge des modèles linguistiques personnalisés avec des transcriptions en streaming.	20 octobre 2021
Nouvelle fonction	Amazon Transcribe peut désormais générer des sous-titres pour vos fichiers vidéo.	16 septembre 2021
Nouvelle fonction	Amazon Transcribe prend désormais en charge la rédaction et l'identification des informations personnelles pour le streaming.	14 septembre 2021

Nouvelle fonction	Amazon Transcribe prend désormais en charge le contexte de AWS KMS chiffrement pour renforcer le niveau de sécurité de vos Compte AWS ressources.	10 septembre 2021
Nouvelles langues	Amazon Transcribe prend désormais en charge l'afrikaans, le danois, le chinois mandarin (traditionnel), le thaï, l'anglais néo-zélandais et l'anglais sud-africain.	26 août 2021
Nouvelle fonction	Amazon Transcribe prend désormais en charge le balisage des ressources.	24 août 2021
Nouvelle fonction	Amazon Transcribe prend désormais en charge Call Analytics pour les tâches de transcription par lots.	4 août 2021
Nouvelle fonction	Amazon Transcribe prend désormais en charge l'utilisation de vocabulaires personnalisés avec des modèles linguistiques personnalisés par lots.	12 mai 2021
Nouvelle fonction	Amazon Transcribe prend désormais en charge la stabilisation partielle des résultats pour la transcription en streaming.	11 mai 2021

Nouvelle fonction	Amazon Transcribe prend désormais en charge l'anglais australien, l'anglais britannique, l'hindi et l'espagnol américain pour les modèles linguistiques personnalisés.	19 mars 2021
Nouvelle fonction	Amazon Transcribe supporte désormais les codecs OGG/OPUS et FLAC pour la transcription audio en streaming.	24 novembre 2020
Nouvelles langues	Amazon Transcribe ajoute le support de l'italien et de l'allemand pour la transcription audio en streaming.	4 novembre 2020
Région AWS expansion	Amazon Transcribe est désormais disponible à Francfort (eu-central-1) et à Londres (eu-west-2).	4 novembre 2020
Nouvelle fonction	Amazon Transcribe ajoute la prise en charge des points de terminaison VPC d'interface dans la transcription par lots.	9 octobre 2020
Nouvelle fonction	Amazon Transcribe ajoute le support pour l'identification des chaînes en streaming.	17 septembre 2020
Nouvelle fonction	Amazon Transcribe ajoute le support pour l'identification automatique de la langue dans la transcription par lots.	15 septembre 2020

Nouvelle fonction	Amazon Transcribe ajoute la prise en charge du partitionnement des haut-parleurs en streaming.	19 août 2020
Nouvelle fonction	Amazon Transcribe ajoute le support pour les modèles linguistiques personnalisés.	5 août 2020
Nouvelle fonction	Amazon Transcribe ajoute la prise en charge des points de terminaison VPC d'interface en streaming.	26 juin 2020
Nouvelle fonction	Amazon Transcribe ajoute la prise en charge du filtrage du vocabulaire en streaming.	20 mai 2020
Nouvelle fonction	Amazon Transcribe ajoute la prise en charge de la suppression automatique des informations personnelles identifiables.	26 février 2020
Nouvelle fonction	Amazon Transcribe ajoute la prise en charge de la création d'un vocabulaire personnalisé de mots à filtrer dans une transcription.	20 décembre 2019
Nouvelle fonction	Amazon Transcribe ajoute la prise en charge de la mise en file d'attente des tâches de transcription.	19 décembre 2019

Nouvelles langues	Amazon Transcribe ajoute le support pour l'arabe du Golfe, l'hébreu, le japonais, le malais, le suisse-allemand, le télougou et le turc.	21 novembre 2019
Région AWS expansion	Amazon Transcribe est désormais disponible dans la région Asie-Pacifique (Tokyo) (ap-northeast-1).	21 novembre 2019
Nouvelle fonction	Amazon Transcribe ajoute le support pour les transcriptions alternatives.	20 novembre 2019
Nouvelles langues	Amazon Transcribe ajoute la prise en charge du néerlandais, du farsi, de l'indonésien, de l'anglais irlandais, du portugais, de l'anglais écossais, du tamoul et de l'anglais gallois.	12 novembre 2019
Nouvelle langue	Amazon Transcribe prend désormais en charge la transcription en streaming pour l'anglais australien (en-AU).	25 octobre 2019
Région AWS expansion	Amazon Transcribe est désormais disponible en Chine (Pékin) (cn-north-1) et en Chine (Ningxia) (cn-north-west-1).	9 octobre 2019

Nouvelle fonction	Amazon Transcribe vous permet de fournir les vôtres KMS key pour crypter vos fichiers de sortie de transcription. Pour plus d'informations, consultez le KeyId paramètre OutputEncryptionKMS de l' StartStreamTranscriptionAPI .	24 septembre 2019
Nouvelles langues	Amazon Transcribe ajoute le support pour le chinois (mandarin), le chinois simplifié , la Chine continentale et le russe.	23 août 2019
Nouvelle fonction	Amazon Transcribe ajoute le support pour la transcription audio en streaming à l'aide du WebSocket protocole.	19 juillet 2019
Nouvelle fonction	AWS CloudTrail enregistre désormais les événements pour l' StartStreamTranscriptionAPI .	19 juillet 2019
Région AWS expansion	Amazon Transcribe est désormais disponible dans l'ouest des États-Unis (Californie du Nord) (us-west-1).	27 juin 2019
Nouvelle langue	Amazon Transcribe ajoute le support pour l'arabe standard moderne.	28 mai 2019

Nouvelle fonction	Amazon Transcribe transcrit désormais des mots numériques en chiffres pour l'anglais américain. Par exemple, « quarante-deux » est transcrit en « 42 ».	23 mai 2019
Nouvelle langue	Amazon Transcribe ajoute le support pour l'hindi et l'anglais indien.	15 mai 2019
Nouveau kit SDK	Le AWS SDK pour C++ prend désormais Amazon Transcribe en charge.	8 mai 2019
Nouvelle langue	Amazon Transcribe ajoute le support pour l'espagnol.	19 avril 2019
Région AWS expansion	Amazon Transcribe est désormais disponible dans l'UE (Francfort) (eu-central-1) et en Asie-Pacifique (Séoul) (ap-northeast-2).	18 avril 2019
Nouvelle langue	Amazon Transcribe ajoute le support pour la transcription en streaming en anglais britannique, en français et en français canadien.	5 avril 2019
Nouvelle fonction	Le AWS SDK pour Ruby V3 prend désormais en charge Amazon Transcribe	25 mars 2019

Nouvelle fonction	Amazon Transcribe autorise les vocabulaires personnalisés, qui sont des listes de mots spécifiques que vous Amazon Transcribe souhaitez reconnaître dans votre entrée audio.	25 mars 2019
Nouvelles langues	Amazon Transcribe ajoute le support pour l'allemand et le coréen.	22 mars 2019
Nouvelle langue	Amazon Transcribe prend désormais en charge la transcription en streaming pour l'espagnol américain (es-US).	7 février 2019
Région AWS expansion	Amazon Transcribe est désormais disponible en Amérique du Sud (São Paulo) (sa-east-1).	7 février 2019
Région AWS expansion	Amazon Transcribe est désormais disponible en Asie-Pacifique (Mumbai) (ap-south-1), en Asie-Pacifique (Singapour) (ap-southeast-1), dans l'UE (Londres) (eu-west-2) et dans l'UE (Paris) (eu-west3).	24 janvier 2019
Nouvelles langues	Amazon Transcribe ajoute le support pour le français, l'italien et le portugais brésilien.	20 décembre 2018

Nouvelle fonction	Amazon Transcribe prend désormais en charge la transcription de flux audio.	19 novembre 2018
Nouvelles langues	Amazon Transcribe ajoute le support pour l'anglais australien, l'anglais britannique et le français canadien.	15 novembre 2018
Région AWS expansion	Amazon Transcribe est désormais disponible au Canada (centre) (ca-central-1) et en Asie-Pacifique (Sydney) (ap-southeast-2).	17 juillet 2018
Nouvelle fonction	Vous pouvez désormais spécifier votre propre emplacement de stockage de la sortie d'une tâche de transcription.	11 juillet 2018
Nouvelle fonction	Ajout AWS CloudTrail et Amazon CloudWatch Events intégration.	28 juin 2018
Nouvelle fonction	Amazon Transcribe ajoute le support pour les vocabulaires personnalisés.	4 avril 2018
Nouveau guide	Il s'agit de la première version du Guide du développeur Amazon Transcribe .	29 novembre 2017

Glossaire AWS

Pour connaître la terminologie la plus récente d'AWS, consultez le [Glossaire AWS](#) dans la Référence Glossaire AWS.

Les traductions sont fournies par des outils de traduction automatique. En cas de conflit entre le contenu d'une traduction et celui de la version originale en anglais, la version anglaise prévaudra.